Roberto Moreno Díaz
Franz Pichler
Alexis Quesada Arencibia (Eds.)

LNCS 3643

Computer Aided Systems Theory – EUROCAST 2005

10th International Conference on Computer Aided Systems Theory
Las Palmas de Gran Canaria, Spain, February 2005
Revised Selected Papers

Springer

Lecture Notes in Computer Science 3643

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison
Lancaster University, UK
Takeo Kanade
Carnegie Mellon University, Pittsburgh, PA, USA
Josef Kittler
University of Surrey, Guildford, UK
Jon M. Kleinberg
Cornell University, Ithaca, NY, USA
Friedemann Mattern
ETH Zurich, Switzerland
John C. Mitchell
Stanford University, CA, USA
Moni Naor
Weizmann Institute of Science, Rehovot, Israel
Oscar Nierstrasz
University of Bern, Switzerland
C. Pandu Rangan
Indian Institute of Technology, Madras, India
Bernhard Steffen
University of Dortmund, Germany
Madhu Sudan
Massachusetts Institute of Technology, MA, USA
Demetri Terzopoulos
New York University, NY, USA
Doug Tygar
University of California, Berkeley, CA, USA
Moshe Y. Vardi
Rice University, Houston, TX, USA
Gerhard Weikum
Max-Planck Institute of Computer Science, Saarbruecken, Germany

Roberto Moreno Díaz   Franz Pichler
Alexis Quesada Arencibia (Eds.)

Computer Aided Systems Theory – EUROCAST 2005

10th International Conference on Computer Aided Systems Theory
Las Palmas de Gran Canaria, Spain, February 7 – 11, 2005
Revised Selected Papers

Springer

Volume Editors

Roberto Moreno Díaz
Universidad de Las Palmas de Gran Canaria
Instituto Universitario de Ciencias y Tecnologícas Cibernéticas
Campus de Tafira, 35017, Las Palmas de Gran Canaria, Las Palmas, Spain
E-mail: rmoreno@ciber.ulpgc.es

Franz Pichler
Johannes Kepler University Linz
Institute of Systems Science
Austria
E-mail: pichler@cast.uni-linz.ac.at

Alexis Quesada Arencibia
Universidad de Las Palmas de Gran Canaria
Instituto Universitario de Ciencias y Tecnologícas Cibernéticas
Campus de Tafira, 35017, Las Palmas de Gran Canaria, Las Palmas, Spain
E-mail: aquesada@dis.ulpgc.es

Library of Congress Control Number: 2005932551

CR Subject Classification (1998): J.6, I.6, I.2, J.7, J.3, C.1.m, C.3, F.4, F.3

ISSN 0302-9743
ISBN-10 3-540-29002-8 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-29002-5 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springeronline.com

© Springer-Verlag Berlin Heidelberg 2005
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11556985 06/3142 5 4 3 2 1 0

Preface

The concept of CAST, computer aided systems Theory, was introduced by F. Pichler of Linz in the late 1980s to include those computer theoretical and practical developments used as tools to solve problems in system science. It was considered as the third component (the other two being CAD and CAM) that would provide for a complete picture of the path from computer and systems sciences to practical developments in science and engineering.

The University of Linz organized the first CAST workshop in April 1988, which demonstrated the acceptance of the concepts by the scientific and technical community. Next, the University of Las Palmas de Gran Canaria joined the University of Linz to organize the first international meeting on CAST (Las Palmas February 1989), under the name EUROCAST 1989, a very successful gathering of systems theorists, computer scientists and engineers from most European countries, North America and Japan.

It was agreed that EUROCAST international conferences would be organized every two years. Thus, the following EUROCAST meetings took place in Krems (1991), Las Palmas (1993), Innsbruck (1995), Las Palmas (1997), Vienna (1999), Las Palmas (2001) and Las Palmas (2003) in addition to an extra-European CAST conference in Ottawa in 1994. Selected papers from those meetings were published as Springer Lecture Notes in Computer Science vols. 410, 585, 763, 1030, 1333, 1728, 2178 and 2809 and in several special issues of Cybernetics and Systems: an lnternational Journal. EUROCAST and CAST meetings are definitely consolidated, as has been demonstrated by the number and quality of the contributions over the years.

EUROCAST 2005 (Las Palmas, February 2005) continued with a new approach to the conferences which was adopted in 2001. Besides the classical core on generic CAST, chaired by Pichler and Moreno-Diaz, there were workshops on Computation and Simulation in Modelling Biological Systems, chaired by Ricciardi (Naples); Cryptography, chaired by Müller (Klagenfurt); Intelligent Information Processing, chaired by Freire (A Coruña); Robotics and Robot Soccer, chaired by Kopacek (Vienna) and Pfalgraf (Salzburg); Spectral Methods, chaired by Astola (Tampere); and Computer Vision and Intelligent Vehicular Systems, chaired by Maravall and García Rosa (Madrid).

This volume contains the full papers selected after the oral presentations of the different sessions. The editors would like to thank all contributors for their quickness in providing their material in hard and electronic forms. Special thanks are due to the staff of Springer Heidelberg for their valuable support.

July 2005 Roberto Moreno-Díaz, Franz Pichler and Alexis Quesada-Arencibia

Table of Contents

Formal Approaches in Modelling

On the Physical Formal and Semantic Frontiers Between Human Knowing and Machine Knowing
José Mira Mira 1

Approximation Problems Categories
Liara Aparecida dos Santos Leal, Dalcidio Moraes Claudio, Laira Vieira Toscani, Paulo Blauth Menezes 9

Computation of Partial Automata Through Span Composition
Karina Girardi Roggia, Marnes Augusto Hoff, Paulo Blauth Menezes 15

Degenerate Arrays: A Framework for Uncertain Data Tables
Margaret Miró-Julià 21

Neural Network Sensitivity Analysis Applied for the Reduction of the Sensor Matrix
Przemyslaw M. Szecówka, Andrzej Szczurek, Maciej Mazurowski, Benedykt W. Licznerski, Franz Pichler 27

Fuzzy Modeling for Coal Seams A Case Study for a Hard-Coal Mine
José Antonio Martin, Teresa de Pedro, Carlos González, Ricardo García, Luís Argüelles, Jose M. Rivas, Javier Toraño 33

Optimization of a Class of Uncertain Systems Based on Uncertain Variables
Zdzislaw Bubnicki 38

Computational Simulation of Categorical Constructions
Rodrigo Born Vieira, Paulo Blauth Menezes 44

Composing Transitions into Transactions in UML Diagrams
Júlio Pereira Machado, Paulo Blauth Menezes 50

Theory-Building with System Dynamics: Principles and Practices
Markus Schwaninger, Thomas K. Hamann 56

Ontology Integration for Statistical Information
Wilfried Grossmann, Markus Moschner 63

Intelligent Information Systems

On Recursive Functions and Well–Founded Relations in the Calculus of Constructions
José L. Freire, Enrique Freire, Antonio Blanco 69

Longest Sorted Sequence Algorithm for Parallel Text Alignment
Tiago Ildefonso, Gabriel Pereira Lopes 81

Information Retrieval and Large Text Structured Corpora
Fco. Mario Barcala, Miguel A. Molinero, Eva Domínguez 91

Meteorological Image Descriptors
José L. Crespo, Pilar Bernardos, Marta E. Zorrilla, Eduardo Mora 101

Towards a Certified and Efficient Computing of Gröbner Bases
J. Santiago Jorge, Víctor M. Gulías, José L. Freire, Juan J. Sánchez 111

CheapTB: A Low Cost of Operation Distributed Filesystem
Javier París, Victor M. Gulías, Carlos Abalde 121

Spelling Correction on Technical Documents
Manuel Vilares, Juan Otero, Jorge Graña 131

Verification of Language Based Fault-Tolerance
Clara Benac Earle, Lars-Åke Fredlund 140

Applying Stacking and Corpus Transformation to a Chunking Task
José A. Troyano, Víctor J. Díaz, Fernando Enríquez, Vicente Carrillo, Fermín Cruz 150

Extracting Computer Algebra Programs from Statements
Jesús Aransay, Clemens Ballarin, Julio Rubio 159

Integrating Syntactic Information by Means of Data Fusion Techniques
Francisco J. Ribadas, Jesús Vilares, Miguel A. Alonso 169

Unsupervised Learning in Information Retrieval Using NOW Architectures
E.F. Combarro, J. Ranilla, R. Mones, N. Vázquez, I. Díaz, E. Montañés 179

An Iterative Method for Mining Frequent Temporal Patterns
Francisco Guil, Antonio Bailón, Alfonso Bosch, Roque Marín 189

Information Applications Components

Data Mining with Scatter Search
I.J. García del Amo, M. García Torres, B. Melián Batista, J.A. Moreno Pérez, J.M. Moreno Vega, Raquel Rivero Martín 199

Web Usage Mining Project for Improving Web-Based Learning Sites
M.E. Zorrilla, E. Menasalvas, D. Marín, E. Mora, J. Segovia 205

Similarity Queries in Data Bases Using Metric Distances - from Modeling Semantics to Its Maintenance
Josef Küng, Roland Wagner 211

A WEB-CASE Tool Prototype for Hybrid Software Development
Francisco J. Orellana, Francisco Guil, Isabel M. del Aguila, Samuel Túnez 217

An Augmentative Communication System Based on Adaptive Evolutionary Hypermedia Systems
M. Visitacíon Hurtado, Nuria Medina, Lina García-Cabrera, María L. Rodríguez 223

The Gaps of the Thesaurus Wordnet Used in Information Retrieval
Javier de la Mata, Jose A. Olivas, Jesús Serrano-Guerrero 229

Fuzzy Adaptive Objects (Logic of Monitors)
Germano Resconi, Javier Alonso, Raul Izquierdo 235

A Model-Based Architecture for Fuzzy Temporal Diagnosis
José M. Juarez, José Palma, Manuel Campos, José Salort, Antonio Morales, Roque Marin 241

Extension of Ontologies Assisted by Automated Reasoning Systems
Joaquín Borrego-Díaz, Antonia M. Chávez-González 247

A Software Architecture for Effective Document Identifier Reassignment
Roi Blanco, Álvaro Barreiro 254

An Ontology for Reusing Synthetic Tasks
Abraham Rodríguez-Rodríguez, Francisca Quintana-Domínguez 263

A Tractable Subclass of Fuzzy Constraint Networks
Alfonso Bosch, Francisco Guil, Roque Marin 269

Parallel State Space Generation and Exploration on Shared-Memory Architectures
Milan Češka, Bohuslav Křena, Tomáš Vojnar 275

Towards Automated Controlling of Human Projectworking Based on Multiagent Systems
Manfred Mauerkirchner, Gerhard Hoefer 281

Cryptography and Spectral Analysis

Tree-Structured Legendre Multi-wavelets
Ekaterina Pogossova, Karen Egiazarian, Atanas Gotchev, Jaakko Astola 291

Remarks on Calculation of Autocorrelation on Finite Dyadic Groups by Local Transformations of Decision Diagrams
Radomir S. Stanković, Mark G. Karpovsky 301

A New Pseudo-Random Generator Based on Gollmann Cascades of Baker-Register-Machines
Dominik Jochinger, Franz Pichler 311

An Excellent Permutation Operator for Cryptographic Applications
Josef Scharinger 317

Fault Cryptanalysis of ElGamal Signature Scheme
Janusz Biernat, Maciej Nikodem 327

Complexity-Theoretical Approaches to the Design and Analysis of Cryptographical Boolean Functions
Juan David González Cobas, José Antonio López Brugos 337

Algorithm for Proving the Knowledge of an Independent Vertex Set
Pino Caballero-Gil, Candelaria Hernández-Goya 346

Improvement of the Edit Distance Attack to Clock-Controlled LFSR-Based Stream Ciphers
Pino Caballero-Gil, Amparo Fúster-Sabater 355

Protocol Analysis for Concrete Environments
Dieter Gollmann 365

Computer Vision

Pattern Recognition in AVHRR Images by Means of Hibryd and Neuro-fuzzy Systems
Jose Antonio Piedra, Francisco Guindos, Alberto Molina, Manuel Canton 373

Image Processing Techniques for Braille Writing Recognition
Néstor Falcón, Carlos M. Travieso, Jesús B. Alonso, Miguel A. Ferrer 379

Retinal Based Authentication via Distributed Web Application
Castor Mariño, Manuel G. Penedo, Marta Penas 386

Skeleton Extraction of 2D Objects Using Shock Wavefront Detection
Rubén Cárdenes, Juan Ruiz-Alzola 392

Cue Combination for Robust Real-Time Multiple Face Detection at Different Resolutions
Modesto Castrillón-Santana, Oscar Déniz-Suárez, Cayetano Guerra-Artal, José Isern-González 398

Evolutionary Color Constancy Algorithm Based on the Gamut Mapping Paradigm
Cristian Munteanu, Agostinho Rosa, Manuel Galan, Enrique Rubio Royo 404

Vision Based Automatic Occupant Classification and Pose Recognition for Smart Airbag Deployment
Min-Soo Jang, Yong-Guk Kim, Sang-Jun Kim, Jeong-Eom Lee, Soek-Joo Lee, Gwi-Tae Park 410

Biocomputing

A Wiener Neuronal Model with Refractoriness
Virginia Giorno, Amelia G. Nobile, Luigi M. Ricciardi 416

On Myosin II Dynamics: From a Pulsating Ratchet to a Washboard Potential
Aniello Buonocore, Ludovico Caputo, Enrica Pirozzi, Luigi M. Ricciardi 426

Feedback Effects in Simulated Stein's Coupled Neurons
Antonio Di Crescenzo, Barbara Martinucci, Enrica Pirozzi 436

Upcrossing First Passage Times for Correlated Gaussian Processes
Virginia Giorno, Amelia G. Nobile, Enrica Pirozzi 447

Convergence of Iterations
Paul Cull 457

Semiautomatic Snake-Based Segmentation of Solid Breast Nodules on Ultrasonography
Miguel Alemán-Flores, Patricia Alemán-Flores, Luis Álvarez-León, M. Belén Esteban-Sánchez, Rafael Fuentes-Pavón, José M. Santana-Montesdeoca 467

Parallel Progressive Multiple Sequence Alignment
Erik Pitzer 473

Concepts and Systems Tools for Modelling Signal Processing in Vertebrate Retina
Roberto Moreno-Díaz, Gabriel de Blasio, Arminda Moreno-Díaz 483

Application of Multichannel Vision Concepts and Mechanisms in an Artificial Industrial Vision System
A. Quesada-Arencibia, J.C. Rodríguez-Rodríguez, Roberto Moreno-Díaz Jr 492

Intelligent Vehicular Sytems

Soft Computing and Geometrical Control for Computer Aided Driving
Javier Alonso Ruiz, Teresa de Pedro, Carlos González, Ricardo García 501

A Monocular Solution to Vision-Based ACC in Road Vehicles
Miguel Ángel Sotelo, Jesús Nuevo, Manuel Ocaña, Luis Miguel Bergasa 507

Multi-objective Dynamic Optimization for Automatic Parallel Parking
Javier de Lope, Darío Maravall 513

Electric Power Steering Automation for Autonomous Driving
José E. Naranjo, Carlos González, Ricardo García, Teresa de Pedro 519

Computer Vision Application: Real Time Smart Traffic Light
Ángel Serrano, Cristina Conde, Licesio J. Rodríguez-Aragón, Raquel Montes, Enrique Cabello 525

Permanency Memories in Scene Depth Analysis
Miguel A. Fernández, José M. López-Valles, Antonio Fernández-Caballero, María T. López, José Mira, Ana E. Delgado 531

Pedestrian Detection for Intelligent Vehicles Based on Active Contour Models and Stereo Vision
C. Hilario, J.M. Collado, J. Ma Armingol, A. de la Escalera 537

Fast Road Sign Detection Using Hough Transform for Assisted Driving of Road Vehicles
Miguel Ángel García-Garrido, Miguel Ángel Sotelo, Ernesto Martín-Gorostiza 543

Robotic Soccer, Robotics and Control

Advances in Robotics
Peter Kopacek 549

Current and Future Trends and Challenges in Robot Soccer
Norman Weiss, Bernd Reusch 559

Strategy and Communication in Robotic Soccer Game
Bobumil Horák, Marek Obitko, Jan Smid, Václav Snášel 565

Rete Algorithm Applied to Robotic Soccer
Manuel Palomo, Francisco J. Martín-Mateos, José A. Alonso 571

Towards a Biomathematical Model of Intentional Autonomous Multiagent Systems
Jochen Pfalzgraf, Bernhard Mitterauer 577

A Controller Network for a Humanoid Robot
Peter Kopacek, Edmund Schierer, Markus Wuerzl 584

Programming by Integration in Robotics
José L. Fernández-Pérez, Antonio C. Domínguez-Brito, Daniel Hernández-Sosa, Jorge Cabrera-Gámez 590

A Mathematical Formalism for the Evaluation of C-Space for Redundant Robots
Roberto Therón, Vidal Moreno, Belén Curto, Francisco J. Blanco 596

Global Modal Logics for Multiagent Systems: A Logical Fibering Approach
Johann Edtmayr 602

Improved Non-standard Discretization Methods for Nonlinear Dynamical Control Systems
Jesús Rodríguez-Millán, Carla González, Anna Patete 608

Hierarchical Control of a Distributed Solar Collector Field
Manuel Berenguel, Cristina M. Cirre, Ryszard Klempous, Henryk Maciejewski, Maciej Nikodem, Jan Nikodem, Imre Rudas, Loreto Valenzuela 614

Explanatory Analysis of Data from a Distributed Solar Collector Field
Manuel Berenguel, Ryszard Klempous, Henryk Maciejewski, Jan Nikodem, Maciej Nikodem, Loreto Valenzuela 621

Author Index 627

On the Physical Formal and Semantic Frontiers Between Human Knowing and Machine Knowing

José Mira Mira

Dpto. de Inteligencia Artificial,
ETSI. Informática. UNED, Madrid, Spain
jmira@dia.uned.es

Abstract. The purpose of this paper is to reflect on the nature of human knowledge and that of the knowledge that finally can dwell in an electronic computer. Three frontiers can be distinguished between these two constitutively different types of knowing: (1) The nature of current physical machines (silicon semiconductor crystal) and its organizational restrictions in relation with the biological tissue, which is autonomous, dynamic, tolerant to failures, self-organizative, and adaptive. (2) The semantics of the available algorithms and programming languages in relation with the evolutionary and reactive (behavior-based) biological programming strategies. (3) The nature of current formal tools in relation with natural language.

1 Problem Statement

A great part of the AI community attempts to interpret and use concepts of the computational paradigm when applied to natural systems as equivalent to the corresponding concepts when applied to artificial systems. These scientists assume that natural language can be reduced to formal language, that biological programming strategies can be reduced to conventional programming languages and that the nervous tissue and a body of meat can be reduced to a CPU of Silicon crystal and an electro-mechanical robot. Unfortunately, fifty years after the christening of AI at the 1956 summer conference at Dartmouth Colleague, this is not the case. There are relevant, constitutive, differences between Computation in Natural Systems (the human way of knowing) and Computation in Artificial Systems (the machine way of knowing). To contribute to the establishment of a clear distinction between these two different ways of knowing we introduce in section two the methodological building of knowledge [11,9,10], which will enable us to distinguish between three levels and two domains of description of the knowledge involved in a calculus. In sections three, four and five, making reverse engineering, we consider the knowledge that each one of these levels can accommodate, from the physical level to the symbol level and, finally, to the knowledge level. Then we conclude, mentioning the topics in which, in our opinion, we should concentrate our efforts to move the frontiers between human intelligence and machine intelligence.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 1–8, 2005.
© Springer-Verlag Berlin Heidelberg 2005

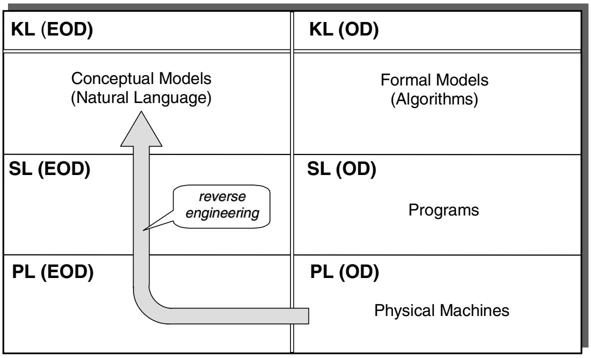


Fig. 1. Reverse engineering in the building of knowledge

2 Direct and Reverse Engineering Inside the Building of Knowledge

The three levels (storey) of the building for knowledge are the physical level where lives the machine hardware, the symbol level where the programs live and the third level, introduced by Allen Newell [12] and David Marr [4], located over the symbol level and named by Newell the knowledge level and by Marr the level of the theory of calculus. Additionally the external observer of a calculus (either in natural or in artificial systems) always can distinguish between two domains of description in each level [6,15,8]: the level's own domain, OD, and the domain of the external observer (EOD), as shown in figure 1.

In direct engineering we start with a conceptual model at the KL and in the EOD of the method used by humans to solve a problem. The architecture of this model depends on the AI paradigm used in our approach (symbolic, connectionist, situated or hybrid). Then we use a table of correspondences to move from this conceptual model to the abstract entities and relations of a formal model situated in the OD of the KL (the 3rd right apartment). Next we program the formal model and a compiler ends the work by producing the final machine language version of the program.

In this formalization process we have left out (in the third left apartments) a great part of the human knowledge used in the conceptual model. This non-computable knowledge establishes the first semantic and formal frontier between human knowing and machine knowing. Only the formal model underlying natural language words enters the computer. The rest of the knowledge always remains in the natural language of the external observer, in the EOD. Obviously the external observer injects this non-computable knowledge when he interprets the results of the calculus (left hand bottom-up pathway in figure 1).

If we now attempt to recover the conceptual model from which this calculus has emerged we have to make reverse engineering, starting on the first right apartment where the program is running in a physical machine, going through the symbol level and ending at the EOD of the KL. In this reverse pathway we will reflect on the type of knowledge that can accommodate each level, both in electronic computers and in humans. Then we propose a comparison frame to enlighten the differences (frontiers) between human knowing and machine knowing. Two considerations are common to the three levels, both in natural and in artificial systems:

1. Each level (KL, SL, PL) is a closed organization and its knowing capacity is constitutively determined by the set of entities and relations that characterize this level as a distinguishable class. That is to say, by its language [5,7].
2. The interpretation (translation) of this language in each level is structurally determined by the architecture and the language of the level below. That is to say, the available programming languages determinate the interpretation of the natural language models and the available machine language determinate the interpretation of the programs. Finally, the available materials determinate the limits of machine languages.

3 The Knowing and the Physical Level (PL)

The constituent entities of a computer hardware (logic circuits) only enable us to establish binary distinctions (0, 1) on the true or falseness of a set of logical expressions. If we consider digital delays the PL can also accommodate all the knowledge related to abstract internal states and state transitions of a finite state automaton (FSA). Finally the superimposed organization of the computer architecture provides computation with the electronic mechanisms necessary to accommodate the knowledge that can be described using the machine language. In conclusion, the PL of machines can only accommodate (OD) the formal knowledge associated with logical expressions, FSAs and machine languages. The semantic tables of these abstract entities always remain at the EOD of the PL, outside the machine.

The capacity to accommodate knowledge at the PL in biological systems also is a consequence of the constitutive entities of the nervous system (ionic channels, synaptic circuits, neurons, neural assemblies, ...) and of the neural mechanisms that evolution has superimposed at the architecture level (oscillatory and regulatory feedback loops, lateral inhibition circuits, reflex arches, adaptive connectivity and routing networks, distributed central-patterns generators, ...) and the emerging functionalities (learning, self-organization, reconfiguration after physical damage, ...) characteristic of an always unfinished architecture.

It seems now clear to us where is the first frontier between human knowing and machines knowing: In the constitutively different nature of its physical elements, mechanisms and architectures (figure 2), and, consequently, in the differences between the emergent semantics of neural networks and the limited semantics of the formal descriptions of combinational logic and FSA.

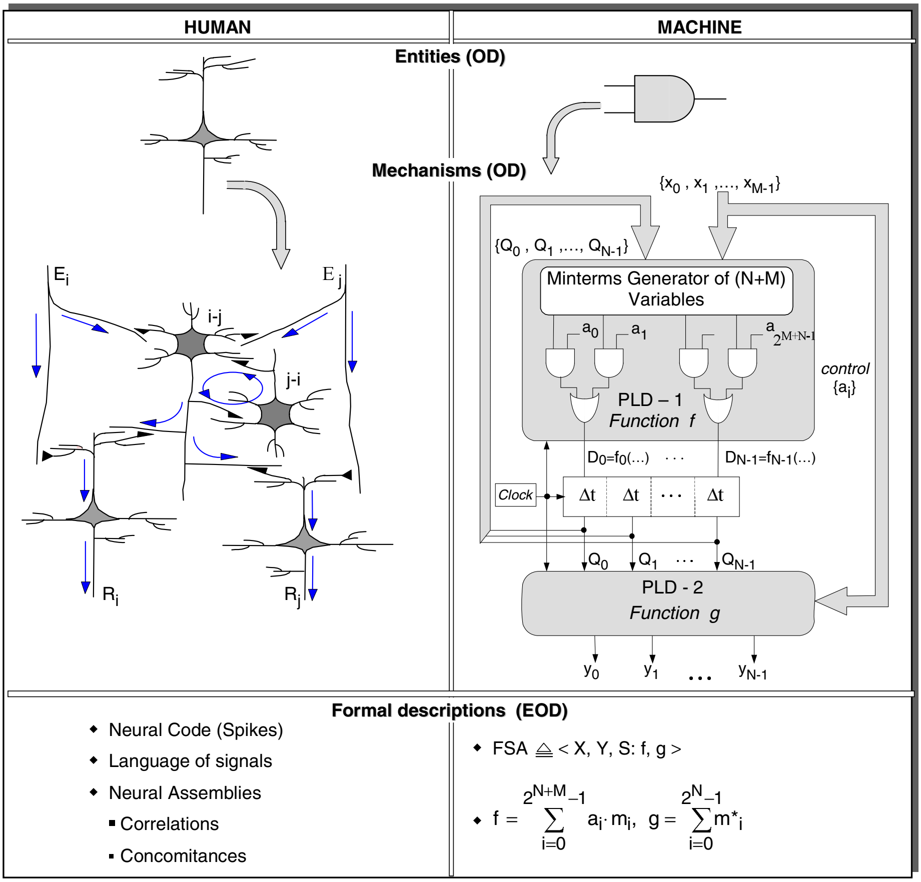
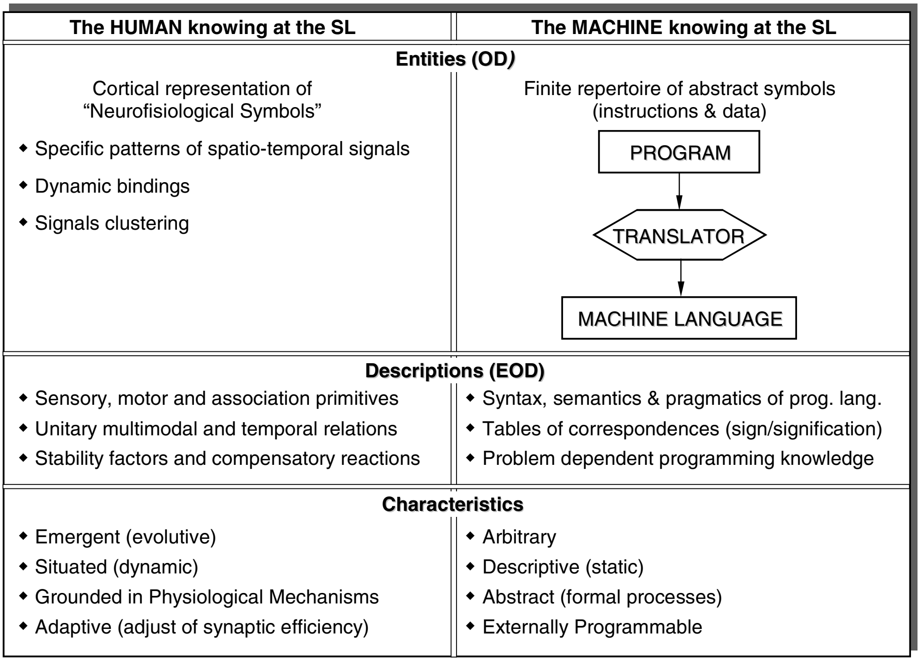


Fig. 2. The knowledge that can accommodate the physical level of human and machines depends on the entities (OD), mechanisms (OD) and formal descriptions (EOD) constitutive of this level

4 The Knowing at the Symbol Level (SL)

Let us assume that we know all the knowledge than can accommodate the circuits of the PL in humans and machines. We still do not know what the brain and the computer are calculating, until the description of the SL and the KL are completed. The knowing capacity of the SL is determined again by the set of entities and relations that characterize this level as a class (by its language). That is to say, the second part of the human knowledge involved in a calculus that finally can dwell in an electronic computer, is determined by the representational and inferential facilities and limits provided by current programming languages. The "Physical Symbol System Hypothesis" proposal of Newell and Simon [13] is representative of these limits (frozen symbols of arbitrary semantics, descriptive, abstract and externally programmable).

If we accept that evolution has accumulated enough mechanisms in the nervous system architecture as to accommodate what an external observer could describe as neurophysiological symbols and programs, then we can compare again the human knowing versus the machine knowing at the symbol level (figure 3). These symbols are connectionist and grounded in the neural mechanisms that generate and recognize them. Also, these symbols and the corresponding programs, are not programmable in the conventional sense, but via dynamic adjustment of a huge number of synaptic contacts. As a consequence, symbols in natural systems are emergent, situated, grounded and adaptive. Per contra, in artificial systems symbols are arbitrary, representational, static and externally programmable. These constitutively different characteristics establish the second semantic and formal frontier between human and machines knowing.



The HUMAN knowing at the SL	The MACHINE knowing at the SL
Entities (OD)	
Cortical representation of “Neurofisiological Symbols” ◆ Specific patterns of spatio-temporal signals ◆ Dynamic bindings ◆ Signals clustering	Finite repertoire of abstract symbols (instructions & data) PROGRAM TRANSLATOR MACHINE LANGUAGE
Descriptions (EOD)	
◆ Sensory, motor and association primitives ◆ Unitary multimodal and temporal relations ◆ Stability factors and compensatory reactions	◆ Syntax, semantics & pragmatics of prog. lang. ◆ Tables of correspondences (sign/signification) ◆ Problem dependent programming knowledge
Characteristics	
◆ Emergent (evolutive) ◆ Situated (dynamic) ◆ Grounded in Physiological Mechanisms ◆ Adaptive (adjust of synaptic efficiency)	◆ Arbitrary ◆ Descriptive (static) ◆ Abstract (formal processes) ◆ Externally Programmable

Fig. 3. The knowledge that can accommodate the SL in humans and machines

Some bio-inspired programming strategies (genetic algorithms, evolutionary and genetic programming) seems promising but are again for away from biology. Other approaches, such as computational ethology, collective emergent calculus (ant colonies, bees, ...) and the reactive (situated) approach to programming [3,2] are also trying to move the frontier between humans and machines at the SL.

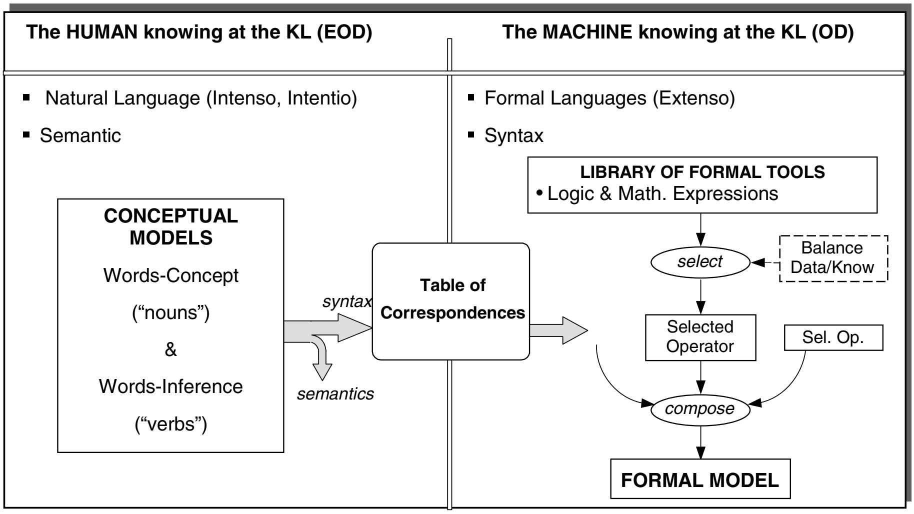


Fig. 4. Semantic frontier between natural and formal languages at the knowledge level. (Adapted from [10]).

5 The Knowing at the Knowledge Level (KL)

Let us finally climb upstairs from the SL (second floor) to the KL, the third floor of the building in which we have distributed the knowledge involved in a calculus. What are the differences between the knowledge that can accommodate at this level humans and machines?. Or, what is equivalent, what are the constitutive differences between the entities and relations of cognition and those of the conceptual and formal models currently used in AI and Knowledge Engineering? [14]. We do not know in deep the architecture of cognition but it is usually accepted that its constitutive entities are what we call perceptions, goals, purposes, intentions, ideas, plans, motivations, emotions, attitudes, actions, and so on. A major part of what we know about cognition has emerged through the observer's natural language. Consequently, we can consider that the accessible part of the architecture of cognition coincides with the architecture of natural language. Then the human knowing at the KL is the knowledge that can accommodate the natural language of the external observer. And here is the main frontier between humans and machines at the KL: Computing machines can only understand formal languages (the formal components underlying natural language models). In figure 4 we show a summary of the distinctive characteristics between these two types of languages as well as the current procedure in AI to reduce natural language to formal descriptions based on logic and mathematics. A table of correspondences between words and abstract entities is always necessary to store the meanings that are non-computable. These meanings are recovered when interpreting the results of the calculus.

6 Conclusions

The purpose of AI is to make human knowledge computable using conceptual models, formal tools, programming languages and electronic machines. Unfortunately, after fifty years and different approaches (cybernetics, heuristics, symbolic or representational stage, connectionistic again, situated and hybrid) this dream has not been yet fulfilled in a satisfactory manner. Two of the possible causes of this failure could be (1) the excess of initial optimism in assuming that cognition can be reduced to computation and (2) the lack of distinction between the constituent elements of humans and machines.

In this paper we have used the methodological building of levels and domains of description of a calculus to establish the constitutive differences at three levels (materials, symbols and languages) and we hope that the acceptance of these differences could help to delimit the real scope of AI and to focus where the real problems are: (i) Development of new materials and new computing architectures. (ii) Development of more powerful programming languages and algorithms and (iii) Development of new modeling tools and formal languages semantically closer to natural language, and still compilable.

Let us assume that these three objectives has been fulfilled. Would we say that human knowing has been then reduced to machine knowing?. Clearly, not. If the brain reasons as a logical machine the signals and symbols that it employs in its reasoning, must constitute a formal language [1], but it is not clear to us neither that cognition could be represented by using only formal languages, nor that the computational paradigm could be the only way to tackle living systems. What about cognition without computation?. Let us see what happens in the next fifty year of Neuroscience and AI.

References

1. J. Bronowski. The logic of the mind. *American Scientist*, 54, 1:1–14, 1966.
2. R.A. Brooks. Intelligence without reason. A.I. Memo 1293, MIT, 1991.
3. W.J. Clancey. *Situated cognition. On human knowledge and computer representation.* Univ. Press, Cambridge, 1997.
4. D. Marr. *Vision.* Freeman, New York, 1982.
5. H. Maturana. Ontology of observing. the biological foundations of self consciousness and the physical domain existence. http://www.inteco.cl/biology/ontology/, 2002.
6. H.R. Maturana. The organization of the living: A theory of the living organization. *Int. J. Man-Machine Studies*, 7:313–332, 1975.
7. J. Mira. Reverse neurophysiology: The embodiments of mind revisited. In R. Moreno-Díaz and J. Mira-Mira, editors, *Brain Processes, Theories and Models*, pages 37–49. The MIT Press, Massachusetts, 1995.
8. J. Mira and A.E. Delgado. Some comments on the antropocentric viewpoint in the neurocybernetic methodology. In *Proc of the Seventh International Congress of Cybernetics and Systems*, pages 891–95, 1987.

9. J. Mira and A.E. Delgado. Where is knowledge in robotics? some methodological issues on symbolic and connectionist perspectives of AI. In Ch. Zhou, D. Maravall, and Da Rua, editors, *Autonomous robotic systems*, page 334. Physical-Verlag. Springer, Berlin, 2003.
10. J. Mira and A.E. Delgado. From modeling with words to computing with numbers. pages 7–13, Budapest, 2004. Proceedings ISDA 2004.
11. J. Mira, A.E. Delgado, J.G. Boticario, and F.J. Díez. *Aspectos básicos de la inteligencia artificial.* Sanz y Torres, SL, Madrid, 1995.
12. A. Newell. The knowledge level. *AI Magazine*, 120, 1981.
13. A. Newell and H.A. Simon. Computer science as empirical inquiry: Symbols and search. *Communications of ACM*, 19:113–126, 1976.
14. G. Schreiber, H. Akkermans, and R. de Anjo Anjewierden. *Engineering and Managing Knowledge: The CommonKADS Methodology.* The MIT Press, Cambridge, Mass, 1999.
15. F.J. Varela. *Principles of Biological Autonomy.* The North Holland Series in General Systems Research, New York, 1979.

Approximation Problems Categories*

Liara Aparecida dos Santos Leal[1], Dalcidio Moraes Claudio[1],
Laira Vieira Toscani[2], and Paulo Blauth Menezes[2]

[1] Mathematics Department, PUCRS – Porto Alegre, Brazil
{liara, dalcidio}@pucrs.br
http://www.mat.pucrs.br/
[2] Computing Institute, UFRGS – Porto Alegre, Brazil
{blauth, laira}@inf.ufrgs.br
http://www.inf.ufrgs.br/

Abstract. In this paper we continue along the same line of research started in earlier works, towards to providing a categorical view of structural complexity to optimization problems. The main aim is to provide a universal language for supporting formalisms to specify the hierarchy approximation system for an abstract NP-hard optimization problem. Categorical shape theory provides the mathematical framework to deal with approximation, enabling comparison of objects of interest and of models. In this context, tractable optimization problems are considered as a class of "models" or "prototypes" within a larger class of objects of interest - the intractable optimization problems class. Standard categorial constructions like universal objects, functors and adjunctions allow to formalize an approximation hierarchy system to optimization problems, besides characterizing NP-hard optimization problems as concrete universal objects.

1 Introduction

The notion of approximation problems was formally introduced by Johnson [4] in his pioneering paper on the approximation of combinatorial optimization problems, and it was also suggested a possible classification of optimization problems on grounds of their approximability properties. Since then, it was clear that, even though the decision versions of most NP-hard optimization problems are polynomial-time reducible to each other, they do not share the same approximability properties. In spite of some remarkable attempts, according to Ausiello [1] the reasons that a problem is approximable or nonapproximable are still unknown. The different behaviour of NP-hard optimization problems with respect to their approximability properties is captured by means of the definition of approximation classes and, under the "P $\neq$ NP" conjecture, these classes form a strict hierarchy whose levels correspond to different degrees of approximation.

In this paper we continue along the same line of research started in [7], towards to providing a categorical view of structural complexity to optimization

---

* This work is partially supported by FAPERGS and CNPq.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 9–14, 2005.
© Springer-Verlag Berlin Heidelberg 2005

problems. The main aim is to provide a universal language for supporting formalisms to specify the hierarchy approximation system for an abstract NP-hard optimization problem, in a general sense. From the observation that, intuitively, there are many connections among categorical concepts and structural complexity notions, we started defining two categories: the OPTS category of polynomial time soluble optimization problems, which morphisms are reductions, and the OPT category of optimization problems, having approximation-preserving reductions as morphisms. The study of approximation implies to create means of comparing optimization problems. The basic idea of approximation by models is a recurrent one in mathematics and in this direction a comparison mechanism between the OPTS and OPT categories has been introduced in [8]. In order to establish a formal ground for the study of the approximation properties of optimization problems, a system approximation to each optimization problem is constructed, based on categorical shape theory [3]. In so doing, we were very much inspired in previous works by Rattray [12,13] on complex systems.

Given a functor K: OPTS $\longrightarrow$ OPT, the category $APX_{B,K}$ of approximations to an optimization problem $B \in$ OPT is the comma category $B \downarrow K$ of K-objects under B. A such kind of limit construction provides a means of forming complex objects from patterns (diagrams) of simpler objects. In particular, by using co-limits in the $APX_{B,K}$ definition, a hierarchical structure can be imposed upon the system of approximation, reaching the best approximation from the system, if it exists. Besides, optimization problems B and B' can be compared by their approximation $B \downarrow K$ and $B' \downarrow K$ more easily. In addition, if K has an adjoint then each $B \downarrow K$ has an initial object, i.e., a best approximation to B. The advantage of initiality conditions is that they imply that each $B \downarrow K$ can be handled as if it were a directed set. Thus the existence of an initial object means that given any two approximation, one can find a mutual refinement of them.

In a sequel of this paper, we have planned to extend the investigation in order to characterize optimization problems in terms of their hardness in being approximated, also exploiting farther on the class of NPO problems.

2 Definitions and Known Results

In this section the results of earlier papers on the subject are summarized with some notational improvements. It is supposed that the basic concepts as of computational complexity theory well as of category theory are well known. The main refereed books are [1,2,6,11]. Next, in analogy to P and NP complexity classes, NPO stands to the class of nondeterministic polynomial time optimization problems, and PO stands to the deterministic one.

2.1 Optimization Problems Categories

The notion of *reductibility* provides the key-concept to this approach. In this context, reductions between optimization problems are considered as morphisms in the categorial sense, being optimization and approximation problems viewed as the obvious objects.

Definition 1 (OPTS Category). *The polynomial time soluble optimization problems category OPTS has PO optimization problems as objects and reductions between optimizations problems as morphisms.*

Definition 2 (OPT Category). *The optimization problems category OPT has NPO optimization problems as objects and approximation-preserving reductions as morphisms.*

NP-hard problems were proved concrete universals objects to OPT category, according to the Theory of Universals [5], confirming that a hard problem captures the essential properties of its class.

After defining both the OPTS and OPT categories in [9], the next step was to identify the relationships between them. In [10] were proposed two basic questions: What does it mean to say that a problem A "approximates" an optimization problem B? What is it understood by the "best approximation" for such an optimization problem?

In order to answer those questions, were provided mechanisms for the comparison between such categories. This led us to the categorical shape theory.

2.2 Categorical Shape Theory Revisited

The first categorical approach to shape theory arose in the beginning seventies. Since then many other works related to the subject have appeared. As was pointed by Cordier and Porter in their introduction to [3], shape theory describes a process which is common in mathematical reasoning. Typically one has a class of objects in which one has a reasonably complete set of information. This class is considered as a class of "models" or "prototypes" within a larger class of objects of interest.

In the context of categorical shape theory, there are three basic defining elements:

1. a category **B** of objects of interest;
2. a category **A** of prototypes or model-objects;
3. a "comparison" of objects with model-objects, ie. a functor $K : \mathbf{A} \longrightarrow \mathbf{B}$.

If B is an object of **B**, one can form the comma category $B \downarrow K$ whose objects are pairs (f, A) with $f : B \longrightarrow KA$. A morphism from (f, A) to (g, A') is a morphism $a : A \longrightarrow A'$ such that $K(a) \circ f = g$. If $h : B \longrightarrow B'$ is a morphism in **B**, there is an induced functor $h^* : B \downarrow K \longrightarrow B' \downarrow K$ obtained by composition in an obvious way. This functor preserves the codomain $h^*(f, A) = (fh, A)$.

A *shape category* is defined introducing new morphisms preserving codomain between objects in **B**. The basic idea behind categorical shape theory is that recognizing and understanding an object of interest B via a comparison $K : \mathbf{A} \longrightarrow \mathbf{B}$ requires the identification of the corresponding prototype A which best represents B. Besides, in any approximating situation, the approximations are what encode the only information that it can analyze.

3 Category of Approximations

Through categorical shape theory and under a few many conditions it is possible to identify the best approximation to an optimization problem B in OPT category, if it exists. The notion of "most closely approximates" is given by a *universal* object. Besides, it is provided the way of comparing NP-hard problems whose are approximated by the same design technique.

Let OPT be the category of objects of interest **B** and OPTS the category of prototypes **A**. We must have some way of comparing hard problems with tractable problems. The fundamental techniques for the design of approximation algorithms are presented in [1]. In many cases it is possible to define an algorithm scheme that can be applied to obtain several algorithms for the same problem with possibly different approximation properties. The most used such design techniques are: relaxation method, greed method, local search, linear programming based algorithm, dynamic programming and randomized algorithm.

Let K: OPTS$\longrightarrow$ OPT be a comparison mechanism related to an approximation method (for instance by using relaxation method). In order to characterize approximation degrees by means of categorical shape theory, the basic idea is the construction of a system approximation to each optimization problem, using the notion of co-limit. In a general case, approximations with their morphisms form a category $B \downarrow K$, the comma category of K-objects under B.

Definition 3 (Approximation Problem). *Given a functor K:OPTS$\longrightarrow$OPT, a problem B $\in$OPT is said an approximation problem if there are a problem A $\in$OPTS and an approximation-preserving reduction f, such that $f : B \longrightarrow KA$.*

In this case, the pair (f, A) is an *approximation* to the problem B. Notice that as a particular K may apply distinct problems from OPTS to the same problem in OPT, it is better to represent such an approximation as a pair (f, A).

Definition 4 (Category of Approximations).
Given a functor K:OPTS$\longrightarrow$OPT, the category $APX_{B,K}$ of approximations to an optimization problem $B \in$ OPT is the comma category $B \downarrow K$ of K-objects under B.

The definition of a morphism $h : (f, A) \longrightarrow (g, A')$ between approximations corresponds to saying that $g : B \longrightarrow KA'$ can be written as a composite $K(a)^\circ f$, where $f : B \longrightarrow KA$ and $a : A \longrightarrow A'$, that is

$$B \rightarrow KA \rightarrow KA'$$

This it means that (f, A) already contains the information encoded in (g, A'). Thus in some way (f, A) is "finer" approximation to B than is (g, A'). The cone-like form of the morphisms in **B** giving the approximations for some problem B, suggests that the best approximation to such problem B, if it exists, is given by a limit object in $APX_{B,K}$. In this case, a hierarchical structure can be imposed upon the system of approximation by using a kind of universal construction in the category of approximations.

3.1 Comparison of NP-Hard Problems

Very often we are faced to comparing two problems related to the approximation issue. Supposing that it is given a comparison functor K:OPTS⟶OPT, and a problem-object B in OPT, the category of approximations $APX_{B,K}$ encodes the only information available on B, by using an approximating-object (f, A). Therefore, if we would compare two problems B and B' in OPT, we should compare the corresponding categories of approximations $APX_{B,K}$ and $APX_{B\ ,K}$.

In this case a morphism preserving codomain from B to B' induces a functor(shape morphism) that compares the information encoded in their corresponding categories of approximations.

The meaning of this categorial construction has to be investigated in more detail and it is in order for further work.

3.2 Approximation Scheme

In the context of complexity theory, the existence of an algorithm scheme to a problem means that there is a best approximation to such problem.

Consider the comparison functor K:OPTS⟶OPT resulting of an algorithm scheme. In the categorical approach, this it means that a problem-object B in OPT has a K-universal prototype A in OPTS. Therefore there is an adjoint functor to K. This fact implies that the corresponding category of approximations $APX_{B,K}$ has an initial subcategory consisting of a single morphism and a single object. Thus such a category can be handled as if it were a directed set. The meaning of this result is of great theoretical significance: it implies that given any two approximations to the problem B, one can find a finer approximation than both of them.

4 Conclusions

Approximation of optimization problems has become a very active area of research. Nowadays it is known that the computational efficiency of approximating different NP-hard optimization problems varies a great deal. It is normal in computational theory to regard a problem as “tractable” if we know of an algorithm that takes time that is bounded above by some polynomial of the size of the problem instance. Unfortunately, for many problems there are complexity theoretic evidence to suggest strongly that they are, in fact, “intractable”. In order to define the structure of problems better, much effort has been turned to classifying computational problems according to how hard they are to solve. However, computational problems are not only things that have to be solved. They are also objects that can be worth studying problems and can be formalized mathematically.

In this paper we extend our previous work on the application of categorical shape theory in order to provide a mathematical framework in dealing with the question outlined above. Our knowledge is by no means complete however, and there remain many open problems. The direction is aimed towards actually exploring the connections among the structural complexity aspects and categorical concepts, which may be viewed in a "high-level", in the sense of a structural complexity approach.

References

1. G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela and M. Protasi, " Complexity and Approximation – Combinatorial Optimization Problems and their Approximability Properties", Springer-Verlag, 1999.
2. M. Barr and C. Wells "Category Theory for Computing Science", Prentice Hall, New York, 1990.
3. J-M. Cordier and T. Porter. Shape Theory: Categorical Approximations Methods. Ellis Horwood Ltd, 1990.
4. D. S. Johnson, "Approximation Algorithms for Combinatorial Problems", J. Comput. System Sci. 9, 1974. p.256-278.
5. D. P. Ellermann. Category Theory and Concrete Universals. ERKENNTNIS, 28. No.3,1988. p.409-429.
6. M. R. Garey and D. S. Johnson, "Computers and Intractability - A guide to the Theory of NP-Completeness", Bell Laboratories Murray Hill, New Jersey, 1979.
7. L. A. S. Leal; P. B. Menezes; D. M. Claudio and L. V. Toscani. Optimization Problems Categories. EUROCAST'01, LNCS 2178. Springer-Verlag, R.Moreno-Diáz, B. Buchberger and J-L. Freire Eds., 2001. p.285 – 299.
8. L. A. S. Leal. Uma fundamentação Teórica para a Complexidade Estrutural de Problemas de Otimização. (PhD Thesis), Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, 2002. 115p.
9. L. A. S. Leal; D. M. Claudio; P. B. Menezes and L. V. Toscani. Modelling the Approximation Hierarchy to Optimisation Problems Through Category Theory. International Journal of Computing Anticipatory Systems, v. 11, Belgium, 2002. p.336 – 349.
10. L. A. S. Leal; D. M. Claudio; L. V. Toscani and P. B. Menezes. A Categorical Approach to NP-Hard Optimization Problems. EUROCAST'03, LNCS 2809. Springer-Verlag, R.Moreno-Diáz and F. Pichler Eds., 2003. p.62 – 73.
11. C. H. Papadimitriou, "Computational Complexity", Addison-Wesley Publishing Company, 1994.
12. C. Rattray. Identification and Recognition Through Shape in Complex Systems. LNCS 1030, Springer-Verlag, 1995. p.19-29.
13. C. Rattray. Abstract Modelling Complex Systems. Advances in Computing Science, Systems: Theory and Practice. R. Albrecht, Ed., 1998. pp. 1-12.

Computation of Partial Automata Through Span Composition*

Karina Girardi Roggia, Marnes Augusto Hoff, and Paulo Blauth Menezes

Instituto de Informática - UFRGS,
Av. Bento Gonçalves, 9500, Campus do Vale, Bloco IV,
Porto Alegre, RS, Brazil – CEP 91501-970
{kaqui, marnes, blauth}@inf.ufrgs.br

Abstract. In this paper a way to have structures with partiality in its internal structure in a categorical approach is presented and, with this, a category of partial graphs $\mathcal{G}r_p$ is given and partial automata are constructed from $\mathcal{G}r_p$. With a simple categorical operation, computations of partial automata are given and can be seen as a part of the structure of partial automata.

Keyworks: computation, partial automata, Category Theory.

1 Introduction

In Computer Science, to express non-terminanting computations and to define partial recursive functions, it is common to use the notion of partiality. Actually, due to partiality, p.g., the class of partial recursive functions becomes equivalent to Turing Machines. Category Theory arrises as an useful tool to formalize abstract concepts making easy to construct proofs and investigate properties in many areas, specially in Semantics and Type Theory. The constructions about universal mappings like limits and adjunctions are getting useful interpretations in terms of compositionality of systems. Besides, in Category Theory there are tools to define structures more complex based in simple ones like Comma Categories, that allows to define a category based in another and to inherit properties. Categories of Graphs and Automata are usually defined by this structure.

Graphs are commonly used to model systems, either by simple graphs or by graph-based structures like Petri nets [1,2,3] and automata [4,5]. Automata is a common formalism used in many areas in Computer Science like compilers, microelectronics, etc. Most of the study about it is in the Formal Language area.

In this paper we define a different category of automata: a category of Partial Automata, named $\mathcal{A}ut_p$. This category is constructed over a category of partial graphs ($\mathcal{G}r_p$). The difference between a graph and a partial graph is in the definitions of the source and target functions that mapped an arc to a node: in graphs these functions are total while in partial graphs source and target functions are partial functions. Due to this difference, automata based in partial

---

* This work is partially supported by CAPES, CNPq and FAPERGS.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 15–20, 2005.
© Springer-Verlag Berlin Heidelberg 2005

graphs can have marks of initial and final states naturally. Beyond that, we can also define constructions like limits in $\mathcal{A}ut_p$ that allows composition of partial automata.

In [6], span Composition [7] is used to compose graphs given semantics of systems with dynamic topology, p.g.. This kind of composition could be used to define the computations of (partial) automata. Briefly, a span composition of two partial automata results in a partial automata where each edge represents a path of length up to two (between nodes), which first half is some edge of the first automaton and which second half is some edge of the second one. It is possible to compose the same automaton with itself several times which is the purpose of this paper. In the case of n successive span compositions, we can obtain all the words of its accepted language whose needs $n+1$ steps of computation in the arcs of the partial automaton that don't have source neither target.

2 Partial Graphs

To define partial automata, we'll first construct a way to define structures with partiality in its internal structure: this is done with the definition of *pComma* – a special kind of comma-category [8]. Then a category of partial graphs is defined. A partial graph is a directed graph where the functions source and target of arcs are **partial** functions. The definition of *pComma*, uses the notion of category of partial morphisms – named *pC* – defined in [9].

Definition 1 (*pComma*). *Consider the finitely complete category* $\mathcal{C}$ *and the functors* $\mathbf{inc_p} : \mathcal{C} \to p\mathcal{C}$ *(the canonical inclusion functor),* $\mathbf{f} : \mathcal{F} \to \mathcal{C}$ *and* $\mathbf{g} : \mathcal{G} \to \mathcal{C}$*. Therefore,* $p\mathcal{C}omma(\mathbf{f}, \mathbf{g})$ *is such that the objects are triples* $S = \langle F, s, G\rangle$*, where* F *is a* $\mathcal{F}$*-object,* G *is a* $\mathcal{G}$*-object and* $s : \mathbf{inc_p} \circ \mathbf{f}F \to \mathbf{inc_p} \circ \mathbf{g}G$ *is a* $p\mathcal{C}$*-morphism; a morphism* $h : S_1 \to S_2$ *where* $S_1 = \langle F_1, s_1, G_1\rangle$*,* $S_2 = \langle F_2, s_2, G_2\rangle$ *is a pair* $h = \langle h_F : F_1 \to F_2, h_G : G_1 \to G_2\rangle$ *where* h_F *and* h_G *are morphisms in* $\mathcal{F}$ *and* $\mathcal{G}$ *respectively, and are such that in* $p\mathcal{C}$ *(see figure 1)* $(\mathbf{inc_p} \circ \mathbf{g}h_G) \circ s_1 = s_2 \circ (\mathbf{inc_p} \circ \mathbf{f}h_F)$*; the identity morphism of an object* $S = \langle F, s, G\rangle$ *is* $\iota_S = \langle \iota_F : F \to F, \iota_G : G \to G\rangle$*; and the composition of* $u = \langle u_F, u_G\rangle : S_1 \to S_2$*,* $v = \langle v_F, v_G\rangle : S_2 \to S_3$ *is* $v \circ u = \langle v_F \circ u_F, v_G \circ u_G\rangle : S_1 \to S_3$*.*

$$\begin{tikzcd}
\mathbf{inc_p} \circ \mathbf{f}F_1 \arrow[r, "s_1"] \arrow[d, "\mathbf{inc_p} \circ \mathbf{f}h_F"'] & \mathbf{inc_p} \circ \mathbf{g}G_1 \arrow[d, "\mathbf{inc_p} \circ \mathbf{g}h_G"] \\
\mathbf{inc_p} \circ \mathbf{f}F_2 \arrow[r, "s_2"] & \mathbf{inc_p} \circ \mathbf{g}G_2
\end{tikzcd}$$

Fig. 1. Diagram of Partial Comma Category

Definition 2 (Category of Partial Graphs). *The category of partial graphs with total homomorphisms, named* $\mathcal{G}r_p$*, is the partial comma category* $p\mathcal{C}omma(\mathbf{\Delta}, \mathbf{\Delta})$ *(beeing* $\mathbf{\Delta} : \mathcal{S}et \to \mathcal{S}et^2$ *the diagonal functor).*

Thus, a partial graph is $\langle V, T, \partial_0, \partial_1\rangle$ respectively set of nodes, set of arcs and source and target *partial* functions. Seeing a graph as a system, arcs with target function defined only can be seen as entry-points, arcs with source function defided only as end-points. And arcs with neither defined can be seen as transactions. We can divide the set of arcs of a given partial graph respecting the type of the arc.

Definition 3 (Division of T). *Let $G = \langle V, T, \partial_0, \partial_1\rangle$ a partial graph, $\varnothing : T \to \{*\}$ the empty partial function, $tot_T : T \to \{*\}, tot_V : V \to \{*\}$ both total functions and $\partial_0^* = tot_V \circ \partial_0$, $\partial_1^* = tot_V \circ \partial_1$. The following subobjects are given by the equalizers in **p**$\mathcal{S}et$ like in figure 2:$\langle K_0, \neg\partial_0\rangle$ equalizer of ∂_0^* and $\varnothing$ – arcs of G with source undefined; $\langle K_1, \neg\partial_1\rangle$ equalizer of ∂_1^* and $\varnothing$ – arcs of G with target undefined; $\langle E_0, ‘\partial'_0\rangle$ equalizer of ∂_0^* and tot – arcs of G with source defined; and $\langle E_1, ‘\partial'_1\rangle$ equalizer of ∂_1^* and tot – arcs of G with target defined. The pullbacks of figure 3 give the division of T in four classes, where:*

$$\begin{tikzcd}
K_0 \arrow[r, tail, "\neg\partial_0"] & T \arrow[r, shift left, "\partial_0^*"] \arrow[r, shift right, "\varnothing"'] & \{*\}
\end{tikzcd}
\qquad
\begin{tikzcd}
K_1 \arrow[r, tail, "\neg\partial_1"] & T \arrow[r, shift left, "\partial_1^*"] \arrow[r, shift right, "\varnothing"'] & \{*\}
\end{tikzcd}$$

$$\begin{tikzcd}
E_0 \arrow[r, tail, "‘\partial'_0"] & T \arrow[r, shift left, "\partial_0^*"] \arrow[r, shift right, "tot_T"'] & \{*\}
\end{tikzcd}
\qquad
\begin{tikzcd}
E_1 \arrow[r, tail, "‘\partial'_1"] & T \arrow[r, shift left, "\partial_1^*"] \arrow[r, shift right, "tot_T"'] & \{*\}
\end{tikzcd}$$

Fig. 2. Equalizers in $p\mathcal{S}et$

$\langle VV, vv\rangle$, being $vv = ‘\partial'_0 \circ vv_0 = ‘\partial'_1 \circ vv_1$, arcs with ∂_0 and ∂_1 defined; $\langle VF, vf\rangle$, being $vf = ‘\partial'_0 \circ vf_0 = \neg\partial_1 \circ vf_1$, arcs with ∂_0 defined only; $\langle FV, fv\rangle$, being $fv = \neg\partial_0 \circ fv_0 = ‘\partial'_1 \circ fv_1$, arcs with ∂_1 defined only; and $\langle FF, ff\rangle$, being $ff = \neg\partial_0 \circ ff_0 = \neg\partial_1 \circ ff_1$, arcs with ∂_0 and ∂_1 undefined.

$$\begin{tikzcd}
& T & \\
K_0 \arrow[ur, tail, "\neg\partial_0"] & pb & K_1 \arrow[ul, tail, "\neg\partial_1"'] \\
& FF \arrow[ul, tail, "ff_0"] \arrow[ur, tail, "ff_1"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
& T & \\
K_0 \arrow[ur, tail, "\neg\partial_0"] & pb & E_1 \arrow[ul, tail, "‘\partial'_1"'] \\
& FV \arrow[ul, tail, "fv_0"] \arrow[ur, tail, "fv_1"'] &
\end{tikzcd}$$

$$\begin{tikzcd}
& T & \\
E_0 \arrow[ur, tail, "‘\partial'_0"] & pb & K_1 \arrow[ul, tail, "\neg\partial_1"'] \\
& VF \arrow[ul, tail, "vf_0"] \arrow[ur, tail, "vf_1"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
& T & \\
E_0 \arrow[ur, tail, "‘\partial'_0"] & pb & E_1 \arrow[ul, tail, "‘\partial'_1"'] \\
& VV \arrow[ul, tail, "vv_0"] \arrow[ur, tail, "vv_1"'] &
\end{tikzcd}$$

Fig. 3. Division of Arcs

3 Partial Automata

The term "partial automaton" had been used before to define an algebraic structure based in the definition of automaton. One of the most frequent reference of this term is given by [10]. This kind of partial automata accepts a different type of language in comparison with the languages in [4] that are one of the subjects of this work. Despite, the term *partial automata* in this paper is different from Rutten. Here, a partial automaton is an automaton (possible non-deterministic) where transitions can occur without the assumption of any state before and/or after them. In other words: is an automaton defined from a partial graph.

To define a partial automata category of partial automata we first define two functors: $\mathbf{arcs_p}$ and $\mathbf{coprod_4}$. The forgetful functor $\mathbf{arcs_p}$ takes a partial graph to its set of arcs T in $\mathcal{S}et$ with a function $m : T \to \Omega^2$ that classifies each arc of T in its type by the definition 3, where $\Omega^2 = \{vv, vf, fv, ff\}$.

Definition 4 (Functor $\mathbf{arcs_p}$). *The functor* $\mathbf{arcs_p} : \mathcal{G}r_p \to \mathcal{S}et/\Omega^2$ *is such that, taking* $\langle V, T, \partial_0, \partial_1\rangle$ *any partial graph,* $\mathbf{arcs_p}(\langle V, T, \partial_0, \partial_1\rangle) = \langle T, m\rangle$ *where* $m : T \to \Omega^2$ *is such that (given* $t \in T$*)* $m(t) = \langle v, v\rangle$ *if* $t \in VV, m(t) = \langle v, f\rangle$ *if* $t \in VF, m(t) = \langle f, v\rangle$ *if* $t \in FV$ *or* $m(t) = \langle f, f\rangle$ *if* $t \in FF$*; and given* $h = \langle h_V, h_T\rangle : \langle V_1, T_1, \partial_0^1, \partial_1^1\rangle \to \langle V_2, T_2, \partial_0^2, \partial_1^2\rangle$ *a total homomorphism of partial graphs,* $\mathbf{arcs_p}(h) = h_T$*.*

The functor $\mathbf{coprod_4}$ does the 4-ary disjoint union of a set and associates a function where each element of a given set A goes to each element of Ω^2.

Definition 5 (Functor $\mathbf{coprod_4}$). *The functor* $\mathbf{coprod_4} : \mathcal{S}et \to \mathcal{S}et/\Omega$ *is such that, given any set* A*,* $\mathbf{coprod_4}(A) = \langle \amalg_A^4, \alpha\rangle$ *where* $\alpha : \amalg_A^4 \to \Omega^2$ *is such that (suppose* $a_i \in \amalg_A^4$ *where* $i \in \{1, 2, 3, 4\}$ *indicate the source of the element in the coproduct – first, second, third or fourth immersion)* $\alpha(a_i) = \langle v, v\rangle$ *if* $i = 1, \alpha(a_i) = \langle v, f\rangle$ *if* $i = 2, \alpha(a_i) = \langle f, v\rangle$ *if* $i = 3$ *or* $\alpha(a_i) = \langle f, f\rangle$ *if* $i = 4$*; and taking* $f : A \to B$ *a function,* $\mathbf{coprod_4}(f) = f^* : \langle \amalg_A^4, \alpha\rangle \to \langle \amalg_B^4, \beta\rangle$ *where (for* $p, q \in \{v, f\}$*)* $f^*(\langle a, \langle p, q\rangle\rangle) = \langle f(a), \langle p, q\rangle\rangle$*.*

Definition 6 (Category $\mathcal{A}ut_p$). *The category of Partial Automata, called* $\mathcal{A}ut_p$*, is the comma-category* $\mathbf{arcs_p} \downarrow \mathbf{coprod_4}$*.*

4 Computation of Partial Automata

We use an extension of span composition to define computations of partial automata in the sense used in [6], where span composition was used to compose graphs as dynamic systems. To have definitions and proprieties of span and span composition see [7,11,6].

Definition 7 (Partial Automata Composition of Transitions). *Given the partial automata* $A_1 = \langle V, T_1, \partial_0^1, \partial_1^1, \Sigma_1, etiq_1\rangle$ *and* $A_2 = \langle V, T_2, \partial_0^2, \partial_1^2, \Sigma_2, etiq_2\rangle$*. The* Binary Composition of Transitions*, or just Composition of Transitions, of* A_1 *and* A_2 *is the partial automaton* $A_1 \triangleright A_2 = \langle V, T, \partial_0, \partial_1, \Sigma, etiq\rangle$ *where* T *is the object from the pullback,* $\Sigma = \Sigma_1 \times \Sigma_2$ *and etiq is the function induced by the product in pSet illustrated in figure 4 and* $\partial_0 = \partial_0^1 \circ p_0$ *and* $\partial_1 = \partial_1^2 \circ p_1$*.*

$$\begin{tikzcd}
V & & V & & V \\
& T_1 \arrow[ul, "\partial_0^1"'] \arrow[ur, "\partial_1^1"] \arrow[dd, "etiq_1"'] & & T_2 \arrow[ul, "\partial_0^2"'] \arrow[ur, "\partial_1^2"] \arrow[dd, "etiq_2"] & \\
& & T \arrow[ul, "p_0"] \arrow[ur, "p_1"'] \arrow[dd, dashed, "etiq"'] & & \\
& \Sigma_1 & & \Sigma_2 & \\
& & \Sigma_1 \times \Sigma_2 \arrow[ul, "\pi_{\Sigma_1}"] \arrow[ur, "\pi_{\Sigma_2}"'] & &
\end{tikzcd}$$

Fig. 4. Partial Automata Composition of Transitions

Example 1 (Partial Automata and Composition of Transitions). Let the partial automaton $A = \langle \{A, B\}, \{t, u, v,\ w, x, y, z\}, \partial_0^A, \partial_1^A, \{0, 1\}, etiq_B \rangle$ from figure 5 (left) where the functions ∂_0^A, ∂_1^A and $etiq_A$ are represented in there. The resulted Composition of Transitions $A \rhd A$ is in figure 5 (right).

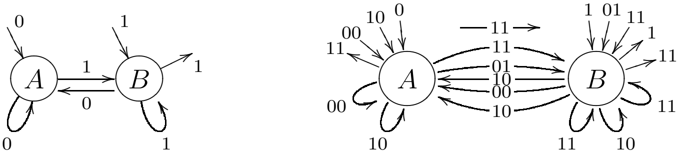


Fig. 5. Partial Automaton A (left) and its Composition of Transitions (right)

Definition 8 (Finite Computation of Partial Automata). *Given a partial automaton $A = \langle V, T, \partial_0, \partial_1, \Sigma, etiq \rangle$, the finite computations of length up to $n+1$ of A is the class of arcs FF (as in the definition 3) of the resulting automaton from the Transitions Composition with itself n times.*

Let a partial automaton that computes the language L (the automaton from example 1 computes the language $L = \{w | w \in \{0, 1\}^* \wedge w$ finishes in $11\}$). Composing itself *ad infinitum* by composition od transistions, the resulting arcs without source neither target nodes can be seen as the transitive closure L^+. If we compose itself n times, this kind of arcs will be the subset of L^+ whose word's length is limited to $n + 1$, i.e., the computations of the automaton with $n + 1$ steps.

5 Concluding Remarks

In this paper we presented a way to define computations of a different type of automata: the partial automata. One of the advantages of this kind of automaton is that initial and final actions are natural in the structure, that is constructed in a categorical approach. Generally, to construct structures like graphs and automata in a categorical way, initial and/or final states are not natural. Besides, we can computes the language of a partial automaton with a simple operation of composition seen in each resulting automaton the steps of the computation.

From this work is possible to research and to develop extensions to more complex structures like, for instance, Petri Nets; to explore partial automata that evolve (by a graph-grammar like approach) using the composition of transitions and to study proprieties of formal languages using this approach.

References

1. Peterson, J.L.: Petri Net Theory and the Modelling of Systems. Prentice-Hall, Englewoods Cliffs, New Jersey (1981)
2. Meseguer, J., Montanari, U.: Petri nets are monoids. Information and Computation **88** (1990) 105–155
3. Menezes, P.B.: Diagonal compositionality of partial petri nets. In: 2nd US-Brazil Joint Workshops on the Formal Foundations of Software Systems, Eletronic Notes in Theoretical Computer Science v.14 (1998)
4. Hopcroft, J.E.: Introduction to automata theory, languages and computation. Addison-Wesley (1979)
5. Adamek, J., Trnkova, V.: Automata and Algebras in Categories. 1 edn. Kluwer, Dordrecht (1990)
6. Hoff, M.A., Roggia, K.G., Menezes, P.B.: Composition of Transformations: A Framework for Systems with Dynamic Topology. International Journal of Computing Anticipatory Systems **14** (2004) 259–270
7. Bénabou, J.: Introduction to bicategories. In: Reports of the Midwest Category Seminar. Number 47 in Springer Lecture Notes in Mathematics. Springer-Verlag (1967) 1–77
8. Borceux, F.: Handbook of Categorical Algebra 1: Basic Category Theory. Volume 50 of Encyclopedia of Mathematics and Its Applications. Cambridge University Press, Cambridge (1994)
9. Asperti, A., Longo, G.: Categories, Types, and Structures - An Introduction to Category Theory for the Working Computer Scientist. MIT Press, Cambridge, USA (1991)
10. Rutten, J.J.M.M.: Automata and coinduction (an exercise in coalgebra). In Sangiorgi, D., de Simone, R., eds.: International Conference on Concurrency Theory, CONCUR, n.9. Volume 1466 of Lecture Notes in Computer Science., Berlin, Springer-Verlag (1998) 194–218
11. Bruni, R., Gadducci, F.: Some algebraic laws for spans. In W. Kahl, D.P., Schmidt, G., eds.: Proceedings of RelMiS 2001, Workshop on Relational Methods in Software. Electronic Notes in Theoretical Computer Science, n.44 v.3, Elsevier Science (2001)

Degenerate Arrays: A Framework for Uncertain Data Tables

Margaret Miró-Julià

Departament de Ciències, Matemàtiques i Informàtica ,
Universitat de les Illes Balears, 07122 Palma de Mallorca, Spain
`margaret.miro@uib.es`

Abstract. Boolean algebra provides an important model for the description of knowledge using a binary language. This paper considers a multivalued language, based on arrays and co-arrays, that allows a multivalued description of the knowledge contained in a data table.

This multivalued algebra has some special elements which are arryas and co-arrays at the same time: the degenerate arrays. These degenerate arrays are singled out and their interpretation analyzed, which gives rise to the introduction of the array projections and co-array projections.

Finally, a relation between array projections, co-array projections and uncertain data tables is examined.

1 Introduction

It is well known that the set of all subsets of a given set C (the power set of C), constitutes a Boolean algebra $< \rho(C), \cup, \cap, \hat{}, \emptyset, C >$. If a symbolic representation or a description of subsets is considered, there is a parallel Boolean algebra $< \mathcal{S}_c, +, \cdot, \hat{}, \vee_c, \wedge_c >$ defined on the set $\mathcal{S}_c$ of all possible symbols describing subsets of C.

The special elements of $\mathcal{S}_c$ are $\vee_c$ the symbol describing the empty set, $\vee_c \looparrowright \emptyset_C$, and $\wedge_c$ the symbol describing set C, $\wedge_c \looparrowright C$. Throughout this paper, the expression $c_i \looparrowright C_i$ may be read as "c_i is the symbol describing set C_i".

All the concepts, operations and special elements introduced above make reference to only one set of values, that is, one attribute. A data table has more than one attribute. Let's consider g sets $G, \ldots, B$ and A, the elements of each of these sets are the 1-spec-sets (one specification). A g-spec-set, $[g_k, \ldots, b_j, a_i]$, is a chain ordered description of g specifications, one from set $G, \ldots$, one from set B and one from set A. Each spec-set represents itself and all possible permutations.

The cross product $G \otimes \cdots \otimes B \otimes A$ is the set of all possible g-spec-sets formed by one element of $G, \ldots$, one element of B and one element of A. The set of all possible g-spec-sets induced by sets $G, \ldots, B$ and A is called the universe and every subset of the universe is called a subuniverse.

It is important to mention that the cross product is not the cartesian product. A g-spec-set represents itself and all possible permutations whereas the elements of the cartesian product are different if the order in which they are written varies.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 21–26, 2005.
© Springer-Verlag Berlin Heidelberg 2005

Recently, a non binary algebra that allows the treatment of multiple valued data tables with systematic algebraic techniques has been introduced [1]. The basic elements of this algebra are the arrays and co-arrays. An array is a description of those subuniverses (subsets of g-spec-sets) that can be written as a cross product. A co-array is a description of those subuniverses (subsets of g-spec-sets) whose complement (respect to the universe) can be written as a cross product.

Definition 1. *Given sets* $G, \ldots, B, A$, *let* $G_i \subseteq G, \ldots, B_i \subseteq B, A_i \subseteq A$, *an array* $|t_i| = |g_i, \ldots, b_i, a_i|$ *is the symbolic representation of the cross product* $G_i \otimes \ldots \otimes B_i \otimes A_i$ *where* $g_i \looparrowright G_i, \ldots, b_i \looparrowright B_i$ *and* $a_i \looparrowright A_i$.

$$|t_i| = |g_i, \ldots, b_i, a_i| \looparrowright G_i \otimes \cdots \otimes B_i \otimes A_i$$

Definition 2. *Given sets* $G, \ldots, B, A$, *let* $G_p \subseteq G, \ldots, B_p \subseteq B, A_p \subseteq A$, *the symbolic representation of the complement (in the universe) of the cross product of subsets* $\hat{G}_p \otimes \ldots \otimes \hat{B}_p \otimes \hat{A}_p$ *where* $g_p \looparrowright G_p, \ldots, b_p \looparrowright B_p$ *and* $a_p \looparrowright A_p$ *is called a co-array.*

$$||t_p|| = ||g_p, \ldots, b_p, a_p|| \looparrowright \sim (\hat{G}_p \otimes \cdots \otimes \hat{B}_p \otimes \hat{A}_p)$$

Arrays and co-arrays are symbolic representations of subuniverses, 2-dimensional (two attributes) arrays and co-arrays can be represented graphically as shown in Fig. 1.

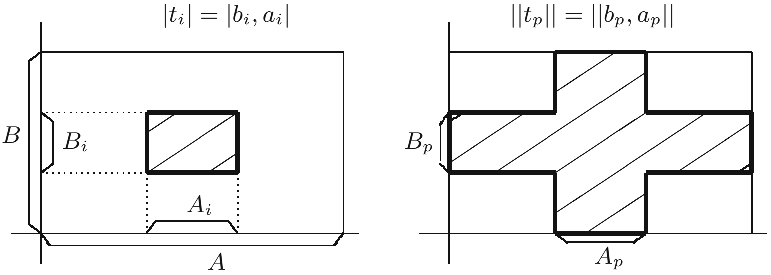


Fig. 1. Arrays and co-arrays in 2 dimensions

2 Degenerate Arrays

This array algebra has some special elements which are arrays and co-arrays at the same time.

Definition 3. *Given sets* $G, \ldots, B, A$, *if* $|t_i| = \|t_i\|$ *then* $|t_i|$ *is called a degenerate array or a degenerate co-array.*

The degenerate arrays were introduced in [2]. There are three types of degenerate arrays:

- The identity array $\bigwedge$, which describes the universe:

$$\bigwedge = |\wedge_g, \ldots, \wedge_b, \wedge_a| = ||\wedge_g, \ldots, \wedge_b, \wedge_a|| \looparrowright U = G \otimes \cdots \otimes B \otimes A$$

- The zero array $\bigvee$, describing the empty universe:

$$\bigvee = |\vee_g, \ldots, \vee_b, \vee_a| = ||\vee_g, \ldots, \vee_b, \vee_a|| \looparrowright \emptyset_U = \emptyset_G \otimes \cdots \otimes \emptyset_B \otimes \emptyset_A$$

- Arrays of the form:

$$|t_i| = |\wedge_g, \ldots, b_i, \wedge_a| = ||\vee_g, \ldots, b_i, \vee_a||$$

The 2-dimensional degenerate arrays can be represented graphically as can be seen in Fig. 2. The identity array describes the universe, whereas the zero array describes the empty universe. The third type of degenerate arrays describes subuniverses with only one distinguising attribute.

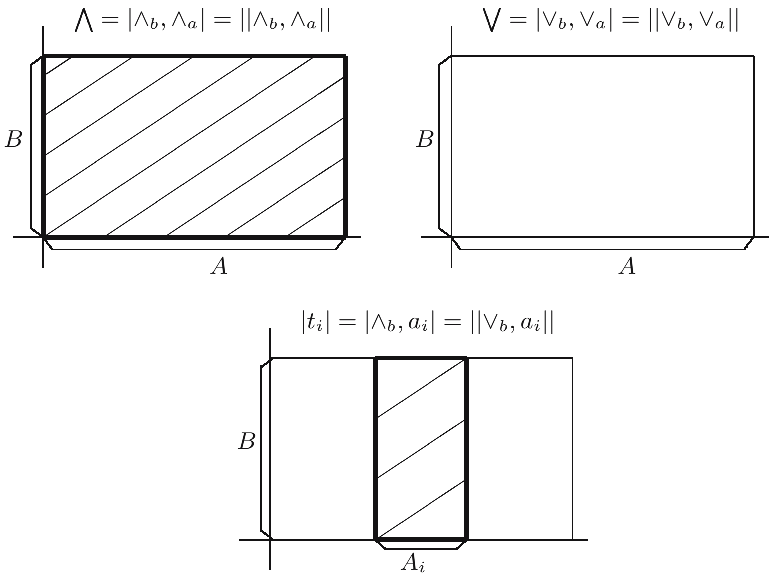

$\bigwedge = |\wedge_b, \wedge_a| = ||\wedge_b, \wedge_a||$
B
A
$\bigvee = |\vee_b, \vee_a| = ||\vee_b, \vee_a||$
B
A
$|t_i| = |\wedge_b, a_i| = ||\vee_b, a_i||$
B
A_i

Fig. 2. Degenerate arrays in 2 dimensions

3 Array Projections and Co-array Projections

Even though the cross product is not the cartesian product it inherits some of its properties. It is well known that the cartesian product of any set by the empty set is the empty set. In array algebra this can be stated as follows: any array with a $\vee$ component is equal to $\bigvee$. The dual statement maintains that any co-array with a $\wedge$ component is equal to $\bigwedge$. These statements have been proved and used throughout the development of the array algebra, however they give rise to some interesting questions.

1. Arrays with a $\vee$ component are equal to $\bigvee$, in other words, just because there is a missing piece of information, we have no information at all. Furthermore,

$$|g_i, \ldots, b_i, \vee_a| = |g_i, \ldots, \vee_b, a_i| = \cdots = |\vee_g, \ldots, b_i, a_i|$$

2. Co-arrays with a $\wedge$ component are equal to $\bigwedge$, that is, just because all values of some attributes appear, we have complete information. Furthermore,

$$||g_i, \ldots, b_i, \wedge_a|| = ||g_i, \ldots, \wedge_b, a_i|| = \cdots = ||\wedge_g, \ldots, b_i, a_i||$$

The first question was studied in [3] and the array projections were introduced.

Definition 4. *Given an array $|t_i| = |g_i, \ldots, b_i, a_i|$, a first order array projection, $|P^1|$, is an array with one $\vee$ component and $(g-1)$ non-zero components, a second order array projection, $|P^2|$, is an array with two $\vee$ components and $(g-2)$ non-zero components, a nth order array projection ($n < g$), $|P^n|$, is an array with n $\vee$ components and $(g-n)$ non-zero components.*

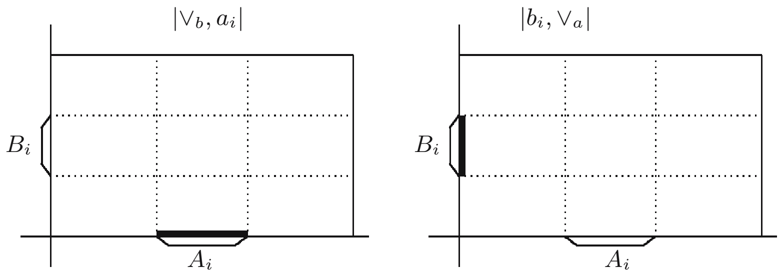


Fig. 3. 2-dimensional array projections

The array projections are descriptions of a reality with missing values for some of the attributes. An nth order array projection, $|P^n|$, is a description with no attribute values for n of the g attributes. The array projections are descriptions of incomplete data tables, some attributes do not take any of the attribute values.

Given a 2-dimensional array $|t_i| = |b_i, a_i|$, the first order array projections describe the following:

$$|P_a^1| = |b_i, \vee_a| \looparrowright B_i \otimes \emptyset_A$$
$$|P_b^1| = |\vee_b, a_i| \looparrowright \emptyset_B \otimes A_i$$

If two dimensions (attributes) are considered, an array $|b_i, a_i|$ can be graphically represented by a rectangle $b_i \times a_i$. When one of the sides becomes zero, then the rectangle has zero area. In this sense $|b_i, \vee_a| = |\vee_b, a_i|$. But even though one of the sides is zero, the other is not. The array $|b_i, \vee_a|$ becomes a line of size b_i, whereas the array $|\vee_b, a_i|$ becomes a line of size a_i. Therefore there is a difference. These lines are the array projections.

These array projections are shown on Fig. 3.

Let's address the second question, by studying co-arrays with a $\wedge$ component.

Definition 5. *Given a co-array* $||t_p|| = ||g_p, \ldots, b_p, a_p||$*, a first order co-array projection,* $||P^1||$*, is a co-array with one* $\wedge$ *component and* $(g-1)$ *non-identity components, a second order co-array projection,* $||P^2||$*, is a co-array with two* $\wedge$ *components and* $(g-2)$ *non-identity components, a* n*th order co-array projection* $(n < g)$*,* $||P^n||$*, is a co-array with* n $\wedge$ *components and* $(g-n)$ *non-identity components.*

The co-array projections are descriptions of a reality with non distinguishing values. An nth order co-array projection, $||P^n||$, is a description with all attribute values for n of the g attributes. The co-array projections are descriptions of ambiguous data tables, where some of the attributes take all possible attribute values.

Given a 2-dimensional co-array $||t_p|| = ||b_p, a_p||$, the first order co-array projections describe the following:

$$||P^1_a|| = ||b_i, \wedge_a|| \looparrowright\!\sim (\hat{B}_i \otimes \emptyset_A)$$

$$||P^1_b|| = ||\wedge_b, a_i|| \looparrowright\!\sim (\emptyset_B \otimes \hat{A}_i$$

If two dimensions (attributes) are considered, a co-array $||b_p, a_p||$ can be graphically represented, as shown in Fig. 1. Its first order co-array projections are: $||P^1_a|| = ||b_p, \wedge_a||$ and $||P^1_b|| = ||\wedge_b, a_p||$. Even though one of the sides is the identity, the other is not. Therefore, the co-array projections are not completely the identity, there is a line missing, as is shown in Fig. 4. This line corresponds to a first order array projection.

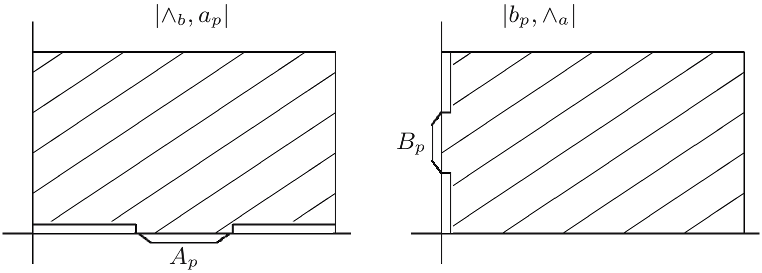


Fig. 4. 2-dimensional co-array projections

The number of nth order co-array projections can be easily found by counting the number of ways n $\wedge$ components can be placed in a co-array $||t_p|| = ||g_p, \ldots, b_p, a_p||$. Depending on the location of the $\wedge$ components, there are $\frac{g!}{n!(g-n)!}$ nth order co-array projections.

4 Conclusion

The multivalued algebra does not handle raw data, it handles declarative descriptions of tha data. The knowledge contained in a data table can be obtained using arrays and co-arrays.

Array projactions and co-array projections allow us to describe uncertain data tables, that is, those data tables that are incomplete (missing attribute values) and those that are ambiguous (non-distinguishing attributes).

On occasions data tables are incomplete, that is, several entries are empty. Data tables with no attribute values can be described by array projections. the order of the array projection is the number of missing attribute values.

Data tables can also be ambiguous, that is, some attributes are non-distinguishing (all attribute values apply). These data tables can be described by co-array projections. The order of the co-array projection is the number of non distinguishing attributes.

The array projections and co-array projections presented in this paper can be seen as a valid strategy for handling uncertain data tables.

Future work will deal with inductive learning [4], and the inclusion of the array projection in the learning process. Furthermore, the fact that $\bigvee$ and $\bigwedge$ are degenerate arrays originates the need to further investigate the third type of degenerate arrays and to try to forsee the relationship between the three types of degenerate arrays, their projections and uncertainty.

Acknowledgements

This work has been partially supported by the Dirección General de Investigación del Ministerio de Educación, Ciencia y Tecnología through the TIN2004-07926 project.

References

1. Miró-Julià M., Fiol-Roig G.: An Algebra for the Treatment of Multivalued Information Systems. Lecture Notes in Computer Science, **2652** (2003) 556–563.
2. Miró-Julià M.: A Contribution to Multivalued Systems. PhD thesis, Universitat de les Illes Balears, 2000.
3. Miró-Julià M.: The Zero Array: A Twilight Zone. Lecture Notes in Computer Science, **2809** (2003) 92–103.
4. Fiol G.: Inductive Learning from Incompletely Specified Examples. Frontiers in Artificial Intelligence and Applications, **100** (2003) 286–295.

Neural Network Sensitivity Analysis Applied for the Reduction of the Sensor Matrix

Przemyslaw M. Szecówka[1], Andrzej Szczurek[2], Maciej A. Mazurowski[1],
Benedykt W. Licznerski[1], and Franz Pichler[3]

[1] Faculty of Microsystem Electronics and Photonics,
Wroclaw University of Technology, Janiszewskiego 11/17, 50-372 Wroclaw, Poland
`przemyslaw.szecowka@pwr.wroc.pl`
[2] Institute of Environmental Protection Engineering,
Wroclaw University of Technology,
Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland
[3] Institut fur Systemwissenschaften, Johannes Kepler University,
Altenbergerstr. 69, A-4040 Linz, Austria

Abstract. The neural network sensitivity analysis, involving neural network training and the calculation of its outputs derivative on inputs, was applied to select the least significant sensor in the multicomponent gas mixtures analysis system. The sensitivity analysis results, collected for various neural network structures were compared with the real significances of the sensors, determined experimentally. The question of the influence of the correlation of the input vector elements on the analysis results was also illustrated and discussed.

1 Introduction

Contemporary gas mixtures analysis technology relies on the matrices of sensing elements and smart data processing techniques applied for their responses analysis, providing the desired information of qualitative or quantitative character. This approach is usually forced by the low selectivity of sensors, which disables simple calibration of 1 sensor for 1 gas appearing in the mixture. It may be observed that most of gas sensor systems described in the literature, although very successful, contain more or less redundant sets of sensing elements [1,2,3,4,5]. Shall be noted that each redundant sensor, applied in the matrix increases the cost of both fabrication and operation of the prospective system. Seems like the main problem is the lack of the reasonable and efficient methods of sensors selection.

If to assume that the preliminary version of the system, providing the acceptable accuracy of measurements is available (what is btw. usually reached using the large enough sensor array) the problem may be transformed to the elimination of the most redundant sensors, as far as the required performance of the system is preserved. The possible solution may be neural network sensitivity analysis [6,7], adopted for the estimation of the significance of the information given to the system by the particular sensors in the matrix. The neural networks

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 27–32, 2005.
© Springer-Verlag Berlin Heidelberg 2005

approach is somewhat unusual here. In the initial phase of the system construction the dummy neural network is trained to provide the sensitivity analysis and judge each sensor. After selection of the reasonable set of sensors the eventual data processing algorithm may be created using either neural network again or any other methodology.

The paper focused on the investigation of the efficiency and reliability of the neural network sensitivity analysis approach in the context of both the real world application in gas sensor system and the data artificially created for the purposes of the experiment.

2 Neural Network Sensitivity Analysis

The operation of the feedforward neural network with a single hidden layer and the sigmoid transfer function applied may be described by (1).

$$y_k^{(2)} = f\left(\sum_{j=0}^{J} w_{kj}^{(2)} f\left(\sum_{i=0}^{I} w_{ji}^{(1)} u_i\right)\right), \text{ where } f(x) = \frac{1}{1+e^{-x}} \tag{1}$$

After the training process, when the weights, denoted by w (with the appropriate indexes) are fixed, the neural network gains the unique approximation capabilities [8]. In the context of sensor systems the vectors of sensors responses u are transformed to the series of outputs $y^{(2)}$, providing desired information of either qualitative or quantitative character, on request.

The k-th neural network output sensitivity for the selected input u_i is defined as a derivative (2), which for the presumed construction (1) gives (3). Calculation of the sensitivity is made for each output-input pair and for each input pattern $u^{(p)}$. Concerning the patterns, the global sensitivity for the whole data set is calculated using for instance the Euclidean formula (4) or by finding the maximum absolute value. Eventually the sensitivity matrix is obtained with e.g. inputs listed in columns and outputs listed in rows. Further analysis may involve the min-max procedure providing the series of parameters describing how much the neural network is sensitive to the particular input. The inputs (i.e. the sensors in our context), with the lower values of sensitivity shall be considered as the candidates to remove.

$$s_{ki} = \frac{dy_k}{du_i} \tag{2}$$

$$s_{ki} = f'\left(x_k^{(2)}\right) \sum_{j=1}^{J} \left(w_{kj}^{(2)} f'\left(x_j^{(1)}\right) w_{ji}^{(1)}\right) \tag{3}$$

$$S_{ki} = \sqrt{\sum_{p=1}^{P} \left(s_{ki}^{(p)}\right)^2} \tag{4}$$

Formula (3) was originally proposed for pruning the redundant inputs of the neural network with the single hidden layer only [6]. The experience with construction of neural networks for the gas sensor arrays shows the need of applying the structures with two hidden layers to obtain the highest performance. Some doubts may appear then whether the sensitivity analysis applied for the too scant neural network structures would give the reliable results. Eventually the extended sensitivity formula, for the neural networks with two hidden layers, was calculated. Starting from (5) describing the appropriate neural network with two hidden layers, where u_h and $y_k^{(3)}$ denote the selected input and output, and the $w^{(1)}$, $w^{(2)}$ and $w^{(3)}$ with the appropriate indexes are the weights of the neurons in the following layers, the eventual sensitivity is given by (6).

$$y_k^{(3)} = f\left(\sum_{j=0}^{J} w_{kj}^{(3)} f\left(\sum_{i=0}^{I} w_{ji}^{(2)} f\left(\sum_{h=0}^{H} u_h\right)\right)\right) \tag{5}$$

$$s_{kh} = \frac{dy_k^{(3)}}{du_h} = f'\left(x_k^{(3)}\right) \sum_{j=1}^{J} \left(w_{kj}^{(3)} f'\left(x_j^{(2)}\right) \sum_{i=1}^{I} \left(w_{ji}^{(2)} f'\left(x_i^{(1)}\right) w_{ih}^{(1)}\right)\right) \tag{6}$$

Analysis of (3) and (6) induces that the extension of the sensitivity formula for the bigger and bigger neural networks may be generalized to the recurrence, somewhat similar to the classic error backpropagation [9], where the sensitivity of the bigger structure may be calculated as a weighted sum of the sensitivities calculated for the appropriate nodes of the smaller structure (7),(8).

$$s_{ki}^{(L)} = \frac{\partial y_k^{(L)}}{\partial u_i} = f'\left(x_k^{(L)}\right) \sum_{j=1}^{J} \left(s_{ji}^{(L-1)} w_{kj}^{(L)}\right) \tag{7}$$

$$s_{ki}^{(1)} = \frac{\partial y_k^{(1)}}{\partial u_i} = f'\left(x_k^{(1)}\right) w_{ki}^{(1)} \tag{8}$$

3 Experimental

The sensitivity analysis methodology was applied to the design of the sensor system providing quantitative analysis of the mixtures of butanol and toluene. The matrix, initially containing six sensors - TGS 800, TGS 822, TGS824, TGS 825, TGS 880, TGS 883 [10], was placed in a test chamber with controlled atmosphere for the multi-component characterisation. The set of vectors containing the gas concentrations and sensors responses was collected this way building up the 148 samples data set [2]. A series of the neural networks was created to provide some estimation of the reliablity of the method, with the sensors responses acting as the input and two gas concentrations as the desired output. The structures were varying between 6-10-2 and 6-40-2. For each neural network the sensitivity analysis was performed, i.e. sensitivities of all outputs to all the

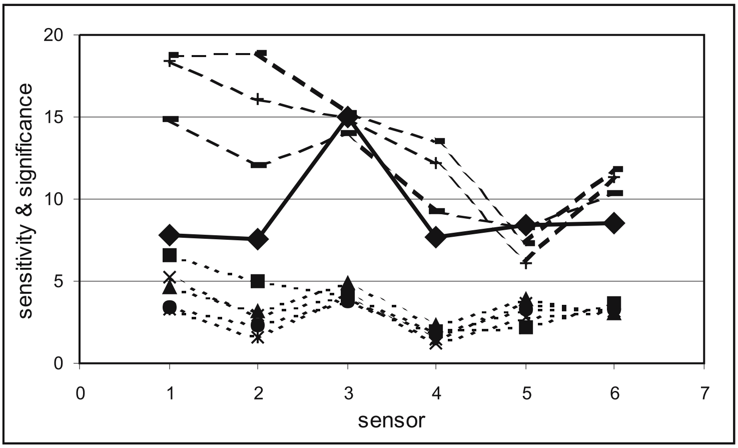


Fig. 1. Neural network sensitivities obtained for the structures with the single hidden layer (dotted lines) and with two hidden layers (dashed lines) compared with the experimental estimation of the sensors significance (thick solid line).

inputs were calculated for all the available samples. The global sensitivities for all patterns were calculated in two variants - Euclidean formula (5) and *maximum*. In-house developed software tools were used for both the neural networks development and sensitivity calculation. In further steps of the analysis the sensor with the lowest absolute value of the sensitivity factor shall be removed and the whole process may be repeated, with the reduced sensor matrix, to point the next one to remove etc. This process shall be stopped when either the system performance decreases dramatically or the results of the sensitivity analysis performed for series of neural networks are no longer coherent.

The sensitivity analysis performed for several neural networks with the single hidden layer consequently pointed to the sensor No. 4 as redundant. The details may be found in [11]. The sensitivity factors, obtained in several trials, are presented in Fig. 1 (the dotted lines). The dashed lines present analogous results obtained for the structures with two hidden layers. Various structures were implemented, starting from 6-12-8-2 up to 6-50-30-2. The results within this group are similar again, but this time sensor No. 5 is commonly recognized as redundant one. Such contradiction could be perceived as a stop condition for the sensor matrix reduction, but it is known from the other experiments that this set of sensors may be reduced indeed without visible loss in accuracy.

Further investigation of this phenomenon involved the introduction of the six data sets deriving from the original one, but with 1 input-sensor removed in each. The series of neural networks were trained for each variant, targeting in the experimental determination of the *significance* factors for all the sensors. These factors were calculated as an average error of the 3 best structures. (The higher error of the neural network trained without particular input denotes its higher significance). The balance of these factors was plotted again in Fig. 1 (thick solid

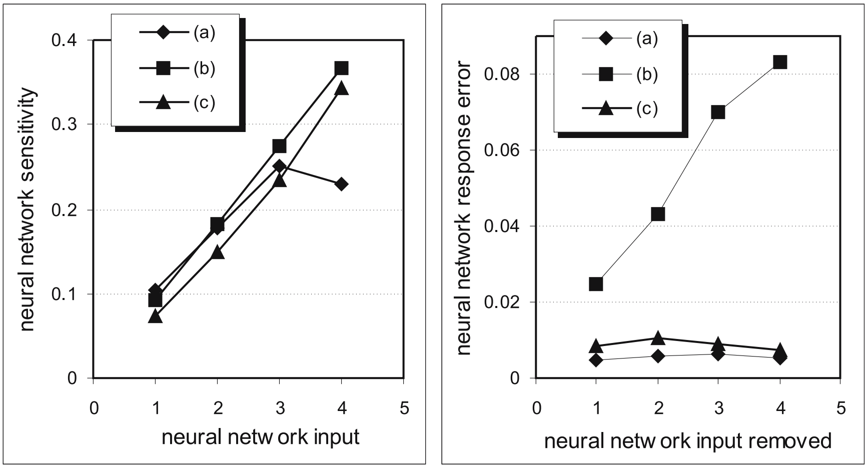


Fig. 2. Sensitivity analysis results (left) compared with the experimental estimation of the input significance (right) for (a) linear dependent, (b) independent and (c) non-linear dependent data

line). If to analyse the precise values of the sensors significance factors obtained this way, No. 2 shall be considered most redundant this time. Eventually the results of the sensitivity analysis performed for various neural networks, especially in the critical *first to remove* context are sometimes contradicting themselves and simultaneously contradicting the experimentally determined significance of the sensors. Shall be noted however that the experimental analysis, which shall be perceived as the most reliable here, estimates the significance of five sensors (i.e. No. 1, 2, 4, 5 and 6) at the very similar level. The insignificant differences mean that in fact any of these sensors could be removed, with similar impact on the system performance, what probably justifies the contradictions mentioned before.

The meaningless differences between the significance of the sensors and consequently the contradictions are probably caused by the correlation of the sensors responses (i.e the elements of the neural network input vector), which is very high. The simple experiment may show how the dependent inputs may keep the sensitivity analysis results far away from the real balance of the inputs significance. Let's take a sample function of $y = x_1 + 2x_2 + 3x_3 + 4x_4$ and generate a data set for the appropriate neural network training, in 3 variants - the first one with independent input variables x_1, x_2, x_3, x_4, the second one with linear dependence $x_4 = x_1 + x_2 + x_3$, and the third one with non-linear $x_4 = x_1^2 + x_2^2 + x_3^2$. The results of the sensitivity analysis are shown in Fig. 2 (on the left). These ones are very similar for all the data sets. And the real significance factors of the input variables, estimated by the "remove the input and train the neural network" procedure, are completely different for the dependent and non-dependent variants as it is shown in Fig. 2 (on the right), matching the intuitive expectations.

4 Conclusions

The neural network sensitivity analysis may be attractive tool for the reduction of the redundant sensor arrays. Presented experiments have shown however, that it does not provide the absolutely reliable results, when some elements of the input vector are dependent. It may be used, with care, as a reasonable heuristics for the construction of the effective gas sensor arrays, where the number of sensors is critical issue, and in many other fields requiring an estimation of the significance of the particular factor.

References

1. Gutierrez F.J., Ares L., Robla J. et al.: Integrated Sensors for Monitoring Contaminant Gases in Atmospheres and Soils, Proc. IEEE International Symposium on Industrial Electronics, Guimaraes, Portugal (1997) SS113- SS115.
2. Szczurek A., Szecówka P.M., Licznerski B.W.: Application of sensor array and neural networks for quantification of organic solvents vapours in air, Sensors and Actuators B, 58 (1999) 427-432.
3. Martin M.A., Santos J.P., Agapito J.A.: Application of artificial neural networks to calculate the partial gas concentrations in a mixture, Sensors and Actuators B 77 (2001) 681-471.
4. Sobański T., Saffarpour N.: Analysis of gas mixture using sensor array and artificial neural networks. In: Cybernetics and systems 2002. Proceedings of the Sixteenth European Meeting on Cybernetics and Systems Research. Vol. 1. (2002) 97-100.
5. Jervis B.W., Desfieux J., Jimenez J. et al.: Quantification of gas concentrations in mixtures of known gases using an array of different tin-oxide sensors, IEE P-Sci. Meas. Tech. 150 (3) (2003) 97-106.
6. Żurada J.M., Malinowski A., Usui S.: Perturbation method for deleting redundant inputs of perceptron networks, Neurocomputing 14 (1997) 177-193.
7. Engelbrecht A.P.: A New Pruning Heuristic Based on Variance Analysis of Sensitivity Information, IEEE Trans. On Neural Networks, Vol. 12, No. 6 (2001) 1386-1399.
8. Kornik K., Stinchcombe M., White H.: Multilayer feedforward networks are universal approximators, Neural Networks, vol. 2 (1989) 359-366.
9. Rumelhart D.E., Hinton G.E., Williams R.J.: Learning Representations by Back-Propagating Errors, Nature, 322, (1986) 533-536.
10. Figaro Gas Sensors, TGS 800 series specification, Figaro Engineering Inc.
11. Szecówka P.M., Szczurek A.: Sensors selection for gas mixtures analysis systems, Proc. 26-th International Spring Seminar on Electronic Technology, Stara Lesna (2003) 398-403.

Fuzzy Modeling for Coal Seams A Case Study for a Hard-Coal Mine

José Antonio Martin[1], Teresa de Pedro[1], Carlos González[1], R. García[1],
Luís Argüelles[2], Jose M. Rivas[2], and Javier Toraño[2]

[1] Industrial Automation Institute, CSIC, La Poveda, Arganda del Rey,
28500 Madrid, Spain
`{jamartin, ricardo, tere, gonzalez}@iai.csic.es`
[2] Carbonar S.A. Cardenal Cienfuegos 8, 33007 Oviedo,
ETSIMO. Independencia, 13, 33004, Oviedo
`{arguelles, jmrivas}@carbonar.es`
`jta@uniovi.es`

Abstract. This work presents a set of linguistic variables for modelling some geologic and morphological features of a coal seam. Since self-advancing winning coal systems and their interactions with coal faces represent extremely complex situations where traditional mathematical models are unable to offer working solutions, we explore the viability of using Fuzzy-logic based techniques in order to obtain a good-enough model that allows technicians at mine to get a better understanding of the variables at play and to foresee the coal production.

1 Introduction

Coal mining is a strategic industrial activity in the North of Spain. An example of an especially competitive mine is that of Carbonar S.A. because of its hard coal seam about 4m in thickness and the planning and use of modern techniques for coal winning. Coal thickness up to 6 meters and other physical characteristics of the seam permits the exploitation with a longwall mining method. This method consists on a set of self-advancing shields (left part of figure) that are located adjacent to one another along a transversal section of the seam. The shields are used to keep the hanging-wall in place while the coal winning machine takes 60 cm off the wall (right part of figure) for every coal winning run. After the shearer passes, the coal is taken off the mine with a continuous system of transport (panzers, conveyor belts). A manually operated mechanism permits the operators to advance the shields. In Figure 1 can be seen the Shield supporting system and the Coal-winning shearer working.

A research project [1] has been awarded in 2004 to Carbonar S. A. by the "Centro para el Desarrollo Tecnológico Industrial" (CDTI) agency of the Spanish Ministry of Industry for financially supporting the research. The goals of the project are to analyze, model and simulate the longwall exploitation system merging finite element modeling (FEM), fuzzy logic techniques (FL) and virtual reality (VR). In this paper

---

[1] "Sistema de Simulación Inteligente en Arranque Mecanizado Integral", CDTI 20040116.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 33 – 37, 2005.
© Springer-Verlag Berlin Heidelberg 2005

we will present the preliminary works of that project. Aside the R+D department of Carbonar S.A., CSIC's Instituto de Automática Industrial and the Mining and Exploration Department of the Mining Engineering School of the University of Oviedo cooperate in the project.

As exposed, one of the goals of the project is to simulate the behavior of the longwall equipment with fuzzy logic. Fuzzy logic is a technique whose power to perform control of systems where the model is not well known has already been proven. In this work we attempt to use fuzzy logic not to control the longwall, but to model its interplay with the existing geomorphologic conditions. In order to do this, the project is planned in three phases:

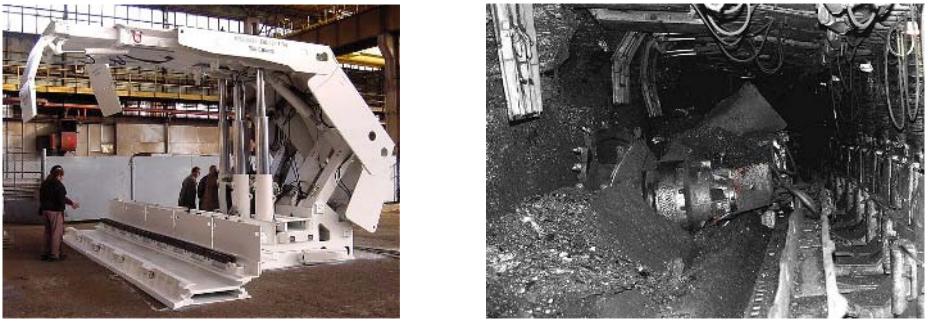


Fig. 1. Shield Supporting system and the Coal-winning shearer working

1. Simulation of the coal winning system. There are many variables influencing the coal winning activities. This part of the project will attempt to identify them and assign linguistic values to them. Several output variables will be established as, for instance, the number of shearer passes per labor shift, or the daily advance in longitude for the coal seam resulting from winning the coal. After that, it will be necessary to establish the rules linking the input and output variables.
2. If problems are detected in the simulation by the fuzzy module under definite conditions, a detailed physical analysis of the shield mechanism will be run for these conditions by means of finite elements modelling techniques.
3. A virtual reality system will show the results of the simulation.

2 Fuzzy Model and Simulation of the Coal Winning System

Our initial model of the coal winning system was developed as a set of geologic and morphologic properties represented by means of a set of input variables to the system. For each of these input variables, we have defined the corresponding linguistic label set or fuzzy partitions. These linguistic labels describe the level of

Features:

- Coal thickness
- Transversal dip
- Longitudinal dip
- Roof fractures
- Barren
- Coal Hardness

For example, the coal thickness is one of the most important features in the mining activity because it determines the kind of exploitation methods that can be performed. In our particular case, the coal thickness is very well suited for a coal winning system based on a semi-automated machine. In Figure 2 we can appreciate the adjustable shield supporting system with its maximum expansion is up to 5 meters.

Other important features are Longitudinal and Transversal Dip, which describe the inclination of the coal seam.

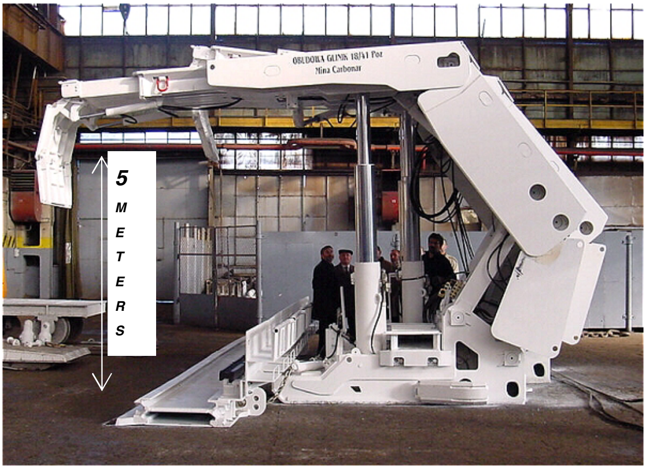


Fig. 2. Representation of the adjustable shields structure

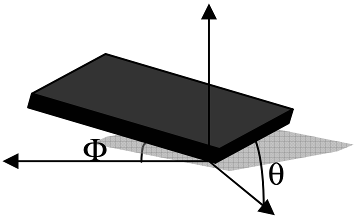


Fig. 3. Longitudinal and Transversal Dip

All of these features are defined by means of trapezoidal membership functions in order to be processed by the ORBEX system [2, 3]. The complete set of input features are defined above with its four values, one for each trapezoid singular point:

INPUTS:

- Coal thickness { low 0 0 2.4 4 high 2.4 4 5 5 }
- Transversal dip { low 0 0 15 30 high 15 30 35 35 }
- Longitudinal dip { low 0 0 5 10 high 5 10 20 20 }
- Roof fractures { low 0 0 1 2 medium 1 2 2 3 high 2 3 5 5 }
- Barren { medium 0 50 50 100 }
- Hardness { coal 0 0 30 40 slate 30 40 60 100 sandstone 50 80 100 100 }

The complete graphical set of fuzzy membership functions can be seen in Figure 4.

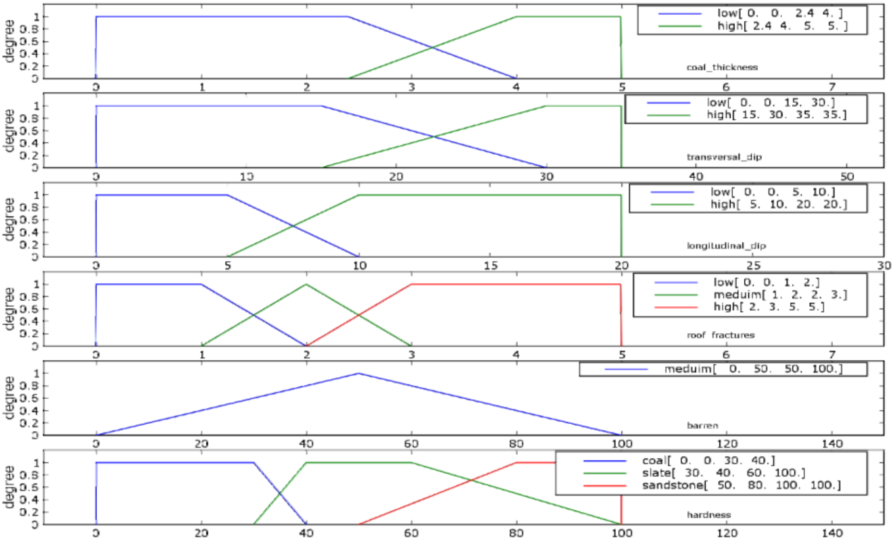


Fig. 4. The complete set of fuzzy sets over the input variables

The output variable of the simulation system is a normalized (0,1) measure of the velocity of advance of the machine.

OUTPUT:

- Advance { slow 0 fast 1 }

In order to obtain a particular value for the output variable {Advance} the values of the inputs variables must be established via a particular probability distribution defined specifically for the simulation and the system must evolve trough a set of fuzzy inference rules which are shown next:

- **IF** transversal dip low **AND** coal thickness high **AND** longitudinal dip low **THEN** Advance fast
- **IF** coal thickness high **AND** longitudinal dip high **AND** transversal dip high **THEN** Advance slow
- **IF** hardness <u>sandstone</u> **THEN** Advance slow

Thus, the ultimate goal of the project can be described by; the quest of a 6-D hypersurface that can be expressed in its general form as the Equation 1.

$$S = f(x_1, x_2, x_3, x_4, x_5) \ldots \; where \tag{1}$$

S = *Advance speed*
x1 = *Coal thickness*
x2 = *Transversal dip*
x3 = *Longitudinal dip*
x4 = *Roof fractures*
x5 = *Barren*
x6 = *Coal Hardness*

Due to the intrinsic uncertainty and complexity of a coal mining activities, such a 6D surface is impossible to express with a traditional closed mathematical models, so we have opted to build a model with the aforementioned linguistic variables and a relatively simple set of expert fuzzy-rules.

References

1. George J. Klir , “Fuzzy Logic in Geology”, Eds. Rober V. Demicco, Elsevier Academic Press, 2004 , ISBN 0-12-415146-9.
2. R. García Rosa y T de Pedro “Modelling a Fuzzy Coprocessor and its Programming Language”, Mathware & Soft Computing 2,3 (1988), 167-174.
3. R. García Rosa y T de Pedro “First Applications of the ORBEX Coprocessor: Control of Unmanned Vehicles”, Mathware & Soft Computing 7 (2000), 265-273.

Optimization of a Class of Uncertain Systems Based on Uncertain Variables

Zdzislaw Bubnicki

Institute of Information Science and Engineering,
Wroclaw University of Technology,
Wyb. Wyspianskiego 27, 50–370 Wroclaw, POLAND
Phone: +48–71–320 33 28, Fax: +48–71–320 38 84
`zdzislaw.bubnicki@pwr.wroc.pl`

Abstract. The uncertain variables have been developed as a tool for decision making in a class of uncertain systems described by traditional models or by relational knowledge representations. The purpose of this paper is to show how the uncertain variables may be applied to specific optimization problems formulated for uncertain static plants. A general approach and the optimization with the given certainty threshold are described in the first part. In the second part the application of the presented approach to an optimal distribution problem is considered. Two examples illustrate the presented concepts.

1 Introduction

The uncertain variables have been introduced and developed as a tool for decision making in a class of uncertain systems [1,4,5,6]. The uncertain variables are described by a certainty distribution given by an expert and describing his/her knowledge on approximate values of the variable. The purpose of this paper is to show how the uncertain variables may be used for specific optimization problems formulated for an uncertain plant described by a functional model or by an inequality with unknown parameters. A general approach to the formulation and solution of the optimization problem is presented in Sect. 3 and its application to an optimal distribution problem — in Sect. 4. Section 2 presents a short description of the uncertain variables and the basic decision problem. Details and examples of different applications may be found in the books [1,4,5].

2 Uncertain Variables and Basic Decision Problem

In the definition of the uncertain variable $\bar{x}$ we consider two soft properties (i.e. such properties $\varphi(x)$ that for the fixed x the logic value $v[\varphi(x)] \in [0,1]$): "$\bar{x} \tilde{=} x$" which means "$\bar{x}$ is approximately equal to x" or "x is the approximate value of x," and "$\bar{x} \tilde{\in} D_x$" which means "$\bar{x}$ approximately belongs to the set D_x" or "the approximate value of $\bar{x}$ belongs to D_x". The *uncertain variable* $\bar{x}$ is defined by a set of values X (real number vector space), the function $h_x(x) = v(\bar{x} \tilde{=} x)$ (i.e.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 38–43, 2005.
© Springer-Verlag Berlin Heidelberg 2005

the certainty index that $\bar{x} \tilde{=} x$, given by an expert) and the following definitions for $D_x, D_1, D_2 \subseteq X$:

$$v(\bar{x} \tilde{\in} D_x) = \max_{x \in D_x} h_x(x)$$

$$v(\bar{x} \tilde{\notin} D_x) = 1 - v(\bar{x} \tilde{\in} D_x) \ ,$$

$$v(\bar{x} \tilde{\in} D_1 \vee \bar{x} \tilde{\in} D_2) = \max\{v(\bar{x} \tilde{\in} D_1), v(\bar{x} \tilde{\in} D_2)\} \ ,$$

$$v(\bar{x} \tilde{\in} D_1 \wedge \bar{x} \tilde{\in} D_2) = \begin{cases} \min\{v(\bar{x} \tilde{\in} D_1), v(\bar{x} \tilde{\in} D_2)\} & \text{for } D_1 \cap D_2 \neq \varnothing \\ 0 & \text{for } D_1 \cap D_2 = \varnothing \ . \end{cases}$$

The function $h_x(x)$ is called a *certainty distribution.*

C–uncertain variable $\bar{x}$ is defined by the set of values X, the function $h_x(x) = v(\bar{x} \tilde{=} x)$ given by an expert, and the following definitions:

$$v_c(\bar{x} \tilde{\in} D_x) = \frac{1}{2}[v(\bar{x} \tilde{\in} D_x) + 1 - v(\bar{x} \tilde{\in} \bar{D}_x)] \tag{1}$$

where $\bar{D}_x = X - D_x$,

$$v_c(\bar{x} \tilde{\notin} D_x) = 1 - v_c(\bar{x} \tilde{\in} D_x) \ ,$$

$$v_c(\bar{x} \tilde{\in} D_1 \vee \bar{x} \tilde{\in} D_2) = v_c(\bar{x} \tilde{\in} D_1 \cup D_2) \ ,$$

$$v_c(\bar{x} \tilde{\in} D_1 \wedge \bar{x} \tilde{\in} D_2) = v_c(\bar{x} \tilde{\in} D_1 \cap D_2) \ .$$

The application of C–uncertain variable means better using of the expert's knowledge, but may be more complicated. Let us consider a static plant with the input vector $u \in U$ and the output vector $y \in Y$, described by a relation $R(u, y; x) \subset U \times Y$ where the vector of unknown parameters $x \in X$ is assumed to be a value of an uncertain variable described by the certainty distribution $h_x(x)$ given by an expert. If the relation R is not a function then the value u determines a set of possible outputs

$$D_y(u; x) = \{y \in Y : (u, y) \in R(u, y; x)\} \ .$$

For the requirement $y \in D_y \subset Y$ given by a user, we can formulate the following **decision problem**: For the given $R(u, y; x)$, $h_x(x)$ and D_y one should find the decision u^* maximizing the certainty index that the set of possible outputs approximately belongs to D_y (i.e. belongs to D_y for an approximate value of $\bar{x}$). Then

$$u^* = \arg\max_{u \in U} v[D_y(u; \bar{x}) \tilde{\subseteq} D_y] = \arg\max_{u \in U} \max_{x \in D_x(u)} h_x(x) \tag{2}$$

where $D_x(u) = \{x \in X : D_y(u; x) \subseteq D_y\}$.

3 Optimization Problems

Let us consider a static plant in which one-dimensional output y denotes a quality index which should be minimized. If the plant is described by a function $y = \phi(u)$ and the function ϕ is known then we can formulate and solve a *deterministic* *optimization problem*: to find $u^* \in U$ minimizing y. For the plant described by the function $y = \phi(u,x)$ or by the inequality $y \le \Phi(u,x)$ (a special form of the relation $R(u,y;x)$ introduced in Sect. 2) where the unknown vector parameter x is assumed to be a value of an uncertain variable characterized by $h_x(x)$ — different formulations of the optimization problem may be considered:

1. For $y = \phi(u,x)$, as an optimal decision one can accept u^* minimizing the mean value [1]:

$$\mathrm{M}[\bar{y}(u)] = \int\limits_{-\infty}^{\infty} y h_y(y;u)dy \cdot [\int\limits_{-\infty}^{\infty} h_y(y;u)dy]^{-1}$$

where

$$h_y(y;u) = v[\bar{x} \tilde{\in} D_x(y)] = \max_{x \in D_x(y)} h_x(x) \ ,$$

$$D_x(y) = \{x \in X : \ y = \Phi(u,x)\} \ .$$

2. For $y = \phi(u,x)$ or $y \le \phi(u,x)$, one can apply an approach presented in Sect. 2. The requirement concerning y may be presented in the form $y \le \alpha$ or $D_y = (-\infty, \alpha]$ where α is given by a user and should be as small as possible. Consequently, the decision $u^*(\alpha)$ determined by (2) and the corresponding certainty index $v[u^*(\alpha)] \triangleq \bar{v}(\alpha)$ are the functions of α. According to (2)

$$v[\Phi(u,\bar{x}) \tilde{\le} \alpha] = v\{\Phi(u,\bar{x}) \tilde{\in} (-\infty,\alpha]\} \triangleq v(u,\alpha) = \max_{x \in D_x(u)} h_x(x) \tag{3}$$

where

$$D_x(u) = \{x \in X : \Phi(u,x) \le \alpha\} \ . \tag{4}$$

The property $\Phi(u,\bar{x}) \tilde{\in} (-\infty,\alpha]$ is denoted here by $[\Phi(u,\bar{x}) \tilde{\le} \alpha]$ and $v(u,\alpha)$ is the certainty index that this property is "approximately satisfied" (is satisfied for an approximate value of $\bar{x}$). In the case of C–uncertain variables, according to (1)

$$\begin{aligned} v_c[\Phi(u,\bar{x}) \tilde{\le} \alpha] &= v_c\{\Phi(u,\bar{x}) \tilde{\in} (-\infty,\alpha]\} \triangleq v_c(u,\alpha) \\ &= \frac{1}{2}[\max_{x \in D_x(u)} h_x(x) + 1 - \max_{x \in X - D_x(u)} h_x(x)] \ . \end{aligned} \tag{5}$$

3. It is easy to note that, as a rule, $\bar{v}(\alpha)$ is an increasing function. For the given certainty threshold $\bar{v}$ determining the required level of the uncertainty, one should solve the equation $\bar{v}(\alpha) = \bar{v}$ with respect to α, and use the solution $\bar{\alpha}$ in the determination of the final results: $\bar{u} = u^*(\bar{\alpha})$ and $\bar{v} = \bar{v}(\bar{\alpha})$. The analogous formulation of the decision problem may be formulated and solved for C–uncertain variables with v_c instead of v. The above statement of the decision problem may be considered as a specific optimization problem where the minimization of y in the deterministic case is replaced by the *maximization of* v *for the given certainty threshold*.

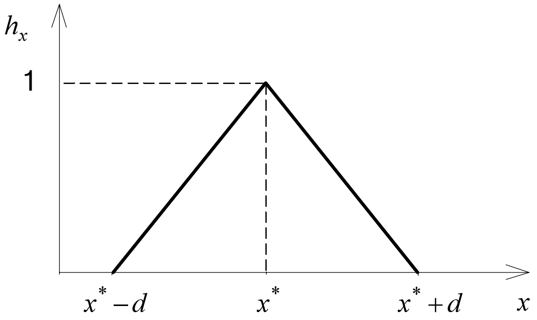


Fig. 1. Example of the certainty distribution

Example 1. Let $u, x \in R^1$ (one-dimensional variables) and

$$y = [1 - (c - xu)^2]^{-1} \ , \quad u \geq 0 \ .$$

The requirement $y \leq \alpha$ is reduced here to

$$(c - \sqrt{1-\alpha^{-1}})u^{-1} \leq x \leq (c + \sqrt{1-\alpha^{-1}})u^{-1} \ .$$

Assume that $h_x(x)$ has the form presented in Fig. 1, with $x^* = d = 0.5$. Using (5) we obtain

$$v_c(x,u) = \begin{cases} (c + \sqrt{1-\alpha^{-1}})u^{-1} & \text{for } u \geq 2c \\ 0 & \text{for } u \leq 1 - (c - \sqrt{1-\alpha^{-1}}) \\ 1 - (c - \sqrt{1-\alpha^{-1}})u^{-1} & \text{otherwise .} \end{cases}$$

It is easy to see that $u_c^*(\alpha) = u_c^* = 2c$ (it does not depend on α) and $v_c(u^*, \alpha) = 0.5(1 + c^{-1}\sqrt{1-\alpha^{-1}})$. From the equation $v_c(u^*, \alpha) = \bar{v}_c$ we obtain

$$\bar{\alpha}_c = [1 - c^2(2\bar{v}_c - 1)^2]^{-1} \ .$$

For the numerical data $c = 1$ and $\bar{v}_c = 0.9$, the results are as follows: $u^* = \bar{u} = 2$, $\bar{\alpha}_c = 2.8$.

4 Optimal Distribution Problem

The presented approach may be applied to allocation problems consisting in the proper task or resource distribution in a complex of operations described by a relational knowledge representation with unknown parameters. Let us consider a complex of k parallel operations described by a set of inequalities

$$T_i \leq \varphi_i(u_i, x_i) \ , \ i = 1, 2, \ldots, k \tag{6}$$

where T_i is the execution time of the i-th operation, u_i is the size of a task (e.g. a raw material) in the problem of task allocation or the amount of a resource in the problem of resource allocation, an unknown parameter $x_i \in R^1$ is a value of an uncertain variable $\bar{x}_i$ described by a certainty distribution $h_i(x_i)$ given by an expert, and $\bar{x}_1, \ldots, \bar{x}_k$ are independent variables. The complex may be considered as a decision plant described in Sects. 2 and 3, where y is the execution time of the whole complex $T = \max\{T_1, \ldots, T_k\}$, $x = (x_1, \ldots, x_k)$, $u = (u_1, \ldots, u_k) \in \bar{U}$. The set $\bar{U} \subset R^k$ is determined by the constraints: $u_i \geq 0$ for each i and $u_1 + \ldots + u_k = U$ where U is the total size of the task or the total amount of the resource to be distributed among the operations. According to the general formulation of the decision problem presented in Sect. 2, the allocation problem may be formulated as an optimization problem consisting in finding the optimal allocation u^* that maximizes the certainty index of the soft property: "the set of possible values T approximately belongs to $[0, \alpha]$" (i.e. belongs to $[0, \alpha]$ for an approximate value of $\bar{x}$).

Optimal distribution problem: For the given φ_i, h_i $(i \in \overline{1,k})$, U and α find

$$u^* = \arg\max_{u \in \bar{U}} v(u)$$

where

$$v(u) = v\{D_T(u; \bar{x}) \tilde{\subseteq} [0, \alpha]\} = v(T(u, \bar{x}) \tilde{\leq} \alpha) \ .$$

The soft property "$D_T(u; \bar{x}) \tilde{\subseteq} [0, \alpha]$" is denoted here by "$T(u, \bar{x}) \tilde{\leq} \alpha$", and D_T $(u; x)$ denotes the set of possible values T for the fixed u, determined by the inequality

$$T \leq \max_i \varphi_i(u_i, x_i) \ . \tag{7}$$

According to (6) and (7)

$$v(u) = v\{[T_1(u_1, \bar{x}_1) \tilde{\leq} \alpha)] \wedge [T_2(u_2, \bar{x}_2) \tilde{\leq} \alpha] \wedge \ldots \wedge [T_k(u_k, \bar{x}_k) \tilde{\leq} \alpha]\} \ .$$

Then

$$u^* = \arg\max_{u \in \bar{U}} \min_i v_i(u_i) \tag{8}$$

where

$$v_i(u_i) = v[T_i(u_i, \bar{x}_i) \tilde{\leq} \alpha)] = v[\varphi_i(u_i, \bar{x}_i) \tilde{\leq} \alpha)] = v[\bar{x}_i \tilde{\in} D_i(u_i)] \ ,$$

$$D_i(u_i) = \{x_i \in R^1 \ : \ \varphi_i(u_i, x_i) \leq \alpha\} \ .$$

Finally, according to (2)

$$v_i(u_i) = \max_{x_i \in D_i(u_i)} h_i(x_i) \tag{9}$$

and

$$u^* = \arg\max_{u \in \bar{U}} \min_i \max_{x_i \in D_i(u_i)} h_i(x_i) \ .$$

It may be shown that if $0 \leq v(u) \leq 1$ then the optimal distribution u satisfies the following set of equations

$$v_1(u_1) = v_2(u_2) = \ldots = v_k(u_k) \triangleq v \ . \tag{10}$$

For the given *certainty threshold* $\bar{v}$, it is possible to determine the value $\bar{\alpha}$ and the optimal distribution $\bar{u}_i = u_i^*(\bar{\alpha})$ in the way described in Sect. 3.

Example 2. Assume that in the case of the task distribution $T_i \le x_i u_i$ and $h_i(x_i)$ has a triangular form presented in Fig. 1 with h_i, x_i^*, d_i in place of h_x, x^*, d. Using (9) one may obtain

$$v_i(u_i) = d_i^{-1}(\alpha u_i^{-1} - x_i^*) + 1$$

for $\alpha(x_i^*)^{-1} \le u_i \le \alpha(x_i^* - d_i)^{-1}$, i.e. for $0 < v_i(u_i) < 1$.
Under the assumption $x_1^* d_1^{-1} = x_2^* d_2^{-1} = \ldots = x_k^* d_k^{-1} \triangleq \gamma$, applying the equation (10) we obtain the following results:

$$v[u^*(\alpha)] \triangleq v^*(\alpha) = 1 + \gamma[\alpha U^{-1} \sum_{i=1}^{k} (x_i^*)^{-1} - 1] \ ,$$

$$u_i^*(\alpha) = \gamma\alpha(x_i^*)^{-1}[v^*(\alpha) + \gamma - 1]^{-1} \ .$$

5 Conclusions

It has been shown how to use the uncertain variables in the formulation and solution of the optimization problems for a static plant with unknown parameters. In particular , a concept of the optimization with the given certainty threshold for the general case and for the optimal task or resource distribution has been described. The presented approach may be extended for complex systems with the distributed knowledge [2] and for closed–loop control systems [3].

References

1. Bubnicki, Z.: Analysis and Decision Making in Uncertain Systems. Springer-Verlag, Berlin London New York (2004)
2. Bubnicki, Z.: Application of uncertain variables to decision making in a class of distributed computer systems. In: Proc. 17th IFIP World Computer Congress, Vol. Intelligent Inf. Proc. Kluwer Academic Publishers, Norwell (2002) 261-264
3. Bubnicki, Z.: Application of uncertain variables to stabilization and parametric optimization of uncertain dynamic systems. In: Proc. 16th IFAC World Congress. Prague, Czech Republic (2005)
4. Bubnicki, Z.: Modern Control Theory. Springer-Verlag, Berlin London New York (2005)
5. Bubnicki, Z.: Uncertain Logics, Variables and Systems. Springer-Verlag, Berlin London New York (2002)
6. Bubnicki, Z.: Uncertain variables and their application to decision making. IEEE Trans. Systems Man Cybernetics, Part A: Systems and Humans **31** (2001) 587-596

Computational Simulation of Categorical Constructions*

Rodrigo Born Vieira and Paulo Blauth Menezes

Computing Institute, UFRGS - Porto Alegre, Brazil
{rodrigor, blauth}@inf.ufrgs.br
http://www.inf.ufrgs.br/

Abstract. Category Theory is a useful topic to the studies of Computer Science. In spite of experiments with children suggest that categorical reasoning is supposed to be natural to humans, the ones who study Category Theory expose difficulties. This paper starts presenting and analyzing this situation. Then it is developed an evaluation of computational models which represents categorical structures as a way to help students in dealing with categorical concepts intuitively. In the context of this evaluation, a computational model is presented.

1 Category Theory

Category Theory is a branch of Pure Mathematics with a scientific field apparently distinct from Computer Science ones. However, some of its properties make this mathematical model very helpful to the studies of Computer Science.

From many characteristics of Category Theory which motivate its use in Computer Science it's relevant to mention implementation independence, duality, inheritance of results, ability to compare expressiveness of other formalisms, strong basis on graphical notation and, above all, expressiveness of its constructions.

One of the main (if it isn't the main) motivations for the use of Category Theory in Computer Science is the expressiveness of categorical constructions, which allows to formalize complex ideas in a simple way. This motivation can be justified by the fact that the development of computational solutions is bounded by the human ability to express the problems and their solutions. So, more expressive formalism produces better solutions with less effort. Additionally, more expressive formalisms help not only in specifications and proofs but also - and mainly - in a better comprehension of problems and in simpler and clearer solutions.

In the context of this paper, implementation independence is an especially relevant property. Since the primitive structures of Category Theory are objects and morphisms, categorical properties must be specified in terms of them. So, when a system is modeled as a category and it produces structured objects or

---

* This work is partially supported by CAPES, CNPq and FAPERGS.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 44–49, 2005.
© Springer-Verlag Berlin Heidelberg 2005

morphisms (e.g. objects representing ordered pairs) their internal structures are categorically irrelevant.

This property allows Category Theory to be seen as a suitable formalization to deal with structure independent abstract properties. Category Theory semantics is able to express a general vision of a system without concerning to irrelevant details (such as implementation), providing operations powerful enough to treat the problem in high level.

2 Piaget's Experiments

Jean Piaget shows in [5] that the idea of composition, typical on categorical system, is in the genesis of human being. Piaget's results conduct for the conclusion that the intuitive human reasoning is categorical. It is explicit exposed in [5] in the following quotation:

> *I wanted to make plausible, in broad outline, the idea that the theory of categories, considered as a theory of mathematical constructions, reflects the genetic*[1] *constitution of man's cognitive tools, that is, the detachment of transferrable schemes from a set of actions, then similar operations on those schemes, then similar operations on schemes of schemes, and so forth. Hence it seems clear to me that the categorical style, as a way of envisioning an important aspect of the genesis of man's cognitive faculties, is not a style imposed on genetic epistemology from the outside but is a style that, by nature, is adequate for describing the constructions discovered by genetic epistemology.*

Additionally, it's possible to make a relation between implementation independence property of Category Theory and this way of reasoning. To deal with a category that is modeling a system means to deal with the abstract semantic of its objects and morphisms and forget the specific structure of the system. The way that Category Theory deals with morphisms and compositions is a construction of a chain of composition of properties. So, the abstract reasoning that is in the genesis of human being is represented in Category Theory.

3 Problems of Categorical Reasoning Development

The conclusions of Piaget's experiments not only show that Category Theory could be taught earlier than in undergraduation level but also that it should be done. The development of logic formal thought on children allows developing skills and reasoning lines which normally aren't developed in childhood, despite of being required in many areas of knowledge. This kind of skill, and in particular general and unified thought, typical of Category Theory, although being intuitive for human being, it is discouraged beside the learning process.

---

[1] In the context of Piaget's experiments, the adjective *genetic* refers to *genesis*, not to *gene*.

Category Theory reasoning is used to be learned in some computer science graduation courses and, less frequently, undergraduation courses. Its mathematical principles, which could be easy, natural and intuitive, become very dense and abstract because of the way we have been stimulated to reason. Additionally, the heterogeneity in the presentation of introductory topics and the lack of published works in other languages than English complete the set of obstacles to be surpassed before applying Category Theory in Computer Science.

4 Computational Simulators

To make this abstract theory become concrete and kind of intuitive is a challenge and an interesting research area. This context has been motivating us to research about computational simulators.

It's well known that there are limits on representing structures and solving problems in a computer. Although the computational simulation of Category Theory is an unexplored area, it's easy to see that isn't possible to construct a simulator able to represent all Category Theory. By deduction, it isnt possible to implement the verification of a property that works for any kind of category.

However, the research in computer science shows we can compute and represent relevant structures and operations, as finite categories with objects being finite sets and morphisms being relations. This kind of simulator is an efficient way to turn concrete these abstract concepts, making this theoretical concept closer to related interests in applied researches.

There is a lack of research in this area. The existent simulators deal only with basic categorical concepts [2] [4] or are inaccessible to laymen [7].

Watching this scenery, we have been researching about technologies to implement categorical constructions and have been implementing simulators. In this paper we present the technologies used to implement these categorical constructions and the theoretical principles we chose to simulate computationally.

5 State of the Art

We have found only three programs produced in the context of this kind of researching:

- Category Construction Program (or DBC - acronym of "A Database of Categories") [4]: it is a text interface program developed in ANSI C, which makes it platform dependent. However, since it was developed in C standard, it's possible to compile its font in distinct platforms. It doesn't have internet support. The objects and morphisms of DBC are atomic. It represents basic categorical elements and is able to store functors.
- Category Theory Database Tools (CTDT) [2]: it's a Java applet - which turns it platform independent and accessible by internet - with graphical interface. It represents the same categorical structures supported by DBC;

- Computational Category Theory Project (CCTP) [7]: it is a text interface software developed in ML, which turns CCTP platform dependent. It doesn't have internet support. To operate this software, the user needs to have notions of functional environment operation. It is much more powerful then the other two applications, supporting several categorical and functorial elements. It supports structured objects and morphisms.

Since the presented simulators deal only with basic categorical concepts or are inaccessible to laymen, they have limitations to achieve our goal: to turn Category Theory easier, more intuitive and more concrete.

Our goal requires an accessible program, which demands as less previous knowledge as possible - graphical interface, platform independence and Internet support are desirable characteristics. Since a software can't represent all Category Theory elements, it's important to make a good selection of which elements are more relevant to be implemented. And our goal will be as near to be achieved as much categorical calculus and characteristics are covered by the application. So, accessibility, relevance of the implemented structures and high coverage of categorical concepts are the elements that should be used to evaluate a categorical simulator.

6 Modeling Categorical Structures

Although the functional paradigm is the most natural one to model categorical structures, the functional languages don't have, in general, good usability. Since accessibility is one of the characteristics that we desire for our software, we've discarded functional paradigm as the way we would model.

Among the other paradigms, the object oriented one has the most appropriate characteristics to model Category Theory elements (especially encapsulation). Additionally, Java is one of the programming languages which most helps in accessibility aspect. And Java is object oriented. The model that is presented in this paper is being implemented by our group using Java.

The object oriented analysis focus in two diagrams: package diagram and class diagram. The package diagram contains 5 packages. Since each package is modeled as a class diagram, there is 5 class diagrams.

br.ufrgs.inf.catres.error. The classes of this package provide support to identify why a given graph isn't a category. This evaluation is divided in three specialized subclasses: IdentityError (it verifies the identity law), CompositionError (it verifies if all possible compositions exist) and AssociativeError (it verifies the associative law).

br.ufrgs.inf.catres.operation. This package is used to make categorical calculus. The ones which we've implemented were cone, equalizer, pre-product and product. Although the dual operations arent implemented, they can simulate them using dual categories as operators.

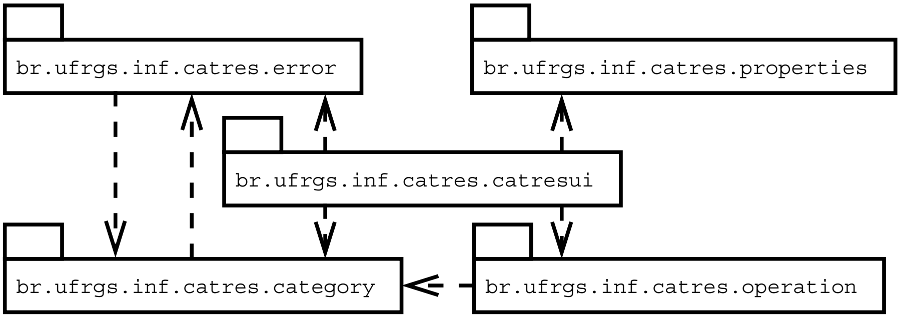

br.ufrgs.inf.catres.error
br.ufrgs.inf.catres.properties
br.ufrgs.inf.catres.catresui
br.ufrgs.inf.catres.category
br.ufrgs.inf.catres.operation

Fig. 1. Package Diagram

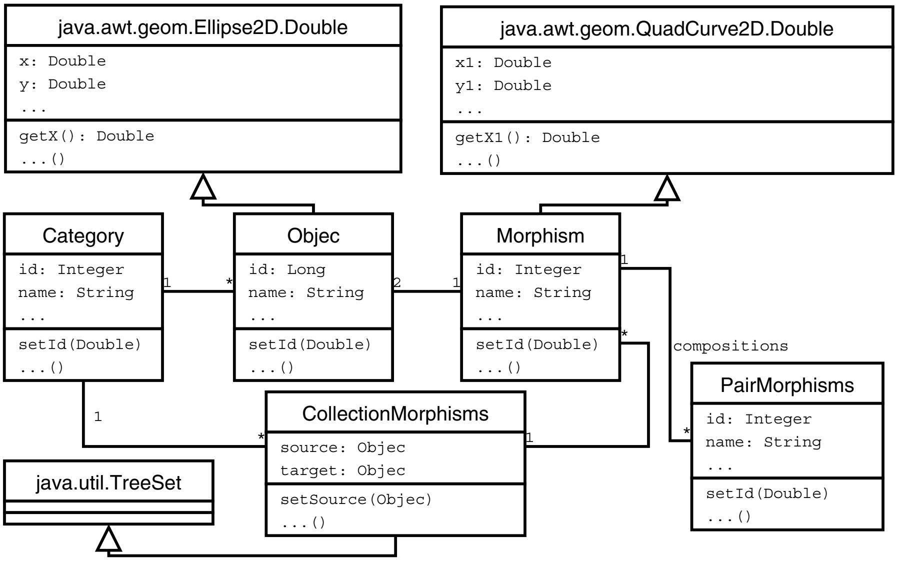


Fig. 2. Class Diagram br.ufrgs.inf.catres.category - Partial Representation

br.ufrgs.inf.catres.category. This package is the kernel of the application. It contains the classes which define a category and its components, such as some auxiliary structures. A category is modeled basically as a collection of morphisms and a collection of objects. While the collection of objects contains instances of the class Objeto, the collection of morphisms contains instances of the class CollectionMorfismos, which is a class which joins all parallel morphisms of a given category (so, we have a collection of parallel morphisms). The compositions are represented as a collection of pair of morphisms.

br.ufrgs.inf.catres.catresui. This package contains all classes developed to construct the graphical interface. The main classes of this package are *Diagram-*

Builder (the application container) and *DiagramBuilderPanel* (the area where the graphs are edited). These two classes are the starting point to all other application actions.

br.ufrgs.inf.catres.properties. - It contains only one class, which provides to the application easy access to configure properties.

7 Conclusions

We've modeled and implemented a categorical simulator following three principles: accessibility, relevant structures and high coverage.

Because of the graphical user interface, the software can be used without any knowledge of special operation environments (e.g. functional). The way that the categories are created and manipulated is very intuitive. It can be accessed by any platform with Java Virtual Machine. It's accessible by Internet. We have been working on mechanisms to implement a good treatment to categories with thousands of objects and morphisms. These aspects improve the accessibility.

The model represents both categories with atomic objects and morphisms and structured objects (sets) and morphisms (relations between sets). Relations have been used in computer science especially in databases. We have been working to implement a support to partial functions and total functions in this tool - implementing restrictions inside relations.

The model represents the most usual structures and calculus of Category Theory - including finite product. The tool is able to represent functor. We have been working to model and implement functorial calculus.

References

1. Saunders Mac Lane. *Categories for the Working Mathematician.* Springer Verlag, New York, USA, 2nd edition, 1998.
2. R. Rosebrugh and J. Bradbury. *Category Theory Database Tools.* Mount Allison University, Canada, 1998. Available at http://cs.mta.ca/research/rosebrugh/ctdt/.
3. Michael Barr and Charles Wells. *Category Theory for Computing Science.* Prentice Hall International Series in Computer Science. Prentice Hall, 2nd edition, 1995.
4. M. Fleming, R. Gunther, R. Rosebrugh. *A Database of Categories.* Mount Allison University, Canada, 1995.
   Available at http://mathcs.mta.ca/research/rosebrugh/dbc/.
5. Jean Piaget. *Morphisms and Categories - Comparing and Transforming.* Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1992.
6. Peter J. Freyd and Andre Scedrov. *Categories, Allegories.* North Holland Publishing Company, 1990.
7. D. E. Rydeheard and R. M. Burstall. *Computational Category Theory Project.* Electronic edition, 1988. Available at http://www.cs.man.ac.uk/ david/categories/.

Composing Transitions into Transactions in UML Diagrams

Júlio Pereira Machado[1] and Paulo Blauth Menezes[2]

[1] Faculdade de Informática, Pontifícia Universidade,
Católica do Rio Grande do Sul, Porto Alegre, RS, Brasil
`juliopm@inf.pucrs.br`
[2] Instituto de Informática, Universidade Federal do Rio Grande do Sul,
Porto Alegre, RS, Brasil
`blauth@inf.ufrgs.br`

Abstract. When modeling concurrent or parallel systems, we must be aware that basic activities of each system may be constituted by smaller activities, i.e. transitions may be conceptually refined into transactions. Nevertheless, the Unified Modeling Language seems to lack compositional constructs for defining atomic actions/activities/operations. We discuss proper extensions for UML behavioral diagrams that are able to cope with the concept of transaction. Transactions are formally defined through a special morphism between automata in a semantic domain called Nonsequential Automata.

1 Introduction

The Unified Modeling Language (UML) [1] may be used to describe both the structure and behavior of object-oriented systems using a combination of notations. For the modeling of the dynamic behavior, a number of different models are offered such as interaction, state and activity diagrams.

When modeling concurrent or parallel systems with such diagrams, we must be aware that basic activities of each system may be constituted by smaller activities, i.e. transitions may be conceptually refined into transactions. This important notion is present in different fields of computer science like operating system's primitives, implementation of synchronization methods for critical regions, database management systems, and protocols, just no name a few. In this sense, when modeling a computational process, we need means of composing subactivities both in a non atomic or atomic way. Nevertheless, the UML seems to lack compositional constructs for defining atomic actions/activities/operations.

In this work[1], we concentrate on describing groups of sequential or concurrent activities that are responsible for performing a computation, and we address the issue of modeling transactions. We remark that in our setting the term

---

[1] This work was partially supported by CTXML/Microsoft/PUCRS, CNPq (Projects HoVer-CAM, GRAPHIT, E-Automaton) and FINEP/CNPq (Project Hyper-Seed) in Brazil.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 50–55, 2005.
© Springer-Verlag Berlin Heidelberg 2005

"transaction" denotes a certain activity of the system that might be composed by many, possibly concurrent, subactivities. Moreover, we require this composition of activities to be considered atomic.

2 Nonsequential Automata

Nonsequential Automata [2,3] constitute a non interleaving semantic domain, with its foundations on category theory, for reactive, communicating and concurrent systems. It follows the so-called "Petri nets are monoids" approach [4] and is similar to Petri nets, but it is a more concrete model - it can be seen as computations from a given place/transition net. In the next definitions **CMon** denotes the category of commutative monoids and $k \in \{0,1\}$ (for simplicity, we omit that $k \in \{0,1\}$).

A nonsequential automaton $NA = \langle \mathrm{V}, \mathrm{T}, \delta_0, \delta_1, \iota, \mathrm{L}, lab \rangle$ is such that $\mathrm{V} = \langle V, \oplus, 0 \rangle$, $\mathrm{T} = \langle T, ||, \tau \rangle$, $\mathrm{L} = \langle L, ||, \tau \rangle$ are **CMon**-objects of states, transitions and labels respectively, $\delta_0, \delta_1 : \mathrm{T} \to \mathrm{V}$ are **CMon**-morphisms called source and target respectively, $\iota : \mathrm{V} \to \mathrm{T}$ is a **CMon**-morphism for mapping identities, and $lab : \mathrm{T} \to \mathrm{L}$ is a **CMon**-morphism for labeling transitions such that $lab(t) = \tau$ whenever there is $v \in V$ where $\iota(v) = t$. Therefore, a nonsequential automaton can be seen as $NA = \langle \mathrm{G}, \mathrm{L}, lab \rangle$ where $\mathrm{G} = \langle \mathrm{V}, \mathrm{T}, \delta_0, \delta_1, \iota \rangle$ is a reflexive graph internal to **CMon** representing the automaton shape, L is a commutative monoid representing the labels of transitions and lab is the labeling morphism associating a label to each transition.

According to the definition, the automaton consists of a reflexive graph with monoidal structure on both states and transitions, initial and final states and labeling on transitions. The interpretation of a structured state is the same as in Petri nets: it is viewed as a "bag" of local states representing a notion of tokens to be consumed or produced. For example, $\langle \{A, B, C\}^{\oplus}, \{t, u\}^{||}, \delta_0, \delta_1, \iota, \{t, u\}^{||}, lab \rangle$ with δ_0, δ_1, ι determined by transitions $t : A \to B$, $u : B \to C$, and labeling $t \mapsto t$, $u \mapsto u$, is represented in figure 1 (identity arcs are omitted and, for a given node A and arcs $t : X \to Y$ and $\iota_A : A \to A$, the structured arc $t||\iota_A : X \oplus A \to Y \oplus A$ is simply noted $t : X \oplus A \to Y \oplus A$). This nonsequential automaton was not completely drawn as it has infinite distinguished nodes, for they are elements of a freely generated monoid chosen to represent its states.

We are able to define atomic composition of transitions through the concept of refinement. It is defined as a special morphism of automata where the target one (more concrete) is enriched with its computational closure (all the conceivable sequential and nonsequential computations that can be split into permutations of original transitions). Considering the previous nonsequential automaton its computational closure is also partially depicted in figure 1 (added transitions were drawn with a dotted pattern).

The computational closure (tc) of a nonsequential automaton is formally defined as the composition of two adjoint functors between the **NAut** category and the category **CNAut** of nonsequential automata enriched with it computations: the first one (nc) basically enriches an automaton with a composition operation

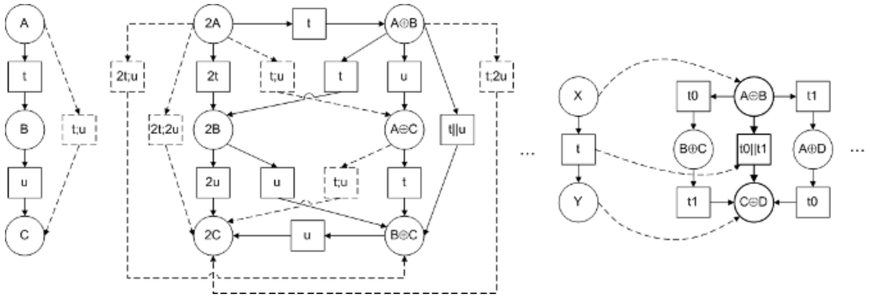


Fig. 1. Nonsequential automaton with computational closure (left) and refinement morphism (right)

on transitions, and the second functor (cn) forgets about the composition operation. Then, the refinement morphism φ from NA into (the computations of) NA' can be defined as $\varphi : NA \to \mathsf{tc}NA'$. Both functors were presented in [5] and due to limitations are not being rephrased here. The transitive closure functor is $\mathsf{tc} = \mathsf{cn} \circ \mathsf{nc} : NAut \to NAut$. To illustrate the refinement morphism, given two nonsequential automata NA and NA' with free monoids on states and labeled transitions respectively induced by transitions $t : X \to Y$, and $t_0 : A \to C$, $t_1 : B \to D$, suppose we want to build a transaction containing both t_0 and t_1. First we apply the transitive closure functor tc. For the last step we build the refinement morphism by mapping the corresponding states and transitions. The refinement $\varphi : NA \to \mathsf{tc}NA'$ is given by $X \mapsto A \oplus B$, $Y \mapsto C \oplus D$, $t \mapsto t_0||t_1$ (see figure 1 - right). Notice that due to the equations, we actually get a class of transitions containing $t_0||t_1$, $t_0;t_1$ and $t_1;t_0$, represented as $t_0||t_1$ in the figure.

3 Transactions in UML Diagrams

In order to correctly introduce the notion of transactions, we need to analyze the UML official documentation. The UML specification by OMG [6,7] posses a semi-formal semantics, composed by a set of metalanguage, restrictions and text in natural language. The metalanguage is basically a set of class diagrams which describe the basic building blocks of UML models (it can be seen as the abstract syntax of the language). The Object Constraint Language (OCL) further defines constraints over models so they can be considered well-formed.

In our approach, a basic set of metamodel elements is selected. The idea is to focus only on constructs for exposing the behavior (to be understood as a sequence of observable actions) of software artifacts. From this set, we extend the metamodel with elements denoting atomic composites. The graphical notations for the new composites are based on the nonatomic ones and are further decorated with proper stereotypes. Also, new OCL expressions are built to define the new constraints over atomic compositions. One example of a new constraint for the atomic composite state in sate diagrams is the one that does not allow internal states to be interrupted by explicit external events. Finally, the

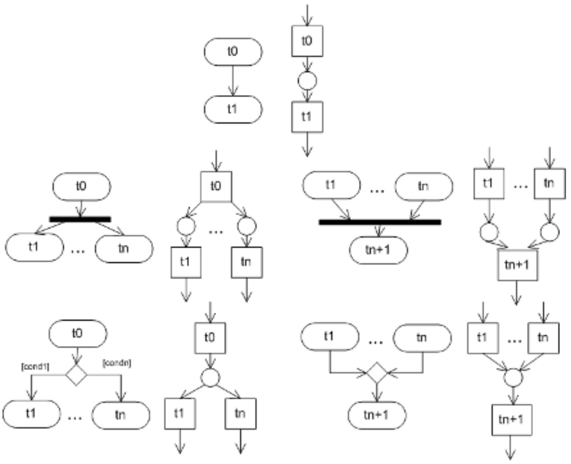


Fig. 2. Semantic mapping examples

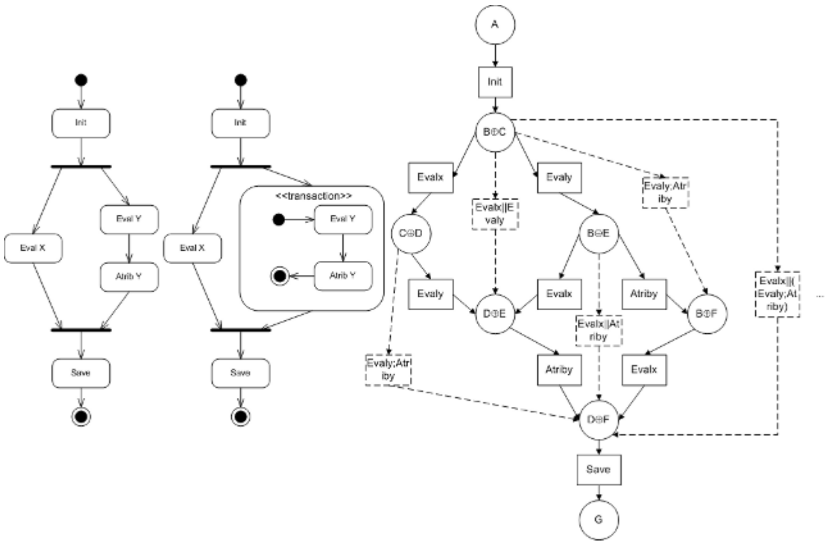


Fig. 3. UML activity diagram without (left) and with composite state (center) , and nonsequential automaton for its semantics (right)

well-formed models are mapped to nonsequential automata, thus formally defining its semantics. We define one new atomic composite for activity diagrams, state diagrams and sequence diagrams, but due to space limitations, our discussion and working example are based only on activity diagrams. A similar approach for state and sequence diagrams have been employed.

Activity diagrams are one of the means for describing behavior of systems within UML focused on the flow of control from activity to activity. The most basic node is the action node, which represents an atomic action. Activities are represented by nonatomic composites of sequential or concurrent actions/activities. The control flow is described by special nodes as fork/join for concurrency, decision/merge for alternative paths of execution and initial/final nodes. Our working example (figure 3 - left) depicts a simple activity diagram for a sequence of operations in a pseudo programing language. Suppose we are interested in defining the sequential sequence of actions "Eval Y" and "Attrib Y" as atomic. To overcome the lack of an atomic activity composite, we introduce a new notation based on the idea of atomic transaction. The new composite activity is decorated with the stereotype $<< transaction >>$ as depicted in figure 3 (center).

The semantics for activity diagrams take into account the fact it comprises a token game similar to Petri nets. So, the semantic mappings from activity diagrams into nonsequential automata are targeted into constructing local transitions for a nonsequential automaton. Before applying the mapping we need to transform the activity diagram in such a way each action node has only one incoming/outgoing edge. We do this as a precaution to avoid misinterpretation of activities control flow because implicit merging/joining of edges has changed from previous UML versions. Each action node consumes/produces control tokens as the steps of computation progresses through the activity diagram. For nonsequential automata, this semantics belongs to transitions. Thus, each action node corresponds to a nonsequential automaton transition, whose origin denotes the necessary tokens for its firing, and whose destiny denotes the tokens produced after its firing. Edges and control nodes are mapped to a consistent set of nonsequential automaton states according to its purpose. Figure 2 depicts the resulting states and transitions in a nonsequential automaton.

The central core of the composite transaction node makes use of nonsequential automata refinement. The source automaton corresponds to the basic translation using the previous mappings, where the composite node is viewed as only one nonsequential automaton transition. The target automaton corresponds to the translation taking into account the subactivity nodes of the composite. The refinement then maps the more abstract transition into the concrete implementation of the transaction obtained via the computational closure of the target automaton. Figure 3 partially depicts the target nonsequential automaton for our working example of activity diagrams. Notice it explicits all possible computational paths, including the transaction state represented by the atomic sequential composition $Evaly; Atriby$.

4 Concluding Remarks

We believe transactions are an important part of today systems and they deserve a first class mechanism in modeling languages, especially UML. Following that premise, this work presented an extension to UML diagrams centered on constructions for defining atomic composition of actions/activities/operations. Its semantics were defined as nonsequential automata refinement morphisms.

Other approaches to translating UML diagrams into formal models have been based on Petri nets [8]. For example, [9] describes a formal translation of activity and collaboration diagrams into place/transition Petri nets and [10] compares different proposals for the semantics based on Petri nets. Also, other works have used formal methods to verify the behavior of UML specifications [11,12]. The main differences between this proposal and related works may be summarized as follows: we are based on the UML 2.0 specification, in which activity diagrams have been decoupled from state diagrams; the applied semantic domain is compositional, in contrast to domains based on Petri nets or statecharts semantics; we are dealing with mechanisms for atomic compositions and not just nonatomic composites.

References

1. Rumbaugh, J., Jacobson, I., Booch, G.: The Unified Modeling Language Reference Manual. 2 edn. Addison-Wesley (2004)
2. Menezes, P.B., Costa, J.F.: Compositional reification of concurrent systems. Journal of the Brazilian Computer Society **2** (1995) 50–67
3. Menezes, P.B., Costa, J.F., Sernadas, A.S.: Refinement mapping for general (discrete event) system theory. In: Lecture Notes in Computer Science - 5th International Conference on Computer Aided Systems Theory and Technology. Volume 1030., Springer-Verlag (1996) 103–116
4. Meseguer, J., Montanari, U.: Petri nets are monoids. Information and Computation **88** (1990) 105–155
5. Machado, J.P., Menezes, P.B.: Modeling transactions in uml activity diagrams via nonsequential automata. In: Actas de la XXX Conferencia Latinoamericana de Informatica, CLEI (2004) 543–553
6. OMG: Uml 2.0 superstructure ftf. Technical Report ptc/04-10-02, Object Management Group (2004)
7. OMG: Uml 2.0 infrastructure final adopted specifcation. Technical Report ptc/03-09-15, Object Management Group (2003)
8. Reisig, W.: Petri Nets: an introduction. Volume 4 of Eatcs Monographs on Theoretical Computer Science. Springer-Verlag (1985)
9. Gehrke, T., Goltz, U., Wehrheim, H.: The dynamic models of UML: Towards a semantics and its application in the development process. Technical Report 11/98, Institut fur Informatik, Universitat Hildesheim (1998)
10. Eshuis, R., Wieringa, R.: Comparing petri net and activity diagram variants for workflow modelling - a quest for reactive petri nets. In: Lecture Notes in Computer Science - Petri Net Technology for Communication Based Systems. Volume 2472., Springer-Verlag (2003) 321–351
11. Shen, W., Compton, K., Huggins, J.: A validation method for uml model based on abstract state machines. In: Proceedings of EUROCAST. (2001) 220 – 223
12. Knapp, A., Merz, S.: Model checking and code generation for uml state machines and collaborations. In: Proceedings of 5th Workshop on Tools for System Design and Verification. (2002) 59 – 64

Theory-Building with System Dynamics: Principles and Practices

Markus Schwaninger[1] and Thomas K. Hamann[2]

[1] University of St. Gallen, Switzerland
[2] McKinsey & Company, Inc., Munich, Germany

Abstract. System Dynamics is a discipline for the modeling, simulation and control of complex dynamic systems. In this contribution, the methodology of System Dynamics-based modeling is argued to be a powerful and rigorous approach to theory-building. The strength of the pertinent process of theory development lies in its high standards for model validation, and in a combination of abductive reasoning with induction and deduction. The argument of the paper is underpinned by an application of System Dynamics to the elaboration of a theory in the new field of Cultural Dynamics.

1 Introduction: Theory-Building in Perspective

Theory-building, in principle, is more than an exercise in academic abstractions. It is an activity fundamental to the survival of societies, organizations and even individuals. Constructing a model, in the sense in which it is used here, consists in building and mathematically formalizing a theory in order to orientate action. It is a device for coping with whatever is complex. As complexity in our time tends to be of a high degree, and often growing, the quest for better theories is a necessity for both academics and practitioners.

Essentially, three different modes of scientific inquiry can be distinguished, namely, deductive, inductive, and abductive [13]. Adopting a deductive research approach entails concluding upon a particular statement derived from theories or laws considered to be universal truths, whereas inductive inquiry involves deriving universal theories or laws from particular observations. Finally, by researching in the abductive mode, possible explanations or interpretations of observed facts are provided, i.e., one generates an understanding of the fundamental driving forces and structures of the phenomenon under consideration. Characteristic outcomes of abductive reasoning are explanatory principles and theories obtained by looking beyond the facts observed in similar cases,thereby taking the longest step of all three modes of scientific inquiry towards the generation of new knowledge. In order to overcome the limitations of each approach, researchers have tried to combine the different modes of theory-building. This is often found to be difficult, or biased in one direction or the other. Hence, there is a need for rigorous theory-building approaches which balance out the trade-off between the quests for genuinely new insights, conceptual stringency, and empirical soundness.

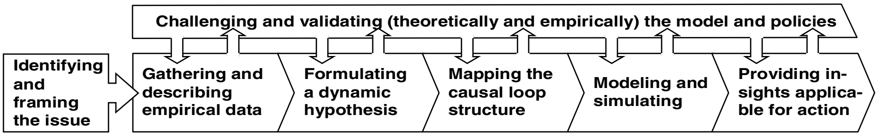
R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 56 – 62, 2005.
© Springer-Verlag Berlin Heidelberg 2005

In this contribution it is demonstrated that the methodology of System Dynamics offers a particularly powerful process and technique for effective theory-building in order to improve decision-making in the context of organizations and society. This methodology is designed to achieve an understanding of the fundamental driving forces and structures underlying a problematic mode of behavior, as well as conceptual and empirical rigor.

2 Theory-Building with System Dynamics

System Dynamics is a discipline for the modeling, simulation and control of complex dynamic systems, founded by Jay W. Forrester [5,6]. A main feature of the SD modeling approach is that the issue modeled is represented by closed feedback loops made up of essentially two kinds of variables – stocks and flows – supplemented by parameters and auxiliary variables. Representation in the form of multiple closed loops, as well as the consideration of delays, enable realistic modeling, which brings the endogenous dynamics generated by the system itself to the fore. Moreover, counterintuitive system behaviors [7] generated in the simulations can lead to important insights for model users.

The methodology of SD is centered around a process which combines modeling and simulation iteratively, thus leading to a continuous improvement of model quality and insights into the domain or issue modeled. Other authors have emphasized the role of modeling as a vehicle for learning, in particular group learning, e.g., Lane [12] and Vennix [17].


Challenging and validating (theoretically and empirically) the model and policies

Identifying and framing the issue	Gathering and describing empirical data	Formulating a dynamic hypothesis	Mapping the causal loop structure	Modeling and simulating	Providing insights applicable for action
Content	• Describing the present situation dynamically • Defining the purpose and the object of the analysis • Formulating the research question(s)	• Analyzing potential causes of the problem dynamically • Generating an initial dynamic hypothesis • Identifying the key elements of the theory/ model to be developed	• Reviewing additional relevant literature • Integrating the isolated theoretical results of previous research to a holistic and consistent theoretical body	• Specifying and documenting the model • Specifying various scenarios • Running simulations • Visualizing and analyzing the simulation results	• Defining the criteria for evaluating the policy design • Assessing alternative policies • Deriving normative recommendations on policy implementation
Intermediate/ end products	• Reference behavior pattern (set of graphs) • Purpose statement • Research question(s)	• Dynamic hypothesis • High-level system diagram	• Causal loop diagram	• Stock and flow map • Computer simulation model • Report of simulation results	• List of criteria for policy evaluation • Action plan

Source: Own representation following High Performance Systems, Inc. (1994) and Sterman (2000)

Fig. 1. Ideal-typical scheme of an SD-based theory-building process

We take a new view by conceiving of modeling and simulation as a powerful approach to theory-building. Figure 1 depicts an ideal-typical scheme of that process. Even though this is a general scheme, the process represented therein is essentially a theory-building process with the sequence of formulating a proposition, then testing it, expanding or refining the proposition, and proceeding with further tests, etc.

The starting point is a framing of the issue at hand, including a rough definition of the scope and purpose of the model to be developed. The ensuing collection of empirical data arranged via a first view of reference patterns then supports the clarification of the goals and the formulation of the research questions to be answered. Proceeding from this, a dynamic hypothesis can be formed which explains the unfolding of the reference behavior pattern over time. Besides empirical data, this dynamic hypothesis is also based on theoretical concepts and constructs which result from previous research efforts. The core of the theory-building process thus consists in elaborating a theory by drawing on that dynamic hypothesis as well as testing, corroborating or refuting it. Model quality is successively enhanced and explanation deepened along the path of this iterative process.

In the following, the theory-building along the various stages of this process will be illustrated by instancing the generation of a holistic and consistent theory about the development of individuals' musical tastes, especially their preference for classical music.

3 Application: The Case of Cultural Dynamics

Identifying and framing the issue: For the long-term success of all kinds of enterprises – and hence also orchestras - it is crucial to discover and take opportunities as early as possible as well as to detect and avoid potential threats before they become uncontrollable. Therefore, the anticipative capturing of future realities by interpreting weak signals of external developments is important and might even be critical for the long-term survival of an organization [1].

Various samples of relevant data collected by Hamann [8] suggested that the classical music audience in Germany and Switzerland has a disproportionate number of elderly people when compared to the population as a whole (significance level = 0.001). This is a weak signal. In order to determine whether it indicates a potential threat to German and Swiss orchestras, the causes for the extremely high proportions of elderly people in classical music audiences needed to be understood properly. Unfortunately, the implications of the weak signal for the future size of such audiences and the resulting demand for live performances of classical music remained unclear, since there was no theory about the formation of individuals' affinity with classical music.

Gathering and describing empirical data: Further analyses of time series of relevant data published by the Institut für Demoskopie Allensbach [10] suggest that the proportion of people older than 59 years of age among those who frequently listened to classical music increased between 1994 and 2002 by 47 percent (from 31.8 to 46.8 percent of the total). During the same time-span, the number of classical music listeners relative to the number of pop music listeners decreased by 9 percent. Therefore, the research question to be answered by the study was: “Why is the classical music audience aging faster than the population as a whole, and why is it decreasing in size?”

Formulating a dynamic hypothesis: As one possible explanation for the behavior pattern recognized, the following dynamic hypothesis was formulated: The development of basic musical taste regarding the various general types of music, e.g., classical music, jazz, pop/rock music, folk music, etc., takes place in a socialization phase during adolescence.

After that phase, the basic musical taste of an individual remains more or less the same into later life – apart from a negligibly small number of people changing genre. With pop and rock emerging in the 1950s and 60s, young people have increasingly been socialized under the influence of these new types of music. Consequently, the proportions of classical music listeners have been falling from a high level with each succeeding younger cohort,which is due rather to genuine cohort differences in participation than to any function of demographic and life-stage factors. This means that the classical music audience will dwindle and die out if no appropriate counteractive measures are taken very soon.

Mapping the causal loop structure: In the next stage of the SD-based theory-building process, the results of previous research relevant to the question as to how music preferences develop were reviewed. The existing research results were thoroughly tested for inconsistencies. Since hardly any contradictions could be identified, the isolated theoretical results of previous research were put together like pieces of a puzzle,finally adding up to a holistic and consistent body of theory. We concluded that the extent to which activities with musical relevance are put into practice (repeatedly listening to classical music, playing an instrument, and attending appreciation classes) during the socialization phase in an individual’s adolescence determines his or her fundamental musical orientation with regard to classical music in later life. The reason is that such activities enhance the development of “listening competence”, i.e., what Behne [3] calls the “cognitive components (‘concentrated’ and ‘distancing’)” of listening. This theory on the development of individuals’ basic musical taste was represented by means of a causal loop diagram [8].

Modeling and simulating: An SD model is a mathematically formalized version of a theory. According to Diekmann [4], there are several reasons for modeling quantitatively: First, it conduces to higher precision of the theory, e.g., by specifying the connections between variables as algebraic functions. Secondly, hypotheses can be derived mathematically from formalized theories by which new and surprising insights are often gained. Thirdly, a model allows of testing the theoretical assumptions for inconsistencies in a more stringent fashion and facilitates checking the deduction for errors. Therefore, the theory developed so far was specified as a quantitative model, and many simulations were run. The simulations clearly corroborated the dynamic hypothesis. In addition, deeper insights into the issues under study were gained, which enabled the elaboration of well-founded recommendations for the management of orchestras.

Challenging and validating the model (theoretically and empirically): In theory-building, the quality and robustness of the theoretical propositions developed, i.e., "scientific rigour", should be the principal concern. We are taking Karl Raimund Popper's [14] logic of scientific discovery – essentially a concept of an evolutionary progress of science – as a benchmark for the design of the theory-building process. This implies that any proposition must be formulated in such a way that it can be disproved if confronted with reality. In other words: it must be proposed in such a way that it can be falsified. The reason for this demand for refutability is that science is advanced by bold propositions or guesses to be subjected to a barrage of criticism. Only hypotheses capable of clashing with facts are regarded as scientifically legitimate.

Thus, every single equation of the model, i.e., propositions regarding causal relationships, had to be carefully examined by drawing on additional theoretical and empirical data. The ability of the model to reproduce the reference behavior pattern is not sufficient. Moreover, as Barlas [2] expresses it, "a system dynamics model must generate the 'right output behavior for the right reasons'", i.e., the internal structure of the model has to be valid as well. Hence, the model structure was tested by comparing the model structure with the knowledge about the structure of the real system (direct structure tests) as well as by testing the behavior patterns generated by the model (indirect structure tests). In a concrete example of a structure test, the proportion of women of child-bearing age (i.e., between 15 and 45) was assumed to be approximately constant over time. Empirical data provided by the Federal Statistical Office in Germany revealed that this proportion actually remained within the very narrow range between 48.5 and 48.9 percent during the time-span from 1978 to 2002. Therefore, the so-called parameter-confirmation test (as part of the indirect structure tests) was considered to have been passed.

Finally, the behavior replication tests were not applied before each of the various forms of structure test had been passed. One of the typical tests in this category was a comparison of the time series based on the statistical data about the evolution of the German population in the different age brackets, with the simulated values based on the SD model. The difference between the two is represented in the following formula:

$$D = \int_{t=1980}^{2000} |a(t) - s(t)|\, dt \cdot$$

where a is the actual development, s designates the simulated results and D a measure for the divergence between the two. The subsequent aim was to find a value for $s(t)$ for which

$$D = D_{\min} = \min_{s} \int_{t=1980}^{2000} |a(t) - s(t)|\, dt \cdot$$

The test was regarded as having been passed only when the difference was no longer significant.

Modes of scientific inquiry applied: In the first two stages of SD-based theory-building, the dominant approach is inductive research: First, from particular observations limited to the sample size the age structure of all classical music listeners in Germany and Switzerland respectively, an imbalance was inferred. Secondly, the time series analyzed covered the period from 1994 to 2002. The formulation of the (initial) dynamic hypothesis clearly puts it into the category of abductive research: Eventually, the observed facts were interpreted and explanatory principles obtained by looking beyond the data, which is the very essence of the abductive approach. Finally, integrating the existing theoretical research results into a consistent theory is a deductive process. This is because particular statements regarding the development of an affinity with classical music are concluded from existing theories which are considered to be universally true. In turn, these theories were accepted as universal truths after they had been inferred inductively from particular empirical observations and had withstood a variety of attempts at falsification. This makes it clear that the different modes of scientific inquiry are inextricably bound up together when one is generating theories based on SD methodology.

4 Conclusions

Theory-building is more than an exercise for academics. It is also an indispensable device for practitioners in organizations, allowing them to test their assumptions and bring their speculations down to earth in order to make better decisions. That is why theory-building is a fundamental prerequisite for effective action.

The SD methodology is a powerful and rigorous approach to the development of theories. This is underpinned by its exceptionally high validation standards: Bold guesses, i.e. abductive theory-building, first crystallize in theory, the model then being submitted to numerous tests. Among the methodologies for the modeling of social systems, none, as far as we know, has validation standards as strict as those for SD. For instance, econometrics operates essentially with statistical validation procedures. In SD, the standard procedure for model validation also involves statistical tests, e.g., the comparison of time-series of data representing the object system versus those generated by the simulation. In order to avoid a model's being considered right for the wrong reasons, SD validation includes a whole set of obligatory procedures designed to build up confidence in a model [15]. The abductively acquired elements of the theories therefore do not remain merely speculative, without empirical corroboration.

We have reported an application of SD to the construction of a theory of Cultural Dynamics, from which substantial insights and recommendations for the management of cultural institutions have been derived. Other cases in point have already been published, e.g., Ulli-Beer [16] and Kopainsky [11].

The SD methodology for modeling, simulation and control is in line with the concepts of evolutionary theory-building as proposed by the theory of science. It must be added, however, that it is also highly appropriate for applications, owing to its intuitive techniques and the user-friendly software available.

Summing up, one may say that the potential of SD as a methodology for theory-building is exceedingly high.

References

1. Ansoff, I. (1975): Managing Strategic Surprise by Response to Weak Signals. In: California Management Review, Vol. 18, No. 2, pp. 21-33.
2. Barlas, Y. (1996): Formal Aspects of Model Validity and Validation in System Dynamics. System Dynamics Review, Vol. 12, No. 3, pp. 183 210.
3. Behne, K.E. (1997): The Development of “Musikerleben” in Adolescence: How and Why Young People Listen to Music. In: Deliège, I. & Sloboda, J. (eds.): Perception and Cognition of Music. Hove: Psychology Press.
4. Diekmann, A. (2000): Empirische Sozialforschung: Grundlagen, Methoden, Anwendungen. Reinbek bei Hamburg: Rowohlt Taschenbuch Verlag.
5. Forrester, J.W. (1961): Industrial Dynamics. Cambridge, Mass.: The MIT Press.
6. Forrester, J.W. (1968): Principles of Systems. Cambridge, Mass.: The MIT Press.
7. Forrester, J.W. (1971): Counterintuitive Behavior of Social Systems. In: Technology Review, Vol. 73, No. 3, pp. 52-68.
8. Hamann, T.K. (2005): Cultural Dynamics: Zur langfristigen Existenzsicherung von Kulturorchestern in Deutschland und der Schweiz. Bamberg: Difo-Druck, doctoral dissertation of the University of St. Gallen.
9. High Performance Systems, Inc. (1994): STELLA II: An Introduction to Systems Thinking. Lebanon, N.H.: High Performance Systems, Inc.
10. Institut für Demoskopie Allensbach (1994-2002): AWA: Allensbacher Marktanalyse/Werbeträgeranalyse. Allensbach: Institut für Demoskopie Allensbach.
11. Kopainsky, B. (2005): A System Dynamics Analysis of Socio-economic Development in Lagging Swiss Regions, Aachen: Shaker-Verlag.
12. Lane, D.C. (1994): Modeling as Learning: A Consultancy Methodology for Enhancing Learning in Management Teams. In: Morecroft, J.D.W./Sterman, J.D. (eds.): Modeling for Learning Organizations. Portland, Oregon: Productivity Press, pp. 85-117.
13. Peirce, C.S. (1878): Deduction, Induction and Hypothesis. In: Popular Science Monthly, No. 13, pp. 470-482.
14. Popper, K.R. (1959): The Logic of Scientific Discovery. New York: McGraw-Hill.
15. Sterman, J.D. (2000): Business Dynamics: Systems Thinking and Modeling for a Complex World. Boston, Mass.: Irwin McGraw-Hill.
16. Ulli-Beer, S. (2004): Citizens' Choice and Public Policy : a System Dynamics Model for Recycling Management at the Local Level, Ph.D. dissertation No. 2918, University of St. Gallen, Switzerland.
17. Vennix, J.A.M. (1996): Group Model Building: Facilitating Team Learning Using System Dynamics. Chichester: Wiley.

Ontology Integration for Statistical Information

Wilfried Grossmann and Markus Moschner

WG Data Analysis and Computing, Dept. of Computer Science,
University of Vienna, A–1010 Vienna, Austria

Abstract. Metadata may be used for convenient handling of statistical information. Thus some metadata standards have emerged as guiding lines for information processing within statistical information systems. Another development stems from documentation development for data archives where metadata requirements of researchers in social sciences are considered. Different metadata standards have emerged out of these streams of science. Basic ideas on integration and translation between such different lines of development are given. Hereby principles of ontology engineering play a key role as starting point.

1 Introduction

Statistical information plays a central role in many business and economic decisions. The term information means here that we have to consider the data itself as well as descriptions of the data, so called metadata, which are necessary for obtaining the information. Hence, it is not surprising that metadata play a key role in statistical information systems for a long time. The earliest reference is Sundgren ([13]), who introduced the concept of metadata in statistics for a so called infological approach towards statistical information. This approach has been developed further in many ways by a number of researchers as well as statistical agencies and has lead to a number of metadata standards. Due to the fact that a lot of statistical information is contained in highly aggregated data, represented nowadays usually in data warehouses, one line of development focussed on metadata standards for such type of data (for example the SDDS-standard [11]). A second stream of development based on ideas from documentation for data archives considered mainly the metadata requirements for scientific researchers in social sciences and economics. These efforts have resulted in different metadata standards, probably the best known example is the DDI standard [3], which is a substantive expansion of the Dublin Core metadata [2]. A well known software tool based on these ideas is NESSTAR [8]. A further development concentrated on proper metadata representation for value domains of the attributes of interest, resulting in the so called Neuchatel Terminology [9] for classifications.

Due to the different starting points of these approaches it is rather cumbersome to integrate data in cases where the definition of the data schemes is based on such different documentation schemes. Following the developments of

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 63–68, 2005.
© Springer-Verlag Berlin Heidelberg 2005

intelligent information processing in recent years the field of statistical information processing has seen a number of efforts to develop the idea of metadata further into the direction of ontology (see for example Sowa [12]). In fact, statistical metadata practice includes to a far extent information needed for ontology - engineering. Probably the most important contribution in this direction was made by the METANET [5,7] project, a thematic network within the fifth EU-research framework. The approach tries to fulfil the requirements of the ontology definition of Gruber [6] (”ontology” is a specification of a conceptualization) by formalizing the statistical data paradigm, taking into account the representational, operational and functional semantics of statistical information.

Starting points are basic statistical objects like dataset, statistical population, or statistical variable, which constitute the categories for the ontology. For all these categories a unified description framework was developed, which is called the view facet of statistical categories. The following four views were distinguished:

- The conceptual category view represents the subject-matter definition of any category instance and builds the bridge to reality. Validity of the definition of the subject matter concept gets usually restricted by temporal and geospatial constraints.
- The statistical (methodological) category view describes the statistical properties of the category instance by using a number of parameters, which have to be defined in such way that specific properties of the different categories are taken into account.
- The data management category view is geared towards machine supported manipulation storage and retrieval of the category instance data.
- The administrative category view addresses management and book-keeping of the structures.

Based on these view facets a representation scheme can be defined, which seems to be sufficient for operational purposes. A first sketch of such a model was presented in Denk et al. [4]. In this paper we present first and fundamental ideas for using this framework for mapping different metadata standards.

2 Methodologies

Though scientists speak different languages there is still communication and consent on subjects in question possible. Different metadata standards are only partially the consequence of unintelligibility on research subjects and basic formulations. There is still enough communication about such differences possible. Here we concentrate only on such standards which basic principles and formulations can be systematically treated by humans from a bird’s eye view. Only from such a unifying treatment formalizations are tackled, as there are knowledge representations, order sorted algebras, data types, mathematical approach(es) and statistical notions (units, population and statistical variable). In the following the main issues are given in systematic order.

2.1 Fundamental Concepts

All metadata standards can get formulated by a common basis of fundamental concepts (foundational basis). There may be different ways to do this, yet one particular way is chosen. Such fundamental concepts can be arranged like in formal ontologies. That means that there are atomar notions within a partial order where such an order means something like sub- or superconcept. Set-like operations (join, meet and complement) get induced by such an "order". The used vocabulary comprises attribute-like properties and restricted quantifications like in description logic ([10]). Herewith the basis for the intended fundamental concepts get formulated such that differences and relevant properties get included. It will not be realistic to aim at a world knowledge nor will some technical issues (following below) be settled. For special cases even different ontological approaches might be chosen.

The aforementioned basic statistical objects (population, units and variables) have to fit into the chosen conceptualization.
Concepts are described by basic notions and binary relations between these notions. Hence a set-like formulation for a fundamental structure lies at hand $K =< BNotion, BRel, Subsump, Meet, Join, Compl, Null >$, a pool Var of Variables (there might be more than one sort), properties of elements of K analogous to predicates (roles) and quantifiers in description logics([10]).

2.2 Order Sorted Algebras and Data Types

Til now only abstract concepts have been considered. Handling of values as numbers is one part of statistical information processing. Analogous to programming languages specification of data types with concrete value domains is mandatory. There are data types like numbers and strings which do not share much or even nothing (from a conceptual viewpoint). Furthermore some data types form (nontrivial) order sorted algebras. One example are natural, integer, rational and real numbers where operations are also extended. Intervals will play a prominent role with respect to numbers. The possibility of (domain) restrictions need the concept of attributes - monadic predicates which may be used as generators for new sorts. One may distinct between sub- and supersorts with respect to the partial order of the conceptual basis and sorts associated with subset properties.

A datatype $D =< DDom, DPred >$ represents a domain with predicates defined on it (operations have to be formulated as some sort of equations - what seems rather naturally for most cases). For each datatype we have an instantiation (or: intended interpretation) DInt which maps D into some grammatical structure DStr.

2.3 Formalization of Mathematical Concepts

There is a strong relation between sorts of formal mathematical content and data types. Formalization brings a large body of ordered sorts. It is not always necessary to have a correspondence between formal sorts and ontological concepts.

Formalization gives (semi-)automatic processing of mathematical statements, thus only basic or important mathematical notions need a correspondence to ontological concepts. Since there is (and will be) a versatile variety for this discipline([1]) reinventions should be avoided.

2.4 Statistical Variables

A statistical variable SV is a (partial) mapping from K into instantiations $DsStr$ of datatype domains.

There is not only ontological meaning behind. Statistical variables give also concrete values for abstract notions. In that sense they play a central role in being a bridge between abstract notions and value domains: Thus it is legitimate to see them as interpretations of metadata standards into the available order sorted algebras. The statistical notions of unit and population get hereby a determination with concrete values.

2.5 Specification for Data Repositories

Hereto belong merely technical specifications like data base or storage organization. Detailed inspection of data management issues is beyond the scope of this paper.

2.6 Metadata Formalization

Concrete metadata standards get reformulated pertaining to the fundamental concepts. The original formulation of metadata standards gets simply mapped onto ontological notions whereas totality is not compulsory. What is likely to appear is that especially for technical notions such a standard might be more fine grained than the used ontology. Either start again with an enhanced foundational basis or construct an according coarse mapping. Such a mapping does not mean to forget such more fine grained standard notions- yet only its aggregation with certain ontological notions. That is not only a matter of lowering the work for ontology construction but sometimes one does not need or has no justification for differentiation between some technical terms. In matters of the ontology these are too near related.

It is to expect that such metadata standards are like taxonomies. At least there has to be a set of TC of concepts which is partially ordered.

If description logics are used for specification then a metadata standard (or a substantial part of it) gets modelled as a T-Box.

2.7 Morphisms Between Metadata Formalizations

When the foundations are established translations between metadata standards may be tackled. Since ontological foundations shall be taken into account such morphisms are not simply between taxonomical standards itself. Yet these are

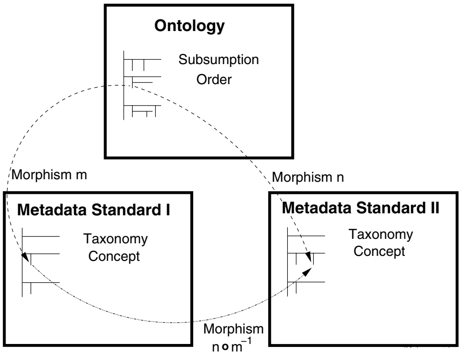


Fig. 1. An arbitrary selection of a translation between metadata standards

between the relations of the fundamental concepts and metadata standards. These morphisms need to be isotone in the sense that the partial order of concepts with respect to subsumption in the foundational basis has to be kept for mapped pairs.

Such a morphism M can be seen technically as subrelation of the support of K and TC. Hereby isotony applies as a basic constraint, yet further may be useful.

The figure exemplifies a simple subcase which would be only a part of a full formalization. Especially for the “metadata standard II” there are two choices. Here experts or other knowledge is necessary for a decision. The morphism between the standards is effectively constructed by use of the fundamental concepts. If done by hand humans play the role of the fundamental concepts. Thus some explication of this activity is demanded here — a task that is too often underestimated.

3 Synopsis

The points of the last section correspond in a certain way to the view facets of statistical categories:

2.1 represents the conceptual category view where matters of knowledge structures are addressed. 2.2-2.4,2.6 and 2.7 comprise statistical approaches as well as concrete object properties. That makes them counterparts of statistical categories. 2.5 resembles the data management and administrative category view.

References

1. Caprotti, O., Dewar, M., Turi, D.: Mathematical Service Matching Using Description Logic and OWL;
   In: Asperti, A., Bancerek, G., Trybulec, A. (eds.): Proc. 3rd Int. Conf. on Mathematical Knowledge Management (MKM 2004), Białowieża, Poland (2004), p. 73–87.
2. DCMI Usage Board, DCMI Metadata Terms, 2003, Dublin Core Metadata Initiative (DCMI), `http://dublincore.org/documents/2003/03/04/dcmi-terms`

3. DDI Data Documentation Initiative — A Project of the Social Science Community–Codebook. `http://www.icpsr.umich.edu/DDI/codebook/index.html`
4. Denk, M. Froeschl, K.A., Grossmann, W.: Statistical Composites: A Transformation Bound Representation of Statistical Datasets.
   In: Kennedy, J. (ed.): Proc. 14th Int. Conf. on Scientific and Statistical Database Management, IEEE Los Alamitos, California/USA (2002), p. 217 - 226.
5. Froeschl, K.A., Grossmann, W., delVecchio, V.: The Concept of Statistical Metadata.
   MetaNet (IST–1999–29093) Work Group 2, Deliverable 5 (2003).
6. Gruber, T. R.: A Translation Approach to Portable Ontologies.
   Knowledge Acquisition, 5(1993), p. 199-220.
7. The *MetaNet* project, `http://www.epros.ed.ac.uk/metanet`.
8. NESSTAR — Networked Social Science Tools and Resources,
   `http://www.nesstar.org`.
9. Neterstrøm, S., et. al.: Neuchâtel Terminology — Classification Data Types, Object Types and Their Attributes. Version 2.0;
   The Neuchâtel Group, Neuchâtel (2002).
10. Pan, J.Z., Horrocks, I.: Web Ontology Reasoning with Datatype Groups;
    In: Fensel, D. et al.(eds.): Proc. of ISWC 2003, LNCS 2870, Springer–Verlag, Berlin (2003).
11. SDDS — Special Data Dissemination Standard, Dissemination Standards Bullentin Bord, `0.http://dsbb.imf.org/Applications/web/sddshome/#metadata`.
12. Sowa, J. F.: Ontology, Metadata, and Semiotics.
    In: Ganter, B., Mineau, G.W. (eds.): Conceptual Structures: Logical, Linguistics, and Computational Issues. LNAI 1867, Springer–Verlag, Berlin (2000).
13. Sundgren, B.: An Infological Approach to Data Bases;
    Urval, nr. 7, National Central Bureau of Statistics (Stockholm, Sweden), 1973.

On Recursive Functions and Well–Founded Relations in the Calculus of Constructions*

J.L. Freire, E. Freire, and A. Blanco

University of A Coruña, Department of Computer Science,
LFCIA, Campus de Elviña, 15071 Coruña, Spain
{freire, rike, blanco}@dc.fi.udc.es

Abstract. This paper presents a reflection about function construction through well-founded recursion in the type theory known as Calculus of Inductive Constructions. It shows the expressive power of this calculus when dealing with concepts as accesibility and noetherianity among others. The properties of the General Recursion Scheme ([2]) and its relation with Structural Recursion in inductive types are analyzed. As a consequence, a methodology arises which is illustrated with some examples.We use the INRIA's compiler of the Calculus of Inductive Constructions: Coq[6].

1 Coq Basics

The Coq logical framework is an implementation of the Calculus of Inductive Constructions (CIC) by G. Huet, T. Coquand and C. Paulin–Mohring, developed at the INRIA. It is a goal-directed and tactic–driven theorem prover where types can be defined by induction. It features a set of predefined tactics, including an `auto` tactic which tries to apply previous proofs. The default logic is intuitionistic but classical logic is also available by importing the `Classical` module.

The system automatically extracts the constructive contents of proofs as an executable ML program. Hence it permits the development of programs consistent with their specification.

The notation `a:A` (`a` is of type `A`) is interpreted as "`a` is a proof of `A`" when `A` is of type Prop, or "`a` is an element of the specification `A`" when `A` is of type Set. Here, Prop and Set are the basic types of the system. By default, Prop is impredicative while Set is not. These two types and a hierarchy of universes Type(i) for any natural i are the elements of the set of sorts. The sorts have the following properties: Prop:Type(0) and Type(i): Type($i+1$).

Allowed constructions are: `x| M N |fun (x:T)=>f|forall x:T,U`, where `x` denotes variables as well as constants; `M N` is the application; the third expression represents the program (λ–expression) with parameter `x` and term `f` as body ($\lambda(x:T) \bullet f$, the abstraction of variable `x` of type `T` in `f`). Finally, the fourth is

---

* Government of Galicia, Spain, Project PGIDT02TIC00101CT and MCyT, Spain, Project TIC 2002-02859, and Xunta de Galicia, Spain, Project PGIDIT03PXIC10502PN.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 69–80, 2005.
© Springer-Verlag Berlin Heidelberg 2005

the type of programs that admit an entry of type `T` and return a result of type `U`. This type is referred to as *product type* and, in type theory, is represented as $\boldsymbol{\Pi}(x : T) \bullet U$ or also as $\forall(x : T) \bullet U$. If `x` is not free in `U`, then this is simply written $T \to U$, the type of the functions between these two types or non-dependent product.

Typing rules provide also proof tactics when reading bottom up. For example:

$$\frac{E[\Gamma] \vdash \forall(x:T) \bullet U : s \quad E[\Gamma :: (x:T)] \vdash f : U}{E[\Gamma] \vdash \lambda(x:T) \bullet f : \forall(x:T) \bullet U}\, Lam$$

expresses that the term (program) $\lambda(x : T) \bullet f$ has the product type $\forall(x : T) \bullet U$ provided that this type has type sort and term f has type U in the environment E and context Γ with the additional hypothesis $x : T$.

Given `forall x:T,U` in Coq, if one looks for some term with that type, the `intro` tactic can be used. This leads to the subgoal `U` for, if we can construct `f` of type `U` then Coq itself builds the term `fun (x:T)=>f` which has type `forall x:T,U` thanks to the *Lam* rule.

1.1 Inductive Types

Under certain constraints, inductive types can be defined in the system Coq and each of them corresponds to an structural induction principle and, possibly, a recursion scheme automatically generated by the system. For instance, the type of natural numbers $\mathbb{N}$ could be defined in a Coq session[1]:

```
Coq < Inductive nat:Set := O:nat | S:nat -> nat.
nat is defined
nat_ind is defined
nat_rec is defined
...
Coq < Parameters P:nat->Set;o:(P O);h:(n:nat)(P n)->(P (S n));n:nat.
Coq < Eval Compute in (nat_rec P o h O).
     = o
     : (P O)
Coq < Eval Compute in (nat_rec P o h (S n)).
     = (h n (nat_rec P o h n))
     : (P (S n))
```

Weak dependent sum is also defined as an inductive type:

```
Coq < Print sig.
Inductive sig (A : Set) (P : A -> Prop) : Set :=
    exist : forall x : A, P x -> sig P
For sig: Argument A is implicit
For exist: Argument A is implicit
Coq < Check sig (gt 0).   (* (gt x y) = x > y *)
```

---

[1] The typewriter font indicates input code and the slanted text corresponds to the output in an interactive session. All input phrases end with a dot.

```
sig (gt 0)
     : Set
Coq < Check {x:nat|(gt x 0)}.
{x : nat | x > 0}
     : Set
Coq < Check exist.
exist
     : forall (A : Set) (P : A -> Prop) (x : A), P x -> sig P
Coq < Parameters (A:Set) (P:A->Prop) (a:A) (H:(P a)).
A is assumed
P is assumed
a is assumed
H is assumed
Coq < Check (exist P a H).
exist P a H
     : sig P
Coq < Check (exist (fun x : A => P x) a H).
exist (fun x : A => P x) a H
     : {x : A | P x}
```

Therefore, given `A:Set` and `P:A->Prop`, the construct `{x:A | P x}` (in abstract syntax `(sig A P)` is a `Set`. We may build elements of this set as `(exist x p)` whenever we have a witness `x:A` with its justification `p:P x`. This type represents also the "constructive existence" of some element which satisfies the predicate `P`.

There also exists the inductive type of the existential quantifier:

```
Coq < Inductive ex (A:Type) (P:A->Prop):Prop:=
Coq < ex_intro : forall (x:A), (P x) -> ex P.
```

2 The Set of Proofs of a Proposition

The type `unit` stands for the inductive type with only one inhabitant.

$$\forall x : unit \bullet x = tt.$$

```
Coq < Print unit.
Inductive unit : Set :=  tt : unit
```

It can be proved that:

```
Lemma ttunit : forall x : unit, x = tt.
```

Let us now define *setof* $(P : Prop)$ as the function that for each $P : Prop$ returns the set of pairs (tt, p) with $p : P$:

```
Definition setof:= fun (P:Prop) => {x_:unit | P}
```

Consequently, a term H of type *setof*(P) (also written in Coq as `sig (fun _ : unit => P)`) is a pair (tt, p) (or `exist P tt p`) where $p : P$.

Fact 1. *There are functions*

$$\begin{tikzcd} P \arrow[r, shift left, "inj(P)"] & \{\_ : unit \mid P\} \arrow[l, shift left, "prj(P)"] \end{tikzcd}$$

inverse one of other[2]*:*

```
Definition prj (P : Prop) (H : {x_ : unit | P}) :=
let (_, p) as H return P := H in p.
Definition inj (P : Prop) (p : P) :=
exist (fun _ : unit => P) tt p.
```

The function *setof* behaves appropriately with respect to the product type:

```
Theorem set_prop:forall P Q:Prop,(P->Q)->(setof P)->(setof Q).
Theorem prop_set: forall P Q:Prop,((setof P)->(setof Q))->P->Q.
Theorem set_pred: forall P Q:A->Prop,(forall x:A,(P x)->(Q x))->(x:A)
(setof (P x))->(setof (Q x)).
Theorem pred_set: forall P Q:A->Prop, (forall x:A,
(setof (P x))->(setof (Q x)))->forall x:A,(P x)->(Q x).
```

3 Well–Founded Relations

Let $\prec$ be a binary relation on a set A. The type $Fin(A, \prec)$ is the set of elements $a \in A$ such that there is no infinite descending sequence $\{a_n\}_{n \in \mathbb{N}}$ verifying

$$\ldots a_{n+1} \prec a_n \prec \ldots a_2 \prec a_1 \prec a. \tag{1}$$

The relation $\prec \subseteq A \times A$ is called noetherian if $A = Fin(A, \prec)$.

Furthermore, given A : *Set* and $\prec \subseteq A \times A$, the concept of *accessibility* can be defined [1] as an inductive predicate: an element $x \in A$ is accessible if every $y \in A$ such that $y \prec x$ is accessible:

$$\forall x : A \bullet \big(\forall y : A \bullet x \prec y \Rightarrow (Acc \ \prec \ y)\big) \Rightarrow (Acc \ \prec \ x) \tag{2}$$

and the relation $\prec \subseteq A \times A$ is *well–founded* if $A = Acc(A, \prec)$.

In the system Coq we represent $\prec$ as `R:A -> A -> Prop`.

```
Coq < Print Acc.
Inductive Acc (A : Set) (R : A -> A -> Prop) : A -> Prop : =
    Acc_intro : forall x : A,
                (forall y : A, R y x -> Acc R y) -> Acc R x
For Acc: Argument A is implicit
For Acc_intro: Arguments A, R are implicit
Coq < Print well_founded.
well_founded =
fun (A : Set) (R : A -> A -> Prop) => forall a : A, Acc R  a
```

---

[2] In classical logic, the irrelevance of the proof implies that there is only one element inhabiting the type $\{x_{\_} : unit \mid P\}$.

```
     : forall A : Set, (A -> A -> Prop) -> Prop
Argument A is implicit
Coq < Print Acc_inv.
Acc_inv =
fun (A : Set) (R : A -> A -> Prop) (x : A) (H : Acc R x) =>
match H in (Acc _ a) return (forall y : A, R y a -> Acc R y)  with
| Acc_intro x0 H0 => H0
end
     : forall (A : Set) (R : A -> A -> Prop) (x : A),
       Acc R x -> forall y : A, R y x -> Acc R y
Arguments A, R, x are implicit
```

Fact 2. *The functions:*

$$\begin{tikzcd}
\forall(y : A) \bullet y \prec x \Rightarrow (Acc\ R\ y) \arrow[r, shift left, "(Acc\_intro\ R\ x)"] & (Acc\ R\ x) \arrow[l, shift left, "Acc\_inv"]
\end{tikzcd}$$

are inverse one of each other[3].

```
Variables (A:Set) (R:A->A->Prop) (B:A->Set).
Variable Phi:forall(x:A),(forall(y:A),(R y x)->(B y))->(B x).
Variable x:A.
Variable h:forall(x:A),forall (y:A),(R y x)->(Acc R y).
Lemma acc_iso1: forall (y:A) (p:(R y x)),
     (Acc_inv (Acc_intro x (h x)) y p)=((h x) y p).
Proof.
 simpl in |- *; auto.
Qed.
Lemma acc_iso2:forall (acc:(Acc R x)),
     (Acc_intro x (Acc_inv acc))=acc.
Proof.
 intros acc; case acc; simpl in |- *; auto.
Qed.
```

We also include the constructive proofs of the following properties: “every minimal element is accessible” and “accessibility is not reflexive”.

```
Definition minimal:=fun (A:Set) (a:A) (R:A->A->Prop) =>
~(ex (fun x:A => R x a)).
Theorem minimAcc: forall (A : Set) (a : A) (R : A -> A -> Prop),
minimal A a R -> Acc R a.
Theorem acc_norefl:forall (A : Set) (a : A) (R : A -> A -> Prop),
Acc R a -> ~(R a a).
```

3.1 Noetherianity and Accessibility

Let us prove now that the notion of noetherianity agrees with that of accessibility.

---

[3] A trivial result in classical logic.

```
Section chains.
Require Import Arith.
Variables (A : Set) (R : A -> A -> Prop).
Definition Desc_seq (s : nat -> A) :=
                  forall i : nat, R (s (S i)) (s i).
Definition Fin (a:A) := forall s : nat -> A, s 0 = a -> ~ Desc_seq s.
Lemma Acc_Fin:(a:A)(Acc A R a)->(Fin a).
Lemma wf_no_Desc_seq:(well_founded A R)->
                       (s:nat->A)~(Desc_seq s).
Theorem not_decreasing:(well_founded A R)->
                  ~(EX s:nat->A | (Desc_seq s)).
Theorem rec_non_decreasing:(EX s:nat->A | (Desc_seq s))
                        -> ~(well_founded A R).
End chains.
```

It is worth highlighting the fact that, only in classical logic it is possible to prove that the non–existence of infinite descending chains guarantees the well–foundedness.

```
Require Classical.
Theorem not_decreasig_inv:~(EX seq:nat->A | (Desc_seq seq))
->(well_founded A R).
```

In the module `Wf_nat` of the system library there is a term `lt_wf` bearing witness to that ($\mathbb{N}$, $<$) is well–founded. On the other hand, in the Cantor space of all infinite sequence of 0 and 1 with the lexicographic ordering $\prec$, the following infinite decreasing sequence can be found: $1000\cdots \succ 0100\cdots \succ 0010\cdots$. Therefore this ordering is not well–founded.

```
Definition Cantor:=nat->bool
Definition R:Cantor->Cantor->Prop:=
fun f g:Cantor => (EX n:nat |
(forall i:nat, (i<n)->(f i = g i)/\(f n < g n)))
Definition s:nat->Cantor:=fun n:nat =>
      fun i:nat => match (eq_nat_dec n i) with
                  |left _ => 1
                  |right _ => 0
                   end.
Lemma inf_dec:(Desc_seq Cantor R s).
Theorem no_well_founded:~(well_founded Cantor R)
```

3.2 The Recursive Scheme

The inductive type `Acc` has the recursive scheme:

```
Coq < Check Acc_rec.
Acc_rec
     : forall (A : Set) (R : A -> A -> Prop) (B : A -> Set),
       (forall x : A,
        (forall y : A, R y x -> Acc R y) ->
```

```
 (forall y : A, R y x -> B y) -> B x) ->
forall x : A, Acc R x -> B x
Arguments A, R, x are implicit
```

With a term Φ of type $\boldsymbol{\Pi}(x : A) \bullet \Big(\boldsymbol{\Pi}(y : A) \bullet (y \prec x) \rightarrow (B\ y)\Big) \rightarrow (B\ x)$, and the notation: $t_\Phi = \lambda(z : A) \bullet \Big(\lambda\big(\_ : \forall(y : A) \bullet (y \prec z) \Rightarrow (Acc\ z)\big) \bullet (\Phi\, z)\Big)$ and $f_{\prec x} : \forall(y : A) \bullet (y \prec x) \rightarrow (Acc\ y)$, the recursive scheme `Acc_rec` has the following well known reduction rule:

Fact 3. $\big(Acc\_rec\ B\ t_\Phi\ x\ (Acc\_intro\ x\ f_{\prec x})\big) =$

$$(\Phi\ x\ \ (\lambda(y : A) \bullet (\lambda(p : (y \prec x)) \bullet (Acc\_rec\ B\ t_\Phi\ y\ (f_{\prec x}\, y\, p))))$$

```
Variable x:A
Variable f:forall(x:A),forall (y:A),(R y x)->(Acc R y).
Variable Phi:forall(x:A),(forall(y:A),(R y x)->(B y))->(B x).
Definition t:=fun (z:A)=>fun (_:forall (y:A),(R y z)->(Acc R y))=>(Phi z).
Lemma fixed_point1:(Acc_rec  B t  (Acc_intro x (f x)))=
(Phi x (fun (y:A) (p:(R y x)) => (Acc_rec B t (f x y p)))).
Proof.
   simpl in |- *.
  auto.
Qed.
```

Let the program $f_{\prec x} : (\forall y : A) \bullet (y \prec x) \rightarrow (Acc\ y)$ be a proof that every "preceding" member of x is accessible, and $h_{\prec x} : (\forall y : A) \bullet (y \prec x) \rightarrow (B\ y)$ a "partial" version of a program – defined only on the elements "preceding" x. Then, the term $t_\Phi : \boldsymbol{\Pi}(z : A) \bullet (\forall(y : A) \bullet y \prec z \Rightarrow Acc\ R\ y) \rightarrow (\forall(y : A) \bullet y \prec z \rightarrow B\ y) \rightarrow B\ z$ allows, for each $x : A$ to obtain a value in $(B\ x)$ and therefore the action of the "complete program" on that element:

$$((t_\Phi\ x)\ f_{\prec x}\ h_{\prec x}) : (B\ x)$$

3.3 Informative Versions of Well–Foundedness

We can define the product type

$$\boldsymbol{\Pi}(B : A \rightarrow Set) \bullet \Big(\boldsymbol{\Pi}(x : A) \bullet (\boldsymbol{\Pi}(y : A) \bullet (y \prec x) \rightarrow B(y)) \rightarrow B(x)\Big) \rightarrow \boldsymbol{\Pi}(a : A) \bullet B(a)$$

and the proposition

$$\forall(B : A \Rightarrow Prop) \bullet \Big(\forall(x : A) \bullet (\forall(y : A) \bullet (y \prec x) \Rightarrow B(y)) \Rightarrow B(x)\Big) \Rightarrow \forall(a : A) \bullet B(a)$$ (well–founded induction principle)

implemented in `Coq` as `wfis` and `wfip` respectively:

```
Definition wfis:=forall B:A->Set,
(forall x:A, (forall y:A, (R y x)->(B y))->(B x))->
forall a:A, (B a).
Definition wfip:=forall B:A->Prop,
(forall x:A, (forall y:A, (R y x)->(B y))->(B x))->
forall a:A, (B a).
```

The Coq standard library includes a module `Coq.Init.Wf` that provides a term (`well_founded_induction`) of type:

$$(well\_founded(\prec)) \Rightarrow \boldsymbol{\Pi}(B : A \Rightarrow Set).$$
$$(\boldsymbol{\Pi}(x : A). (\boldsymbol{\Pi}(y : A).(y \prec x) \Rightarrow B(y)) \Rightarrow B(x)) \Rightarrow \boldsymbol{\Pi}(a : A).B(a)$$

```
Coq < Check well_founded_induction.
well_founded_induction
     : forall (A : Set) (R : A -> A -> Prop),
       well_founded R ->
       forall B : A -> Set,
       (forall x : A, (forall y : A, R y x -> B y) -> B x) - >
       forall a : A, B a
Arguments A, R are implicit
```

We construct a term with the same type as `well_founded_induction`. Let us call it `genrec :forall A:S ,forall R:A->A->Prop,(well_founded A R)->wfis`:

```
Variables (A:Set) (R:A->A->Prop).
Definition genrec:well_founded R -> forall P : A -> Set,
      (forall x : A, (forall y : A, R y x -> P y) -> P x) ->
       forall a : A, P a.
Proof.
   intros wf family wfp element.
   elim (wf element).
   intros; auto.
Defined.
```

and we prove that this program behaves the same as the one in the library.

```
Theorem equiv:
forall wfR:well_founded R,
forall P : A -> Set,
forall Phi:(forall x : A, (forall y : A, R y x -> P y) -> P x),
forall a:A,
(well_founded_induction wfR P Phi a)=
(genrec wfR P Phi a).
Proof.
   case (wfR a);intros;simpl;auto.
Qed.
```

We also construct a proof of the reciprocal theorem:

$$wfis \Rightarrow (well\_founded(\prec))$$

For this purpose, we use an auxiliary lemma that employs the function *setof* as illustrated in the following code:

```
Lemma aux:forall (x:A),(forall (y:A),(R y x)->
(setof (Acc R y)))->(setof (Acc R x)).
Theorem wfp_wf :wfis -> (well_founded  R).
Proof.
   red in |- *.
   unfold wfis in |- *.
   intros H a.
   apply (prj (Acc R a) (H (fun x : A => setof (Acc R x)) aux a)).
Qed.
```

The following proofs are also constructed:

```
Theorem wf_wfip:(well_founded R) -> wfip.
Theorem wfip_wf:wfip -> (well_founded  R).
Theorem wfip_wfis:wfip ->wfis.
Theorem wfis_wfip:wfis ->wfip.
```

4 Results and Methods

To summarize, given a family $B : A \to Set$ of types indexed by $A : Set$, a binary relation $\prec \subseteq A \times A$ with a proof *wfR* of its well–foundedness, then for each

$$\Phi : \boldsymbol{\Pi}(x : A) \bullet \left(\boldsymbol{\Pi}(y : A) \bullet (y \prec x) \to (B\ y)\right) \to (B\ x),$$

the term

$$genrec\ \ A\ \ R\ \ wfR\ \ B\ \ \Phi$$

inhabits the type of all programs which takes an element $a : A$ and returns something of type $B(a)$

$$\boldsymbol{\Pi}(a : A) \bullet (B\ a).$$

This provides a method to build a program on A using the existence of a well–founded relation on it. This is the well known [2] [7] General Recursion method. This represents an alternative way for constructing programs over an inductive type A when the restriction imposed by the syntactic termination test in Coq difficulties straight Structural Recursion.

Balaa, in her interesting PhD dissertation [2], proved that `genrec` satisfies the following fixed–point equation[4]:

Fact 4. $genrec\, A\, wfR\, B\, \Phi\, x = \Phi\, x\, (\lambda(y : A) \bullet (\lambda(\_ : (y \prec x)) \bullet (genrec\, A\ \ wfR\, B\, \Phi\, y))$

```
Variables (A:Set) (R:A->A->Prop)
(wfR:well_founded R)
(B:A->Set)
(Phi:forall(x:A),(forall(y:A),(R y x)->(B y))->(B x)).
Theorem fixed_point2:genrec A R wfR B Phi a =
Phi a (fun y:A=>fun _:(R y a)=>(genrec A R wfR B Phi y)).
```

---

[4] the term *genrec A wfR B* corresponds to Rec_{wf_R} in [2].

Supposing that the domain of a program is an inductive type, we analyze the relation between constructing the program with general recursion and with the alternative structural recursion. This will be illustrated with some examples that show a possible methodology for development.

4.1 The Case of Naturals $\mathbb{N}$

Given a relation `R:nat->nat->Prop` and the hypothesis `wfR:well_founded R`, we can instantiate *genrec* as `N_genrec:=(genrec nat R wfR)`.

Then, given $B : \mathbb{N} \to Set$ y $\Phi : \boldsymbol{\Pi}(x : A) \bullet (\boldsymbol{\Pi}(y : \mathbb{N}) \bullet (R\ y\ x) \to B(y)) \to B(x)$, by defining $N\_0 := (N\_genrec\ B\ \Phi\ 0) : B(0)$ it is possible to construct two terms, h_N and H_N with types $\boldsymbol{\Pi}(x, y : \mathbb{N}) \bullet (R\ y\ x) \to B(y)$ and $\boldsymbol{\Pi}(n : \mathbb{N}) \bullet B(n) \to B(n+1)$, respectively. With them and thanks to the fact 4 it follows that:

Fact 5. $\forall (x : \mathbb{N}) \bullet nat\_rec\ B\ N\_0\ H\_N\ x = N\_genrec\ B\ \Phi\ x.$

Interestingly, fact 4 is used in the proof, and it seems necessary.

```
Section N.
Variable B:nat->Set.
Variable R:nat->nat->Prop.
Variable wfR:well_founded R.
Variable Phi:forall x:nat, ( forall y:nat, (R y x)->(B y)) -> (B x).
Definition N_genrec:=(genrec nat R wfR).
Definition N_0:=(N_genrec B Phi 0).
Definition h_N:forall x y:nat, (R y x) -> (B y).
Proof.
intros.
exact (N_genrec B Phi y).
Defined.
Definition H_N:forall n:nat, (B n)->(B (S n)).
Proof.
intros n H.
cut (forall x y : nat, R y x -> B y).
 intro H0.
 apply (Phi (S n) (H0 (S n))).
 exact h_N.
Defined.
Theorem N_recur : forall x : nat, nat_rec B N_0 H_N x = N_genrec B Phi x.
Proof.
induction x; simpl in |- *; auto.
unfold H_N in |- *.
unfold h_N in |- *.
unfold N_genrec in |- *.
rewrite (fixed_point2 nat R B Phi wfR (S x)).
auto.
Defined.
End N.
```

This proof suggests a general method to construct terms *Φ* and *N_0* that allow us to obtain *h_N* and *H_N*, from which we can define the corresponding program using the left side of the equation in fact 5. In every case we adapt this proof to the concrete problem.

It can be shown that the fact 5 applies also for other inductive types (lists, ordinals, etc.). For reasons of space, only a couple of examples on $\mathbb{N}$ are included.

Example 1. *The type* $\boldsymbol{\Pi}(x : \mathbb{N}) \bullet \{(s, r) : \mathbb{N} \times \mathbb{N} \mid x = s^2 + r \wedge x < (s+1)^2\}$ *specifies the discrete square root. A function of type* $\mathbb{N} \to \mathbb{N} \times \mathbb{N}$*. Therefore, in this case,* $B(n) = \mathbb{N} \times \mathbb{N}$ *for every* $n : \mathbb{N}$*.*

We begin with the corresponding term $\Phi : \boldsymbol{\Pi}(x : \mathbb{N}) \bullet (\boldsymbol{\Pi}(y : \mathbb{N}) \bullet (y < x) \to \mathbb{N} \times \mathbb{N}) \to \mathbb{N} \times \mathbb{N}$.

```
Definition Phi:forall x : nat,
(forall y : nat, y < x -> nat*nat)
-> nat*nat
```

and then:

```
Definition B:=fun _:nat => (prod nat nat)
Fixpoint disc_sqrt (n:nat):(prod nat nat):=
match n with
 |0 => (0,0)
 |(S p) => (H_lt_N B Phi p) (disc_sqrt p)
end.
```

where *H_lt_N* is the *H_N* that corresponds to the relation `lt:nat->nat->Prop`.

Example 2. *To implement the Knuth version of the McCarthy*91 *function, we must take the following relation in* $\mathbb{N}$ *[7]*

$$n \prec m \equiv m < n \wedge n \leq k$$

where k *is a constant greater than 0. In this case, we provide a proof that* $\prec$ *is well–founded by exhibiting directly a term of type* `wfis`*. Then, we define the desired function using* $B(x) = \mathbb{N}$ *for every* x*, and the corresponding Φ:*

```
Inductive R :nat->nat->Prop:=
R1 : forall z n : nat, n < z /\ z <= k -> R z n.
Definition Phi:forall(x:nat),(forall(y:nat),(R y x)->nat)->nat.
Definition B:=fun _:nat => nat
Fixpoint McCarthyK (k0:nat) (n:nat):nat:=
nat_rec B N_R_0 (H_R_N k0 B Phi) n.
```

where *N_R_0* and *H_R_N* are the corresponding terms for the relation R and $k = k0 + 1$.

The source code of the proofs which has not been included for reasons of space, is available in `http://www.dc.fi.udc.es/staff/freire/publications`

References

1. Aczel, P,: Introduction to Inductive definitions in J. Barwise, editor, Handbook of Mathematical Logic. North Holland (1997)
2. Balaa, A.: Fonctions récursives générales dans le calcul des constructions. PhD. Theése. Université de Nice–Sophia Antipolis (2002)
3. Bertot, I., Castéran, P.: Interactive Theorem Proving and Program Development. Springer (2004)
4. Bove, A.: Simple General Recursion in Type Theory. Technical Report. Chalmers University of Technology, Goteborg. (2000)
5. Coquand, T.: An Introduction to Type Theory. Notes of the FPCL summer school, Glasgow. (1989)
6. Dowek, G. et al: The Coq Proof Assistant Reference Manual. V8.0 INRIA 2004
7. Nordström, B.: Terminating General Recursion. BIT, Vol. **28** (1988)

Longest Sorted Sequence Algorithm for Parallel Text Alignment

Tiago Ildefonso and Gabriel Pereira Lopes

Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,
Centre of Informatics and Information Technologies (CITI),
Quinta da Torre, 2829-516 Caparica, Portugal
`{tiago.ildefonso, gpl}@di.fct.unl.pt`

Abstract. This paper describes a language independent method for aligning parallel texts (texts that are translations of each other, or of a common source text), statistically supported. This new approach is inspired on previous work by Ribeiro et al (2000). The application of the second statistical filter, proposed by Ribeiro et al, based on Confidence Bands (CB), is substituted by the application of the Longest Sorted Sequence algorithm (LSSA). LSSA is described in this paper. As a result, 35% decrease in processing time and 18% increase in the number of aligned segments was obtained, for Portuguese-French alignments. Similar results were obtained regarding Portuguese-English alignments. Both methods are compared and evaluated, over a large parallel corpus made up of Portuguese, English and French parallel texts (approximately 250Mb of text per language).

1 Introduction

It is our aim to automatically support the construction of machine translation and cross language information retrieval systems. All over the world millions of texts are translated every day and a lot of them are made public through the web. So, the existence of parallel corpora is not a problem any more. However, such corpora needs to be intelligently used for machines learning how to translate from any language to any other language.

By using this parallel corpora freely available on the web, any system, capable of aligning those texts, i.e., breaking them into smaller parallel text segments (which should be mutual translations of each other), can also automatically extract token and term translations, and feed them into the alignment system, thus reducing the size and augmenting quality of parallel text segments, and output those results to machine translation systems. Another reason for exploring parallel texts on the web is related to the diversity of domains covered by them, and the impossibility to find, most of the time, term translations, in existing manually compiled bilingual dictionaries.

The alignment process proposed in this paper was inspired by the work proposed by Ribeiro et al (2000). As a consequence, they share some properties. Both are language independent. Both use, as starting alignment candidates, equally frequent homographic tokens (acronyms, proper names, country and city names, digits, punctuation and other symbols). Both use alignment candidates, represented by their

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 81 – 90, 2005.
© Springer-Verlag Berlin Heidelberg 2005

positions in parallel texts, to define, through linear regression, the straight line that best fits possible alignment points. Both use a Histogram of Distances for filtering out outliers. Both methods are recursive. As a matter of fact, possible alignment candidates, not used at an earlier stage, because they have different frequencies in the texts to align, may be used later, at another stage, because they have equal frequencies in some parallel segments, obtained meanwhile. Differences occur on the use of the second statistical filter, named Confidence Bands (CB), proposed by Ribeiro et al (2000).

By using the CB algorithm, Ribeiro et al aimed at getting reliable alignment points, which would provide reliable segments. However a finer analysis made apparent problems that were earlier overlooked. This analysis will be subdivided, in this paper, into 5 problems. These problems led us to drop Confidence Bands Algorithm (CBA) and replace it by the Longest Sorted Sequence Algorithm (LSSA). This problems are thoroughly addressed in section 3.All of them contribute to the worst results obtained when we apply the CB algorithm instead of LSSA. Moreover, extra processing time required by CBA is due to two additional factors. First, the calculation of CB's is heavier than the application of LSSA. Last, but not the least, prior to CBA application, the method proposed by Ribeiro et al requires an additional linear regression. LLSA drops this need. So, LSSA is faster and improves the number of parallel segments.

In the next section, we will detail our Longest Sorted Sequence algorithm. At section 3, we will discuss properties of CBA versus LSSA. Is section 4, results are evaluated. In section 5, conclusions are drawn and finally, in section 6, future work is addressed.

2 LSS Algorithm

For illustration purposes, consider the vector of positions of possible aligners in a second language, L2, sorted by the increasing order of corresponding positions in language L1, as depicted at Table 1. Let us name this vector as *L2_array_pos*. Assign a weight to each element of *L2_array_pos*, representing the length of the longest sorted sequence that might end at that element. A new array of weights (*L2_array_weights*), in second line of Table 1, is created, containing in each position the weight of the element in *L2_array_pos* which has the same position. The algorithm takes *L2_array_pos* as an argument and returns another array with the longest sorted sequence, denominated *L2_array_lss*.

Table 1. Illustration of the application of LSS algorithm for selecting aligners

L2_array_pos	1	10	65	100	200	41	45	50	*54*
L2_array_weights	1	2	3	4	5	3	4	5	6
L2_array_lss	1	10				41	45	50	54

The algorithm can be described as follows:

1 – Initialize all weights with 1 (this corresponds to having every possible aligner ending a segment)
2 – For all elements in the array, we compute the corresponding weight. This is done by comparing the value of an element and its current weight with the value and weight of all elements preceding that element. If current element is greater than one of it's predecessors and has a lower or equal weight, then we update the weight of the current element to the weight of it's predecessor added by 1.
3 – Each time a weight is calculated for one element (as described in 2), we keep track of the element which has maximum weight until now.
4 – When all weights are computed, we travel from right to left both in *L2_array_weights* and in *L2_array_pos,* starting in the index of the element which scored maximum weight (as described in 3): the Longest Sorted Sequence consists of picking the next element with weight equal to the weight of the current element minus 1.

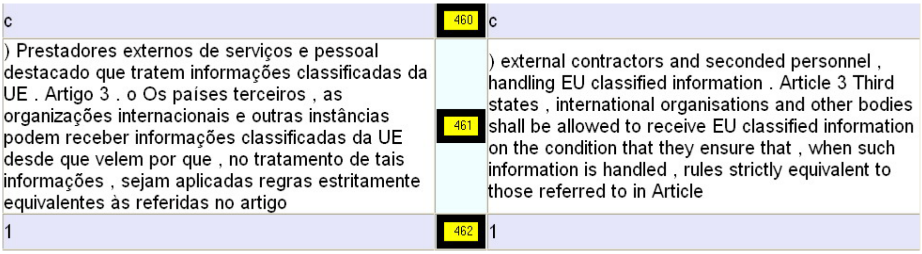
We should notice that, in our implementation, if there are two possible sequences with maximum length, the algorithm selects the rightmost one. From example in Table1, suppose we remove the last element (54) from *L2_array_pos;* then, returned sequence would be [1, 10, 41, 45, 50].

3 CBA vs. LSSA in Parallel Texts Alignment

Having detailed the LSS algorithm, we will now explain how we use it in the process of parallel texts alignment, and also the advantages of LSSA over CBA. Our process of alignment follows Ribeiro et al, with the only exception of replacing CBA with LSSA.

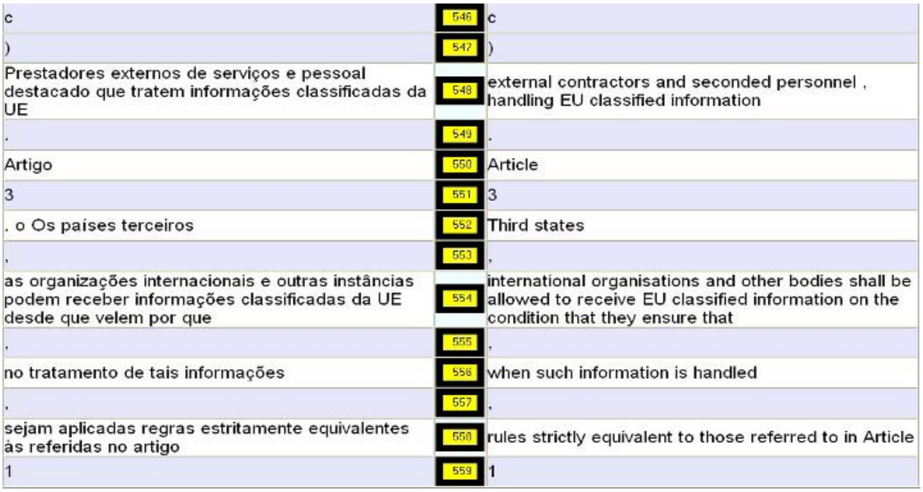
LSSA is applied at the same stage, the same way, and it's objectives remain the same of CBA. After the Histogram Filter has been applied and filtered outlier points, another finer grained filter still needs to be applied in order to obtain more reliable points. We now point out 5 problems which CBA suffers from, and LSSA does not.

Problem 1. Restrictiveness
CBA is too restrictive. It rejects a large number of good alignment candidate points. According to us the best alignment algorithm should satisfy the conditions: *(1) parallel segments must be translations of each other.* This is related with precision. *(2) Segments must have a grain as thin as possible.* This is related with what we might call recall. Comparing results in Fig. 1, obtained using CBA with those of Fig. 2, obtained using LSSA, we notice that CBA refuses 6 good points of alignment. In this case we may say that precision in Fig. 1 is 100%. In Fig. 2, precision is 92,8%, due to an alignment error in segment 552. But recall of LSSA is six times larger than the one using CBA. The thinner the alignment, the easier will be the extraction of word and multi-word term translations.



c	460	c
) Prestadores externos de serviços e pessoal destacado que tratem informações classificadas da UE . Artigo 3 . o Os países terceiros , as organizações internacionais e outras instâncias podem receber informações classificadas da UE desde que velem por que , no tratamento de tais informações , sejam aplicadas regras estritamente equivalentes às referidas no artigo	461	) external contractors and seconded personnel , handling EU classified information . Article 3 Third states , international organisations and other bodies shall be allowed to receive EU classified information on the condition that they ensure that , when such information is handled , rules strictly equivalent to those referred to in Article
1	462	1

Fig. 1. Alignment between "pt_301D0844.txt" and "en_301D0844.txt", using the CBA



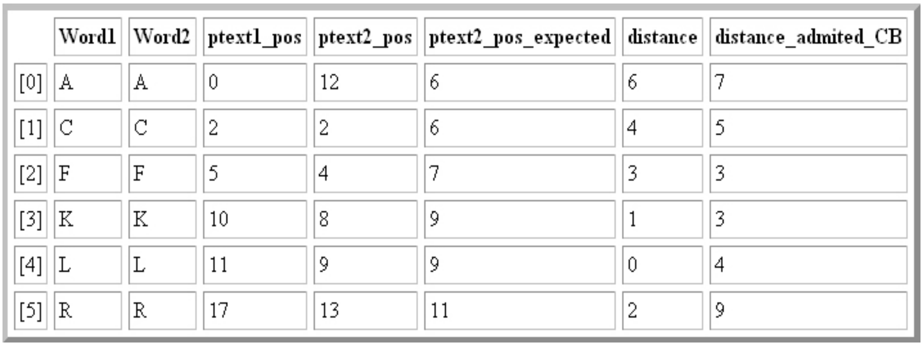
c	546	c
)	547	)
Prestadores externos de serviços e pessoal destacado que tratem informações classificadas da UE	548	external contractors and seconded personnel , handling EU classified information
.	549	.
Artigo	550	Article
3	551	3
. o Os países terceiros	552	Third states
,	553	,
as organizações internacionais e outras instâncias podem receber informações classificadas da UE desde que velem por que	554	international organisations and other bodies shall be allowed to receive EU classified information on the condition that they ensure that
,	555	,
no tratamento de tais informações	556	when such information is handled
,	557	,
sejam aplicadas regras estritamente equivalentes às referidas no artigo	558	rules strictly equivalent to those referred to in Article
1	559	1

Fig. 2. Alignment between "pt_301D0844.txt" and "en_301D0844.txt", using the LSSA

Problem 2. The Aligned Language Disordering Pitfall

The following example illustrates how our algorithm behaves after the application of the histogram filter in comparison with the application of CBA. Confidence Bands determines the maximum admitted deviation from the expected value calculated using the results of linear regression. This is represented in column named "*distance_admitted_CB*" in Fig. 3. The distance between observed and expected positions is represented in column named "*distance*" (difference between values of columns named "ptext2_pos" and "*ptext2_pos_expected*").

For the sake of clarity and simplicity, the example presented bellow is a synthetic one. Yet, it corresponds to observations in real corpora and explains the reason why we get a larger number of segments with LSSA. As already mentioned, our candidate aligners are ordered by their coordinates in "*ptext1_pos*". A segment will be built with all words between one aligner and the next one. Exceptions may occur at the beginning and at the end of texts. If first aligner is not the first text word, first segment will be defined from the first word till the first aligner. Similar reasoning applies to the end of the texts.



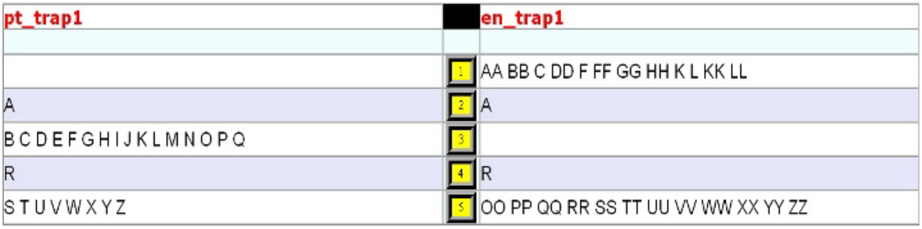
	Word1	Word2	ptext1_pos	ptext2_pos	ptext2_pos_expected	distance	distance_admited_CB
[0]	A	A	0	12	6	6	7
[1]	C	C	2	2	6	4	5
[2]	F	F	5	4	7	3	3
[3]	K	K	10	8	9	1	3
[4]	L	L	11	9	9	0	4
[5]	R	R	17	13	11	2	9

Fig. 3. Table_7_cols

For the base text (in this example, Portuguese) we always have our segments well defined as candidate aligners' coordinates define the sorting criteria. From Fig.3, first segment would be defined from word 0 to word 11, its second segment would be the constituted by the aligner "A" itself (word 12). *But we can not say the third segment will range from word 13 to word 2.* So, any algorithm we use will have to deal with this kind of situation, no matter the other criterions used. Going around this would by force mean we were changing the order of the words in the text, which would make no sense at all.

Observing Fig. 3, we conclude immediately that candidate aligner at position [0] should be discarded. Let us see how LSSA deals with the problem. Fig. 4 represents the results obtained using CBA. Fig. 5 represents the results obtained by using the LSSA. For these results, no recursion was involved. No candidate has been filtered by the Histogram Filter in the previous step of the algorithm. So, for understanding this alignment, we need to look at "*table_7cols*" presented in Fig. 3.

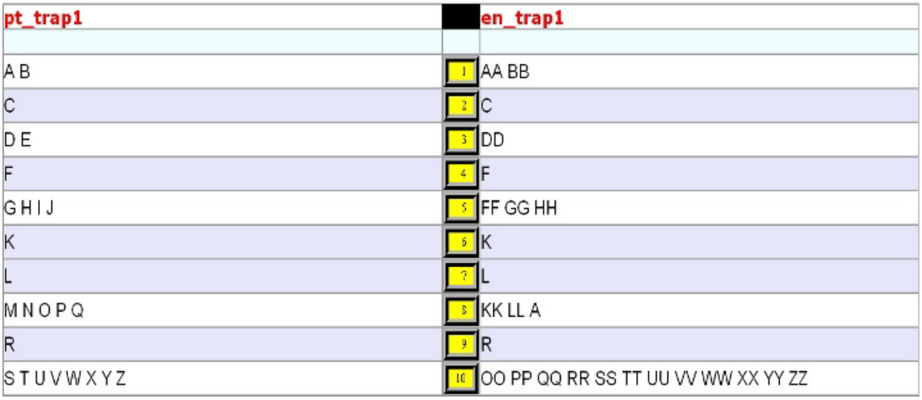
The CB algorithm does not reject any of the candidate alignment points. Looking at last 2 columns of "*table_7cols*", we see that all of them obey the restriction "*distance ≤ distance_admitted_CB*".



pt_trap1		en_trap1
	1	AA BB C DD F FF GG HH K L KK LL
A	2	A
B C D E F G H I J K L M N O P Q	3	
R	4	R
S T U V W X Y Z	5	OO PP QQ RR SS TT UU VV WW XX YY ZZ

Fig. 4. Alignment between "pt_trap1.txt" and "en_trap1.txt", using the CB Algorithm

But the critical issue here is that the Confidence Bands are unable identify the first candidate (word "A") as a bad candidate, which leads to the rejection of candidates [1], [2], [3] and [4], not only because of the filter criteria, but also because the algorithm doesn't deal with unordered candidates. Examples of points like [0] occur in sentences where something is mentioned somewhere in one text, and much later at the parallel text. This kind of occurrences is more or less frequent, depending on the languages involved. Still, we could see how dramatic these situations are, regarding both n.º of segments obtained and quality of those segments. In example above, again, CBA might lead to lower precision and recall.



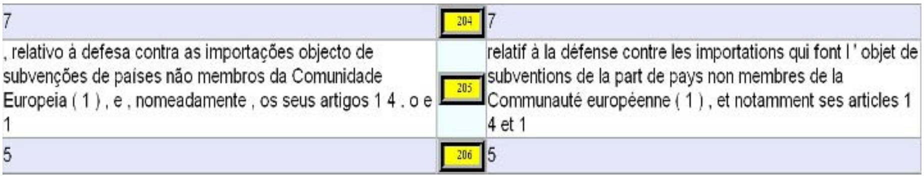
pt_trap1		en_trap1
A B	1	AA BB
C	2	C
D E	3	DD
F	4	F
G H I J	5	FF GG HH
K	6	K
L	7	L
M N O P Q	8	KK LL A
R	9	R
S T U V W X Y Z	10	OO PP QQ RR SS TT UU VV WW XX YY ZZ

Fig. 5. Alignment between "pt_trap1.txt" and "en_trap1.txt", using the LSS Algorithm

Problem 3. CB Model does not Fit the Needs of Parallel Texts Alignment
Previous example shows this clearly. Even ignoring the consequences of CBA being not able to deal with unordered candidates, as we have seen in the previous problem, point [0] ("A") would always be a bad aligner, as it stands far apart in both texts. However, it was accepted by CB filter. Why? Ribeiro et al (2000), refer that "The band is typically wider in the extremes and narrower in the middle of the regression line". And this was exactly the problem in the example presented. If the band is too restrictive in the middle (as discussed in Problem 1), it can also be too permissive in the extremes of the linear regression line.

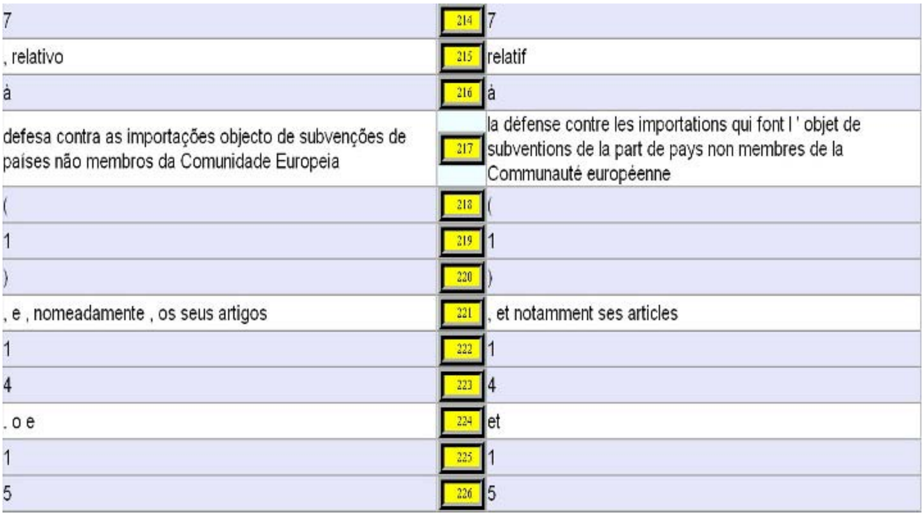
Confidence Bands always accept that good alignment candidates may appear more distant in the beginning/end of the text (although not too distant…), and must occur quite close in the middle of the text. But, our experience in parallel text alignment clearly shows that this does not fit the problem we have to solve. Nothing leads us to be more or less strict with aligners, based only on the fact that they occur in the beginning or end of a text.

Problem 4. Three Candidate Points or Nothing
Consider the PT-FR alignment example at Fig. 6, applying CBA. The LSSA version, in figure 7, splits segment 205 in 11 segments. What happened here? In segment 205, only 2 candidate points are made available by the Histogram Filter to CBA filter. As we can not apply CBA when we have less than three candidate points we have got no further subdivision of the text. From Ribeiro et al (2000), parameter *s* (standard deviation) has a denominator equal to *"n-2"*, where *n* is the number of candidate points.



7	204	7
, relativo à defesa contra as importações objecto de subvenções de países não membros da Comunidade Europeia (1) , e , nomeadamente , os seus artigos 1 4 . o e 1	205	relatif à la défense contre les importations qui font l ' objet de subventions de la part de pays non membres de la Communauté européenne (1) , et notamment ses articles 1 4 et 1
5	206	5

Fig. 6. Piece of alignment from "pt_300R1523..txt" and "fr_300R1523.txt", using the CBA



7	214	7
, relativo	215	relatif
à	216	à
defesa contra as importações objecto de subvenções de países não membros da Comunidade Europeia	217	la défense contre les importations qui font l ' objet de subventions de la part de pays non membres de la Communauté européenne
(	218	(
1	219	1
)	220	)
, e , nomeadamente , os seus artigos	221	, et notamment ses articles
1	222	1
4	223	4
. o e	224	et
1	225	1
5	226	5

Fig. 7. Piece of alignment from "pt_300R1523..txt" and "fr_300R1523.txt", using the LSSA

So, *n* must be identical or superior to 3. LSSA has no restrictions regarding the minimum number of points to accept.

Consequences for CBA are visible in Fig. 6 and Fig. 7. This problem is especially dramatic due to recursivity of the algorithm. Without recursivity it would be impossible to obtain the high recall we are proposing. Moreover we have observed that 25% of the occasions when recursion is called, just one or two points are expected as aligner candidates and these are always rejected by the CB algorithm.

Problem 5. The t_students value (above 120 points)

Ribeiro et al also say that, in respect to the Confidence Bands formulas, they use a t_students value which is *3.27* for "large samples of points (above 120)".But this value should be adjusted dynamically to the number of points we have for each case we're applying the algorithm on. It is obvious that being the algorithm recursive, the majority of the runs of the CB filter will have less than 120 candidate aligners. Even without recursion, some smaller texts won't have 120 candidate points when considering the whole texts.

4 Evaluation

We run our texts on a machine with the following configuration

CPU: Intel Pentium III 933 MHz; **Memory:** 1 Gb SDRAM PC-133; **Hard drive:** IBM 75GXP ATA100 7200rpm; **OS:** Red Hat Fedora Linux Core2

Table 2. PT-EN alignment results

Alignment: PT-EN {15194 documents} {PT=259,EN=258} Mbytes	***CB***	***LSS***	***GAIN***
TOTAL SEGMENTS	18439185	21361199	**+15,8%**
TOTAL TIME (seconds)	344374	258555	**-33,2%**
WORDS/SEGMENT (PT)	2,88	2,49	**-15,7%**
WORDS/SEGMENT (EN)	2,73	2,36	**-15,7%**

Table 3. PT-FR alignment results

Alignment: PT-FR {15210 documents} {PT=258,EN=267} Mbytes	***CB***	***LSS***	***GAIN***
TOTAL SEGMENTS	20534218	24204975	**+17,9%**
TOTAL TIME (seconds)	385298	286204	**-34,6%**
WORDS/SEGMENT (PT)	2,58	2,19	**-17,8%**
WORDS/SEGMENT (FR)	2,75	2,33	**-18,0%**

The corpus used was taken from http://europa.eu.int/, and it consists of documents from the European Parliament, Court of Law, and other European Instances.

In tables 2 and 3, bellow, the number of documents is not the same for both language pairs. Some documents have not been translated in all languages. But we can ignore this. Approximately 99,8% of the total number of documents aligned is common to both language pairs. So, for the purpose of comparison of alignments between language pairs and language pairs similarity, we can consider that the source of both alignments (PT-EN and PT-FR) is the same.

In this paper we evaluate two parameters: processing time and number of segments obtained. Quality of obtained aligned segments must still be thoroughly done. But precision for LSSA in the PT-EN pair is slightly higher than 95%.

The previous two tables hold the evaluation results for both PT-EN and PT-FR alignments. For both pairs of languages, we can clearly see that we obtain considerably larger number of segments (averaging both languages pairs, we get 16,85% more segments), and we reduce the processing time in more than one third (averaging again, LSS computes in less 33,9% of the time).The fact that Portuguese language is more similar to French than to English is also visible. We obtain for the PT-FR pair more 2.1% segments. The word per segment measure gives us an idea of the small granularity we are dealing with.

5 Conclusions

We presented a method for parallel texts alignment that is language independent, uses statistical support and achieves very good performance in the number of segments obtained and the time of processing. As explained above, we have augmented the recall of the alignment procedure and reduced the processing time. We have got more 16.85% of segments in 33.9% less time. The innovation is the use of the Longest Sorted Sequence algorithm instead of the Confidence Bands proposed by Ribeiro at al. This algorithm solves the problems fully discussed in this paper regarding the application of Confidence Bands. Moreover, we have also shown that this new algorithm handles well all the problems we have found when dealing with parallel texts alignment.

An in depth study was made to support our claims, and a very large corpora with 2 distinct pairs of alignment languages was used for evaluation proposes, so one can further rely on the results obtained. As in Ribeiro et al, no heuristics are used. The fact that the aligner is available through the World Wide Web and is usable via any web browser is also a crucial factor. This way, any user can easily submit their texts for alignment.

Moreover, the aligner can be run in batch mode. This is an important feature, because the finely grained segments we obtain, is the basis for the extraction of word or multi-word translations.

6 Future Work

Regarding the results we have obtained just using homograph tokens, we consider that there are still some key features that would greatly benefit our aligner.

1 - Use of possible cognates as candidates to aligners: considering that some authors already stressed that the use of cognates significantly improves the number of good aligner candidates (Danielsson et al, 2000; Ribeiro et al, 2001; Simmard et al, 1992), we plan to include cognates in the alignment process. Due to LSSA features, we expect an increase of the number of alignment points. As this algorithm maximizes the number of accepted aligners, the number of words per segment will decrease dramatically.

2 - In-depth quality evaluation: as previously explained, in this paper, we have not presented an evaluation regarding the quality of the segments obtained by using LSSA.

3 - Use of relevant expressions: As reported by Ferreira da Silva et al (1999), multi-words as “European Parliament” should be considered as textual units. Currently, our aligner does not use this concept. As a consequence it occurs that in “social policy”, “social” aligns with “social” and leaves policy aligning with nothing as well as “política”, leading to precision decrease.

4 - Dropping the restriction of equal frequencies.

5 - Checking the proportionality of segments obtained.

6 - Using more languages for evaluation: We plan to use our aligner with the 20 languages of the European Union, as well as with other languages such as Bulgarian, Chinese, Arabic and other.

7 - Extracting translation equivalents: With a prototype already built, we plan to achieve another application that is able to provide the extraction of word and multi-word translations, using the aligned texts base.

References

1. António Ribeiro, Gabriel Lopes and João Mexia (2000) Using Confidence Bands for Parallel Texts Alignment. In “Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL2000)”, Hong Kong, China, 2000 October 3-6.
2. António Ribeiro, Gaël Dias, Gabriel Lopes and João Mexia (2001) "Cognates Alignment". In: Bente Maegaard (ed.). Proceedings of the Machine Translation Summit VIII (MT Summit VIII), Santiago de Compostela, Spain, September 18-22, 2001. European Association of Machine Translation. pp. 287-292.
3. J.F.Silva, Gaël Dias, Sylvie Guilloré, José Gabriel P. Lopes (1999) "Using Local Maxs Algorithm for the Extraction of Contiguous and Non-contiguous Multiword Lexical Units". In: P. Barahona (ed.) Progress in Artificial Intelligence: 9th Portuguese Conference on AI, EPIA'93, Évora Portugal, September 1999, Proceedings. Lecture Notes in Artificial Intelligence, Springer-Verlag, Vol. 1695, p. 113-132 (1999).
4. Danielsson, P. and Muhlenbock, K. (2000). "Small but efficient: The misconception of high frequency words in Scandinavian translation". In J. White (ed) "Envisioning Machine Translation in the Information Future --- Proceedings of the 4th Conf. of AMTA". Lecture Noten in Artificial Intelligence vol. 1934. p. 158-168. Berlin: Springer Verlag.
5. Simard, M. , Foster, G. and Isabelle, P. (1992). Using cognates to align sentences in bilingual corpora. Proceedings of the 4th International Conference on Theoretical and Methodological Issues in Machine Translation, TMI-92. p. 67-91.

Information Retrieval and Large Text Structured Corpora*

Fco. Mario Barcala[1], Miguel A. Molinero[2], and Eva Domínguez[1]

[1] Centro Ramón Piñeiro, Ctra. Santiago-Noia km. 3, A Barcia,
15896 Santiago de Compostela, Spain
`barcala@freeresearch.org`, `edomin@cirp.es`
[2] Depto. de Informática, Universidade de Vigo,
Campus As Lagoas, s/n, 32004 Ourense, Spain
`molinero@uvigo.es`

1 Introduction

Conventional Information Retrieval Systems (IRSs), also called text indexers, deal with plain text documents or ones with a very elementary structure. These kinds of system are able to solve queries in a very efficient way, but they cannot take into account tags which mark different sections, or at best this capability is very limited.

In contrast with this, nowadays, documents which are part of a corpus often have a rich structure. They are structured using XML (Extensible Markup Language) [1] or in some other format which can be converted to XML in a more or less simple way. So, building classical IRSs to work with these kinds of corpus will not benefit from this structure and results will not be improved.

In addition, several of these corpora are very large and include hundreds or thousands of documents which in turn include millions or hundreds of millions of words. Therefore, there is the need to build efficient and flexible IRSs which work with large structured corpora.

There are several examples of IRSs based on corpora [2] [3], of search methods over large corpora [4], and Chaudhri et al. [5] even introduce a review of different technologies that can be used to build generic IRSs based on XML. However, there are no comparative analyses or studies about technologies that can be used to build IRSs based on large structured corpora.

Since these IRSs can be wide ranging, in this work we will focus on those which work with corpora that do not include any morphosyntactic annotation and are structured in XML format. All topics studied in this paper will also be useful for annotated corpora (although the study need to be completed for the latter) or for corpora without XML format (because if corpora are correctly structured, they can be easily converted to XML format).

---

* Partially supported by Ministerio de Educación y Ciencia (MEC) and FEDER (TIN2004-07246-C02-01 and TIN2004-07246-C02-02), by MEC (HF2002-81), and by Xunta de Galicia (PGIDIT02PXIB30501PR, PGIDIT02SIN01E and PGIDIT03SIN30501PR).

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 91–100, 2005.
© Springer-Verlag Berlin Heidelberg 2005

So, first we will introduce the corpus used to illustrate the study. Next we will present the main alternatives for building IRSs based on large structured corpora. After that we will evaluate two technologies that would seem to be the flagships in this research field, previously defining the needs of that kind of system: on one hand Oracle[1] [6], a Relational Database Management System (RDBMS) that includes XML facilities, and on the other hand, Tamino[2] [7], a native XML indexer. Finally, we will show conclusions and the technology which best fits the established needs.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!ELEMENT document (document_header, document_content)>
<!ELEMENT document_header(identifier,medium,name,publishing_year,publisher)>
<!ELEMENT document_content (section+)>
<!ELEMENT section (name, new+)>
<!ELEMENT new (new_header, new_content)>
<!ELEMENT new_header (identifier, author+, theme+)>
<!ELEMENT new_content (headline?, abstract?, caption*, body)>
<!ELEMENT headline (paragraph+)>
<!ELEMENT abstract (paragraph+)>
<!ELEMENT caption (paragraph+)>
<!ELEMENT body (paragraph+, note*)>
<!ELEMENT paragraph (sentence+)>
<!ATTLIST paragraph distinct (other_language) #IMPLIED>
<!ELEMENT sentence (#PCDATA|note_reference|distinct)+>
<!ATTLIST sentence distinct (other_language) #IMPLIED>
<!ELEMENT note (paragraph+)>
<!ATTLIST note identifier ID #IMPLIED
<!ELEMENT note_reference EMPTY>
<!ATTLIST note_reference reference IDREF #REQUIRED>
```

Fig. 1. Newspaper DTD. Elements not defined are of type #PCDATA.

2 Definition of the Target Corpus

In this work we show technologies, techniques and methods to build IRSs based on large structured corpora which will be illustrated over a real corpus used as example, the CORGA: "Reference Corpus from Present-day Galician"[3]. This, which in its XML version has more than eight million words distributed in hundreds of documents, will also be used to evaluate the technologies studied.

CORGA documents can be newspapers, magazines or books, so we have a DTD (Document Type Definition) [1] for each kind of document. Figures 1 and 2 show the DTDs for newspapers and books respectively[4].

---

[1] Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

[2] Tamino is a Software AG product.

[3] Freely available at http://corpus.cirp.es/corga

[4] Actually, the DTDs used in CORGA are more complex. Here we only show a simplification of them to increase the clarity of explanations.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!ELEMENT document (document_header, document_content)>
<!ELEMENT document_header (identifier, medium, title, author+,
          publishing_year, publisher, theme+)>
<!ELEMENT document_content (preface*, dedication*, cite*, caption*,
          body, appendix*)>
<!ELEMENT preface (header?, content)>
<!ELEMENT header (author+, theme*)>
<!ELEMENT content (head?, caption*, body)>
<!ELEMENT body (part+, note*)>
<!ELEMENT appendix (header?, content)>
<!ELEMENT part (head, dedication*, cite*, caption*, part_body)>
<!ATTLIST part type (chapter|part) #IMPLIED>
<!ELEMENT head (paragraph+)>
<!ELEMENT part_body (paragraph+ | part+)>
<!ELEMENT dedication (paragraph+)>
<!ELEMENT cite (paragraph+)>
```

Fig. 2. Book DTD. Elements not defined correspond to those for newspaper DTD (see figure 1).

3 Building Alternatives

To build IRSs based on corpora several aspects have to be taken into account:

- It is very important to separate the corpus document structure from that of the IRS, that is, corpus document structure cannot condition the IRS one and vice versa, and we have to think of two different systems. In this way, we avoid penalising either the expressiveness of corpus DTDs or system performance.
- With large corpora, a common structure of documents for the IRS has to be designed. Typically there are different kinds of document in the corpus (newspapers, magazines, books, etc.). So, if this variability is maintained in the system, more queries (or more complex ones) have to be made to solve users' queries, and performance will be penalised.
- In these systems the priority is speed in retrieval. Normally these systems about corpora are updated once every three or six months (or never if the corpus is closed), so, in general, it is not important if system updates take several hours or days.

Nowadays two research lines are being followed to build IRSs which work with XML documents. The first one is based on the adaptation of XML documents to the relational model, which are then inserted into a RDBMS. The another one is to introduce XML documents directly into an XML-native or XML-enabled Management System, which allows the documents to be worked on the original XML format.

3.1 Relational Database Management System

There are several ways to represent XML documents in an entity-relation model [5], and in full in a relational model: generic automatic approaches, which allow any kind of XML document to be introduced into a database, or ad-hoc manual ones, which allow only some kinds of XML document to be introduced into it.

Although the second approach force us to define ad-hoc structures and procedures used to introduce our kinds of document into the database, for large corpora it is better to choose this alternative, because it allows us to obtain a higher performance in the retrieval process.

There are several ways to define this ad-hoc structure, and we have to define one which minimises relations in order to improve performance as much as possible. Figure 3 shows the entity-relation model chosen to test this technology.

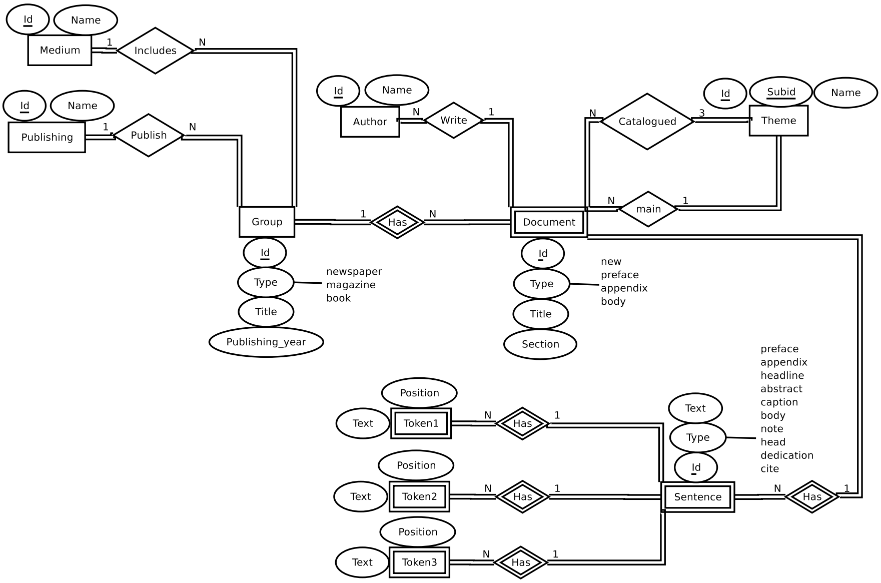


Fig. 3. Entity-relational model

3.2 Native XML Manager

In the case of a native XML manager, it is also necessary to define a common DTD which includes all kinds of document of the corpus. This will avoid a loss of performance associated with the query of different kinds of document, as we have mentioned earlier.

This common structure must also be simple and homogeneous, and have few hierarchies in order to obtain the highest performance, since queries will include fewer structural elements. Figure 4 shows the common XML format DTD chosen to test this technology.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!ELEMENT group (group_header, document+)>
<!ELEMENT group_header (title, type, publishing_year, publisher,
         author*, theme*, main_theme)>
<!ELEMENT document (author+, theme+, main_theme, sentence+)>
<!ATTLIST document type (New|Preface|Appendix|Body) #REQUIRED
          section CDATA #IMPLIED
                  title CDATA #IMPLIED>
<!ELEMENT sentence ((#PCDATA|reference_note|distinct)*, tokens)>
<!ATTLIST sentence type (Preface|Appendix|Headline|Abstract|Caption|
                        Body|Note|Head|Dedication|Cite) #REQUIRED>
<!ELEMENT tokens (token1+, token2*, token3*)>
<!ELEMENT token1 (#PCDATA)>
<!ATTLIST token1 pos CDATA #REQUIRED>
<!-- Ditto for token2 and token3-->
<!ELEMENT distinct (#PCDATA|note_reference)*>
<!ELEMENT theme (id, subid)>
<!ELEMENT main_theme (id, subid)>
```

Fig. 4. Common DTD

4 Evaluation

We have tested the main product for each the two technologies mentioned: Oracle (9ir2 version), a widespread RDBMS, and Tamino (4.2.1 version), a native XML manager with many capabilities. Two main criteria were used in the evaluation:

- **Flexibility:** Taking into account the requirements of IRSs based on corpora, we determine how deeply each technology verifies them.
- **Performance:** Using a representative set of queries, we test the performance of the system running them and measure the time needed to obtain the results.

  Despite the fact that there are some benchmarks for XML IRSs [5], they are too generic and not oriented to the needs of systems based on corpora. Furthermore, these benchmarks are not valid for both kinds of technologies, thus there are benchmarks for RDBMSs and different ones for XML native indexers, but there are no benchmarks to measure the performance on both technologies at the same time. Therefore, we have designed a representative set of queries to make these tests.

4.1 Flexibility

IRSs based on corpora can have a heterogeneous range of requirements, and types of query can be very different from one system to another, but there are some generic needs that most systems should have:

1. **Statistical capabilities:** Capabilities to obtain numerical values at different levels. For example, to count the number of cases and documents which match a query.

2. **Additional information:** Capabilities to offer supplementary information with the returned results. For instance, to return sentences which verify search criteria but also showing the author of the relevant document, publisher name, etc., that is, additional data taken from the corpus document structure.
3. **Match highlighting:** To highlight the term or terms which produced the matching.
4. **Context:** To show not only the matching cases but their context as well. For instance, in the CORGA example it must be possible to show the whole sentence where the match was found, or n words before and after the match, or even n sentences before and after the sentence with the match.
5. **Index/Search methods:**
   (a) Exact matching queries.
   (b) Diacritical mark sensitive or insensitive queries.
   (c) Case sensitive or insensitive queries.
   (d) Boolean queries.
   (e) Proximity queries. That is, queries about words separated by less than a given distance.
   (f) Several tokenization and indexation criteria in the same database. For example, using the same database, a user could decide to make a query that is sensitive or non-sensitive to accents.
   (g) Excluding marked text from index. For example, in the CORGA example, exclude from indexation all text within the *distinct* element.
   (h) Ignoring certain characters on indexation. Sometimes corpora builders use in-line marks that we want to see in the results, but we do not want to index them. For instance, the use of brackets to mark recomposed text from a medieval edition.
6. **Charsets:** To allow several of the most commonly used charsets.
7. **Use of wildcards:** To allow the use of wildcards or regular expressions in queries.
8. **Results browsing and navigation:** It must be possible to navigate between pages of results without having to repeat the query.
9. **Ordering:** To order the results using one or several criteria at a time.
10. **Structural relationships:** The query language should be flexible enough to build a wide range of queries.

Evaluation

In figure 5 we show the evaluation of Oracle-SQL-Relational and two query languages of Tamino XML Native database: XQuery, a working draft of World Wide Web Consortium [1] and X-Query, a Tamino proprietary language. The following conclusions can be extracted:

- X-Query language has the lowest flexibility, so we will take into account only the XQuery alternative in the following discussions.

Requirement	**Oracle**	**Tamino XQuery**	**Tamino X-Query**
1	several at a time (tokens needed)	one at a time (tokens needed)	**No**
2	Yes	Yes	limited
3	Yes	Yes	**No**
4	Yes (several possibilities)	Yes	**No**
5a	Yes	Yes	Yes
5b	Yes	Yes	Yes
5c	Yes	**No**	**No**
5d	Yes	Yes	**No**
5e	NEAR	PROXIMITY-CONTAINS	ADJ,NEAR
5f	Yes	**No**	**No**
5g	Yes	Yes	Yes
5h	SKIPJOINS	**No**	**No**
6	Yes	Yes	Yes
7	%,_	*,?	*,?
8	Yes	Yes	Yes
9	Yes (several criteria)	Yes (several criteria)	Yes (several criteria)
10	SQL (high flexibility)	XQuery (very high)	X-Query (low)

Fig. 5. Flexibility evaluation. Numbers in the first column correspond to the requirements listed in section 4.1.

- Both Tamino and Oracle can obtain statistical information about any level of documents, whenever those levels are included in the structure of the representation of documents.

  This requirement can be problematic, because it could force the replication of data. For example, in the entity-relation model in figure 3, a document is broken down until word (token) level. *Token1*, *token2* and *token3* contain sequences of one, two and three words from each sentence (the structure is needed to count the number of cases that match a query). So, the text is replicated in sentence and token structures.

  Moreover, only Oracle allows us to obtain different statistical information by sharing calculations, thus considerably improving system performance.
- Oracle offers more possibilities to show the context of searched terms.
- Both technologies have proximity operators, but Tamino does not allow us to build case-sensitive indexes, or define different criteria for tokenization and indexation for the same database, or ignore certain characters in the indexing process.
- Both Tamino and Oracle allow the use of different charsets, including UTF-8.
- Both technologies allow us to order the results by several criteria at a time.
- XQuery is even more flexible than SQL, since it has a more complex syntax that allows more complex structures to be represented and manipulated.

Although it would appear that Tamino has almost all the required needs, we must take into account that the proposed needs are minimal for several IRS based on corpora, and the absence of one of them could make all the difference between a useful or useless system.

Moreover, XQuery is still a working draft and is not yet completely implemented. Oracle would therefore seem to be the best alternative, from the flexibility point of view, for building these kinds of system.

4.2 Performance

To evaluate performance, first we define a representative query set, covering the topics explained in the flexibility section, and then we show the elapsed time needed for each technology to solve the queries. These queries are based on the studied corpus (CORGA), but the topics covered are generic enough to be applied to any other IRS based on structured text corpora.

Query set

Queries are defined as follows:

- **Q1:** To obtain the number of documents and cases from sentences which contain the expression "sen embargo" for each type of medium, main theme and lustrum ("sen embargo" means however in the Galician language).
  The system includes three kinds of medium, six different main themes and six lustrum, so thirty numbers must be obtained. We measure the time elapsed between starting the query and checking all results. It is a measure of efficiency in sharing calculations.
- **Q2:** To obtain the sentences which contain the expression "sen embargo" with their document title, publisher, authors, medium, themes, publishing year, and type of sentence ordered by publishing year, medium and main theme. Match words are returned highlighted.
  We measure the time elapsed between starting the query and showing all required values. It is a computational cost measure about showing all associated elements and highlight and sort operations.
- **Q3a:** To obtain the number of documents and cases from sentences which contain a word with "pre" prefix.
- **Q3b:** To obtain the number of documents and cases from sentences which contain a word with "ado" suffix (the "ado" termination in Galician is usually used for participles of verbs).
- **Q3c:** To obtain the number of documents and cases from sentences which contain a word with "poñ" infix (in Galician "poñer" means to put, "repoñer" means to put back, and "pospoñer" means to postpone).
  Q3 queries are different measures of text index efficiency.
- **Q4:** To obtain cases from sentences which contain a word with "in" or "im" prefix, but only in news headlines from newspapers and with a context of fifteen words. As neither technology supports built-in word context, in this

case the queries return all the necessary information to perform this operation at application level.

We combine a structural complex query with word context retrieval to measure its computational cost.

Evaluation

We have tested these queries on a PC Intel Celeron at 2.4GHz, with 512 MB of memory and Windows XP Professional over a CORGA six hundred thousand word subcorpus (uniformly distributed between newspapers, magazines and books) with previously shown structures.

Although this hardware is not the most suitable for testing these kinds of system, which need powerful servers to manage millions of words, it provides us with conclusive results about the performance difference between both technologies tested.

As we can see in figure 6, the most important difference is in Q1 query, due to the inability of Tamino to calculate several statistic values by sharing calculations. So, once again Oracle leads the results of the tests, confirming it as the best option for building this kind of system.

Query	**Tamino XQuery(4.2.1)**	**Oracle (9ir2)**
Q1	585.203s	10.891s
Q2	33.922s	14.531s
Q3a	106.719s	5.828s
Q3b	118.266s	34.172s
Q3c	98.750s	3.687s
Q4	26.545s	11.187s

Fig. 6. Performance evaluation. It shows time duration of proposed queries in seconds.

5 Conclusions

First, it is necessary to emphasise that it is mandatory to transform documents of the corpora into a common format when managing large amounts of information. This will allow us to query all documents using a unique query and to improve the performance of the system. By doing so we will avoid problems with performance and result management.

Furthermore, nowadays, the technologies used to build IRSs are not prepared to satisfy corpora users' requirements. So, in the near future the development of new add-ons which take them into account is needed. There are some timid attempts to include basic linguistic operations (sensitivity to accents, umlauts, etc., theme searches, etc.) based on localization, but it is time to incorporate syntactic techniques [8] [9] into commercial systems to enable the building of more versatile IRSs based on corpora.

Furthermore, traditional technologies manage updatable systems, but companies need to develop more optimisation options for systems which give very high priority to query performance regardless of the very high penalty in updates.

Finally, XML technologies are being developed very rapidly, but they still have to settle. The speed of this evolution prevents robustness and clear development of systems, making it difficult to put them in production.

Against this, RDBMSs have a high robustness, flexibility and performance, due to their long life in the market. Taking into account our test results, the fact that these systems include text index capabilities and the drawbacks of other technologies, at the moment we propose Oracle as the first option for building IRSs based on large structured corpora.

References

1. XML. In *http://www.w3c.org*, 2/5/2005.
2. María Sol López Martínez. CORGA (Corpus de Referencia del Gallego Actual). In *Proc. of Hizkuntza-corpusak. Oraina eta feroa*, pages 500–504, Borovets, Bulgaria, Sept. 2003.
3. Mark Davies. Un corpus anotado de 100.000.000 palabras del español histórico y moderno. In *Proceedings of Sociedad Española para el Procesamiento del Lenguaje Natural* pages 21–27, Valladolid, Spain, 2002.
4. Mark Davies. Relational n-gram databases as a basis for unlimited annotation on large corpora. In *Proceedings from the Workshop on Shallow Processing of Large Corpora*, Lancaster, England, pages 23–33, March 2003.
5. Akmal B. Chaudhri, Awais Rashid and Roberto Zicari. XML Data Management, Native XML and XML-Enabled Database Systems, *Addison-Wesley*, March 2003.
6. Oracle. In *http://www.oracle.com*, 2/5/2005.
7. Tamino. In *http://www.softwareag.com*, 2/5/2005.
8. Jesús Vilares, Miguel A. Alonso and Manuel Vilares. Morphological and syntactic processing for Text Retrieval. In *Database and Expert Systems Applications, volume 3180 of Lecture Notes in Computer Science, Springer-Verlag*, Berlin-Heidelberg-New York, pages 371–380, 2004.
9. Miguel A. Alonso, Jesús Vilares, and Víctor M. Darriba. On the Usefulness of Extracting Syntactic Dependencies for Text Indexing. In *Artificial Intelligence and Cognitive Science, volume 2464 of Lecture Notes in Artificial Intelligence, Springer-Verlag*, Berlin-Heidelberg-New York, pages 3–11, 2002.

Meteorological Image Descriptors

J.L. Crespo, P. Bernardos, M.E. Zorrilla, and E. Mora

Department of Applied Mathematics and Computer Sciences, University of Cantabria,
Avda. de los Castros s/n 39005 Santander, Spain
{crespoj, bernardp, zorrillm, morae}@unican.es

Abstract. The objective of this paper is to get a visual characterization of time evolution images, in particular, synoptic maps taken from Meteorology. Preliminary tasks required before image processing are reviewed. Two different types of numerical descriptors are extracted for characterizing the images, the called *low level* numerical descriptors, and the *high level* corresponding ones. The latter will be subsequently used for prediction tasks, meanwhile the former will be used for classification tasks. Three different relevant information sources in the images are identified as their *low level* descriptors. These are defined by the local density and orientation of the isobar lines, and the number of centres of high (H) and low (L) pressure. Regarding the *high level* descriptors, two main features are taken into account. The different procedures carried out to extract the previous descriptors for our images of interest are discussed.

1 Introduction

One of the PIETSI research project's main objectives is to predict the temporal evolution of images of physical processes or phenomena for which experimental information in image files is available. Specific developments have been initiated, focusing on weather forecasting using synoptic map images. For this purpose, the University of Cantabria group has worked in collaboration with the EUVE (European Virtual Engineering) Technology Centre, and more specifically its Meteorological Centre, with its head office in Vitoria, Spain.

The preprocessing phase in this project was explained in [1]. Common preliminary tasks to single out the image information of interest, such as useless background removing, information separation and memory usage were discussed there. The synoptic maps that we work with are of the type shown in Figure 1.

In particular, regarding the background removing, we eliminated the parallel and meridian lines, the contours of the land over the sea and the upper left and lower right captions in the image by removing the common pixels of several images. Moreover, we separated into different images the isobar (thin) lines and the front (thick) lines, and finally, an efficient storage strategy was able to achieve a saving in memory per image of around 99.5%.

As Figure 2 shows, the isobar line image is not perfect, due to intersections with the thick lines. We made an attempt to overcome this with active contours [2] that would fit to each isobar, but found the following problems:

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 101 – 110, 2005.
© Springer-Verlag Berlin Heidelberg 2005

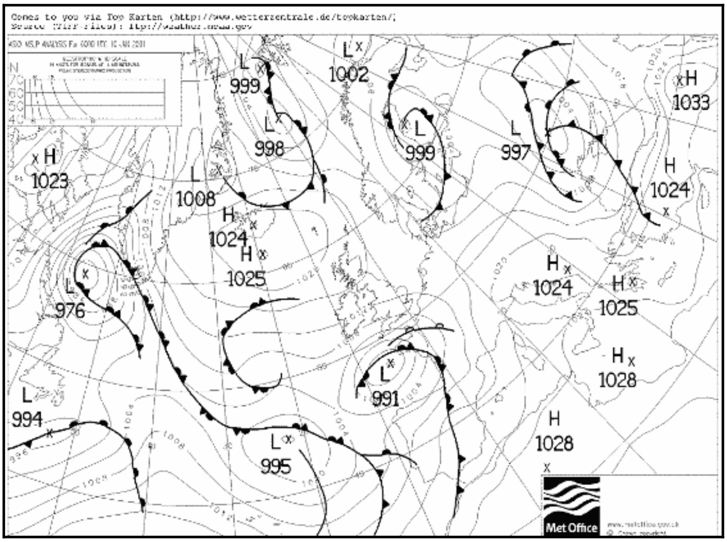


Fig. 1. Usual synoptic map

- The starting point is not easily guessed automatically.
- Since there are many isobars in the image and the distance between them can be similar to the gaps in each of them, the active contour would often try to fit several isobars together, messing the map.

We did not pursue this issue any further since the isobar image is not a final result, but an intermediate step, and its original quality is enough for our purposes, as we will show below.

The next step in our project is studied in this paper and consists in extracting numerical descriptors for the relevant features in the images, that is to say, the isobar lines and the centres of low (L) and high (H) pressures, in order to obtain a visual characterisation of them. As far as we know, no similar work has been found to date in the computer vision literature for classification or prediction of this kind of image, besides those that treat the problem with complicated numerical models in Meteorology or statistical mathematical methods.

For further processing of the images, the amount of information should be as large as possible, but it should avoid the overfitting problem, the risk of which increases with the number of parameters, which in turn is related to the amount of features being processed [3]. Since we are dealing with meteorological images, we focus on features that are related to atmospherical properties [4]. In this sense, given the big problem of predicting the evolution of a synoptic map, two types of numerical descriptors are obtained: the *low level* descriptors and the *high level* ones. The latter will be subsequently used for prediction tasks, while the former will be used for classification tasks.

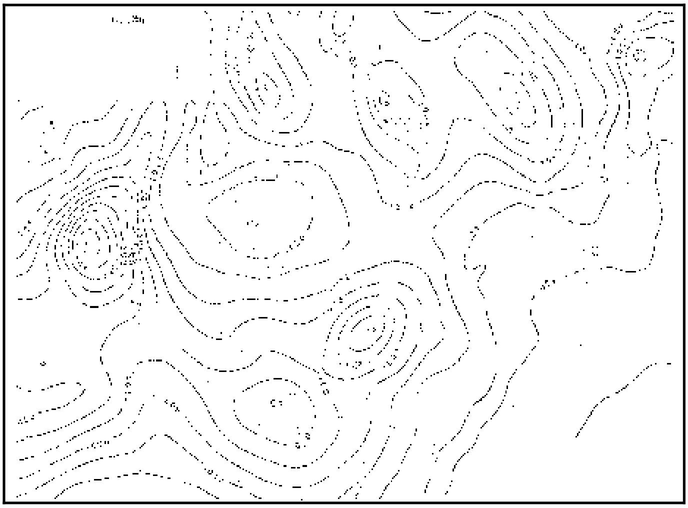


Fig. 2. Isobar image

2 Isobar Image Definition

To accomplish this task, we study separately the isobar lines on the one hand and the centres of high and low pressures on the other hand.

2.1 Isobar Lines

The isobar image, see Figure 2, is a line image, analogous to an edge image and many descriptors can be derived from it [5]. We attempt to describe it here by the following *low level* descriptors: the density of the lines and their average orientation in local areas.

The density of the lines is actually a downscaled image, following the approach of pyramid representations [6]. For this purpose, we consider the isobar image to be divided into adequate small windows. On the one hand, the size of the windows must be small enough to identify the corresponding isobar image. However, on the other hand, the window must be as big as possible in order to minimize the memory used per isobar image, which will be important for the subsequent analysis, given the high number of images to be processed and stored.

Calculations are carried out for square windows 100, 50 and 30 pixels wide, for comparison. Hence, the initial binary isobar image, having 864x1074 pixels, is described with three 9x11, 18x22 and 29x36 arrays, respectively. One of these arrays quantifies the local density of the lines in each window and the two remaining arrays define the x- and y-coordinates, respectively, of the vector that represents the local orientation.

The density of the lines for a given *W* window is easily obtained as the quotient between the number of binary pixels belonging to the lines and the total number of pixels inside that window. That is, it is equivalent to an image reduction:

$$d = \frac{\sum_{p \in lines} p}{\sum_{p \in W} p} . \tag{1}$$

Next, a local orientation for a window is defined. In this case, the optimum orientation is taken, which corresponds to the orientation that shows the least deviations from the directions of the gradient [7]. Thus, the orientation vector for a *W* window is given by:

$$\vec{o} = \frac{\sum_{p \in W / |\overrightarrow{o_p}| > min} (Jxx - Jyy, 2Jxy)}{\sum_{p \in W / |\overrightarrow{o_p}| > min} p} , \tag{2}$$

where

$$J_{ij} = \frac{\partial g(p)}{\partial i} \frac{\partial g(p)}{\partial j} \; with \; i,j = x,y . \tag{3}$$

The *g(p)* function in Equation (3) gives the binary value changes of the *p* pixel in its 2-dimensional neighbourhood and can be obtained, for example, by the well-known Sobel filter [8]. The summation given in Equation (2) is made over every *p* pixel inside the chosen *W* window which verifies that the corresponding orientation vector magnitude for that pixel is bigger than a threshold *min* value. The magnitude of the orientation vector represents a certainty measure of local orientation.

2.2 Centres of High (H) and Low (L) Pressures

Another important aspect for a complete isobar image meteorological definition is the location of high (H) and low (L) pressure points. For this purpose, the isobar-and-front or front image is required, because the H and L letters are both represented with thick lines (see Figure 3).

The detection of H and L letters is a typical OCR task, but since no other letters are involved, we use and 'ad-hoc' algorithm, instead of generic OCR techniques [chapter 2 of 9].

The proposed procedure in the previous image may be carried out by following the next steps:

- Separating regions. This is done by segmenting [10] and labelling the 8-connected objects found in the image with different integer values.
- Extracting the feature measurements for the image regions obtained in the previous step. Among the many features that can be analyzed [11], we concentrate on:
  1. The *area*, with the actual number of pixels in the region.
  2. The 2-Dimensional correlation coefficient computed between two regions of the same size.
  3. The *centroid*, or x- and y- coordinates of the centre of mass of the region.

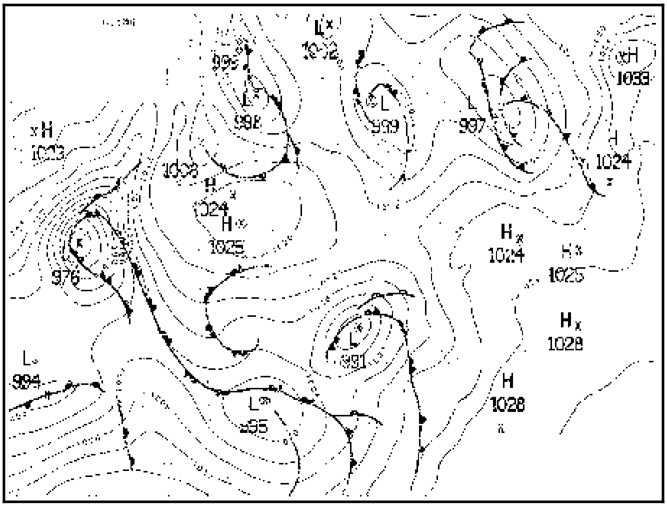


Fig. 3. Isobar-and-front image

4. The *major-axis-length*, or the length, in pixels, of the major axis of the ellipse that has the same second-moments as the region.
5. The *minor-axis-length*, or the length, in pixels, of the minor axis of the ellipse that has the same second-moments as the region.
6. The *orientation*, or the angle between the x-axis and the major axis of the ellipse that has the same second-moments as the region.

When considering their areas, we have found that not all H letters are identical in these particular images, and the same applies for L's; we have found inside the same image, differences of as much as 40% in their respective area values. We found that, to identify the H and L regions by means of their corresponding areas, the intervals of values needed were not restrictive enough to leave out other objects. As a onesequence, this method was dropped.

Another possible region feature to deal with is the 2-Dimensional correlation coefficient. To compute this parameter, two bidimensional arrays of the same size are needed: one array representing the H (L) model chosen, the other array being the region to be studied for the extraction of H (L) region.

Given the variability of H and L letters in our images, their models cannot be chosen at random. Obviously, other regions different from H and L ones must not come out. To make comparisons easier, two H and two L letters are extracted from Figure 3 and shown in Figures 4, 5 and 6, 7, respectively. The different forms and thickness of the vertical and horizontal pieces are evident.

Another important aspect is that reliable values for the correlation coefficient are only obtained when the two arrays compared are constructed around the centre of mass of their regions. As a consequence, the centroid of the region has to be measured too. In practice, the following conditions are found for an image:

$$\text{If } cc(Hm, Reg)>0.53, \text{ then } Reg \equiv H, \quad (4)$$

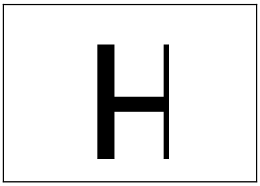


Fig. 4. Region extracted from Fig. 3

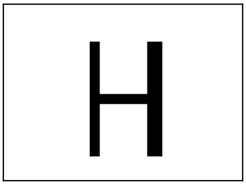


Fig. 5. Region extracted from Fig. 3

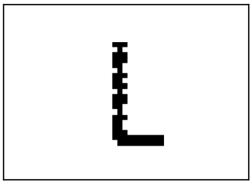


Fig. 6. Region extracted from Fig. 3

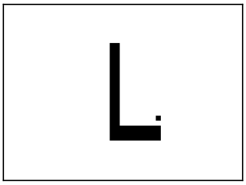


Fig. 7. Region extracted from Fig. 3

where cc stands for the 2D correlation coefficient, Hm is the model for H, and Reg is the region being tested, and

$$\text{if cc(Lm, Reg)>0.62, then Reg}\equiv\text{L,} \tag{5}$$

with analogous notation, which means that if the correlation coefficient computed between the H model and any region in the image is bigger than 0.53, the region is identified as an H region. In the same way, if the correlation coefficient computed between the L model and any region in the image is bigger than 0.62, the region is identified as an L region.

However, the previous limiting values found for the correlation coefficient of H and L regions in an image, written in Equations (4) and (5), have to be lowered when dealing with another image, if the same H and L models obtained for the previous image are kept. Depending on the case, the decrease in the limiting values is so high that unwanted regions become falsely identified, such as special portions of fronts or numbers 9 and 5.

One possibility for solving the problem would be to redefine the H and L models for each image being studied. Both tasks, redefining the limiting correlation coefficient values or the H and L models, would assume a manual work stage with each individual image. Hence, the effort achieved for a big number of images would be unreasonable. For this reason, this second method was also dropped.

Finally, we use as feature measurements for a region: the major- and the minor-axis-lengths of the ellipse that has the same second-moments as the region. Taking the appropriate intervals of those parameters, one manages to identify solely the H and L regions, the locations of which are determined with the corresponding region centroid coordinates. It is worth mentioning that a few images have presented a problem: some number 7's may be identified as an L region. To avoid this, another feature, the orientation of the ellipse, is measured, as defined in the previous list for the last item.

Once the centres of masses of the H and L regions are known, their corresponding windows are calculated and the found quantity of each one is saved in an array. Hence, for a given window, the number of H and L are saved so that two arrays are needed for the H and L *low level* descriptors.

Regarding the *high level* numerical descriptors, two main features are represented per image: the big regions of locally profound high (H) and low (L) pressure and those of nearly constantly oriented circulation.

3 Isobar Image Recovering

To verify if the extracted variables indicated in the previous section are good *low level* descriptors for the isobar image, we develop the inverse procedure, that is, we use the five arrays obtained before rebuilding the image, and compare it with the original one.

Figures 8, 9 and 10 show the recovered images obtained from the original image (Figure 1) or the isobar-and-front image (Figure 3) after axis reductions with windows of 100, 50 and 30 pixels wide, respectively. The background is added to the images for convenience.

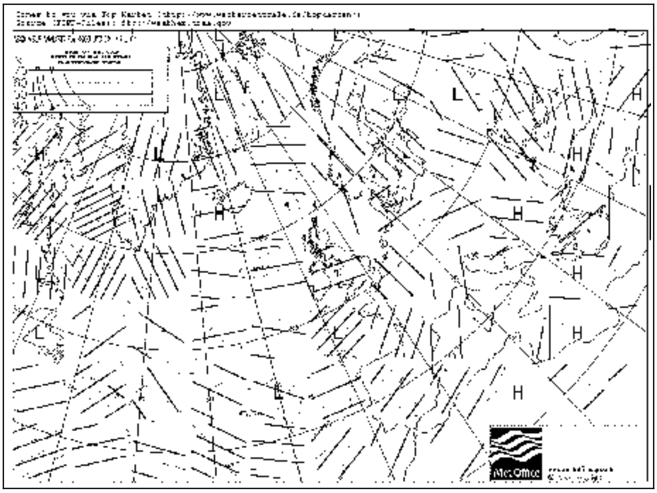


Fig. 8. Recovered isobar image from Figure 3 using 100 pixels wide windows

The window size in each case is easily seen from the figures. For each window, the density of the lines gives how tight or separate the lines drawn inside it are, while the orientation vector gives the line rotation, ignoring the portions that come out the window, using nearest neighbour interpolation. Regarding the H and L descriptors, each window may contain one and only one of these species or not. In the affirmative case, the corresponding letter is drawn using a 22x14 model array inside the window, centred in the window.

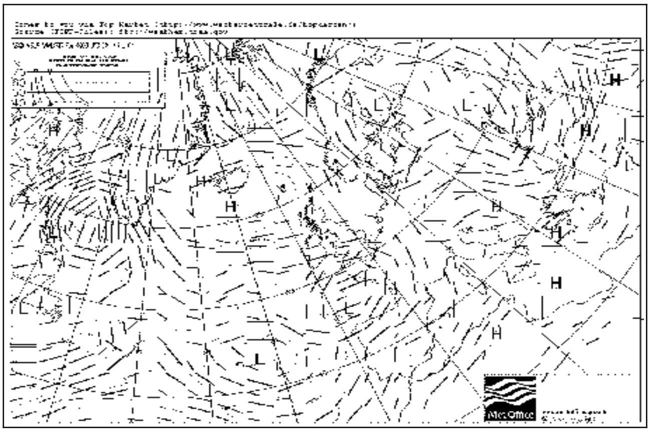


Fig. 9. Recovered isobar image from Figure 3 using 50 pixels wide windows

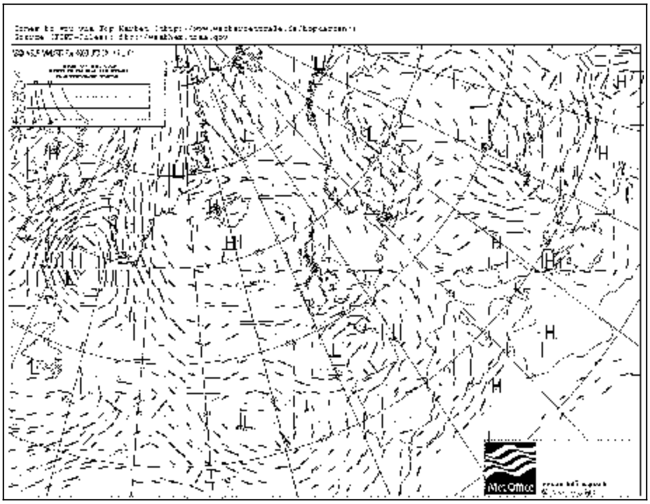


Fig. 10. Recovered isobar image from Figure 3 using 30 pixels wide windows

Focusing on the isobar lines, we can see that a width of 100 is too big to identify the original image visually. However, satisfactory descriptions are obtained when the width comes down to 50 or 30. Obviously, a more accurate image is reproduced when a 30 width is considered. However, the size of the arrays is in this case larger, which is not desirable, given the high number of images to be processed, and, consequently, we select the 50 pixels wide windows.

Regarding the centres of high (H) and low (L) pressure, one can see that the bigger the axis reductions are, the more accurate the positions of the H and L centres are.

4 Conclusions

In this paper, we propose one possible definition for the synoptic map images that we work with. Three different *low level* descriptors are studied:

- The density of the isobar lines,
- the orientation of the isobar lines and
- the centres of high (H) and low (L) pressures.

To extract the corresponding variables, we consider the isobar-and-front images divided into adequate smaller arrays, where each element represents a fixed window in the image. Calculations are carried out for square windows 100, 50 and 30 pixels wide, for comparison. A satisfactory description is obtained when the width is 50 or smaller. Hence, the initial binary isobar images, having 864x1074 pixels, are described with five 18x22 arrays:

- One array quantifies the local density of the isobar lines.
- Two arrays define the x- and y-coordinates, respectively, of the vector that represents the local orientation of the chosen window.
- The two remaining arrays contain the number of H and L per window.

Concerning the H and L identification in our images, which might be imagined easy in principle, in practice, as usually happens in many other real applications that work with photocopied or scanned images for instance, simple methods do not work, as we have seen, because H and L letters present in our images are not equal. For this reason, an analysis of the discriminatory power of several features has been necessary.

Acknowledgments

The authors are deeply grateful to the Spanish Interministerial Board for Science and Technology (Comisión Interministerial de Ciencia y Tecnología CICYT) which supports Project TIC2002-01306, as a part of which this paper has been produced. They would also like to thank the EUVE (European Virtual Engineering) Technology Centre for their collaboration.

References

1. Crespo, J. L., Bernardos, P., Zorrilla, M. E., Mora, E.: Preprocessing Phase in the PIETSI Project (Prediction of Time Evolution Images Using Intelligent Systems). 9th International Workshop on Computer Aided Systems Theory: Eurocast 2003. Eds. Roberto Moreno-Díaz and Franz R. Pichler. Lecture Notes in Computer Science 2809, 651-660. Springer-Verlag, Berlin Heidelberg New York (2003)
2. Kass M., Witkin A., Terzopoulos D.: Snakes: active contour models. Int. J. Comput. Vision, 1 (4) 321-331(1988)
3. Waller, W. G., Jain, A. K.: On the monotonicity of the performance of a Bayesian classifier. IEEE Transactions on Information Theory, 24 (3) 392-394 (1978)

4. A. Naya. Meteorología superior. Espasa Calpe (1984)
5. Sonka, M., Hlavac, V., Boyle, R.: Image processing, analysis and machine vision. International Thomson Computer, London, Madrid (1996)
6. Shalkoff, R. J.: Digital Image Processing and computer vision. Wiley, New York (1989)
7. Jähne, B.: Digital Image Processing. Concepts, Algorithms and Scientific Aplications. 4th Edition. Springer-Verlag, Berlin Heidelberg New York (1997)
8. González, J.: Visión por Computador. Paraninfo, Madrid (1999)
9. Cole, R. A. (ed. in chief): Survey of the State of the Art in Human Language Technology. Cambridge University Press (1996)
10. Young, I. T., Gerbrands, J. J., van Vliet, L. J.: Fundamentals of Image Processing. Delft University of Technology (1998)
11. Rafael C. González, Richard E. Woods. Digital Image Processing, 2nd Edition. Prentice Hall (2002)

Towards a Certified and Efficient Computing of Gröbner Bases*

J. Santiago Jorge, Víctor M. Gulías, José L. Freire, and Juan J. Sánchez

MADS - LFCIA, Dept. de Computación, Universidade da Coruña,
Campus de Elviña, s/n., 15071 A, Coruña, Spain
{sjorge, gulias, freire, juanjo}@dc.fi.udc.es

Abstract. In this paper, we present an example of the implementation and verification of a functional program. We expose an experience in developing an application in the area of symbolic computation: the computing of Gröbner basis of a set of multivariate polynomials. Our aim is the formal certification of several aspects of the program written in the functional language Caml. In addition, efficient computing of the algorithm is another issue to take into account.

1 Introduction

Certifying the correctness of a program is a difficult and expensive labor and, at the same time, it is one of the most important activities for a software engineer. Debugging and testing techniques can detect errors, but they cannot guarantee the correctness of the software. Formal methods complement those techniques assuring that some relevant property holds in the program.

This work aims to contribute to the construction of a methodology to produce certified software, i.e., we intend to find the way to strengthen two different notions: First, there is a need for formal verification of the correctness of algorithms which goes together with their construction; and second, programs are mathematical objects which can be handled using logico-mathematical tools. We present the way in which one can formally verify several aspects of the application implemented following the functional paradigm. These techniques will be exemplified by means of the verification of a program which calculates the *Gröbner basis* of a set of multivariate polynomials [1]. Here, both the certification of the program and its efficiency will be taken into consideration. We study the development of algorithms formally proved for an efficient computing. The steps are: formalization of a multivariate polynomial ring; construction of a well-founded polynomial ordering; and finally, definition of the reduction relation and Buchberger's algorithm.

Functional programming [2,3,4] has often been suggested as a suitable tool for writing programs which can be analyzed formally, and whose correctness can be assured [5]. This assertion is due to *referential transparency*, a powerful mathematical property of the functional paradigm that assures that equational

---

* Supported by MCyT TIC2002-02859 and Xunta de Galicia PGIDIT03PXIC10502PN.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 111–120, 2005.
© Springer-Verlag Berlin Heidelberg 2005

reasoning makes sense. The mathematical way of proving theorems can be successfully applied to computer science. As programming language, Objective Caml [6,7] was chosen due to its efficiency and its wide coverage both in the research and academic environments.

Properties are proved by means of the formalization of theorems in an *abstract* model of *actual* code in the Coq [8,9] proof assistant and also, in a manual but exhaustive style directly applied to the final code. Programs are treated as mathematical objects and each step of those proofs is justified by means of mathematical reasoning. Significant progress has been made (see [10,11,12]) in the automated verification of the proof of Gröbner bases algorithm by proof checkers. Theorem provers assist us in proving the correctness of a program. They not only help us in the development of the proofs but also guarantee the correctness of such proofs; thus they prevent bugs that could be introduced in a hand-made certification. The logical framework Coq is an implementation of the Calculus of Inductive Constructions [13].

The paper is organized as follows. In section 2, a reusable multivariate polynomial library is certified, and a well-founded polynomial ordering is also verified. Section 3 states the correctness of the reduction relation on polynomials. Section 4 reasons about Buchberger's algorithm, and presents some results. Finally, we conclude.

2 Multivariate Polynomials Using Dependent Types

A reusable polynomial library is going to be formally verified. It will make further work in verification of polynomial algorithms less difficult. A multivariate polynomial ring over a coefficient field is our target. We not only want to prove the fundamental properties of polynomial rings, but we also search for an adequate implementation allowing efficient computing. Canonical representations of polynomials are used, and it can be decided if two polynomials are equal by studying if their representations are equal.

Although there are previous works on the formalization of multivariate polynomials by proof checkers (see [14,15]), they neither work directly with canonical representations of polynomials, nor do they use dependent types. The certification of this library has been carried out both reasoning directly about the actual functional program and also proving laws in the abstract model in Coq. Below, the main laws stated with the help of the proof assistant are presented.

A *term* in the variables $X = \{x_1, x_2, \ldots, x_n\}$ is represented by a list of the exponents of each variable.

```
type term = int list
```

In Coq, they are described as lists of fixed length with *dependent types*. The use of dependent types improves the accuracy and clarity of the specification.

```
Inductive Dlist [A: Set]: nat->Set :=
    Dnil: (Dlist A (O))
```

```
  | Dcons: (n:nat)A->(Dlist A n)->(Dlist A (S n)).
Definition term: nat->Set:= [n: nat] (Dlist nat n).
```

Dependent types provide accuracy in the specifications, which is an additional reliability. But, sometimes this precision can impose a heavy manipulation of expressions.

Two terms are multiplied by adding the respective exponents of each variable.

```
(*val mult_term : term->term->term*)
let mult_term xs ys = map2 (+) xs ys
```

This Caml code is very similiar to to the abstract model built in Coq:

```
Definition mult_term:(n:nat)(term n)->(term n)->(term n):=
 [n:nat;t,s:(term n)](map2 nat nat nat plus n t s).
```

Terms form a commutative monoid under multiplication.

```
Lemma mult_term_sym: (n: nat) (t,s: (term n))
 (mult_term n t s)=(mult_term n s t).

Lemma mult_term_assoc_l: (n:nat) (t,s,r: (term n))
 (mult_term n t (mult_term n s r))
   = (mult_term n (mult_term n t s) r).

Lemma mult_term_1_t: (n: nat; t,s: (term n))
 (null_term n t) -> (mult_term n t s)=s.
```

Other results on terms are also proved.

```
Lemma div_term_mult_term: (n: nat) (t, s: (term n))
 (div_term n (mult_term n t s) s)=t.
```

We state the lexicographical order on terms, and we show that it is an admissible order.

$$[a_1, \dots, a_n] > [b_1, \dots, b_n] \Leftrightarrow \exists i \text{ con } a_j = b_j \text{ for } 1 \leq j < i \text{ and } a_i > b_i$$

1. There exists a first element, $t >_{lex} 1$, $\forall t \in T_X$, $1 \neq t$

   ```
   Theorem ltlex_term_admissibility_1: (n: nat; e, t: (term n))
    (null_term n e) -> ~ (null_term n t) -> (ltlex_term n e t).
   ```

2. The ordering respects multiplication, $t >_{lex} s \Rightarrow t \cdot r >_{lex} s \cdot r$, $\forall t, s, r \in T_X$

   ```
   Theorem ltlex_term_admissibility_2: (n: nat; t, s: (term n))
    (ltlex_term n t s) -> (r:(term n))
      (ltlex_term n (mult_term n t r) (mult_term n s r)).
   ```

Monomials are represented as coefficient-term pairs. In the Caml program, absolute precision numbers (`num` library) are used as coefficients. In the model built in Coq, on the other hand, the coefficients are axiomatized. Monomials form a commutative monoid under multiplication.

```
Lemma mult_mon_sym: (m1, m2: mon)
 (eq_mon (mult_mon m1 m2) (mult_mon m2 m1)).

Lemma mult_mon_assoc_l: (m1, m2, m3: mon)
 (eq_mon (mult_mon m1 (mult_mon m2 m3))
         (mult_mon (mult_mon m1 m2) m3)).

Lemma mult_mon_1_m: (e, m: mon)
 (mon1 e) -> (eq_mon (mult_mon e m) m).
```

Polynomials are represented as lists of monomials. The representation is canonical: terms are strictly ordered by a decreasing term order, and the list contains no null monomial. Hence, two polynomials are equals if their representations are syntactically equal. As we axiomatize the coefficients in the model built in Coq, an explicit equality has to be used.

```
Inductive eq_pol  : pol->pol->Prop :=
   eq_pol_1 : (eq_pol (nil mon) (nil mon))
 | eq_pol_2 : (m1,m2:mon; p1,p2:pol)
                (eq_mon m1 m2) -> (eq_pol p1 p2) ->
                  (eq_pol (cons m1 p1) (cons m2 p2)).
```

Sometimes two different versions of a polynomial function (for instance, addition) are implemented: one efficient, the other simple. We prove they are equivalent.

Functions over polynomials always act on canonical objects to yield canonical results. Thus, on the one hand more efficient programs are obtained from an algorithmic point of view. However, on the other hand, polynomial functions become more complex because there are lots of alternatives, and consequently proofs get more complex and tedious.

```
Lemma add_pol_canonical: (p1, p2: pol)
 (canonical p1)->(canonical p2)->(canonical (add_pol p1 p2)).

Lemma mult_pol_canonical: (p1, p2: pol)
 (canonical p1)->(canonical p2)->(canonical (mult_pol p1 p2)).
```

Polynomials with addition and negation form an Abelian group.

```
Lemma add_pol_p_0: (p: pol)
 (canonical p) -> (eq_pol (add_pol p pol0) p).

Lemma add_pol_sym: (p1, p2: pol)
 (canonical p1) -> (canonical p2) ->
   (eq_pol (add_pol p1 p2) (add_pol p2 p1)).

Lemma add_pol_assoc_l: (p1, p2, p3: pol)
 (canonical p1) -> (canonical p2) -> (canonical p3) ->
   (eq_pol (add_pol p1 (add_pol p2 p3))
           (add_pol (add_pol p1 p2) p3)).
```

```
Lemma add_pol_minus_pol: (p: pol)
 (canonical p) -> (eq_pol (add_pol p (minus_pol p)) pol0).
```

With the multiplication they form a ring.

```
Lemma mult_pol_1_p: (e, p: pol)
 (canonical p) -> (pol1 e) -> (eq_pol (mult_pol e p) p).

Lemma mult_pol_sym: (p1, p2: pol)
 (canonical p1) -> (canonical p2) ->
   (eq_pol (mult_pol p1 p2) (mult_pol p2 p1)).

Lemma mult_pol_assoc_l: (p1, p2, p3: pol)
 (canonical p1) -> (canonical p2) -> (canonical p3) ->
   (eq_pol (mult_pol p1 (mult_pol p2 p3))
           (mult_pol (mult_pol p1 p2) p3)).
```

And multiplication distributes over addition.

```
Lemma mult_pol_add_mon_pol_distr: (p1, p2: pol; m: mon)
  (canonical p1) -> (canonical p2) -> (not_mon0 m) ->
      (eq_pol (mult_pol (add_mon_pol m p1) p2)
              (add_pol (mult_mon_pol m p2) (mult_pol p1 p2))).
```

2.1 Well-Founded Polynomial Ordering

The well-foundedness of the lexicographical order on terms has been verified both on the CAML program, and on the abstract model in COQ. Reasoning over the CAML code, the well-foundedness of the total degree order was also proved.

We extend the term order on monomials. With polynomials in canonical form, the monomial ordering is extended to polynomials in a straightforward

Theories	*Lines*	*Defs.*	*Laws*	*Prop.*	*Size*
Dlist	101	9	5	7.21	15K
Term	331	6	18	13.79	155K
LtlexTerm	191	1	8	21.22	146K
Coef	155	6	32	4.08	11K
Mon	171	14	18	5.34	42K
Pol	175	7	15	7.95	73K
AddMonPol	862	1	10	78.36	348K
AddPol	311	2	18	15.55	35K
MultMonPol	260	1	14	17.33	61K
MultPol	338	1	16	19.88	32K
total	2895	48	154	14.33	918K

(a) Polynomial Theories

Theories	*Lines*	*Defs.*	*Laws*	*Prop.*	*Size*
WfLtlexTerm	49	2	2	12.25	18K
WfLtlexMon	23	1	3	5.75	3K
Desc	110	0	6	18.33	18K
WfLtlexPol	98	6	7	7.54	33K
total	280	9	18	10.37	72K

(b) Well-founded Theories

Fig. 1. Quantitative Information on the Development in Coq

way. In this proof, we use a lexicografic exponentiation theory from Paulson [16] that requires monomials to be strictly ordered in a decreasing order. So, the lexicographic relation induced on *polynomials* is well-founded. In this work, we started with the lexicographic order on terms, but the development is generic. Any well-founded term ordering can be used.

Figures 1(a) and 1(b) show quantitative information on the Coq theories. The columns correspond to the number of lines of code in each theory, the number of definitions (including tactics) and the number of laws, the proportion between the number of lines and the quantity of laws and definitions (which can be used as a measure of the complexity of the theory), and the size of each compiled Coq theory, respectively.

3 Polynomial Reduction

The reduction relation on polynomials involves subtracting an appropriate multiple of one polynomial from another. Below it can be seen the Caml code.

$$red(p, q) = p - \frac{hcoef(p) \cdot hterm(p)}{hcoef(q) \cdot hterm(q)} \cdot q$$

```
(*val nred : (term->term->bool) -> pol -> pol -> pol*)
let nred gt_term f g = match (f, g) with
  ((c, t)::_, (b, s)::_) ->
    sub_pol gt_term f (mult_pol gt_term [c//b, div_term t s] g)
```

In the following example, the act of reducing p by r implies subtracting a multiple of r from p so that the head term of p is canceled: $p = 2x^2yz^3 - 7xy^{10} + z$, $r = 5xyz - 3$, the polynomial r reduces p to $p' = p - (\frac{2}{5}xz^2)r = -7xy^{10} + \frac{6}{5}xz^2 + z$.

Reduction of polynomials is not a total function because term division is not a total function. A polynomial p is reducible by q if the heading term of q divides the heading term of p. The verification of the reduction relation covers two facts:

1. $is\_reducible(p, q) \Rightarrow hterm(red(p, q)) <_{T_X} hterm(p)$
2. $is\_reducible(p, q) \Rightarrow \exists r$ such that $p = red(p, q) + r \cdot q$

Next subsection contains the certification of the two conditions above, carried out with manual proofs that treat the Caml program as a mathematical object.

3.1 Laws About Reduction

Theorem 1. *For every nonzero polynomials* $p = [(c_1,t_1);\ldots;(c_m,t_m)]$ *and* $q = [(b_1,s_1);\ldots;(b_l,s_l)]$*, both in canonical form with respect to a term order* *`gt_term`, and such that* *`is_reducible p q`, it holds:*

`gt_term (ht p) (ht (red gt_term p q))`

Proof. By equational reasoning, using two previous results, and with:

```
redp = nred gt_term          divt = div_term
multp = mult_pol gt_term     multt = mult_term
subp = sub_pol gt_term
```

```
  gt_term (ht p) (ht (redp p q))
=     { 1) by definition of nred (left to right) }
  gt_term (ht p) (ht (subp p (multp [($c_1$/$b_1$, divt $t_1$ $s_1$)] q)))
=     { 2) by definition of mult_pol (left to right) }
  gt_term (ht p) (ht (subp p ((($c_1$/$b_1$)*$b_1$, multt (divt $t_1$ $s_1$) $s_1$)::...)))
=     { 3) by arithmetic on num and the law (t/s)*s=t on terms }
  gt_term (ht p) (ht (subp p  (($c_1$,$t_1$)::...)))
=     { 4) ht f = ht g ⇒ gt_term (ht f (ht (subp f g)) }
  true
```

□

Theorem 2. *For every nonzero polynomials in canonical form with respect to a term order* `gt_term`*,* `p = ((`c_1`,`t_1`)::f')` *and* `q = ((`b_1`,`s_1`)::g')`*, such that* `is_reducible p q`*, it holds:*

```
p = add_pol gt_term (red gt_term p q) (mult_pol r q)
```

where: `r = [(`c_1`/`b_1`, div_term `t_1` `s_1`)]`

Proof. By equational reasoning, using two previous results and with:

```
redp = nred gt_term          multp = mult_pol gt_term
addp = add_pol gt_term       subp = sub_pol gt_term
minusp = minus_pol           divt = div_term
```

```
  addp (redp (($c_1$,$t_1$)::f') (($b_1$,$s_1$)::g'))
         (multp [($c_1$/$b_1$,divt $t_1$ $s_1$)] (($b_1$,$s_1$)::g'))
=     { 1) by definition of nred (left to right) }
  addp (subp (($c_1$,$t_1$)::f') (multp [($c_1$/$b_1$,divt $t_1$ $s_1$)] (($b_1$,$s_1$)::g')))
         (multp [($c_1$/$b_1$,divt $t_1$ $s_1$)] (($b_1$,$s_1$)::g'))
=     { 2) by definition of sub_pol (left to right) }
  addp (addp (($c_1$,$t_1$)::f')
             (minusp (multp [($c_1$/$b_1$,divt $t_1$ $s_1$)] (($b_1$,$s_1$)::g'))))
         (multp [($c_1$/$b_1$,divt $t_1$ $s_1$)] (($b_1$,$s_1$)::g'))
=     { 3) by associativity of add_pol }
  addp (($c_1$,$t_1$)::f')
         (addp (minusp (multp [($c_1$/$b_1$,divt $t_1$ $s_1$)] (($b_1$,$s_1$)::g'))))
             (multp [($c_1$/$b_1$,divt $t_1$ $s_1$)] (($b_1$,$s_1$)::g'))
=     { 4) by the law: eq_pol (add_pol p (minus_pol p)) pol0 }
  addp (($c_1$,$t_1$)::f') []
=     { 5) by definition of add_pol (left to right) }
  (($c_1$,$t_1$)::f')
```

□

3.2 Extension of the Reduction Relation

A polynomial p is reducible modulo $Q = \{q_1, q_2, \ldots, q_m\}$, if there exists q_i such that p is reducible by q_i. We define recursively the closure of reduction.

$$full\_red(p, Q) = \begin{cases} p & \text{if } \neg(is\_reducible(p, Q)) \\ full\_red(red(p, Q), Q) & \text{otherwise} \end{cases}$$

```
(*val full_red : (term->term->bool) -> pol -> pol list -> pol*)
let rec full_red gt_term f gs = match sred gt_term f gs with
    None -> f
  | Some [] -> []
  | Some h -> full_red gt_term h gs
```

There is no infinite sequence of reductions because $<_{T_X}$ is well-founded and $hterm(red(p, Q)) <_{T_X} hterm(p)$. In addition, the result is not reducible modulo Q.

4 Buchberger's Algorithm

Buchberger's algorithm [1] is a generalization of Gaussian elimination. Given a set of polynomials, it produces another set of polynomials with the same roots and additional properties which ease the computation of those roots. The new set, called the Gröbner basis, is analogous to a triangular set of linear equations, which can be solved by substitution. The two basic operations in computing a Gröbner basis are: to eliminate one of the terms of two polynomials obtaining a new polynomial (S-polynomial), and to simplify a polynomial by subtracting multiples of other polynomials.

We implement the Buchberger's algorithm in CAML. In each recursion a pair of polynomials is selected, and the reduction of the S-polynomial is added to the set only if it is nonzero. The polynomial added is then smaller than the two selected polynomials, thus the algorithm always ends.

```
(* val buch: (term->term->bool) -> pol list -> pol list *)
let buch gt fs =
  let rec buch_aux gs = function
      [] -> gs
    | (f,g)::ps -> let h = spol gt f g in
        match full_red gt h gs with
          [] -> buch_aux gs ps
        | h' -> buch_aux (gs @ [h'])
                    (ps @ (map (fun g->(g,h')) gs)) in
    buch_aux fs (allpairs fs)
```

The function `allpairs` computes all possible pairs of the elements of a list. Function `buch_aux` is the recursive implementation of the loop of the algorithm. In each recursion, it is selected a pair and, the reduction of the S-polynomial by

	Linux 2.2.20		*Windows 98*	
	gcalc	gcalcopt	gcalc	Maple V
(1)	2.99 s.	1.05 s.	4.13 s.	12.75 s.
(2)	23.32 s.	8.25 s.	27.19 s.	69.98 s.
(3)	243.43 s.	91.12 s.	288.58 s.	519.45 s.

(a) Pentium 200MMX/48M

Linux 2.4.22			
	gcalc	gcalcopt	Gap
(1)	1.20 s.	0.23 s.	4.10 s.
(2)	9.48 s.	1.80 s.	20.98 s.
(3)	100.59 s.	19.89 s.	117.06 s.

(b) AMD Duron 800/768M

Fig. 2. A simple benchmark of the program

gs is chosen if it is nonzero. The function always terminates because the chosen polynomial is always less than the two ones we have studied.

Figure 2 presents some measurements of the program. An execution times comparaison between our program and Maple is shown in figure 2(a). Executions have been carried out on a Pentium 200MMX/48M. Running under Linux, two different versions of our program were used: gcalc and gcalcopt, the former obtained with the bytecode compiler of Objective Caml and the latter generated with the high-performance native-code compiler [7]. Running under Windows 98 with the same hardware, we execute both Maple implementation and gcalc. See below the examples that were used employing the lexicographical order.

$$\{x^{25} - y^{25}zt,\ xz^{25} - y^{25},\ x^{25}y - z^{25}t\} \tag{1}$$

$$\{x^{50} - y^{50}zt,\ xz^{50} - y^{50},\ x^{50}y - z^{50}t\} \tag{2}$$

$$\{x^{100} - y^{100}zt,\ xz^{100} - y^{100},\ x^{100}y - z^{100}t\} \tag{3}$$

In addition, figure 2(b) presents an execution comparison between both versions of our program and GAP on a AMD Duron 800/768M running under Linux. In all cases we have repeated three times each execution and the best one was selected.

5 Conclusions

We have exposed the development of an efficient functional program for computing Gröbner bases of a set of multivariate polynomials, assuring that some relevant properties hold in the program.

We suggest to develop programs using a side-effect free language, a functional language for instance, where tools like equational reasoning make sense. Proofs of properties have been carried out both in an informal and exhaustive style (on Caml programs), and in (with the help of) the Coq proof assistant.

Elaboration of proofs is based on the syntactic structure of the program. Complex proofs are carried out with help of auxiliary laws.

The developments are kept as general as possible. Different reusable modules are implemented as, for example, an efficient multivariate polynomial library.

Program efficiency is taken into account. Sometimes two different versions of a function are implemented: one efficient, the other simple, proving their equivalence. Canonical representations that allow us to use syntactical equality and efficient algorithms are used. Formalizing the canonical representation of polynomials is not complex, but we run into dificulties when defining operations which become more complicated causing more complex and tedious proofs.

Formal methods are not intended to provide *absolute* reliability, but to *increase* software reliability. Formal methods can be used to improve the design of systems, its efficiency, and to certify its correctness. It is often difficult to apply formal methods to a whole system. As future work, we should look for compositional techniques.

We think that the future of program verification heads for a general proposal: to obtain certified software libraries.

References

1. Buchberger, B.: An Algorithm for Finding a Basis for the Residue Class Ring of a Zero-Dimensional Polynomial Ideal. PhD thesis, Univ. of Innsbruck, Austria (1965)
2. Bird, R., Wadler, P.: Introduction to Functional Programming. Prentice Hall (1988)
3. Hudak, P.: Conception, evolution, and application of functional programming languages. ACM Computing Surveys **21** (1989)
4. Paulson, L.C.: ML for the Working Programmer. 2nd edn. Cambridge University Press (1996)
5. Jorge, J.S.: Estudio de la verificación de propiedades de programas funcionales: de las pruebas manuales al uso de asistentes de pruebas. PhD thesis, University of A Coruña, Spain (2004)
6. Weis, P., Leroy, X.: Le langage Caml. 2nd edn. Dunod (1999)
7. Leroy, X., et al.: The Objective Caml system: Documentation and User's Manual, Release 3.08. INRIA, `http://caml.inria.fr`. (2004)
8. The Coq Development Team: The Coq Proof Assistant Reference Manual, Version 7.3. INRIA, `http://coq.inria.fr`. (2002)
9. Bertot, Y., Casteran, P.: Interactive Theorem Proving and Program Development, Coq'Art: The Calculus of Inductive Constructions. Springer-Verlag (2004)
10. Théry, L.: A machine-checked implementation of Buchberger's algorithm. Journal of Automated Reasoning **26** (2001)
11. Medina-Bulo, I., Palomo-Lozano, F., Alonso-Jiménez, J.A., Ruiz-Reina, J.L.: Verified computer algebra in ACL2 (Gröbner bases computation). In: 7th International Conference on Artificial Intelligence and Symbolic Computation (AISC 2004). Volume 3249 of Lecture Notes in Artificial Intelligence., Springer-Verlag (2004)
12. Pérez, G.: Bases de Gröbner: Desarrollo formal en Coq. PhD thesis, University of A Coruña, Spain (2005)
13. Coquand, T., Huet, G.: The calculus of constructions. Information and Computation **76** (1988)
14. Barja, J.M., Pérez, G.: Demostración en implementaciones concretas de anillos de polinomios. RSME (2000)
15. Medina-Bulo, I., Alonso-Jiménez, J.A., Palomo-Lozano, F.: Automatic verification of polynomial rings fundamental properties in ACL2. In: 2nd International Workshop on the ACL2 Theorem Prover and Its Applications. (2000)
16. Paulson, L.C.: Constructing recursion operators in intuitionistic type theory. Journal of Symbolic Computation **2** (1986)

CheapTB: A Low Cost of Operation Distributed Filesystem

Javier París, Victor M. Gulías, and Carlos Abalde

Laboratorio de Fundamentos de Computación e Inteligencia Artificial,
MADS Group. Universidade de A Coruña
`{paris, gulias, carlos}@dc.fi.udc.es`

Abstract. The overall computing power available today has made possible for small enterprises and laboratories to develop applications that need big amounts of storage. This storage has traditionally been expensive, using propietary technologies. With the recent increase in storage space of cheap IDE and SCSI disks it is now possible to build a cluster to be used as a storage solution. CheapTB is a distributed filesystem for this environment, with an emphasys towards low cost operation rather than performance.

1 Introduction

There are several applications that have developed in recent years which need a great amount of storage space, often bigger than what a single workstation can provide. Examples of this applications are the storage of digital video, scientific experiments data or the data needed for weather forecasting.

The traditional way of solving this problem is the use of some costly storage service, like a SAN. Business which sell those systems often include maintenance so it is an adequate solution for companies and research centers without budget constraints. However, these big costs make it impossible for small business or research labs to afford them.

As the storage capacity of IDE and SCSI disk grows it is increasingly possible to put several disks in one machine. A cluster of these machines can be used as a storage service. The cost of a storage cluster like the one described is much lower than that of a SAN solution, although performance will also be lower. Still, there are many applications which need high storage space that don't have critical requirements in throughput or latency.

A distributed filesystem is needed to use all of the space offered by the disks as if it was a single disk. In CheapTB several features were required:

1. It should be accessible through some standard network filesystem protocol, like NFS[2,3] or SMB[4].
2. It should translate these accesses in data retrievals or data storage in the different disks the system has.
3. It must manage access to the disks in all of the storage nodes.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 121–130, 2005.
© Springer-Verlag Berlin Heidelberg 2005

As CheapTB controls the way data is stored in the cluster, it can control power usage by the system powering off disks and machines which are not in use. This makes the system cheaper to operate, and it probably makes the machines last longer.

Although it is possible to shut down machines in the cluster which are not in use, at least a control node must always remain on to answer to external requests. This node will be the same that translates external requests into accesses to the different nodes, as it must know where all the data is in order to know which machines must be powered on to answer an external request.

2 Context

There are several projects which have similar goals to CheapTB.

The google filesystem[5] uses a similar structure to CheapTB. There is a central node which keeps metainformation, and a big number of servers that store data in cheap disks. All request go through the master node. The main differences are the block size, which is 64Mb in GoogleFs, and that the API for applications is a library instead of common network protocols as it is in CheapTB. This makes sense for google since most of their applications are developed in-house. This way they can also design the API towards their most common operations, that will be reading sequentially and appending. Most of the metadata is kept in the storage nodes instead of in the control node. This is not practical in CheapTB as most of the time the storage nodes will be powered off.

Frangipani[7] is a distributed filesystem which also uses the concept of a virtual block device called petal[8]. This virtual block device is a bit different from the one in Cheaptb, as it is not a composition of other virtual block devices. However, frangipani works with several nodes managing the filesystem, and synchronizes them storing the metadata in petal. CheapTB keeps metadata in a control node to be able to power data storage nodes off.

GPFS[6] is a filesystem by the IBM Almaden Research Center. This filesystem works in a shared disk environment, where several disks are used by servers connected to them with some switching fabric. This means that this filesystem requieres some expensive hardware to run on. It is optimized for high performance storage, even on concurrent accesses to the same file. However, this filesystem requires hardware that is not easily affordable.

3 Design

3.1 Software Architecture

The system has three layers as shown on figure 1:

1. Client Interface: The system has a common API to which protocol adapters may be plugged. For example, NFS and SMB adapters have been developed for the system.

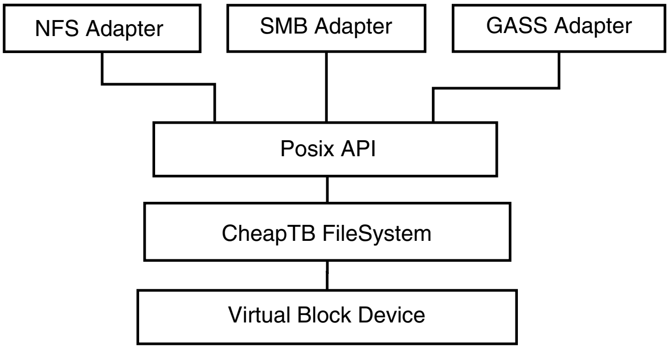


Fig. 1. CheapTB Architecture

2. File System: The file system manages the information about all the stored files and directories. This layer of the system must keep the consistence of the stored data to concurrent requests by clients.
3. Virtual Block Device: This device shows all the storage space in the different disks and machines appear as a single disk to the file system. The filesystem doesn't need to know where the data is nor how to transfer and store it.

The file system is the layer that warrants the coherence of the stored information. The metainformation is stored in a distributed database to make it easy for different nodes to access it.

3.2 Block Device

The block device hides all the storage nodes and disks and makes them appear as a single device to the upper layers. This way the filesystem doesn't need to know the distribution of the storage space available. It also makes the system independent of the storage media used.

There are two types of servers as it can be seen in figure 2:

1. Aggregators: The aggregators group the space made available by several block devices and make them appear as a single device. Using different aggregators the way the data is distributed in the cluster may be changed, or backup copies may be created.
2. Drivers: Drivers know how to store data in a certain kind of storage device, for example a disk, a tape or a file in a filesystem.

The servers must offer several services to the upper layers:

1. Writing and Reading Blocks.
2. Turning on and off a device.
3. Getting information about the device.

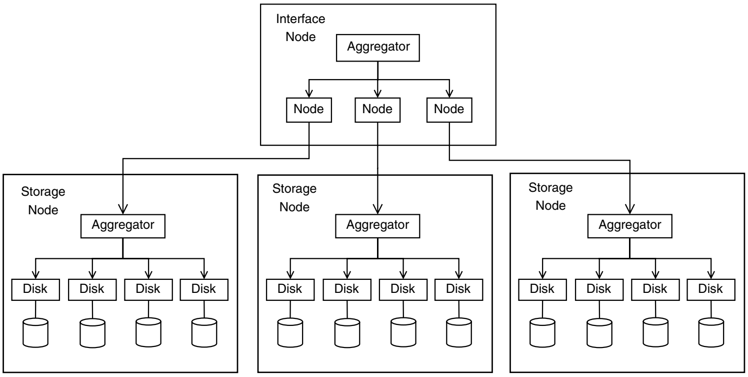


Fig. 2. Block Device Architecture

The possibility of using different aggregators helps to adapt the system to different situations. If efficiency is preferred over powersaving, the data may be aggregated using stripping. To save power, data is stored sequentially so that accesses will be localized in a set of disks.

3.3 File System Description

The CheapTB file system lays between the network protocol adapters and the block devices. The previous level offered an interface that looked like a block device. This level will offer a filesystem API to the protocol adapters.

The filesystem has several subsystems (that can be seen in figure 3) which perform all the different tasks the filesystem is responsible of:

1. Namespace Manager, which keeps information about the directory structure.
2. File Descriptor Manager, that manages access to file descriptors, and keeps copies in memory of the inodes of the open files. It also keeps the open mode and the current access offset.

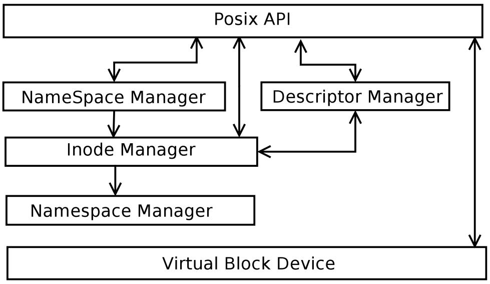


Fig. 3. File System Architecture

3. Inode Manager, which stores the inodes and the information needed to translate file block numbers (relative to file start) to disk offsets (relative to disk start).
4. Space Manager, that stores information about disk usage and manages the allocation and frees of disk blocks.

Metadata is stored in the control node and does not use the space in the storage nodes.

FileSystem API. Among the goals of CheapTB is the possibility of building adapter to allow its use with any protocol deemed necessary. To make it easy to develop adapters the filesystem has a common API to all of them.

As there are free implementations of the most commonly used protocols, it would be desirable to make them adaptable for CheapTB. The way this is achieved is offering an API similar to the one these servers are alredy using. CheapTB runs on Unix machines, so the interface is based on the part of the POSIX[1] standard that defines the API to the filesystem.

This API, as most of CheapTB, is implemented in Erlang in order to make coding servers in Erlang easier. This is good for building servers for scratch, as coding in Erlang is generally faster than in imperative languages.

As most network filesystem servers are generally coded in C, there is a C wrapper for the Erlang API. This wrapper provides an easy way to use existing servers whith CheapTB.

Protocol Adapters. Protocol adapters are the external interface of CheapTB. They provide access to CheapTB through some common protocol like NFS[2,3] or SMB[4]. The goal is that CheapTB may be used by a wide range of clients without needing to install additional software. The use of an API similar to POSIX makes adding new protocol adapters easily as they are needed. Free implementations for this protocols may be used.

To give access to CheapTB through NFS a NFSv2 implemented in Erlang was used. As it expected an API similar to POSIX it was easy to adapt. Most of the calls were just swapped for a CheapTB equivalent.

Another protocol adapter was implemented for SMB, using Samba. Samba is a SMB server for Unix systems. One of its features is its virtual file system, which lets an external module catch the calls to the filesystem. Using this vfs the accesses to disk were changed to calls to the C Wrapper of the CheapTB API.

Power Management. One of the goals of CheapTB is to minimize power consumption powering off disks and nodes that are not going to be used for a while. The approach used to manage power consumption is to distribute responsibility between the filesystem and the virtual block device.

The filesystem knows which blocks are likely to be used at a given time because it has it in the open file table and in the tables that translate file blocks to disk blocks. Using this information the filesystem can tell the virtual block device which blocks should be ready to be used soon. The virtual block device may power off disks and nodes which do not have blocks in use for a while.

The virtual block device knows how to power on and off disks and nodes. It does so either by doing it itself, or delegating it in another block device it aggregates. Using the information provided by the filesystem the virtual block device can decide which nodes may be turned off at a given time. When the filesystem notifies the virtual block device that some block may be used soon, the virtual block device will turn on the disks and nodes where they are stored.

As most reads are sequential the filesystem can notify the virtual block device in advance of the blocks that will be used. This way the disks and nodes will be powered on and off as the reads advance through the file.

4 Performance

The filesystem was tested using several common operations with different files and records sizes [1]. The test was done using three machines. Two of them had a virtual block device which aggregates two files of 2GB each. The total test space was 8 GB. The other machine had the filesystem and the samba server used as the interface of the system. All were connected with a 100Mbps ethernet switched network.

The iozone software was used to perform this test. Iozone is a filesystem performance measurement tool.

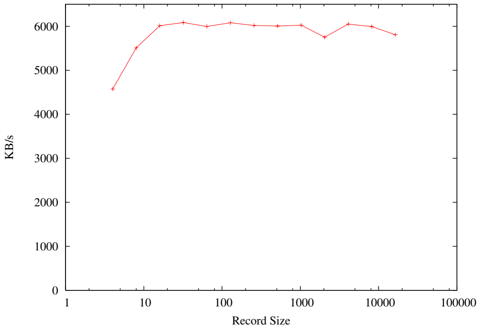


Fig. 4. Writing Throughput With a 16Mb file

4.1 Sequential Writing

Several tests were performed writing to a file sequentially through CheapTB. The files used ranged from 64 Kbytes to 512 Mbytes, and were read using records with sizes from 4 Kbytes to 16 Mbytes. In most cases the throughput grows until the record size reaches the block size (8 Kb).

---

[1] A record in this context is the size of the buffer used in each operation with the filesystem.

The test for 16Mb files can be seen in figure 4.

The result just shows that the virtual block device doesn't cache data. Performance would be mostly constant if there was a block cache because data would only be read from storage nodes once.

Better performance could be achieved if the system was not implemented using adapters. The adapters introduce an additional layer for the traffic of data in the system. In the special case of adapters implemented using the C Wrapper performance is slowed by the need of copying the data between the C program and Erlang.

4.2 Random Writing

In this test the writes to the files are not sequential but random. Any part of the created file may be accessed. Performance, as seen in figure 5 is however similar to sequential writing. This happens due to two reasons:

1. Writing of full blocks is as fast randomly as sequentially because no previous reading of the block is needed.
2. Writing of data is asynchronous, the system doesn't wait for the operation to complete.

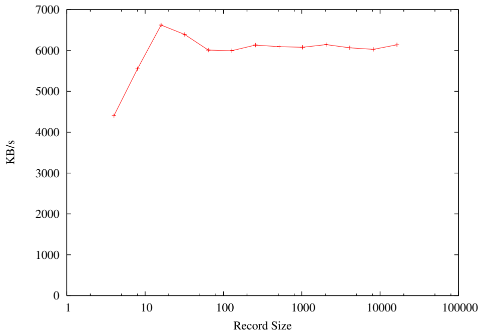


Fig. 5. Throughput in random writing

4.3 Sequential Reading

Reading performance was tested similarly to writing. The files previously created for the writing test were read to test filesystem performance. As it can be seen in figure 6 reading is slower until the record size is the same as the block size. Again, this could be enhanced using a block cache.

Performance is worse for reading than for writing. This is due to the need to read the block from the machines that store them synchronously while there is no need in writes to wait for the data to be sent to storage.

The cache could also help performance here because it would make reading blocks in advance possible. As most of the lost performance comes from the time

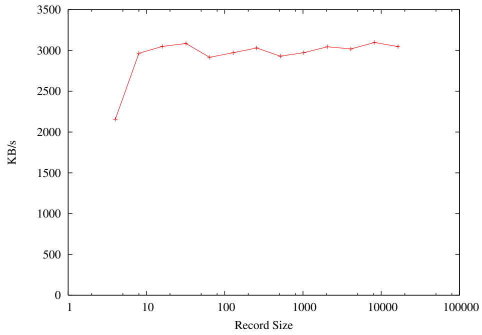


Fig. 6. Reading Throughput

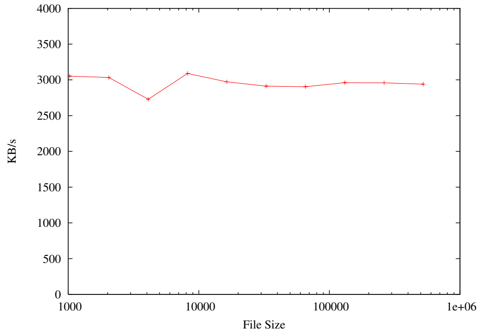


Fig. 7. Reading Throughput With a Record Size of 1 MB

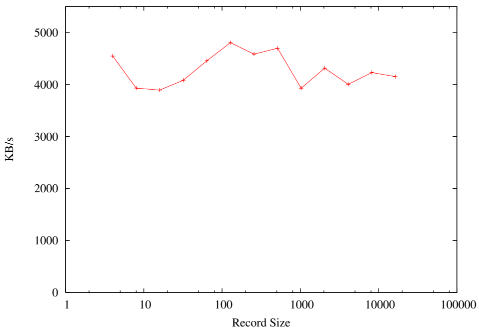


Fig. 8. Throughput in random reads

the request must wait for the data to arrive, reading data before using it would make it possible to put performance in the same levels as for writes.

Reading performance isn't affected by file size. Increasing file size doesn't cause a decrease in performance as shown in figure 7.

4.4 Random Reading

As there is no read ahead of block in CheapTB, random reading performance is a bit higher than sequential reading. The figure 8 shows this. This is because random reading may read the same block twice in the same test, which would hit the buffer cache of the filesystem. Reading blocks in advance would make performance higher for sequential reading, but it would hurt performance for random reading.

4.5 Concurrent Access

The concurrent access was tried with several proccesses writing to disk at once. The results can be seen in figure 9. Total throughput was constant through the test because the Posix API is serialized in CheapTB, and performance is likely to be independent of the number of proccesses.

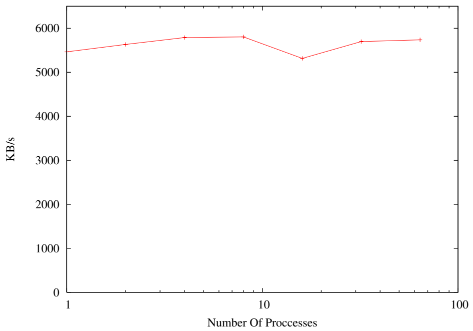


Fig. 9. Total throughput in concurrent writes

5 Conclusions

There are a set of unique points in CheapTB that are not seen in other information storage systems. The power management makes it possible to have a big cluster of machines without having it on at all times. As CheapTB is targeted to small bussines and laboratories, this is likely to be a good feature for them. The problem with power management is that it makes more difficult to have more information distributed. This is so because storage nodes cannot be trusted to hold data which may be needed at any time. The pratical effect is that a node must have the task of keeping metadata at all times.

CheapTB is completely implemented in userspace. This is not usual in similar systems, and makes it easier to debug at the cost of a loss of eficiency. The protocol adapters give flexibility in the access to the data, but also loosing efficiency. However, the possibility of using CheapTB without needing to install aditional software is something that helps making an installation easier.

There are several optimizations that could be made to the system if performance turned out to be a problem.

References

1. IEEE, "Institute of Electrical and Electronics Engineers", Standard 1003.1, 2003 Edition http://www.unix.org/single_unix_specification/
2. Sun Microsystems, Inc., NFS: Network File System Protocol Specification http://www.ietf.org/rfc/rfc1094.txt
3. B. Callaghan, B. Pawlowski, P. Staubach and Sun Microsystems, Inc., NFS Version 3 Protocol Specification, http://www.ietf.org/rfc/rfc1813.txt
4. Christopher R. Hertel, Implementing CIFS, http://www.ubiqx.org/cifs/
5. Sanjay Ghemawat, Howard Gobioff and Shun-Tak Leung, The Google Filesystem,
6. Frank Schmuck and Roger Haskin, GPFS: A Shared-Disk File System for Large Computing Clusters,
7. Chandramohan Thekkath, Timothy Mann and Edward Lee, Frangipani: A Scalable Distributed File System,
8. Edward Lee and Chandramohan Thekkath, Petal: Distributed Virtual Disks,
9. C.A. Stein and Michael Tucker and Margo Seltzer, Building a Reliable Mutable File System on Peer-to-Peer Storage,

Spelling Correction on Technical Documents*

M. Vilares[1], J. Otero[1], and J. Graña[2]

[1] Department of Computer Science, University of Vigo,
Campus As Lagoas s/n, 32004 Ourense, Spain
{vilares, jop}@uvigo.es
[2] Department of Computer Science, University of A Coruña,
Campus de Elviña s/n, 15071 A Coruña, Spain
grana@udc.es

Abstract. We describe a novel approach to spelling correction applied on technical documents, a task that requires a number of especific properties such as efficiency, safety and maintenance. In opposite to previous works, we explore the region close to the point at which the recognition halts, gathering all relevant information for the repair process in order to avoid the phenomenom of errors in cascade. Our approach seems to reach the same quality provided by the most performance classic techniques, but with a significant reduction on both time and space costs.

1 Introduction

Although much effort has gone into the problem of spelling error correction over the years, the devising estrategies remains a research challenge, and even there is a renewal interest on it due to the increasing amount of available information in electronic format or placed on line. In particular, spelling correction is a critical task in dealing with the elaboration of technical documents, for which efficiency, safety and maintenance are properties that cannot be negligented.

In opposite to fully automatic correction [4], most commercial systems assist users by offering a set of candidate corrections that are close to misspelled word. This allows to avoid wrong interpretations in a field where the meaning of a word cannot often be approximately defined and the semantic relations in the vocabulary are very strict. At this point, any technique allowing the repair process to reduce the number of candidates considered for correction translates in an improved efficiency and maintenance, that should not have side effects on safety. In order to get this last, we focus on limit the size of the repair region in the word, in contrast to previous proposals considering a global one.

2 The Operational Model

Our aim is to parse a word $w_{1..n} = w_1 \dots w_n$ according to a RG $\mathcal{G} = (N, \Sigma, P, S)$. We denote by w_0 (resp. w_{n+1}) the position in the string, $w_{1..n}$, previous to

---

* Research supported by the Spanish Government under projects TIN2004-07246-C03-01, TIN2004-07246-C03-02 and HP2002-0081, and the Autonomous Government of Galicia under projects PGIDIT03SIN30501PR and PGIDIT02SIN01E.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 131–139, 2005.
© Springer-Verlag Berlin Heidelberg 2005

w_1 (resp. following w_n). We generate from $\mathcal{G}$ a *numbered minimal acyclic finite automaton* for the language $\mathcal{L}(\mathcal{G})$. A *finite automaton* (FA) is a 5-tuple $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ where: $\mathcal{Q}$ is the set of states, Σ the set of input symbols, δ is a function of $\mathcal{Q} \times \Sigma$ into $2^{\mathcal{Q}}$ defining the transitions of the automaton, q_0 the initial state and $\mathcal{Q}_f$ the set of final states. We denote $\delta(q,a)$ by $q.a$, and we say that $\mathcal{A}$ is *deterministic* iff $\mid q.a \mid \leq 1,\ \forall q \in \mathcal{Q},\ a \in \Sigma$. The notation is transitive, $q.w_{1..n}$ denotes the state $(\overset{n-2}{\ldots}(q.w_1)\overset{n-2}{\ldots}).w_n$. As a consequence, w is *accepted* iff $q_0.w \in \mathcal{Q}_f$, that is, the *language accepted by* $\mathcal{A}$ is defined as $\mathcal{L}(\mathcal{A}) = \{w,\ \text{such that } q_0.w \in \mathcal{Q}_f\}$. A FA is *acyclic* when the underlying graph it is. We define a *path in the* FA as a sequence of states $\{q_1, \ldots, q_n\}$, such that $\forall i \in \{1, \ldots, n-1\},\ \exists a_i \in \Sigma,\ q_i.a_i = q_{i+1}$.

In order to reduce the memory requirements, we apply a minimization process [2]. In this sense, we say that two FA's are *equivalent* iff they recognize the same language. Two states, p and q, are *equivalent* iff the FA with p as initial state, and the one that starts in q recognize the same language. An FA is *minimal* iff no pair in $\mathcal{Q}$ is equivalent. Although the standard recognition is deterministic, the repair one could introduce non-determinism by exploring alternatives associated to possibly more than one recovery strategy. So, in order to get polynomial complexity, we avoid duplicating intermediate computations in the repair of $w_{1..n} \in \Sigma^+$, storing them in a table $\mathcal{I}$ of *items*, $\mathcal{I} = \{[q,i],\ q \in \mathcal{Q},\ i \in [1, n+1]\}$, where $[q,i]$ looks for the suffix $w_{i..n}$ to be analyzed from $q \in \mathcal{Q}$.

We describe our proposal using *parsing schemata* [7], a triplet $\langle \mathcal{I}, \mathcal{H}, \mathcal{D} \rangle$, with $\mathcal{H} = \{[a,i],\ a = w_i\}$ an initial set of items called *hypothesis* that encodes the word to be recognized[1], and $\mathcal{D}$ a set of *deduction steps* that allow to derive items from previous ones. These are of the form $\{\eta_1, \ldots, \eta_k \vdash \xi \,/ \mathit{conds}\}$, meaning that if all antecedents η_i are present and the conditions *conds* are satisfied, then the consequent ξ is generated. In our case, $\mathcal{D} = \mathcal{D}^{\text{Init}} \cup \mathcal{D}^{\text{Shift}}$, where:

$$\mathcal{D}^{\text{Init}} = \{ \vdash [q_0, 1] \} \qquad \mathcal{D}^{\text{Shift}} = \{ [p,i] \vdash [q, i+1] \,/ \exists [a,i] \in \mathcal{H},\ q = p.a \}$$

The recognition associates a set of items S_p^w, called *itemset*, to each $p \in \mathcal{Q}$; and applies these deduction steps until no new application is possible. The word is recognized iff a *final item* $[q_f, n+1],\ q_f \in \mathcal{Q}_f$ has been generated. We can assume, without lost of generality, that $\mathcal{Q}_f = \{q_f\}$, and that exists an only transition from (resp. to) q_0 (resp. q_f). To get this, we augment the FA with two states becoming the new initial and final states, and relied to the original ones through empty transitions, our only concession to the notion of minimal FA.

3 Spelling Correction

We talk about the *error* in a portion of the word to mean the difference between what was intended and what actually appears in the word. So, we can talk about the *point of error* as the point at which the difference occurs.

In this context, a *repair* should be understood as a modification on the input string allowing the recognizer both, to recover the standard process and to avoid

---

[1] A word $w_{1...n} \in \Sigma^+$, $n \geq 1$ is represented by $\{[w_1, 1],\ [w_2, 2],\ \ldots\ ,\ [w_n, n]\}$.

the phenomenom of cascaded errors, that is, errors precipitated by a previous erroneous repair diagnostic. That is, precisely, the goal of the notion of *regional repair* defined in [10], which we succintly introduce now.

Given that we work with acyclic FAs, we can consider a simple order relation between two states, p and q, in such a way that $p < q$ iff exists a path in the FA from p to q. We say that a pair of states (p, q) is a *region* in the FA when it defines a sub-automaton with initial (resp. final) state in p (resp. q). So, we say that a state r is in the region defined by the pair (p, q), denotated by $\mathcal{R}_p^q$, iff there exists a path ρ in $\mathcal{R}_p^q$, such that $r \in \rho$. Given $r \in \mathcal{Q}$, it can be proved that there exists an only *minimal region* in the FA containing it.

To begin with, we assume that we are dealing with the first error detected. We extend the initial estructure of items, as a pair $[p, i]$, with an error counter e; resulting in a new estructure of the form $[p, i, e]$. For a given *point of error*, w_j, we locate the associated *point of detection*, defined as the source of the minimal region, $\mathcal{M}(w_j) = \mathcal{R}_p^q$, containing w_j. Associated to the point of error, w_j, (resp. point of detection, w_i) we consider the corresponding *error* (resp. *detection*) *item* iff is of the form $[q, j]$ (resp. $[p, i]$). To filter out undesirable repairs, we introduce criteria to select those with minimal cost. For each $a, b \in \Sigma$ we assume insert, $I(a)$; delete, $D(a)$, replace, $R(a, b)$, and transpose, $T(a, b)$, costs. Once the detection item has been fixed, we apply from it the deduction steps $\mathcal{D}_{\text{error}} = \mathcal{D}^{\text{Shift}} \cup \mathcal{D}_{\text{error}}^{\text{Insert}} \cup \mathcal{D}_{\text{error}}^{\text{Delete}} \cup \mathcal{D}_{\text{error}}^{\text{Replace}} \cup \mathcal{D}_{\text{error}}^{\text{Transpose}}$, defined as follows:

$$\begin{aligned}
\mathcal{D}^{\text{Shift}} &= \{[p, i, e] \vdash [q, i+1, e],\ \exists [a, i] \in \mathcal{H},\ q = p.a\} \\
\mathcal{D}_{\text{error}}^{\text{Insert}} &= \{[p, i, e] \vdash [p, i+1, e + I(a)]\} \\
\mathcal{D}_{\text{error}}^{\text{Delete}} &= \{[p, i, e] \vdash [q, i, e + D(w_i)] \Big/ \begin{array}{l} \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}_{q_s}^{q_d} \\ p.w_i = q \in \mathcal{R}_{q_s}^{q_d} \textbf{ or } q = q_d \end{array} \} \\
\mathcal{D}_{\text{error}}^{\text{Replace}} &= \{[p, i, e] \vdash [q, i+1, e + R(w_i, a)], \Big/ \begin{array}{l} \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}_{q_s}^{q_d} \\ p.a = q \in \mathcal{R}_{q_s}^{q_d} \textbf{ or } q = q_d \end{array} \} \\
\mathcal{D}_{\text{error}}^{\text{Transpose}} &= \{[p, i, e] \vdash [q, i+2, e + T(w_i, w_{i+1})] \Big/ \begin{array}{l} \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}_{q_s}^{q_d} \\ p.w_{i+1}.w_i = q \in \mathcal{R}_{q_s}^{q_d} \textbf{ or } q = q_d \end{array} \}
\end{aligned}$$

where $w_{1..j}$ looks for the current point of error. Note that, in any case, the error hypotheses apply on transitions behind the repair region. The process continues until a repair covers that region, accepting a character in the remaining string. To avoid the generation of items only differenciated by the error counter, we apply a principle of optimization, saving only for computation purposes those with minimal counters.

When the current repair is not the first one, it can modify a previous repair in order to avoid cascaded errors, by adding the cost of the new error hypotheses to profit from the experience gained from previous ones. This allows us to get, in a simple manner, an asymptotic behavior close to global repair methods [10]. The time complexity is, in the worst case

$$\mathcal{O}(\frac{n!}{\tau! * (n-\tau)!} * (n+\tau) * 2^{\tau} * \text{fan-out}_{\mu}^{\tau})$$

where τ and *fan-out*$_{\mu}$ are, respectively, the maximal error counter computed and the maximal fan-out of the automaton in the scope of the repairs considered. The input string is recognized iff a final item $[q_f, n+1, e]$, $q_f \in \mathcal{Q}_f$, is generated.

4 The Experimental Frame

Our aim now is to validate the practical interest of our proposal in relation to classic global repair strategies. We think that this is an objective criterion since the point of reference is a technique that guarantees the best quality for a given error metric when all contextual information is available.

4.1 The Running Language

Our running language is Spanish, with a great variety of morphological processes, making it adequate for our description. The most outstanding features are to be found in verbs, with a highly complex conjugation paradigm, including nine simple tenses and nine compound tenses, all of which have six different persons. If we add the present imperative with two forms, the infinitive, the compound infinitive, the gerund, the compound gerund, and the participle with four forms, then 118 inflected forms are possible for each verb. In addition, irregularities are present in both stems and endings. So, very common verbs, such as `hacer` (*to do*), have up to seven different stems: `hac-er`, `hag-o`, `hic-e`, `har-é`, `hiz-o`, `haz`, `hech-o`. Approximately 30% of Spanish verbs are irregular, and can be grouped around 38 different models. Verbs also include enclitic pronouns producing changes in the stem due to the presence of accents: `da` (*give*), `dame` (*give me*), `dámelo` (*give it to me*). We have considered forms with up to three enclitic pronouns, like `tráetemelo` (*bring it for you and me*). There exist some highly irregular verbs that cannot be classified in any irregular model, such as `ir` (*to go*) or `ser` (*to be*); and others include gaps in which some forms are missing or simply not used. For instance, meteorological verbs such as `nevar` (*to snow*) are conjugated only in third person singular. Finally, verbs can present duplicate past participles, like `impreso` and `imprimido` (*printed*).

This complexity extends to gender inflection, with words considering only one gender, such as `hombre` (*man*) and `mujer` (*woman*), and words with the same form for both genders, such as `azul` (*blue*). In relation to words with separate forms for masculine and feminine, we have a lot of models: `autor`, `autora` (*author*); `jefe`, `jefa` (*boss*); `poeta`, `poetisa` (*poet*); `rey`, `reina` (*king*) or `actor`, `actriz` (*actor*). We have considered 20 variation groups for gender. We can also refer to number inflection, with words presenting only the singular form, as `estrés` (*stress*), and others where only the plural form is correct, as `matemáticas` (*mathematics*). The construction of different forms does not involve as many variants as in the case of gender, but we can also consider a certain number of models: `rojo`, `rojos` (*red*); `luz`, `luces` (*light*); `lord`, `lores` (*lord*) or `frac`, `fraques` (*dress coat*). We have considered 10 variation groups for number.

4.2 The Corpus

We choose to work with the Itu corpus[2], the main collection of texts about telecommunications. In particular, we have considered a sub-corpus of 17.423

---

[2] For *International Telecommunications Union CCITT Handbook.*

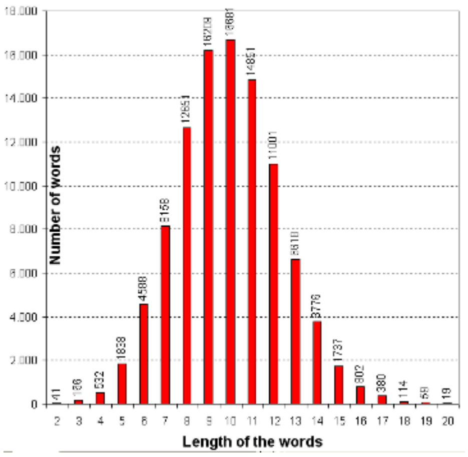


Fig. 1. Statistics on the lexicon

words that have been used as a part of the support for the Crater project [1]. The recognizer is a *finite automaton* (fa) containing 11.193 states connected by 23.278 transitions built from Galena [3]. The distribution, in terms of lengths of the words dealt with, is shown in Fig. 1.

For each length-category, errors have been randomly generated in a number and position for the first error in the input string that are shown in Fig. 2. This is of importance since, as the authors claim, the performance of previous proposals depend on these factors, which has no practical sense. No other dependencies, for example in terms of lexical categories, have been detected at morphological level and, therefore, they have not been considered.

In this context, our testing framework seems to be well balanced, from both viewpoints operational and linguistic, in order to estimate the practical performance of error repair algorithms on fa architectures. It only remains to decide what repair algorithms will be tested. We choose to compare our proposal with the Savary's global approach [6], an evolution of the Oflazer's algorithm [5] and, in the best of our knowledge, the most efficient method of error-tolerant look-up in finite-state dictionaries. The comparison has been done from three complementary viewpoints: the size of the repair region considered, the computational cost and the quality exhibed.

4.3 The Computational Cost

Practical results are compiled in Fig. 2, using as unity to measure the computational effort the concept of item previously defined. In order to take the *edit distance* [5] as the error metric for measuring the quality of a repair, it is sufficient to consider discrete costs $I(a) = D(a) = 1, \ \forall a \in \Sigma$ and $R(a,b) = T(a,b) = 1, \ \forall a, b \in \Sigma, \ a \neq b$. We here consider two complementary approaches illustrating the dependence on both the position of the first point of error in the

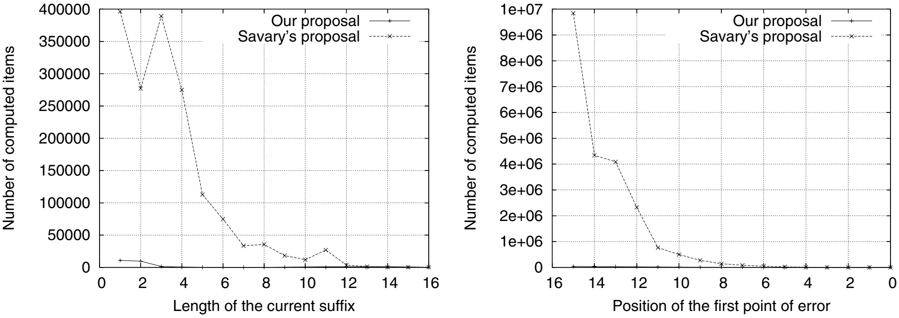


Fig. 2. Number of items generated in error mode

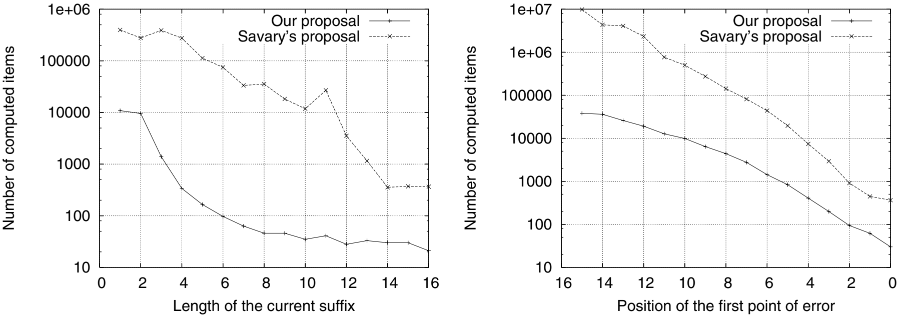


Fig. 3. Number of items generated in error mode. Logarithmic scale.

word and the length of the suffix from it. So, in any case, we are sure to take into account the degree of penetration in the FA at that point, which determines the effectiveness of the repair estrategy. In effect, working on regional methods, the penetration determines the number of regions in the FA including the point of error and, as a consequence, the possibility to consider a non-global resolution.

In order to clearly show the detail of the tests on errors located at the end of the word, which is not easy to observe from the decimal scale of Fig. 2, we include in Fig. 3 the same results using a logarithmic scale. So, both graphics perfectly illustrate our apportation, in terms of computational effort saved, from two viewpoints which are of interest in real systems: First, our proposal shows in practice a linear-like behavior, in opposite to the Savary's one that seems to be of exponential type. In particular, this translates in an essential property in industrial applications, the independence of the the time of response on the initial conditions for the repair process. Second, in any case, the number of computations is significantly reduced when we apply our regional criterion.

4.4 The Performance

However, statistics on computational cost only provide a partial view of the repair process that must also take into account data related to the performance from both the user's and the system's viewpoint. In order to get this, we have introduced the following two measures, for a given word, w, containing an error:

$$performance(w) = \frac{useful\ items}{total\ items} \qquad recall(w) = \frac{proposed\ corrections}{total\ corrections}$$

that we complement with a global measure on the *precision* of the error repair approach in each case, that is, the rate reflecting when the algorithm provides the correction attended by the user. We use the term *useful items* to refer to the number of generated items that finally contribute to obtain a repair, and *total items* to refer to the number of these structures generated during the process. We denote by *proposed corrections* the number of corrections provided by the algorithm, and by *total corrections* the number of possible ones, absolutely.

These results are shown in Fig. 4, illustrating some interesting aspects in relation with the asymptotic behavior we want to put into evidence in the regional approach. So, considering the running example, the performance in our case is not only better than Savary's; but the existing difference between them increases with the location of the first point of error. Intuitively this is due to the fact that closer is this point to the beginning of the word and greater is the number of useless items generated in error mode, a simple consequence of the higher availability of different repair paths in the FA when we are working in a region close to q_0. In effect, given that the concept of region is associated to the definition of corresponding source and drain points, this implies that this kind of regions are often equivalent to the total one since the disposition of these regions is always concentric. At this point, regional and repair approaches apply the same error hypotheses not only on a same region, but also from close states given that, in any case, one of the starting points for these hypotheses would be q_0 or a state close to it. That is, in the worst case, both algorithms converge.

The same reasoning could be considered in relation to points of error associated to a state in the recognition that is close to q_f, in order to estimate the repair region. However, in this case, the number of items generated is greater for the global technique, which is due to the fact that the morphology of the language often results on the generation of regions which concentrate near of q_f, a simple consequence of the common derivational mechanisms applied on suffixes defining gender, number or verbal conjugation groups. So, it is possible to find a regional repair just implicating some error hypotheses from the state associated to the point of error or from the associated detection point and, although this regional repair could be different of the global one; its computational cost would be usually minor.

A similar behavior can be observed with respect to the recall relation. Here, Savary's algorithm shows a constant graph since the approach applied is global and, as consequence, the set of corrections provided is always the entire one for a fixed error counter. In our proposal, the results prove that the recall is smaller than for Savary's, which illustrates the gain in computational efficiency in opposite to the global method. Related to the convergence between regional and global approaches, we must again search around points of detection close to

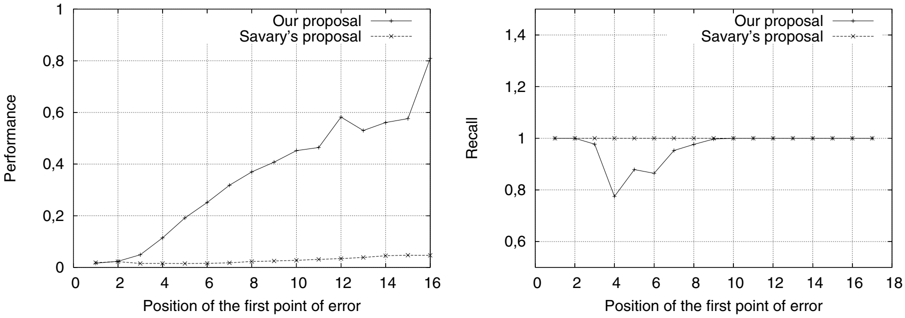


Fig. 4. Performance and recall results

the beginning of the word, which often also implies repair regions be equivalent to the total one and repairs starting around of q_0, such as is illustrated in Fig. 4.

However, in opposite to the case of performance, we remark that for recall the convergence between global and regional proposals seems also extend to processes where the point of error is associated to states close to q_f, that is, when this point is located near of the end of the word. To understand this, it is sufficient to take into account that we are not now computing the number of items generated in the repair, but the number of corrections finally proposed. So, given that closer to the end of the word we are and smaller is the number of alternatives for a repair process, both global and regional approaches converge also towards the right of the graph for recall.

Finally, the regional (resp. the global) approach provided as correction the word from which the error was randomly included in a 77% (resp. 81%) of the cases. Although this could be interpreted as a justification to use global methods, it is necessary to remember that we are now only taking into account morphological information, which has an impact in the precision for a regional approach, but not for a global one that always provide all the repair alternatives without exclusion. So, the consideration of the precision concept represents, in the exclusive morphological context considered, a clear disadvantage for our proposal since it bases its efficiency in the limitation of the search space. We attend that the integration of linguistic information from both, syntactic and semantic viewpoints will reduce significantly this gap, less than 4%, around the precision; or even will eliminate it.

5 Conclusion

We have illustrated how a least-cost regional error repair technique can be applied to the task of spelling correction on technical texts, a field where isolated-word estrategies can be preferable to context-sensitive ones and, as a consequence, morphological aspects strongly impact the performance.

Our proposal adapts to any FA-based frame and no particular requeriments are needed to put it into running. The system was evaluated against a set of texts artificially generated from the ITU corpus in telecommunications, and preliminary results seem to indicate that the system can be used for removing many of the lexical errors in the input of a technical document.

References

1. Thorsten Brants. Some experiments with the CRATER corpus. Technical report, Universität des Saarlandes, Saarbrücken, 1995.
2. J. Daciuk, S. Mihov, B.W. Watson, and R.E. Watson. Incremental construction of minimal acyclic finite-state automata. *Computational Linguistics*, 26(1):3–16, 2000.
3. J. Graña, F.M. Barcala, and M.A. Alonso. Compilation methods of minimal acyclic automata for large dictionaries. *Lecture Notes in Computer Science*, 2494:135–148, 2002.
4. K. Kukich. Techniques for automatically correcting words in text. *ACM Computing Surveys*, 24(4):377–439, December 1992.
5. K. Oflazer. Error-tolerant finite-state recognition with applications to morphological analysis and spelling correction. *Computational Linguistics*, 22(1):73–89, 1996.
6. A. Savary. Typographical nearest-neighbor search in a finite-state lexicon and its application to spelling correction. *Lecture Notes in Computer Science*, 2494:251–260, 2001.
7. K. Sikkel. *Parsing Schemata*. PhD thesis, Univ. of Twente, The Netherlands, 1993.
8. J. Vilares, F.M. Barcala, and M.A. Alonso. Using syntactic dependency-pairs conflation to improve retrieval performance in spanish. *Lecture Notes in Computer Science*, 2276:381–390, 2002.
9. J. Vilares, D. Cabrero, and M.A. Alonso. Applying productive derivational morphology to term indexing of Spanish texts. *Lecture Notes in Artificial Intelligence*, 2004:336–348, 2001.
10. M. Vilares, J. Otero, and J. Graña. Regional finite-state error repair. In *Proc. of the Ninth Int. Conf. on Implementation and Application of Automata (CIAA'04)*, Kingston, Canada, 2004.

Verification of Language Based Fault-Tolerance

Clara Benac Earle[1] and Lars-Åke Fredlund[2,3]

[1] Computing Laboratory, University of Kent, England
[2] LSIIS, Facultad de Informática, Universidad Politécnica de Madrid*
[3] Swedish Institute of Computer Science, Sweden

Abstract. In this paper we target the verification of fault tolerant aspects of distributed applications written in the Erlang programming language. Erlang programmers mostly work with ready-made language components. Our approach to verification of fault tolerance is to verify systems built using a central component of most Erlang software, a generic server component with fault tolerance handling.

To verify such Erlang programs we automatically translate them into processes of the μCRL process algebra, generate their state spaces, and use a model checker to determine whether they satisfy correctness properties specified in the μ-calculus.

The key observation of this paper is that, due to the usage of these higher-level design patterns, the state space generated from a Erlang program, even with failures occurring, is relatively small, and can be generated automatically.

1 Introduction

As software based critical systems are now becoming widely deployed, it is a crucial development task to analyse whether such systems will survive the inevitable faults (in hardware, in network links, in software components) that occur. However, this remains very difficult. It is hard to develop realistic failure models for simulation and testing (often the environment characteristics are not fully known), and to test and simulate for all possible faults would be very time consuming. Consequently, here is an area where there is a need for formal verification. But that too is hard. Most earlier work on verifying fault-tolerance target a single application only, are ad-hoc, and do not provide a reusable verification method. In this paper, instead, we propose a verification method based on model checking, that, since it addresses programs developed using higher-level design patterns which address fault-tolerance in a structured way, can be reused for a large set of such applications.

Erlang is a programming language developed at the Ericsson corporation for implementing telecommunication systems [1]. It provides a functional sub-language, enriched with constructs for dealing with side effects such as process creation and inter–process communication. Today many commercially available products offered by Ericsson are at least partly programmed in Erlang. The software of such products is typically organised into many, relatively small, source modules, which at runtime execute

---

* The second author was supported by an ERCIM grant and a Spanish Governmental Grant from the Ministerio de Educación Y Ciencia, with reference SB2004–0195.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 140–149, 2005.
© Springer-Verlag Berlin Heidelberg 2005

as a dynamically varying number of processes operating in parallel and communicating through asynchronous message passing. The highly concurrent and dynamic nature of such software makes it particularly hard to debug and test. We therefore explore the alternative of software verification based on a formal proof system.

A key feature of the systems for which Erlang was primarily created is fault-tolerance. Switching systems should provide an acceptable level of service in the presence of faults. Erlang implements fault-tolerance in a simple way. Links between two processes A and B can be set up so that process B is notified of the termination of process A and vice versa. The default behaviour of a process that is informed of the abnormal termination (e. g., due to an exception) of another process is to terminate abnormally itself, although this behaviour can be modified. This process linking feature can be used to build hierarchical process structures where some processes are supervising other processes, for example restarting them if they terminate abnormally.

We start, in Section 2, by explaining the software components that are used to build quality Erlang software. The basic mechanisms for error handling in Erlang are described in Section 3. In Section 4, we describe how the generic server component of Erlang is extended with fault tolerance, and the actual translation from Erlang to μCRL is given in Section 5. The checking of correctness properties of fault-tolerant systems is discussed in Section 6, where as an example we analyse mutual exclusion and starvation properties of a server implementing a locking service for a number of client processes.

2 Erlang Components

A key aspect of the Erlang approach to development is the use of design patterns (provided by Erlang/OTP) which are encapsulated in terms of generic components. This approach simplifies the development cycle, as well as our verification of fault-tolerance.

Erlang/OTP provides a convenient component, the *generic server*, for programming server processes. A server is a process that waits for a message from another process, computes a response message and sends that back to the original process. Normally the server will have an internal state, which is initialised when starting the server and updated whenever a message has been received.

The behaviour module (`gen_server`) implements the common parts of a generic server process, providing a standard set of interface functions, for example, the function `gen_server:call` for synchronous communication with the server. The specific parts of the concrete client-server system are given in a call-back module.

We illustrate the functionality provided by the generic server component using a server in Figure 1 which also serves to introduce the concrete Erlang syntax. Informally the server implements a locking facility for a set of client processes. A client can acquire the lock by sending a `request` message, and release it using a `release` message.

Names of functions and atoms begin with a lowercase letter, while variables begin with an uppercase letter. The usual data types are provided, e.g., lists, tuples (enclosed in curly braces) and numbers. Matching a value against a sequence of patterns, which happens in function applications and in the `case` expression, is sequential.

A programmer that uses the generic server component essentially has to provide two functions: `init` which is invoked when the generic server starts, and which should

```
init(A) -> {ok,[]}.

handle_call(request, Client, Pending) ->
    case Pending of
         [] -> {reply, ok, [Client]};
         _  -> {noreply, Pending ++ [Client]}
    end;
handle_call(release, Client, [_|Pending]) ->
    case Pending of
         [] -> {reply, done, []};
         _  -> gen_server:reply(hd(Pending), ok), {reply, done, Pending}
    end.
```

Fig. 1. The source code of an Erlang generic server

return the initial state of the server (the empty list in the example), and `handle_call` which is invoked when a call is made to the generic server, and a reply is expected by the caller. The `handle_call` function is invoked with three arguments, the message submitted in the call, a value that is used to reply to the message, and the current state of the generic server. It returns the new state of the server upon completion. The processing of calls by a generic server is sequential, i.e., there are never concurrent invocations of the callback functions; a generic server thus offers a convenient way of controlling the amount of concurrency in an application and of protecting the state of the server.

In the example the server may be called with a `request` or a `release` message. If the message is a request, and if `Pending` is the empty list, it replies to the caller with the atom `ok`, and the new state of the server is `[Client]`. If `Pending` is not empty, then the reply is postponed (until more messages arrive) and the new state of the server is obtained by adding `Client` to the end of `Pending`. In case of a `release`, the server may issue a reply to the waiting caller, using `gen_server:reply`.

Client processes use a uniform way of communicating with the server; when a reply is expected they issue a call `gen_server:call(Locker, Message)` where `Locker` is the process identifier of a generic server. The client process suspends until a value is returned by the server.

Note that the semantics of communication using the generic server component is less complex that the communication paradigm of the underlying Erlang language. Generic servers always receive messages sequentially, i.e., in FIFO (first-in first-out) order. Erlang processes in contrast can potentially receive messages sent to them in arbitrary order. Thus by focusing on the higher-level components, rather than the underlying language primitives, our verification task becomes easier (concretely, state spaces are reduced). We will see the same thing happening when considering fault tolerance.

3 Fault-Tolerance in Erlang

In Erlang, bidirectional links are created between processes by invoking the `link` function with the process identifier of the process to link to as argument. There is also a function `spawn_link` which atomically both spawns a new process, and creates a bidirectional link to it.

Terminating processes will emit exit signals to all linked processes. Erlang distinguishes between normal process termination (the toplevel function of the process returned a value) from abnormal process termination (e.g. a runtime error such as attempting to divide by zero). If a process terminates abnormally, linked process will by default terminate abnormally as well. However, a linked process can trap exit such exit signals, and thus escape termination, by calling `process_flag(trap_exit,true)`.

In this case, when an exit signal reaches the process it is transformed into an exit message and delivered to the process mailbox like any other message. Exit messages are of the form {`'EXIT',Pid,Reason`}, with `Pid` the process identifier of the process that terminated, and `Reason` the reason for termination. If a process terminates normally `Reason` is equal to `normal`.

This basic mechanism of Erlang for error handling is exploited by the Erlang generic server behaviour in order to build fault-tolerant client-server systems. The Erlang programmer that implements a server process using the generic server component has to take several possible types of faults into account. First, the server itself may be faulty and crash. Recovery should be implemented by designating a supervisor process that restarts the server process (or takes some other corrective action).

Another error condition occurs when the server may communicate with remote processes, or hardware devices, that can malfunction without crashing, and moreover without generating exit signals to linked processes. Such error conditions should be handled in a traditional manner using timeouts. We focus instead on the error condition when an explicit exit signal reaches the generic server process. For the Erlang programmer such signals are handled by providing a new callback function, `handle_info(Signal,State)` that gets passed the exit signal as argument, together with the current state of the server. The `handle_info` function should, similarly to the other callback functions, either return the new state of the server or stop. This function will be called only if no call to the server is being processed.

In the client-server applications that we want to verify using the fault-tolerant extension, the state of the server contains information about the state in which its clients are in, for example, in the locker in Figure 1, the state of the locker reflects whether a client is accessing a resource or whether is waiting to get access to it. If a client terminates abnormally, the system should be able to recover gracefully without a complete restart, i.e., the state of the server process should be cleaned up accordingly.

4 Fault-Tolerance in Generic Servers

Our goal is to check the correctness of generic servers in the presence of crashing clients. The class of servers that we can analyse for fault tolerance have the following characteristics: (i) the server expects to receive an exit message whenever a linked client crashes, and (ii) the server establishes a process link to every client that issues a generic server call to it.

Although the above conditions may appear arbitrary, they are in fact indicative of a class of servers that safely implement a stateful protocol between itself and its clients, through call and reply exchanges. Thus, in a sense, these conditions give rise to a new Erlang high-level component which refines the basic Erlang generic server component.

As an example of a fault-tolerant server let us reconsider the simple server in Figure 1. The main loop of a client that accesses the locker is given below. Every client process sends a `request` message followed by a `release` message.

```
loop(Locker) ->
    gen_server:call(Locker, request),
    gen_server:call(Locker, release),
    loop(Locker).
```

We implement a locker which recovers from the abnormal termination of a client process by first adding the functions `process_flag` and `link` to the call-back module of the locker given in Figure 1 as shown below.

```
init(A) -> process_flag(trap_exit,true), {ok,[]}.

handle_call(request, {ClientPid,Tag}, Pending) ->
    link(ClientPid),
    case Pending of
         [] -> {reply, ok, [Client]};
         _  -> {noreply, Pending ++ [Client]}
    end;
```

The locker process now gets linked to the clients when they request a resource. If a client crashes, the locker will receive an exit message. As previously mentioned, exit messages are handled by the generic server function `handle_info` provided by the Erlang generic server behaviour. A trivial implementation of this function just returns the state of the server.

```
handle_info({'EXIT',ClientPid,Reason},Pending) -> {noreply, Pending}.
```

Now, if a client process crashes immediately after sending the `request` message to the locker, then the locker will process the `request` message before the exit signal. If there the resource is available, then the locker will send an `ok` message to the client that crashed and will put the client in the pending list. Since this client has crashed, it cannot release the resource, therefore, all other clients requesting the resource are put in the pending list and will eventually starve. If the resource is not available, the client will be put in the pending list, and when the resource is available, we have the same starving situation described before. Starvation also occurs if the client crashes while accessing the resource and before releasing it. However, if the client crashes after releasing, then the program behaves correctly. Of course, more than one client process may crash, therefore, we need to consider all the combinations of clients crashing at different points in the program execution. Already we can see that testing fault-tolerant code for a simple protocol like the one presented here is quite complex. Our goal is to use a high-level language, a process algebra, and use tools to automatically generate all these combinations and to check that key properties, deadlock-freedom, mutual exclusion, and non-starvation, are fulfilled.

The implementation of the `handle_info` function for the locker is given below.

```
handle_info({'EXIT',ClientPid,Reason},Pending) ->
    NewPending = remove(ClientPid,Pending),
    case available(ClientPid,Pending) of
         true -> gen_server:reply(hd(NewPending), ok),
                 {noreply, NewPending};
```

```
          _    -> {noreply, NewPending}
    end.

remove(ClientPid,[]) -> [];
remove(ClientPid,[{ClientPend,TagPending}|Rest]) ->
      case ClientPid == ClientPend of
        true  -> Rest;
        false -> [{ClientPend,TagPending}|remove(ClientPid,Rest)]
      end.

available(ClientPid,[]) -> false;
available(ClientPid,[{ClientPend,TagPending}]) -> false;
available(ClientPid,[{ClientPend,TagPending}|Rest]) -> ClientPid == ClientPend.
```

When the locker receives an exit message, i.e., a client process has terminated abnormally, then if the client is in the pending list, then it is removed from it. Moreover, if the client was accessing the resource (i.e., it was in the head of the pending list), then, the resource is available and therefore the locker gives access to the resource to a client which was waiting for it. This is similar to when a client sends a `release` message.

5 Translating Fault-Tolerant Systems to μCRL

In this section we briefly review the translation to μCRL of Erlang fault-tolerant client-server systems, full details are provided in [3].

For the purpose of verification Erlang programs are translated into the μCRL process algebra [5] by an automatic translator tool [2]. In μCRL behaviour is described on two levels, as traditional process behaviour using the process algebra operators of μCRL (sequencing, parallel composition, recursion, communication using synchronisation, etc), and data kept by processes and exchanged in communications. Functions can be defined over data types using rewrite rules.

The translation of Erlang mimics the separation between process behaviour and functional behaviour present in μCRL. A pre-analysis step partitions Erlang functions into two categories: the ones with pure functional computation, and the ones with side effects (e.g., communication to/from a generic server). The side-effect free Erlang functions are translated into μCRL functions, which are defined using a set of rewrite rules. Thus such Erlang functions do not generate any state. In contrast the side-effect Erlang functions are translated into μCRL processes, using the process operators.

The translation of communications with a generic server uses an intermediate buffer process implemented in μCRL, which stores sent messages until the translated generic server process is ready to receive them. Thus the asynchronous nature of communication in Erlang is kept in the translated code. The translation of non-tail recursive side-effect functions uses an explicit call-stack to keep track of recursive calls.

Which processes (e.g., generic servers and clients) to translate is computed by analysing the code for setting up the system. The generic server processes are found by analysing which processes initially execute a function in a module with the generic server behaviour attribute.

The fault-tolerant extension of Erlang only affects the process part of Erlang, hence, the translation of the functional part of fault-tolerant Erlang remains the same. For the process part, the fault-tolerant extension of Erlang assumes that a server expects to receive an exit message in its mailbox whenever a linked client crashes, and that this exit message is received and handled by the generic server primitive `handle_info`. The translation to μCRL therefore needs to take into account this implicit communication between the client and the server, and the translation of the `handle_info` function.

The μCRL toolset [4] is used to generate a state space from the μCRL translation. Obviously, the state space generated for a client-server system with this client process is larger than the one where the client cannot crash. For example, the state space generated in a scenario with two client processes which cannot crash contains 33 states and 48 transitions, while the state space for the same scenario with crashing clients consists of 326 states and 584 transitions.

6 Model Checking Properties in Fault-Tolerant Systems

Once the labelled transition system has been generated by the μCRL toolset from the μCRL specification (the result of translating the Erlang program), the CADP toolset is used to check whether safety and liveness properties hold. Such correctness properties are formulated in the regular alternation free μ-calculus [8, 7]. Informally, the modalities in the logic are relaxed to sequences of actions characterised by regular expressions.

Action label are enclosed in quotes (e.g., $'\texttt{crash}'$) and can contain wildcards (e.g., $'.*\texttt{crash}.*'$ matches any action that has the text string `crash` somewhere in its name), $\neg \mathit{regaction}$ matches any action that does not match the action regular expression $\mathit{regaction}$, $\mathit{regaction}_1 \vee \mathit{regaction}_2$ is disjunction. Actions can be composed using the normal regular expression operators, i.e., $|$ denotes alternative, $*$ zero or more occurencies, . is sequencing, and $-$ matches any action. Comments can be enclosed in formulas using the `(* comment *)` notation.

6.1 Deadlock Freedom

Since we model crashing of client processes, actually we are introducing deadlock states. To verify that a client-server system is deadlock-free except for the states where all clients have crashed, we formulate a fault-tolerant version of the classical deadlock-freedom property. The property we are interested in states that no deadlocks occurs as long as not all the processes in the system have crashed. This property can be expressed by explicitly stating the crash actions in the formula.

For instance, supposing there are three processes in the system. Then we define a action sequence macro denoting the sequences containing 0, 1, or 2 crashes:

```
BETWEEN_0_AND_2_CRASHES() =
 ((¬ . info.  )  (* 0 crashes *)                          |
 (¬ . info.    . info.  .(¬ . info.  )  (* 1 crash *)      |
 (¬ . info.   . . info.  .(¬ . info.  ) . . info.  .(¬ . info.  ) ))
```

Using the macro, the deadlock freedom property becomes:

$$[\mathit{BETWEEN\_0\_AND\_2\_CRASHES}()]\langle - \rangle \mathit{true}$$

This formula will spot the deadlocks unrelated to complete crashes of the system. In general, for N processes in the system, one must write N-1 lines of the form $('.^*\texttt{info}.^*'.(\neg'.^*\texttt{info}.^*')^*)$ in the macro above.

This example highlights the need to reconsider the properties used to verify nonfault-tolerant systems in order to verify fault-tolerant systems. In the following two subsections we discussed how mutual exclusion and non-starvation can be verified.

6.2 Mutual Exclusion

The formulation of the mutual exclusion property for the non-fault-tolerant locker is given below. To make verification easier two actions are introduced in the Erlang code of the client to signal the entering (`use`) and the exiting (`free`) of the critical section.

$$\begin{gathered} BETWEEN(\mathsf{a}_1,\ \mathsf{a}_2,\ \mathsf{a}_3) = [\text{-}\ \ .\ \mathsf{a}_1\ .\ (\neg \mathsf{a}_2)\ \ .\ \mathsf{a}_3]\mathit{false} \\ MUTEX() = BETWEEN(\ \texttt{use(.\ )}\ ,\ \texttt{free(.\ )}\ ,\ \texttt{use(.\ )}\ ) \end{gathered}$$

The formula states that 'on all possible paths, after an `use` action, any further `use` action must be preceded by an `free` action'. Intuitively, the formula means that if a client process is accessing the resource, then no other client process can access it until the resource has been freed. This formula does not hold in the state space generated for the a scenario with two crashing clients. The CADP model checker gives the following counter-example.

```
"call(locker,request,C1)"
"reply(C1,ok,locker)"
"action_use(C1)"
"info(locker,{EXIT,C1,EXIT))"
"call(locker,request,C2)"
"reply(C2,ok,locker)"
"action_use(C2)"
```

The counter-example shows that the mutual exclusion property is violated, since the resource is accessed by two process clients, client 1 and client 2, without being freed. However, the counter-example is also showing that, client 2 is accessing the resource after client 1 has crashed, therefore, strictly speaking, client 1 is not accessing the resource because it is dead.

In order to show that the mutual exclusion property is verified in the fault-tolerant first version of the locker case-study, we need to take the client crashes into account, as is done in the property below.

$$\begin{gathered} FT - BETWEEN(\mathsf{a}_1,\ \mathsf{a}_2,\ \mathsf{a}_3,\ \mathsf{a}_4) = [\text{-}\ \ .\ \mathsf{a}_1\ .\ (\neg \mathsf{a}_2 \vee \mathsf{a}_3)\ \ .\ \mathsf{a}_4]\mathit{false} \\ FT - MUTEX() = FT - BETWEEN(\ \texttt{use(.\ )}\ ,\ \texttt{free(.\ )}\ ,,\ \texttt{use(.\ )}\ ) \end{gathered}$$

To illustrate the power of model checking as a debugging tool, consider the following erroneous implementation of the `handle_info` function of the locker. After a client crashes, access to the resource is given to the client that was waiting to get access in the head of the pending list.

```
handle_info({'EXIT',ClientPid,Reason},Pending) ->
    NewPending = remove(ClientPid,Pending),
    case NewPending == [] of
         false -> gen_server:reply(hd(NewPending), ok),
                  {noreply, NewPending};
         _ ->     {noreply, []}
    end.
```

This code is correct for the case where a client crashes after obtaining access to the resource, but it is wrong if the client crashes after releasing the resource. Testing concurrent code is tricky, in particular, in this example, only the right combination of more than three clients, a client crashing after releasing the resource and the other two or more clients waiting in the pending list triggers the error in the fault-tolerant code.

6.3 Non-Starvation

As with the verification of mutual exclusion, the fault-tolerant behaviour of the system we want to verify needs to be taken into account in order to prove the non-starvation property. Thus, instead of the following property that checks for non-starvation of the client C, which because of crashes is not satisfied,

$$\begin{aligned}&\mathit{NONSTARVATION}(C) = \\ &\quad [\texttt{-}\ \ .\ \texttt{gen\_server:call}(.\ \texttt{request}.\ , C)\ ] \\ &\qquad \mu X.(\langle\texttt{-}\rangle \mathit{true}\ \wedge\ [\neg\ \texttt{reply}(\texttt{ok},C)\ ]X)\end{aligned}$$

we use the following “fault-tolerant” one, which is satisfied by the locker:

$$\begin{aligned}&\mathit{NONSTARVATION}(C) = \\ &\quad [\texttt{-}\ \ .\ \texttt{gen\_server:call}(.\ \texttt{request}.\ , C)\ ] \\ &\qquad \mu X.(\langle\texttt{-}\rangle \mathit{true}\ \wedge\ [\neg\ \texttt{reply}(\texttt{ok},C)\ \vee\ \texttt{info}(.\ ,\texttt{C},.\ )\ ]X)\end{aligned}$$

7 Conclusions and Related Work

One of the aspects that makes the programming language Erlang popular among developers of business-critical systems is the inclusion of constructs to handle fault-tolerance. Our approach to verification of such fault-tolerant systems has several components. First, Erlang systems are translated into μCRL specifications. Next, the μCRL toolset generates the state space from the algebraic specification, and finally, the CADP toolset is used to check whether the system satisfies correctness properties specified in a the alternation-free μ-calculus.

To enable analysis of fault behaviour we introduce during the translation phase to μCRL explicit failure points in the algebraic specification, in a systematic way, where the system processes may fail. The key observation is that, due to the usage of higher-level design pattern that structure process communication and fault recovery, the number of such failure points that needs to be inserted can be relatively few, and can be inserted in an application independent manner. In other words, the state spaces generated from a failure model can be generated automatically, are relatively small, and are thus amenable to model checking.

We have demonstrated the approach in a case study where a server, built using the generic server design pattern, implements a locking service for the client processes accessing it. The server necessarily contains code to handle the situation where clients can fail; if it did not the server would quickly deadlock. In the study we verify, using the automated translation and model checking tool, systems composed of a server and a set of clients with regards to crucial correctness properties such as deadlock freedom, mutual exclusion and liveness.

The formal verification of fault-tolerant systems has been studied in several case-studies such as e.g. [9, 10]. In contrast to our approach, they target a single application only, are ad-hoc, and often do not provide a reusable verification method.

General models for the verification of fault-tolerant algorithms are also present in the literature, for example [6]. The main difference with our approach is that our models (similar to the software) are on a higher-abstraction level than those works; there is more intelligence built-in the Erlang component programming model than in general model, and it is interesting to see, that using such a model actually makes it easier to verify the correctness of the solution.

References

[1] J.L. Armstrong, S.R. Virding, M.C. Williams, and C. Wikström. *Concurrent Programming in Erlang*. Prentice Hall International, 2nd edition, 1996.

[2] T. Arts, C. Benac Earle and J. J. Sánchez-Penas. Translating Erlang to μCRL. Application of Concurrency to System Design, 2004. ACSD 2004. Proceedings. Fourth International Conference on, Vol., Iss., 16-18, pp. 135-144, June 2004.

[3] C. Benac Earle. Model Checking the Interaction of Erlang Components. PhD thesis, University of Kent, UK. February 2005.

[4] J.-C. Fernandez, H. Garavel, A. Kerbrat, R. Mateescu, L. Mounier, and M. Sighireau. CADP (CÆSAR/ALDÉBARAN development package): A protocol validation and verification toolbox. In *Proc. of CAV*, LNCS 1102, p. 437–440, Springer-Verlag, Berlin, 1996.

[5] J. F. Groote. The syntax and semantics of timed mCRL. Technical report SEN-R9709, CWI, Amsterdam, 1997.

[6] T. Janowski and M. Joseph. Dynamic Scheduling and Fault-tolerance: Specification and Verification. Real-Time Systems. Vol. 20, Issue 1, Kluwer Academic Publishers. 2001.

[7] D. Kozen. Results on the propositional μ-calculus. *TCS*, **27**:333-354, 1983.

[8] R. Mateescu. Local Model-Checking of an Alternation-free Value-Based Modal Mu-Calculus. *Proceedings of the International Workshop on Software Tools for Technology Transfer STTT'98*, Aalborg, Denmark, July 1998.

[9] J. Rushby. Systematic Formal Verification for Fault-Tolerant Time-Triggered Algorithms. IEEE Transactions on Software Engineering, volume 25, number 5, 1999.

[10] F. Schneider, S. M. Easterbrook, J. R. Callahan and G. H. Holzmann, Validating Requirements for Fault Tolerant Systems using Model Checking. *Proceedings, 3rd International Conference on Requirements Engineering*, 4-13, Colorado, Springs, Colorado, April 1998.

Applying Stacking and Corpus Transformation to a Chunking Task

José A. Troyano, Víctor J. Díaz, Fernando Enríquez,
Vicente Carrillo, and Fermín Cruz

Department of Languages and Computer Systems,
University of Seville, Av. Reina, Mercedes s/n 41012, Sevilla (Spain)
`troyano@lsi.us.es`

Abstract. In this paper we present an application of the stacking technique to a chunking task: named entity recognition. Stacking consists in applying machine learning techniques for combining the results of different models. Instead of using several corpus or several tagger generators to obtain the models needed in stacking, we have applied three transformations to a single training corpus and then we have used the four versions of the corpus to train a single tagger generator. Taking as baseline the results obtained with the original corpus ($F_{\beta=1}$ value of 81.84), our experiments show that the three transformations improve this baseline (the best one reaches 84.51), and that applying stacking also improves this baseline reaching an $F_{\beta=1}$ measure of 88.43.

1 Introduction

There are many tasks in natural language processing that consist in associating some kind of category to a group of words. Named Entity Extraction, Shallow Parsing, or Semantic Role Identification are three good examples. In this type of tasks, we can identify two subtasks: one that finds the boundaries of the group of words (chunk) and a second process that associates the correct tag to this group. In this paper we present a series of experiments on a clear example of chunking: the NER (Named Entity Recognition) problem. We show that corpus transformation and system combination techniques improve the performance in this task.

The NER task consists in the identification of the group of words that form a named entity. IOB notation is usually employed to mark the entities in a corpus. In this notation, the B tag denotes the beginning of a name, the I tag is assigned to those words that are within (or at the end of) a name, and the O tag is reserved for those words that do not belong to any named entity.

In the development of our experiments we have used a Spanish corpus tagged with NER information, and a re-trainable tagger generator based on Markov Models. In order to improve the performance of the NER task we have defined three transformations that give us modified versions of the training corpus, and we have trained the tagger generator with them to obtain different taggers.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 150–158, 2005.
© Springer-Verlag Berlin Heidelberg 2005

Finally we have applied a stacking (machine learning) scheme to combine the results of the models.

Experiments show that the three transformations improve the results of the NER task, and that system combination achieves better results than the best of the participant models in isolation.

2 Resources, Evaluation and Baseline

The two main resources employed in our experiments are the corpus and the tagger generator. The corpus provides a wide set of named entity examples in Spanish. It was used in the Named Entity Recognition shared task of CoNLL-02 [14] and it is distributed in three different files, a train corpus, and two test corpus. We have used the additional test corpus in stacking experiments to generate the training database.

There are four categories in the corpus taxonomy: PER (people), LOC (places), ORG (organizations) and MISC (rest of entities). However, the NER task does not need the category information, so we have simplified the corpus by removing the category information from the tags. Figure 1 shows a fragment of the original corpus, and its simplified version used in the NER task.

The other main resource is the tagger generator. We have chosen TnT [1], one of the most widely used re-trainable tagger in NLP applications. It is based upon second order Markov Models, consisting of word emission probabilities and tag transition probabilities computed from trigrams of tags. As a first step it computes the probabilities from a tagged corpus through maximum likelihood estimation, then it implements a linear interpolation smoothing method to manage the sparse data problem. It also incorporates a suffix analysis for dealing with unknown words, assigning tag probabilities according to the word ending.

Word	*Tag*
La	O
Delegación	B-ORG
de	I-ORG
la	I-ORG
Agencia	I-ORG
EFE	I-ORG
en	O
Extremadura	B-LOC
transmitirá	O
hoy	O
...	...

NEE corpus

Word	*Tag*
La	O
Delegación	B
de	I
la	I
Agencia	I
EFE	I
en	O
Extremadura	B
transmitirá	O
hoy	O
...	...

NER corpus

Fig. 1. Original corpus and corpus tagged only for the recognition subtask

Table 1. Baseline, TnT trained with *NER coprpus*

	Precision	Recall	$F_{\beta=1}$
Baseline	81.40%	82.28%	81.84

To evaluate our experiments, we have used the classical measures *precision*, *recall* and $F_{\beta=1}$. *Precision* is defined as the percentage of correctly extracted entities. *Recall* is defined as the proportion of entities that the system has been able to recognize from the total correct entities in the test corpus. The overall $F_{\beta=1}$ measure combines recall and precision, giving to both the same relevance:

$$F_{\beta=1} = \frac{2\, Precision\ Recall}{Precision + Recall}$$

We will use $F_{\beta=1}$ measure for comparing the results of our experiments. It is a good performance indicator of a system and it is usually used as comparison criterion. Table 1 shows the results obtained when TnT is trained with the *NER corpus* , we will adopt these results as the baseline for further experiments in this paper.

3 Corpus Transformation

In order to have different views of the NER problem, we have defined three transformations that applied to the original corpus give us three additional versions of it. This way, the tagger generator learns in four different ways and the resulting models can specialize in the recognition of named entities of different nature.

3.1 Vocabulary Reduction

In this transformation we employ a technique similar to that used in [12] replacing the words in the corpus with tokens that contain relevant information for recognition. One of the problems that we try to solve is the treatment of unknown words: the words that do not appear in the training corpus and, therefore, the tagger can not make any assumption about them. In the NER task, the lack of information of an unknown word can be mitigated with its typographic information because capitalization is a good indicator of the presence of a proper name. We also include in this transformation the knowledge given by non-capitalized words that frequently appear before, after or inside named entities. We call them trigger words and they are of great help in the identification of entity boundaries. Both pieces of information, trigger words and typographic clues, are extracted from the original corpus through the application of the following rules:

- Each word is replaced by a representative token, for example, it _starts_cap_ for capitalized words. These word patterns are identified using a small set of regular expressions.

Word	*Tag*
La	O
Delegación	B
de	I
la	I
Agencia	I
EFE	I
en	O
Extremadura	B
transmitirá	O
hoy	O
...	...

NER corpus

Word	*Tag*
La	O
_starts_cap_	B
de	I
la	I
_starts_cap_	I
_all_cap_	I
en	O
_starts_cap_	B
transmitirá	O
lower	O
...	...

NER-V corpus

Word	*Tag*
La_det_	O
_starts_cap__noun_	B
de_prep_	I
la_det_	I
_starts_cap__noun_	I
_all_cap__noun_	I
en_prep_	O
_starts_cap__noun_	B
transmitirá_verb_	O
_lower__adv_	O
...	...

NER-P corpus

Fig. 2. Changing the words

- Not all words are replaced with its corresponding token, the trigger words remain as they appear in the original corpus. The list of trigger words is computed automatically counting the words that most frequently appear around or inside an entity.

Figure 2 shows the result of applying vocabulary reduction (*NER-V corpus*). The results of the experiment *TnT-V* are presented in Table 2, we can see that this transformation makes TnT improve from 81.84 to 83.63.

3.2 Addition of Part-of-Speech Information

In this case we will make use of external knowledge to add new information to the original corpus. Each word will be replaced with a compound tag that integrates two pieces of information:

- The result of applying the first transformation (vocabulary reduction).
- The part-of-speech (POS) tag of the word.

To obtain the POS tag of a word we have trained TnT with the Spanish corpus CLiC-TALP [4]. We make use of a compound tag in the substitution

Table 2. Results of corpus transformation

	Precision	Recall	$F_{\beta=1}$
Baseline	81.40%	82.28%	81.84
TnT-V	81.76%	85.59%	83.63
TnT-P	81.51%	84.79%	83.12
TnT-N	82.77%	86.33%	84.51

because the POS tag does not provide enough information to recognize an entity. We complete this information with the knowledge given by typographical features and trigger words. Figure 2 shows the result of the application of this transformation (*NER-P corpus*). Adding POS information also results in a performance improvement of TnT in the NER task. Table 2 presents the results of the experiment *TnT-P*, in this case TnT reaches an $F_{\beta=1}$ measure of 83.12.

3.3 Changing the Tags

We replace the original IOB notation with a more expressive one that includes information about the position of words inside and around entities. In order to consider the position inside entities, we have added two new tags E and BE that are assigned, respectively, to words that end a multi-word named entity and to single-word named entities. The meaning of the tags assigned to words inside entities are now:

- B, that denotes the beginning of a named entity with more than one word.
- BE, that is assigned to a single-word named entity.
- I, that is assigned to words that are inside of a multiple-word named entity, except to the last word.
- E, assigned to the last word of a multiple-word named entity.

We can also add more information to words outside entities, particularly we are interested in those words that appear just before or after an entity. We split the meaning of the non-informative O tag into four tags:

- BEF, that is assigned to those words that appear before an entity.
- AFT, assigned to words that appear after an entity.

Word	*Tag*
La	O
Delegación	B
de	I
la	I
Agencia	I
EFE	I
en	O
Extremadura	B
transmitirá	O
hoy	O
...	...

NER corpus

Word	*Tag*
La	BEF
Delegación	B
de	I
la	I
Agencia	I
EFE	E
en	BET
Extremadura	BE
transmitirá	AFT
hoy	O
...	...

NER-N corpus

Fig. 3. Changing the tags

- BET, for words that are between two entities.
- O, for words outside entities and not adjacent to entities

This new tag set give more relevance to the position of a word, forcing the taggers to learn which words appear more frequently at the beginning, at the end, inside or around a named entity.

Figure 3 shows the result of applying this new tag set to a corpus fragment. Changing the tag set also leads to better results in the NER task than those obtained with the original corpus. The results of the experiment *TnT-N* are showed in Table 2. In this case, TnT improves from 81.84 to 84.51, the best result of all the transformations studied.

4 System Combination

System combination is not a new approach in NPL tasks, it has been used in several problems like part of speech tagging [7], word sense disambiguation [10], parsing [8], noun phrase identification [13] and even in named entity extraction [6]. The most popular techniques are voting and stacking (machine learning methods), and the different views of the problem are usually obtained using several taggers or several training corpora. In this paper, however, we are interested in investigate how stacking behaves when the combined systems are obtained with transformed versions of the same training corpus.

4.1 Stacking

Stacking consists in applying machine learning techniques for combining the results of different models. The main idea is to build a system that learns the way in which each model is right or makes a mistake. In this way the final decision is taken according to a pattern of correct and wrong answers.

In order to be able to learn the way in which every model is right or wrong, we use a training database. Each example in the training database includes the four tags proposed by the models for a given word and the actual tag. From this point of view, deciding the tag given the tags proposed by several models is a typical classification problem. Figure 4 shows a small database written in “arff” format, the notation employed by *weka* [16] to represent training databases. *Weka* is a collection of machine learning algorithms for data mining tasks, and is the tool that we have used in our stacking experiments.

An important advantage of using stacking as combining method is that we can include in the database heterogeneous information. Making use of this feature, we do not only include the tags of a given word in its register, but the tags assigned by the four models to its previous and following words are also included. This way, the registers of our database have twelve features instead of just four corresponding to the four tags of the word we are interested in.

We have used a corpus with new examples to generate the database, so we can ensure that de database used in stacking is independent of the models (training corpus) and it is also independent of the evaluation process (test corpus).

```
@relation combination
@attribute TnT          {O, B, I}
@attribute TnT-V        {O, B, I}
@attribute TnT-N        {O, B, I}
@attribute TnT-P        {O, B, I}
@data
I, I, I, B,      I
O, O, O, O,      O
B, B, B, B,      B
I, I, I, I,      I
O, I, I, I,      I
B, I, I, I,      I
O, O, O, O,      O
O, O, O, O,      O
B, B, B, O,      O
```

Fig. 4. A training data base. Each register corresponds to a word

Table 3. Results of stacking with a decision tree as learning technique

	Precision	Recall	$F_{\beta=1}$
Baseline	81.40%	82.28%	81.84
Decision Tree	87.96%	88.44%	88.20

Table 3 shows the results of the experiment *Decision Tree*, carried out using a decision tree [11] as stacking technique.

A decision tree uses a binary tree to predict the value of a target variable from those of a set of predictor variables. The tree is built by successively splitting nodes according to an information gain criterion. A pruning criterion is also applied to confine the tree size to appropriate limits. This technique is one of the best and most commonly used learning algorithm in classification.

The $F_{\beta=1}$ measure is 88.20, which is better than the baseline (81.84) and also better than the best of participant models in the stacking experiment (*TnT-N* with 84.59).

4.2 Using Other Machine Learning Algorithms

Apart from allowing the use of heterogeneous information, the use of machine learning as combination method has another important advantage: it is possible to choose among a large variety of schemes and techniques to find the most suitable for a specific problem. We have experimented with several machine learning algorithms included in the *weka* package to compare their performance when they are trained with the database that we have created. Most of them are rule-based because this kind of classifiers behaves better with discrete databases:

- Bagging [2]is based on the generation of several training data sets taking as base a unique data set. Each new version is obtained by sampling with

Table 4. Results of stacking with different classifiers

	Precision	Recall	$F_{\beta=1}$
Baseline	81.40%	82.28%	81.84
Decision Table	86.52%	87.59%	87.05
Random Tree	86.43%	87.84%	87.13
Part	87.70%	87.84%	87.72
Bagging	88.20%	88.42%	88.31
Ripper	88.88%	87.98%	88.43

replacement the original database. Each new data set can be used to train a model and the answers of all models can be combined to obtain a joint answer. Generally, bagging leads to better results than those obtained with a single classifier. The price to pay is that this kind of combination methods increase the computational cost associated to learning. In our experiment we have used decision trees as base learner with this scheme.

- Decision Table [9] is a rule-based classifier. The model consists of a schema, in which only the most representative attributes of the database are included, and a body that has labelled instances of the database defined by the features of the schema.
- Part [15] is the rule-based version of decision trees, it uses a divide and conquer strategy, building a partial decision tree in each iteration and converting the best leaf of the tree into a rule.
- Ripper [5] applies an iterative and incremental pruning process to obtain an error reduction. At a first stage it generates a set of rules that is optimized by generating new rules with randomized data and pruning them.
- Random Tree [16] is an adaptation of decision tree in which every node consider only a subset of the attributes of the database, this subset is chosen randomly.

Table 4 shows the results of the experiments. All of them present good results, the best one is achieved with Ripper (88.43) improving more than six percent points the baseline. This performance is similar to state-of-the-art recognizers, with comparable results to those obtained by one of the best NER systems for Spanish texts [3].

5 Conclusions and Future Work

In this paper we have shown that the combination of several taggers is an effective technique for improving a chunking task like named entity recognition. Taking as baseline the results obtained when a tagger generator (TnT) is trained with a corpus, we have investigated alternative methods for taking more advantage of the knowledge provided by the corpus. By means of corpus transformation we have obtained three different views of the training corpus, with them we have obtained three taggers that improve the results obtained with the original version of the corpus.

Once we had four different taggers we have applied stacking, combining them by generating a training database of examples and applying machine learning. We have experimented with several classifiers reaching a best result of 88.43 in the $F_{\beta=1}$ measure, more than six percent points better than the baseline (81.84). This performance is similar to state of the art NER systems, with comparable results to those obtained by the best system in the CoNLL-02 competition [3].

Much future work remains. We are interested in applying the ideas of this paper in the recognition of entities in specific domains, and in the growth of corpus, using the jointly assigned tag as agreement criterion in co-training or active learning schemes.

References

1. Brants, T.: TnT. A statistical part-of-speech tagger. In *Proceedings of the 6th Applied NLP Conference (ANLP00).* USA (2000) 224–231
2. Breiman, L.: Bagging predictors. In *Machine Learning Journal* 24 (1996) 123–140
3. Carreras, X., L. Màrquez y L. Padró: Named Entity Extraction using AdaBoost. In *CoNLL02 Computational Natural Language Learning.* Taiwan (2002) 167–170
4. Civit, M.: Guía para la anotación morfosintáctica del corpus CLiC-TALP. *X-TRACT Working Paper WP-00/06.* (2000)
5. Cohen, W. W.: Fast Effective Rule Induction. In *Proceedings of the 12th International Conference on Machine Learning.* Morgan Kaufmann (1995) 115–123
6. Florian, R., Ittycheriah, A., Jing, H., Zhang, T.: Named Entity Recognition through Classifier Combination. In *Proceedings of CoNLL-2003.* Canada (2003) 168–171
7. Halteren, v. H., Zavrel, J. , Daelemans, W.: Improving accuracy in word class tagging through the combination of machine learning systems. *Computational Linguistics 27* (2001) 199–230
8. Henderson, J. C., Brill, E.: Exploiting diversity in natural language processing. Combining parsers. In *1999 Joint Sigdat Conference on Empirical Methods in Natural Language Processing and Very Large Corpora. ACL.* USA (1999) 187–194
9. Kohavi, R.: The Power of Decision Tables. In *Proceedings of the European Conference on Machine Learning.* LNCS 914. (1995) 174–189.
10. Pedersen, T.: A simple approach to building ensembles of naive bayesian classifiers for word sense disambiguation. In *Proceedings of NAACL00.* USA (2000) 63–69
11. Quinlan, J.R.: Induction of decision trees. In *Machine Learning* 1 (1986) 81–106.
12. Rössler, M.: Using Markov Models for Named Entity recognition in German newspapers. In *Proceedings of the Workshop on Machine Learning Approaches in Computational Linguistics.* Italy (2002) 29–37
13. Tjong Kim Sang, E.F., Daelemans, W., Dejean, H., Koeling, R., Krymolowsky, Y., Punyakanok, V., Roth, D.: Applying system combination to base noun phrase identification. In *Proceedings of COLING00.* Germany (2000) 857–863
14. Tjong Kim Sang, E.F.: Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2002.* Taiwan (2002) 155–158
15. Witten, I.H., Frank, E.: Generating accurate rule sets without global optimization. In *Proceedings of the 15th International Conference on Machine Learning.* Morgan Kaufman (1998). 144–151
16. Witten, I.H., Frank, E.: Data Mining. Machine Learning Algorithms in Java. Morgan Kaufmann Publishers (2000)

Extracting Computer Algebra Programs from Statements*

Jesús Aransay[1], Clemens Ballarin[2], and Julio Rubio[1]

[1] Dpto. de Matemáticas y Computación, Univ. de La Rioja, 26004 Logroño, Spain
{jesus-maria.aransay, julio.rubio}@dmc.unirioja.es
[2] Institut für Informatik. Technische Univ. München. D-85748 Garching, Germany
ballarin@in.tum.de

Abstract. In this paper, an approach to synthesize correct programs from specifications is presented. The idea is to extract code from definitions appearing in statements which have been mechanically proved with the help of a proof assistant. This approach has been found when proving the correctness of certain Computer Algebra programs (for Algebraic Topology) by using the Isabelle proof assistant. To ease the understanding of our techniques, they are illustrated by means of examples in elementary arithmetic.

1 Introduction

Kenzo is a Common Lisp program created by Sergeraert [10], for Computer Algebra computations in the field of Algebraic Topology. Its main characteristics are its handling of infinite spaces (by using functional programming), and that Kenzo has found results unreachable by any other means (see [17]).

Taking into account these features, a project to analyze formally fragments of Kenzo was undertaken, with the aim of increasing the reliability of the system. Two kinds of formal methods have been applied. First, algebraic specification techniques have been used to model the data structures in algorithmic Algebraic Topology (see [13]). Second, we are using proof assistants to mechanize the reasoning needed in this area of Computer Algebra (see [1], [2], [3]).

Several approaches have been introduced to deal with objects (as chain complexes, morphisms, and so on) of Algebraic Topology in the Isabelle proof assistant [15]. One of the fundamental theorems in algorithmic Homological Algebra, namely the so-called *Basic Perturbation Lemma* [6], was chosen as a test case for these experiments. The final formalization considers morphisms as records encoding the real map, the potential source and target chain complexes, an also the real domain of definition and the image (see details in [3]).

Despite of the success of this approach, the price to be paid was that we got a formalization (and proof mechanization) of the *theorems*, but not of the *programs* appearing in Kenzo. To bridge the gap between (mechanized) theorems and programs, we considered to use Berghofer's tool [4] for extracting ML programs

---

* Partially supported by SEUI-MEC, project TIC2002-01626.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 159–168, 2005.
© Springer-Verlag Berlin Heidelberg 2005

from Isabelle theories. We suspected that the proving efforts previously done could be perhaps unsuitable, due to additional constraints on the constructive nature of the proofs (up to now, we have chosen a *classical* way of proving in Isabelle, trying to emulate the proofs-by-hand from Homological Algebra). Surprisingly enough, we observed that most of our already formalized theorems had *constructive statements* (in a sense which will be explained in this paper), even if proofs are not necessarily expressed in a constructive manner. This simple observation allows to apply Berghofer's tool to some of our Isabelle theories, extracting ML programs equivalent to some (small) fragments of Kenzo. Even if preliminary (the programs extracted so forth are extremely simple, compared with Kenzo as a whole), these results invite to explore further this research line. This paper is devoted to illustrate, through simple examples, this approach.

The paper is organized as follows. Sections 2 and 3 briefly introduce, respectively, some mathematical definitions and the case study chosen from Kenzo: the composition of two morphisms. In Section 4, we move to a well-known domain, namely elementary arithmetic, to work out a simple example related to Euclid's proof on the existence of infinitely many primes. The aim of this section is to introduce some key ideas, without the complexities of Homological Algebra and Algebraic Topology. Basics on formalization, automated theorem proving and program extraction are commented on in Section 5, where our notion of *constructive statement* is (informally) introduced, too. Then, this notion is applied in Section 6 to the elementary arithmetic example, showing how Berghofer's tool can be used to obtain an ML program (certified correct) computing a prime number bigger than its input. In Section 7, we go back to Computer Algebra, applying the same techniques to obtain an ML program (certified correct with Isabelle) to compose two morphisms. The paper ends with a section devoted to conclusions and open problems.

2 Mathematical Preliminaries

The mathematical machinery necessary to deal with the objects in the Kenzo system is quite complicated. In order to make easier the reading of this short paper, we focus only on the composition of morphisms, used as an illustrating example. The essential definitions are the following.

Definition 1. *A* graded group C_* *is a family of abelian groups indexed by the integer numbers,* $C_* = \{C_n\}_{n \in \mathbb{Z}}$*, with each* C_n *an abelian group. A* graded group morphism $f \colon A_* \to B_*$ of degree k *(*$\in \mathbb{Z}$*) between two graded groups* A_* *and* B_* *is a family of group morphisms,* $f = \{f_n\}_{n \in \mathbb{Z}}$*, with* $f_n \colon A_n \to B_{n+k}$ *a group morphism* $\forall n \in \mathbb{Z}$*. A* chain complex *is a pair* (C_*, d_{C_*})*, where* C_* *is a graded group, and* d_{C_*} (the differential map) *is a graded group morphism* $d_{C_*} \colon C_* \to C_*$ *of degree -1 such that* $d_{C_*} d_{C_*} = 0$*. A* chain complex morphism $f \colon (A_*, d_{A_*}) \to (B_*, d_{B_*})$ *between two chain complexes* (A_*, d_{A_*}) *and* (B_*, d_{B_*}) *is a graded group morphism* $f \colon A_* \to B_*$ *(degree 0) such that* $f d_{A_*} = d_{B_*} f$*.*

The mathematical result whose proof will be considered in the following sections is elementary: the composition of two morphisms is again a morphism.

Even if elementary, it is a fundamental fact which is used intensively in more interesting applications. In particular, it is instrumental in the proof (and even in the statement and in the algorithm) of the Basic Perturbation Lemma (BPL, from now on), central theorem [6] which has been studied in previous works, from several points of view (see [17], [3]).

3 The Kenzo Program

In [16] a fragment of the implementation in Kenzo of the BPL is presented. This is a quite complex Common Lisp program. As explained above, we focus here on the simpler case of the composition of morphisms. Even in this case, some explanations are needed in order to understand the code.

In Kenzo every chain complex is *free*. That implies a graded group in Kenzo is defined simply by a set of *generators* and an *equality test* among them, in *each degree*. The generators are used to form *combinations* (that are linear combinations of generators, with coefficients over the integer numbers), which are the real elements of the group in each degree. To add two combinations, the equality test between generators is used. With this organization, to define a morphism between chain complexes, it is enough to give the image of each generator, that will be a combination in the target group. In order to extend this map to combinations on the source group, the equality test *in the target group* is used. This is the most frequent *strategy* to define a morphism in Kenzo. But there are situations where this way of working is really wasteful from the performance point of view: think of the identity or the null morphisms, where obviously no equality checking is needed. So, Sergeraert considered a second strategy, called *by combination* (`:cmbn` as keyword in Common Lisp), to cover this case (the first, more frequent, strategy is called *by generator*, and denoted in Common Lisp by `:gnrt`). This explains the optional argument (called `strt` for *strategy*) in the following Kenzo program.

```
(DEFMETHOD CMPS ((mrph1 morphism) (mrph2 morphism) &optional strt)
   ;;; ... lines skipped
  (build-mrph :sorc sorc2 :trgt trgt1 :degr (+ degr1 degr2)
              :intr #'(lambda (cmbn)
                        (declare (type cmbn cmbn))
                          (the cmbn
                               (cmbn-? mrph1 (cmbn-? mrph2 cmbn))))
              :strt :cmbn :orgn `(2mrph-cmps ,mrph1 ,mrph2 ,strt))
   ;;; ... lines skipped
```

The program is a *method*, since Kenzo is built on CLOS (the Common Lisp Object System). This allowed Sergeraert, by using the inheritance mechanism of object-oriented programming, to give the same name to similar operations which compose two morphisms of coalgebras and another related algebraic structures. The lines skipped correspond to the cases in which one of the two morphisms has (or both have) a strategy by generator. The fragment showed here corresponds to the case in which at least one of the morphisms has strategy by combination, and this is also the strategy in the result morphism: `:strt :cmbn`. The symbols `sorc2`, `trgt1`, `degr1` and `degr2`, correspond, respectively, to the source of the second morphism `mrph2`, to the target of the first one, and to the degrees of the morphisms, which have been previously extracted from the morphisms in the lines skipped. The fragments `(declare (type cmbn cmbn))` and `the cmbn`, show that we are in a *typed* context. The keyword `:orgn` is used to store some information on the *origin* of the new morphism (that is to say, on the method and arguments constructing the new object), for software engineering purposes. Finally, the Kenzo function `cmbn-?` allows the programmer to invoke a morphism on a combination, and it is used here (twice) to accomplish the actual composition.

Apart from technicalities, the essence of the previous method, in the particular case of chain complexes morphisms of degree zero, is equivalent to the following Common Lisp function.

```
(defun CMPS (g f)
  (build-mrph :sorc (sorc f) :trgt (trgt g)
              :intr #'(lambda (cmbn)
                           (cmbn-? g (cmbn-? f cmbn)))))
```

The formalization work will be illustrated with this simplified version. But, instead of start with this algebraic case, we deal in the next section with an elementary example in number theory.

4 An Elementary Example

In this section, the same question of proving the correctness of a program is studied, but in the domain of elementary arithmetic, with the aim that our ideas can be understood without the complexities of Homological Algebra.

The example is related to prime numbers. The following Common Lisp program computes the next prime to a given positive integer number $x > 1$.

```
(defun nextprime (x) ; x integer > 1
  (if (isprime (+ x 1))
      (+ x 1)
    (nextprime (+ x 1))))
```

Here, we are assuming the existence of a Common Lisp program `isprime` used to determine if its input is a prime number. In addition, we assume that `isprime` is correct. This will be used in the following elementary result.

Theorem 1. *The program* `nextprime` *is correct with respect to the following specification:*
Input specification: x *integer,* $x > 1$.
Output specification: `(isprime (nextprime x))` *returns* true.

Proof. In two parts:

1) *Partial correctness.* If the program stops, `(nextprime x)` $= z$, with $z =$ `(+ y 1)` and `(isprime (+ y 1))` $=$ *true* (by the semantics of `if`). Then, the output specification follows.

2) *Proof of termination.* By contradiction.

Let us assume that there exists x integer, $x > 1$, such that `(nextprime x)` does not stop. This implies that $\forall y > x$, y is not a prime number. Let $P = \{p_1, \ldots, p_n\}$ be the set of primes smaller than x. By hypothesis, this is the set of *all* prime numbers. Let us consider $m = p_1 * \ldots * p_n + 1$. Then we have that p_i does not divide m, $\forall i = 1, \ldots, n$. We conclude, by applying Lemma 1 (see below), that m is prime. But $m > p_i$, $\forall i = 1, \ldots, n$, and thus $m \notin P$ and m is not a prime number. *Contradiction.* □

Lemma 1. *Let m be an integer number, $m > 1$. Then, m is prime if and only if for all prime number $p < m$, p does not divide m.*

Everyone will perceive in this proof Euclid's argument to show that there are infinitely many primes. Nevertheless, there are several variations on Euclid's idea. This one has the property that the computational content is more hidden than in other variants, which are based on the following Lemma 2, instead of on Lemma 1. In those proofs, even if presented in a "reductio ad absurdum" manner, the computational content is quite explicit[1]. More on that in Section 6.

Lemma 2. *For all integers $m > 1$, there exists a prime number $p \leq m$ such that p divides m.*

We have tried to write down a very detailed proof of correctness, since our aim is not only to give such a proof, but also a *mechanized certificate of correctness*. To be precise, we are looking for Isabelle [15] scripts containing proofs of correctness for our programs. To this aim, it is necessary to link in some way the running code (or, at least, the source code) with some formalization of it. In general, this is a task far from trivial (the lack of the sought link has been illustrated graphically in the previous proof, where `x` and x have been used in an indistinguished manner). Different approaches to formalize and mechanize such proofs are explored in the next section.

5 Formalization and Mechanization

In order to build correctness certificates for programs, a first necessary task is to formalize the objects of study in a computer-aided mathematical tool. There are many such tools: for instance, ACL2 [11], Coq [7], Mizar [18] or Isabelle [15]. The choice of one of them depends on several criteria, including the expressiveness

---

[1] Thanks are due to W. Bosma, who explained in the Map e-list (see `http://www.disi.unige.it/map/`) that the same is not true in the proof presented above.

of the underlying logic, the degree of automation or the libraries previously developed for our domain of interest. In the elementary example of the section above both Mizar and Isabelle could be convenient (in fact, the example and proof have been extracted from [21], where they are used to compare Mizar and Isabelle). However, once the objects of study have been formalized, more efforts are needed in order to link the formalization with a *real* program.

One well-known strategy is to establish the formalization in a (mechanized) *constructive* logic, and then extract a program from the proof. This is the point of view when using the Coq system [7] for this task. Nevertheless, it is not the *only* way to get certificates for programs.

In particular, the Common Lisp program `nextprime` in Section 4, with its non-constructive proof, seems to be far from this kind of approach. This is true from a strict-constructivist perspective, but it is not in Markov's *constructive recursive mathematics*[2]. In fact, the program `nextprime` is a paradigmatic example of a (correct) program generated by *Markov's principle* [14]. These issues are also discussed in [12], where Markov's principle is integrated with constructive type theory, and by Berghofer in [5], in the context of his tool to extract code from Isabelle scripts.

Taking into account this example it seems that Davenport's observation in [8] (or, even more explicitly, in page 140 of [9]), claiming that *Computer Algebra* correctness proofs can be done by not neccesarily *strict-constructive* methods, is quite accurate (up to our knowledge, if proofs in Computer Algebra can go beyond Markov's constructivism is an open problem).

In fact, in the field of algorithmic homological algebra it has been observed that the theorems to be formalized have *constructive statements.* That is to say: a new object is defined (the composite of two morphisms, in our running example) and some property of this object is asserted (namely, it is a morphism). Then, code can be extracted from the definition or specification appearing in the statement. Thus, this approach seems to indicate that the underlying logic in the proof is not mandatory to be constructive (since the program is extracted from definitions and not from proofs), in the vein of Davenport's observation. This strategy will be presented in the case of the composition in Section 7. But, first, we go back to the arithmetical example, to show how in this case statements can be rendered *constructive*, allowing program extraction in presence of classical proofs.

6 The Elementary Example Revisited

In this section the example in Section 4 is reconsidered. The idea is that the computational content of a proof (presented in a constructive or in a non-constructive manner) can be made explicit in order to build a new statement, which is *constructive* in the informal sense introduced above.

---

[2] We are grateful to F. Sergeraert who attracted our attention to this important variant of constructivism.

We use Lemma 2 in Section 4 above to define a primitive recursive function “some-prime-divisor” that for each integer number greater than 1 computes a prime divisor of it.

```
primrec
some-prime-divisor-aux x 0 = 0
some-prime-divisor-aux x (Suc n) =
 (if (prime-cons (Suc n) ∧ dvd-cons (Suc n) x) then (Suc n)
  else some-prime-divisor-aux x n)
consts
some-prime-divisor :: nat => nat
defs
some-prime-divisor-def: some-prime-divisor x == some-prime-divisor-aux x x
consts
some-big-prime :: nat => nat
defs
some-big-prime-def: some-big-prime x == some-prime-divisor (x! + 1)
```

The same task of *rebuilding* must be done with the predicates “divides” and “isprime”, which are converted to two (Boolean-valued) functions “dvd-cons” and “prime-cons” (here *cons* stands for *constructive*). In addition, it is necessary to prove in Isabelle that these functions have the right properties (by relating them, for instance, to the predicates “dvd” and “prime” from the Isabelle libraries on prime numbers). With these preparations, the following lemma can be proved in Isabelle.

```
theorem x < (some-big-prime x)
proof (unfold some-big-prime-def)
  let ?p = some-prime-divisor (x! + 1)
  from some-prime-divisor-gt-zero
  have prime-cons: prime-cons ?p and dvd-cons: dvd-cons ?p (x! + 1)
    by (simp-all add: some-prime-divisor-properties)
  from prime-cons have prime-p: ?p ∈ prime
    by (simp add: prime-equiv-prime-cons)
  from dvd-cons have dvd: ?p dvd (x! + 1)
    by (simp add: dvd-cons-impl-dvd)
  show x < ?p
  proof -
    have ¬ ?p ≤ x
    proof
      assume ?p ≤ x
      with prime-g-zero and prime-p have ?p dvd x!
        by (simp add: dvd-factorial)
      with dvd have ?p dvd (x! + 1) - x! by (rule dvd-diff)
      then have ?p dvd 1 by simp
      with prime-p show False using prime-nd-one by auto
    qed
    then show ?thesis by simp
 qed
qed
```

This proof, a variant of Euclid's argument, is an adaption of a proof script presented in [21]. It obviously has a non-constructive presentation. It might be considered unsuitable from a constructivist's point of view, but it is well-known and easily understood. The important observation now is that, no matter the logic underlying the proof, the *statement* is constructive, and Berghofer's tool [4] can be applied on the complete theory (only the final fragment is displayed here) to obtain the following ML program (the names for certain constants have been modified, in order to make the code more readable):

```
fun some_prime_divisor_aux x 0 = 0
  | some_prime_divisor_aux x (Suc n) =
    (if (prime_cons (Suc n) andalso dvd_cons (Suc n) x) then Suc n
      else some_prime_divisor_aux x n);

fun some_prime_divisor x = some_prime_divisor_aux x x;

fun fact 0 = 1 | fact (Suc n) = times (fact n) (Suc n);

fun some_big_prime x = some_prime_divisor (plus (fact x) 1);
```

This ML code is to be compared to the Common Lisp program `nextprime` in Section 4. Both functions compute a prime number greater than its argument x. But of course, `nextprime` is quite more efficient.

7 Application to Computer Algebra

When applying these ideas to the Kenzo program, the first observation to be made is that the elaboration done in the previous section is unnecessary: many of the theorems to be proved have already a *constructive statement* (or can be easily transformed into such a statement). See details in [3]. Therefore, the same work already made for the formalization, can be reused for extracting code from statements.

As explained at the end of Section 3, we will work with a simplified version of the Kenzo composition. More concretely, we do not consider the degree in the structures (that is, the degree of morphisms is 0) and, in addition, we do not assume that groups are free (that is to say, the strategy is always `:cmbn`, *by combination*, following Kenzo terminology). One fragment of such a formalization can be found here:

```
constdefs
  group-mrp-comp :: [ ('b, 'c) group-mrp-type, ( 'a, 'b)  group-mrp-type] =>
                      ('a , 'c) group-mrp -type
  group-mrp-comp  g f  ==
 (| src =  src f, trg = trg g, morph =   (morph g) ∘ (morph  f),
  src-comm-gr =  src-comm-gr f,  trg-comm-gr =  trg-comm-gr g |)
```

lemma *group-mrp-composition*:
assumes *A1*: *group-mrp A*
and *B1*: *group-mrp B*
and *C1*: *trg-comm-gr A* = *src-comm-gr B*
and *D1*: *trg A* = *src B*
shows *group-mrp* ($B \circ A$)

We can now apply Berghofer's extraction tool [4] on the definition appearing in the statement, obtaining the following ML program (only the most relevant part is shown here):

```
fun comp g f = (fn x => g (f x));

fun group_mrp_comp g f =
  group_mrp_type_ext (src f) (trg g) (comp (morph g) (morph f))
                     (src_comm_gr f) (trg_comm_gr g) Unity;
```

In this case, the proof of the previous Isabelle lemma is to be considered as a proof of correctness for the ML program (assuming, as usual, the soundness of Berghofer's translation). This (certified correct) ML program should be compared with the *real* corresponding Kenzo program or better, to ease the reading, with the simplified Common Lisp version given at the end of Section 3.

8 Conclusions and Future Work

In this paper we have applied Berghofer's extraction tool for Isabelle scripts to obtain Computer Algebra programs, with a certificate of correctness. Our contribution lies on extracting code from *statements* and not from *proofs*, as usual in constructive type theory. From a technical point of view, it would be more accurate to say that code is extracted from *definitions* appearing in statements, but it has been considered more appealing to exploit the couple statement/proof. In fact, in the arithmetic example in Section 6, it is clear that we are *programming* inside Isabelle, by transforming predicates into recursive functions, and proving, at the same time, the correctness of these transformations.

Even if it seems that in the elementary examples from Computer Algebra in Algebraic Topology, this work of *programming in Isabelle* is not necessary, important challenges are still open. It would be necessary to bridge the gap between the ML and Common Lisp programming language, and even more difficult, the gap between the ML programs extracted (that could be very inefficient) and the corresponding performing programs which are really usable. With respect to this, the toy example with prime numbers could illustrate the strong difficulties (in terms of proving efforts within Isabelle) to obtain a reasonably efficient program. Thus, finally it will be perhaps unavoidable to *program in Isabelle* in order to get usable programs. From our point of view, this problem of using proof assistants to synthetize "real-life" programs is a central one in intelligent information processing (see, for instance, [19] and [20]).

References

1. J. Aransay, C. Ballarin and J. Rubio, *Deduction and Computation in Algebraic Topology*, Proceedings IDEIA 2002, Universidad de Sevilla (2002) 47-54.
2. J. Aransay, C. Ballarin and J. Rubio, *Towards a higher reasoning level in formalized Homological Algebra*, Proceedings Calculemus 2003, Aracné Éditrice (2003) 84-88.
3. J. Aransay, C. Ballarin and J. Rubio, *Four approaches to automated reasoning with differential algebraic structures*, Lecture Notes in Artificial Intelligence 3249 (2004) 222-235.
4. S. Berghofer, *Program Extraction in Simply-Typed Higher Order Logic*, Lecture Notes in Computer Science 2646 (2002) 21-38.
5. S. Berghofer, *Answer to Tom Ridge*, `isabelle-users@cl.cam.ac.uk`, February 18, 2005.
6. R. Brown, *The twisted Eilenberg-Zilber theorem*, Celebrazioni Arch. Secolo XX, Simp. Top. (1967) 34-37.
7. *The Coq Proof Assistant Reference Manual*, `http://coq.inria.fr/doc/main.html`
8. J. M. Davenport, *Effective Mathematics: the Computer Algebra viewpoint*, Lecture Notes in Mathematics 873 (1981) 31-43.
9. J. M. Davenport, *Algebraic computations and structures*, Lecture Notes in Pure and Applied Mathematics 113, Marcel Dekker (1989) 129-144.
10. X. Dousson, F. Sergeraert and Y. Siret, *The Kenzo program*, `http://www-fourier.ujf-grenoble.fr/~sergerar/Kenzo/`
11. M. Kaufmann, P. Manolios and J. Strother Moore, *Computer-Aided Reasoning: An Approach*, Kluwer Academic Publishers, 2000.
12. A. Kopylov and A. Nogin, *Markov's principle for propositional type theory*, Lecture Notes in Computer Science 2142 (2001) 570-584.
13. L. Lambán, V. Pascual and J. Rubio, *An object-oriented interpretation of the EAT system*, Applicable Algebra in Engineering, Communication and Computing 14 (2003) 187-215.
14. A. A. Markov, *On constructive mathematics*, AMS Translations 98 (1971) 1-9.
15. T. Nipkow, L. C. Paulson and M. Wenzel, *Isabelle/HOL: A proof assistant for higher order logic*, Lecture Notes in Computer Science 2283, 2002.
16. J. Rubio, *Constructive proofs or constructive statements?*, Dagstuhl Proceedings 05021, 2005. `http://www.dagstuhl.de/05021/`
17. J. Rubio and F. Sergeraert, *Constructive Algebraic Topology*, Bulletin des Sciences Mathématiques 126 (2002) 389-412.
18. P. Rudnicki, *An Overview of the MIZAR Project*, In Proceedings Workshop on Types for Proofs and Programs (1992) 311-330. `http://web.cs.ualberta.ca/piotr/Mizar/MizarOverview.ps`
19. N. Schirmer, *A verification environment for sequential imperative programs in Isabelle/HOL*, Lecture Notes in Computer Science 3452 (2005) 398-414.
20. M. Takeyama, Q. Haiyan, P. Dybjer, *Verifying Haskell programs by combining testing, model checking and interactive theorem proving*, Information and software technology 15 (2004) 1011-1025.
21. M. Wenzel and F. Wiedijk, *A Comparison of Mizar and Isar*, Journal of Automated Reasoning 29 (2002) 389-411.

Integrating Syntactic Information by Means of Data Fusion Techniques[⋆]

Francisco J. Ribadas[1], Jesús Vilares[2], and Miguel A. Alonso[2]

[1] Escuela Superior de Ingeniería Informática, Universidade de Vigo,
Campus de As Lagoas, 32004 Orense, Spain
ribadas@uvigo.es
[2] Departamento de Computación, Universidade da Coruña,
Campus de Elviña s/n, 15071 La Coruña, Spain
{jvilares, alonso}@udc.es

Abstract. In Information Retrieval (IR) systems, the correct representation of a document through an accurate set of index terms is the basis for obtaining a good performance. If we are not able to both extract and weight appropriately the terms which capture the semantics of the text, this shortcoming will have an effect on all the subsequent processing.

1 Introduction

One of the major limitations we have to deal with in text retrieval applications is the linguistic variation of natural languages, particularly those with more complex morphologic and syntactic structures than English, as in the case of Spanish. When managing this type of phenomena, the employment of Natural Language Processing (NLP) techniques becomes feasible. Previous work has shown that linguistic variation at word level can be effectively managed by taggers-lemmatizers, improving the average precision attained by stemmers [5].

The next step consists of applying phrase-level analysis techniques to reduce the *syntactic variation* present in both documents and queries, identifying the syntactic structure of the text by means of shallow parsing techniques that provide an effective balance between descriptive power and computational cost [3]. However, the integration of the information obtained by syntactic processing (usually in the form of complex index terms) into the indices of an IR system is an open issue. As indicated in [1], when simple and complex terms are used together, the assumption of term independence is violated because words forming a complex term also occur in the documents from which the syntactic dependency has been extracted and deviations are introduced in the obtained results.

In our work we propose to apply *data fusion*, a set of techniques for combining the results retrieved by different representations of queries and documents, or by

---

[⋆] The research reported in this article has been partially supported by Ministerio de Educacin y Ciencia (MEC) and FEDER (TIN2004-07246-C02-01 and TIN2004-07246-C02-02), by MEC (HF2002-81), and by Xunta de Galicia (PGIDIT02PXIB30501PR, PGIDIT02SIN01E and PGIDIT03SIN30501PR).

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 169–178, 2005.
© Springer-Verlag Berlin Heidelberg 2005

different retrieval techniques, to integrate index terms coming from different text processing techniques. This article is structured as follows. Section 2 describes the linguistically motivated techniques we have proposed for index term extraction. In Section 3 we introduce the three data fusion methods we have evaluated and show the results obtained in our experiments with the CLEF'2003 Spanish corpus. Finally, in Section 4 we sumarize our conclusions.

2 Linguistic Processing for Information Retrieval

The usual approaches followed in IR systems are based on using index term or keywords to represent documents and user queries. In our case, we propose a set of linguistically motivated methods to extract this kind of index terms. We have employed NLP tools to identify single word terms, by means of a lemmatization process, and complex word terms, using a partial parser to recognize pairs of related words. We have also used a locality based retrieval approach to combine related words, based on word proximity measures.

2.1 Index Term Extraction by Lemmatization

The first index term extraction method we have studied exploits lexical level information and uses lemmatization to select and normalize single word terms, using the canonical form, or *lemma*, of the content words –nouns, adjectives and verbs, the grammatical categories which concentrate the semantics of a text– as index terms.

The process we follow for single word term identification starts by tagging the document. The first step consists of applying our linguistically-motivated preprocessor module [7] in order to perform tasks such as format conversion, tokenization, sentence segmentation, morphological pretagging, contraction splitting, separation of enclitic pronouns from verbal stems, expression identification, numeral identification and proper noun recognition.

The output generated by our preprocessor is then taken as input by our tagger-lemmatizer, `MrTagoo` [11], based on a second order Hidden Markov Model (HMM). This tagger also incorporates certain advanced capabilities, such as a very efficient structure for storage and search —based on finite-state automata and able to support automatic error correction features [13]—, management of unknown words, the possibility of integrating external dictionaries in the probabilistic frame defined by the HMM and the possibility of managing ambiguous segmentations. Once text has been tagged, the lemmas of the content words (nouns, verbs and adjectives) are extracted to be indexed.

2.2 Syntactic Dependency Pairs

To extract complex index terms we have followed a syntactically based approach. This method employs a shallow parser to identify the syntactic dependencies present in documents and queries, that will we used to build the complex index

Table 1. Basic retrieval results

	lem	*dep*	*dep-fb*	*loc*
# Docs.	57k	57k	57k	57k
Relevant retrieved	2221	2218	2256	2221
Non-interp. precision	.4681	.4014	.4741	.4873
Doc. precision	.5431	.4647	.5538	.5154
R-precision	.4471	.3961	.4425	.4438
precision at 5 docs.	.5965	.4842	.5825	.6104
precision at 10 docs.	.5000	.4333	.5070	.5296
precision at 30 docs.	.3813	.3205	.3731	.4157
precision at 100 docs.	.2314	.2053	.2328	.2571

terms used during the retrieval process. We propose a shallow parser based on a cascade of finite-state transducers consisting of the following five layers, whose input is the output of the tagger-lemmatizer.

Layer 0: preprocessing. Its function is the management of certain linguistic constructions, like *numerals in non-numerical format*, *quantity expressions* and *expressions with a verbal function*, in order to minimize the noise generated during the subsequent parsing.

Layer 1: adverbial phrases and first level verbal groups. This layer identifies the *adverbial phrases* of the text, either those with an adverbial head —e.g., *rápidamente* (quickly)—, or those expressions having an equivalent function —e.g., *de forma rápida* (in a quick way). It also processes non-periphrastic verbal groups, which we call *first level verbal groups*, both simple and compound forms, and both active and passive forms.

Layer 2: adjectival phrases and second level verbal groups. Adjectival phrases —e.g., *muy alto* (very high)— are managed here, together with periphrastic verbal groups —e.g., *tengo que ir* (I have to go)—, which we call *second level verbal groups.*

Layer 3: noun phrases. We have considered some complex phenomena in noun phrases, such as the existence of *partitive complements* —e.g., *ninguno de* (none of)—, in order to cover complex nominal structures —e.g., *cualquiera de aquellos coches nuevos* (any of those new cars).

Layer 4: prepositional phrases. Formed by a noun phrase preceded by a preposition, we have considered three different types according to this preposition: those preceded by the preposition *por* (by), those preceded by *de* (of) and the rest of prepositional phrases.

These layers and the rules of the grammar employed by the parser are explained in detail in [3]. Each of the rules involved in the different stages of the parsing process has been implemented through a finite-state transducer.

To extract the multiword terms from documents and queries, we identify the syntactic roles of the basic phrases returned by the shallow parser. We only consider six basic syntactic roles: prepositional noun complement, subject, attribute, direct object, agent and prepositional verb complement. Once we have identified

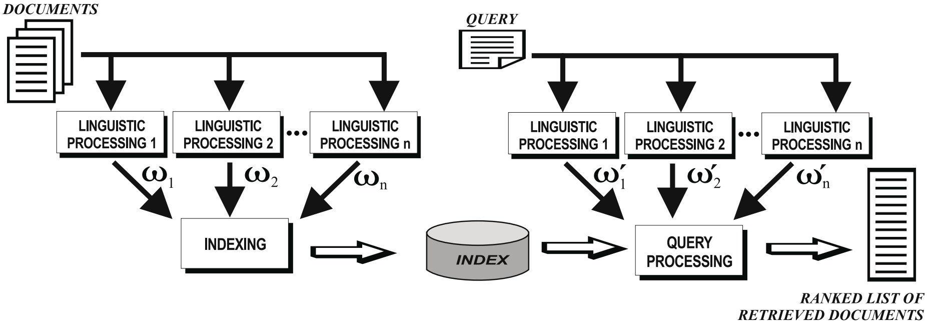


Fig. 1. Weighted mixing of index terms

these syntactic roles, we use the following four types of syntactic dependencies to create pairs of related terms:

- Noun-Adjective/Prepositional noun complement
- Verb-Object/Prepositional verb complement
- Subject-Verb
- Subject-Attribute

Once the dependencies have been extracted, they are conflated into *complex index terms*. In our case, we have used a conflation technique based on the employment of morphological relations[1] in order to improve the management of syntactic and morphosyntactic variation [12].

2.3 Syntactic Dependency Pairs Filtered by Relevance Feed-Back

In previous experiments, the direct combination of single terms extracted by lemmatization, with complex terms obtained by shallow parsing, is not able to get good performaces due to term interdependency, specially when queries are processed. In order to overcome this problem, we have improved the selection of query terms through a two steps method, based on pseudo-relevance feedback (blind-query expansion), to get more representative complex terms for queries.

The indexing process is the same described in the previous case, we simple merge single and complex terms in document representation. In contrast the querying process is performed in two stages:

1. The lemmatized query is submitted to the system, and we perform a first retrieval using lemmas only.
2. From those results, the n top documents retrieved by this initial query are employed to select the most informative dependencies, which are used to expand the query in order to obtain the final set of documents. These dependencies are selected automatically using Rocchio's approach [8] to feedback.

As a result of a tuning process, the best results were obtained expanding the original topic with the best 10 multiword terms of the 5 top ranked documents.

---

[1] In this way we can partially overcome the kind of morphosyntactic variation shown in `''cambio en el clima''`(*change of the climate*) and `''cambio climatico''` (*climatic change*).

Table 2. Weighted mixing of index terms

	lem	***lem, dep***	***lem, dep-fb***
# Docs.	57k	57k	57k
Relevant retrieved	2221	2242	2255
Non-interp. precision	.4681	.4710	.5151
Doc. precision	.5431	.5475	.6047
R-precision	.4471	.4454	.4752
precision at 5 docs.	.5965	.6070	.5930
precision at 10 docs.	.5000	.5053	.5070
precision at 30 docs.	.3813	.3789	.3835
precision at 100 docs.	.2314	.2337	.2354

2.4 Locality Based Retrieval

Another approach to manage multiword terms is inspired on the locality based retrieval, a pseudo-syntactic approach [4] based on the proximity between index terms. The locality based model considers the collection to be indexed not as a set of documents, but as a sequence of words where each occurrence of a query term has an influence on the surrounding terms. Such influences are additive, thus, the contributions of different occurrences of query terms are summed, yielding a similarity measure. As a result, those areas of the texts with a higher density of query terms, or with important query terms, show peaks in the resulting influence graph, highlighting those positions of the text which are potentially relevant with respect to the query.

We have adapted the original proposal of Kretser and Moffat [6]. The contribution to the similarity graph of a given query term is determined by a *similarity contribution function*. In such a way that the degree of similarity or relevance associated with a given location is the sum of all the influences exerted by the rest of query terms within whose spread the term is located. The final relevance score assigned to every document is given as the sum of the similarities corresponding to occurrences of query terms that this document contains.

The approach we have chosen for integrating distance-based similarity in our IR system consists in postprocessing the documents obtained by a document-based retrieval system. This initial set of documents is obtained through a base IR system which employs content word lemmas as index terms. This list of documents is then processed using the locality-based model, taking the final ranking obtained from the distance-based similarity as the final output to be returned to the user.

3 Data Fusion Methods

In order to get an effective combination of the available term extraction methods and to overcome the violation of term independence assumption, that raises when single and complex terms are used together, we have tested different *data fusion* techniques. The basic idea behind data fusion [10,9] is well know principle in IR.

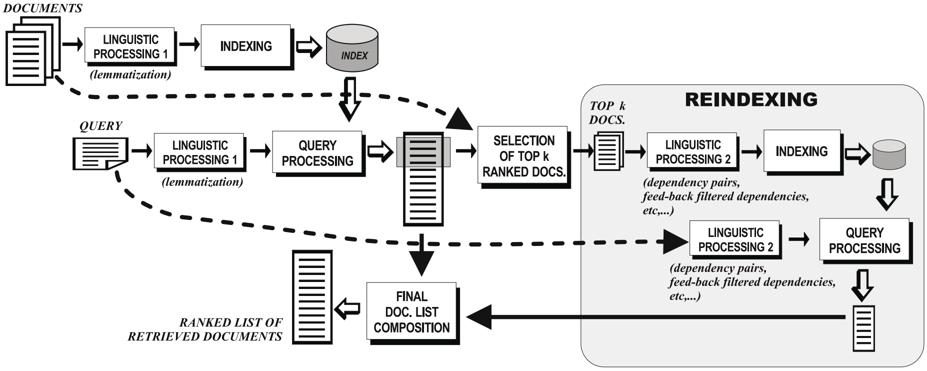

DOCUMENTS
LINGUISTIC PROCESSING 1
(lemmatization)
INDEXING
INDEX
QUERY
LINGUISTIC PROCESSING 1
(lemmatization)
QUERY PROCESSING
SELECTION OF TOP k RANKED DOCS.
REINDEXING
TOP k DOCS.
LINGUISTIC PROCESSING 2
(dependency pairs, feed-back filtered dependencies, etc,...)
INDEXING
LINGUISTIC PROCESSING 2
(dependency pairs, feed-back filtered dependencies, etc,...)
QUERY PROCESSING
FINAL DOC. LIST COMPOSITION
RANKED LIST OF RETRIEVED DOCUMENTS

Fig. 2. Reindexing retrieved documents

In general, it should be expected that the combination of multiple evidences will make possible to improve the effectiveness of the retrieved results.

In our case we are combining the results retrieved by different representations for documents and queries, as result of different index term extraction techniques. Our objective is to take advantage of the term extraction methods shown in previous sections to improve the distribution and the ranking of relevant documents in the final document list and get better precision values, specially partial precision measures on top positions.

To evaluate the performance of the proposed data fusion methods we have employed the Spanish monolingual corpus from CLEF (*Cross Language Evaluation Forum*), composed by 215,738 news reports and by 100 queries, numbered from 41 to 140. Every query is formed by three fields: a brief *title*, a one-sentence *description*, and a more complex *narrative* specifying the relevance criteria. In our experiments only the title and the narrative fields have been employed to build the running queries, using the retrieval facilities of the SMART [2] vectorial indexing engine. The results obtained in a single run with each one of the four term extraction methods described in section 2 are shown in Table 1. The best results were obtained by single lemmas and by reranking using locality based measures, while the results with pure and filtered dependency pairs were poor in terms of global precision, because they assign too relevance to complex terms, that leads to deviations in the final results.

3.1 Weighted Mixing of Index Terms

In our first experiments we have used a weighted combination of single terms extracted by lemmatization and complex dependency pairs. This fusion method, illustrated in Fig. 1, merges the sets of index terms extracted from the documents by the different text processing techniques and assigns to each of them its own weights, using the support for multiple indexing schemes offered by the Smart vectorial IR engine, which will take them into account when computing vector distances during the retrieval phase.

Table 3. Reindexing retrieved documents

	lem	***lem, lem***	***lem, dep***	***lem, dep-fb***	***lem, loc***
# Docs.	57k	57k	57k	57k	57k
Relevant retrieved	2221	2221	2221	2221	2221
Non-interp. precision	.4681	.4629	.4587	.4639	.4712
Doc. precision	.5431	.5123	.5084	.5124	.5445
R-precision	.4471	.4128	.4095	.4107	.4439
precision at 5 docs.	.5965	.5337	.5221	.5287	.5980
precision at 10 docs.	.5000	.4709	.4593	.4612	.5034
precision at 30 docs.	.3813	.3648	.3581	.3596	.3832
precision at 100 docs.	.2314	.2295	.2279	.2291	.2335

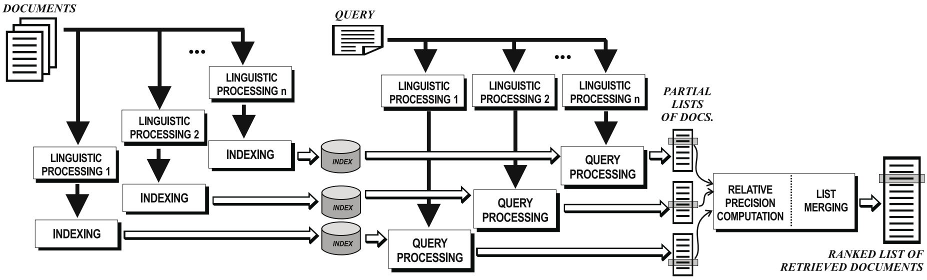


Fig. 3. List merging based on mutual precision

In our evaluation we have performed exhaustive tests, trying several weights for each indexing method and also using different formulae to compute vector components. The best results were obtained when single terms were assigned weights much more higher than the ones assigned to complex index terms . In Table 2 we shown the best precision values obtained with this fusion technique. We use as baseline the results for lemmatization based terms, comparing it with the weighted combination of lemmas with dependency pairs and with the combination of lemmas with dependency pairs filtered by relevance feedback.

The main drawback of this approach is how to find the right set of weights, capable of offering a uniform behaviour with all of the queries being considered and able to ensure good precision values across different queries, because weights with good results in some queries do not necessarily maintain their performance with other queries.

3.2 Reindexing Retrieved Documents

In this second method we propose a two-step approach. As shown in Fig. 2, we perform a first retrieval process using single terms extracted by lemmatization. Then, we select the top k ranked documents and reindex them using complex terms based on dependency pairs. The final list of documents is built by re-

Table 4. List merging based on mutual precision

	lem	***lem, dep, dep-fb***	***lem, dep, loc***	***lem, dep-fbloc***
# Docs.	57k	57k	57k	57k
Relevant retrieved	2221	2256	2221	2256
Non-interp. precision	.4681	.4691	.4701	.4931
doc. precision	.5431	.5534	.5611	.5693
R-precision	.4471	.4394	.4486	.4512
precision at 5 docs.	.5965	.6039	.6103	.6140
precision at 10 docs.	.5000	.5079	.5140	.5202
precision at 30 docs.	.3813	.3849	.3901	.3924
precision at 100 docs.	.2314	.2385	.2430	.2509

ordering the top documents retrieved in the first step according to the results of the second phase. The idea behind this approach is to perform a more precise processing on the most promising documents and use it to improve the final ranking.

We have tested this proposal by reindenxing different amounts of top ranked documents and using different methods to extract dependency pairs for the reindexing step. In Table 3 we show the results obtained with this data fusion approach, where the top 100 ranked documents for each query where selected and reindexed using lemmatization, dependency pairs, filtered dependency pairs and the locality based retrieval method, respectively.

The results obtained were not as good as it could be expected in terms of global precision and also in terms of partial precision values at top positions in the ranked list. The main reason for this poor behaviour seems to be the high degree of intercorrelation shown by the index terms extracted in the second phase. In this situation it is not clear if classical formulae to compute vector components in vector based engines are able to offer good rankings and performances.

3.3 List Merging Based on Rank Positions and Mutual Precision

The last method considered in this paper tries, in contrast with the previous ones, to perform a fusion of results adapted to each individual query. What this proposal does is to merge lists of retrieved documents obtained independently using different indexing methods. As shown in Fig. 3, the final ranked list takes into account both the ranking position of each document in the lists that retrieved it and a value that measures the confidence that each list assigns to a document in that position. These confidence measures are computed independently for each query and for each list of retrieved documents, using the precision values obtained at different levels for different subsets of the list, taking a combination of the other document lists as a reference to compute them.

As a consequence of the way the confidence measures are computed and the method employed to get the final ranked list, the main drawback of this approach is that it tends improve the ranking position of both relevant and non-relevant documents that are common to different lists. This is a well know fenomenon in data fusion, and in the case of IR applications it is not very important in practice, since the overlap factor among relevant documents uses to be much higher than among non-relevant ones. So, in general, it can be expected that different retrieval approaches will return a similar set of relevant documents, but a different set of non-relevant documents. From a practical point of view, this approach is able to offer promising results when dealing with high precision lists of documents, especially if they do not share non-relevant retrieved documents.

In Table 4 we can see the results of applying this fusion method to combine the list of retrieved document offered by three independent runs of different retrieval approaches explained in section 2. The best results were obtained by merging the documents retrieved by the lemmas based method, the filtered dependency pairs method and by the locality based approach.

From our point of view, the most important advantage of this data fusion proposal is that it owns an uniform behaviour across different queries, being able to improve the precision for almost all of the queries in our experimental corpus. In Fig. 4 we have an illustrative example of this property. Here, we show the relative improvements in precision values at 10 documents for some queries taken from our corpus, using the best results obtained by the data fusion methods described in this work (weighting, reindexing and list merging using mutual precision). As we stated before, list merging using mutual precision has a more uniform behaviour, offering slightly better precision improvement in most of these queries.

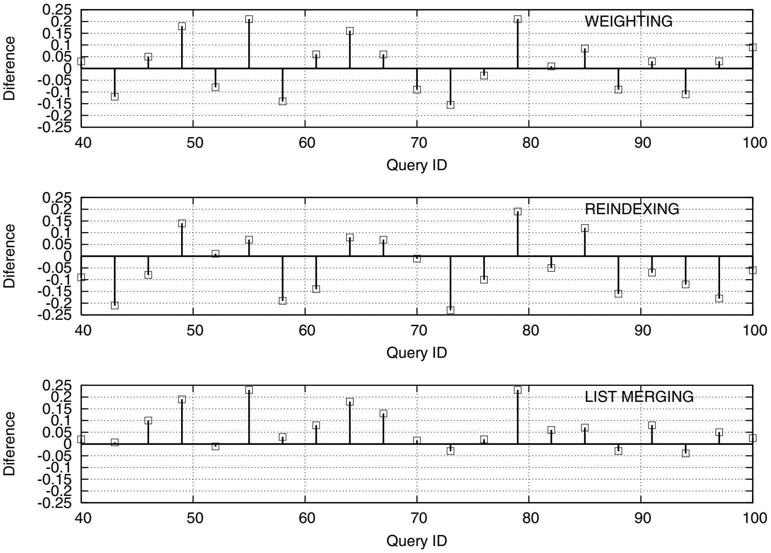


Fig. 4. Improvement in precision at 10 docs. for weighting, reindexing and list merging using mutual precision

4 Conclusions

In this article we have shown the application of different NLP techniques to text retrieval in Spanish, based in both single and complex index terms, that try to overcome the problems derivated from the linguistic variation present in natural languages. We have also proposed the application of data fusion methods to integrate different index term extraction techniques and different retrieval models. Three fusion approaches were proposed for this purpose and evaluated using the CLEF corpus. The most promising approach was based on list merging based on rank position, using mutual precision measures to ponderate the different evidences, that, as shown in the previous section, is able to obtain good improvements in a wide rage of queries.

References

1. M. Narita and Y. Ogawa. The use of phrases from query texts in information retrieval. In *ACM SIGIR 2000*, pages 318–320, Athens, Greece, 2000.
2. C. Buckley. Implementation of the SMART information retrieval system. *Technical report*, Department of Computer Science, Cornell University, 1985.
3. J. Vilares and M. A. Alonso. A grammatical approach to the extraction of index terms. In *Proc. of RANLP 2003*, pages 500–504, Borovets, Bulgaria, Sept. 2003.
4. J. Vilares and M. A. Alonso. Dealing with syntactic variation through a locality-based approach. *Lecture Notes in Computer Science (LNCS)*, 3246:??–??, 2004.
5. J. Vilares, M. A. Alonso, and F. J. Ribadas. COLE experiments at CLEF 2003 Spanish monolingual track. In *Working Notes for the CLEF 2003 Workshop*, pages 197–206, Trondheim, Norway, Sept. 2003.
6. O. de Kretser and A. Moffat. Locality-based information retrieval. In *Proc. of 10th Australasian Database Conference (ADC'99)*, pages 177-188, Singapore, 1999.
7. F. M. Barcala, J. Vilares, M. A. Alonso, J. Grana, and M. Vilares. Tokenization and proper noun recognition for information retrieval. In *DEXA Workshop 2002*, pages 246250. IEEE Computer Society Press, 2002.
8. J. J. Rocchio. Relevance Feedback in Information Retrieval. In G. Salton, ed. *The SMART Retrieval System–Experiments in Automatic Document Processing.* Prentice Hall, Englewood Cliffs, NJ, 1971
9. J. Lee. Analyses of multiple evidence combination. In *Proc. of SIGIR '97, July 27-31, Philadelphia, PA, USA*, pages 267–276. ACM Press, 1997.
10. E. Fox and J. Shaw. Combination of multiple searches. In D. K. Harman, ed., *NIST Special Publication 500-215: The 2nd Text REtrieval Conference (TREC-2)*, pp 243–252, USA, 1994. National Institute of Standards and Technology.
11. J. Graña, J.-C. Chappelier, and M. Vilares. Integrating external dictionaries into stochastic part-of-speech taggers. In *Proc. of the Euroconference Recent Advances in Natural Language Processing (RANLP 2001)*, pp 122–128, Bulgaria, 2001.
12. C. Jacquemin and E. Tzoukermann. NLP for term variant extraction: synergy between morphology, lexicon and syntax. In T. Strzalkowski, editor, *Natural Language Information Retrieval*, pages 25–74. Kluwer Academic, Dordrecht, 1999.
13. Manuel Vilares, Juan Otero Fco. Mario Barcala, and Eva Domnguez, *Automatic Spelling Correction in Galician*, in Advances in Natural Language Processing, vol. 3230 of Lecture Notes in Artificial Intelligence, pp. 51-57, Springer-Verlag, 2004. ISSN 0302-9743

Unsupervised Learning in Information Retrieval Using NOW Architectures*

E.F. Combarro[1], J. Ranilla[1], R. Mones[1], N. Vázquez[1],
I. Díaz[1], and E. Montañés[1]

Artificial Intelligence Center, University of Oviedo,
Campus de Viesques, Gijón (Asturias) Spain
`ir@aic.uniovi.es`

Abstract. The efficiency and effectiveness of the retrieval of documents which are relevant to a certain topic or user query can be improved by means of the clustering of similar documents as well as by introducing parallel strategies. In this paper we explore the use of unsupervised learning, using clustering algorithms based on neural networks, as well as the introduction of *NOW Architectures*, a kind of low-cost parallel architecture, and study the impact on Information Retrieval.

1 Introduction

In many situations we have a large collection of text files and we want to retrieve those documents relevant to a certain topic. This is known with the name of Information Retrieval (IR) [1].

Usually, the user of an IR system writes a query which is confronted to all the files of the collection and, then, those most similar to the content of the query are returned to her/him. The different ways of comparing the documents with the query and the different notions of relevance give raise to different IR systems.

A simple approach to this task consists of representing the documents and the query in a similar way (for instance, by a vector of real numbers) and then computing the distance between the query and all the documents, returning to the user the closest ones [2]. When the number of documents is large, this approach requires a lot of computation time.

In order to reduce the number of comparisons needed, the documents can be clustered and a representative of each resultant group can be chosen. Then, the query has to be matched only against the representatives, and not against the whole collection. The documents of the clusters with representatives closer to the query are selected. Sometimes, this clustering also has the effect of improving the efficiency of the retrieval [3], since similar documents are usually clustered together.

---

* The research reported in this paper has been supported in part under MEC and FEDER grant TIN2004-05920.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 179–188, 2005.
© Springer-Verlag Berlin Heidelberg 2005

The existence of hardware-software parallel solutions of low cost and high performance, like network of workstations using free open-source software like Linux Terminal Server Project (LTSP) [4], allows us to study a solution which involves unsupervised learning and parallel computing with a well defined goal: The IR system must be highly scalable, efficient, cheaper than traditional ones and low-maintainable without forgetting to improve the overall IR process.

To this extent, some clustering algorithms based on self-organizing neural networks (as for example Self Organizing Maps (SOM) [5]) are studied in first place. After that, to study the effect of parallel architectures over the IR systems, these clustering algorithms are compared to their parallel version as well as with a simple IR method named Vector Space Model (VSM) [2]. Therefore, the clustering algorithms as well as the traditional approaches are parallelized by distributing the examples in several ways.

2 Previous Work

The algorithm that we will use in this paper is an unsupervised clustering method based on neural networks. The networks consist in a set of *neurons* or *cells* with some connections among them. Each neuron has a numerical vector associated to it. The elements that are going to be clustered are also represented by vectors and presented to the network in the training phase. Each time that a new element is presented to the network the most similar neurons (their vectors) are slightly adjusted to resemble the element. After this training phase, each cell is the representative of a cluster formed by the elements more similar to it. This process is explained in more detailed in Section 3.2.

This kind of maps has been successfully applied to a wide variety of problems [6], including document management [7,8] and IR [3]. We have also applied SOM's to build an IR system from a massive collection of documents [9,10].

The typical use of these networks in IR is to cluster similar documents together. The search is then performed over the neurons instead of over the documents themselves and, hence, the computational complexity can be reduced several orders of magnitude.

The clustering has also the effect of returning documents that may be relevant even though they do not have many words in common with the query. There is evidence that this might lead to an improvement in the document retrieval [3] at least compared with simple IR methods such as VSM and Latent Semantic Indexing (LSI) [11]. However, to our knowledge, in the literature there is no systematic study about the influence of this kind of clustering and of the choice of parameters. In fact, there exist only a few works [3,9,10] that study the use of SOM's in IR.

3 The System

3.1 NOW Architectures

The construction and use of the system involve different stages. First, we must select the parallel architecture. As we say above, we are interested in an archi-

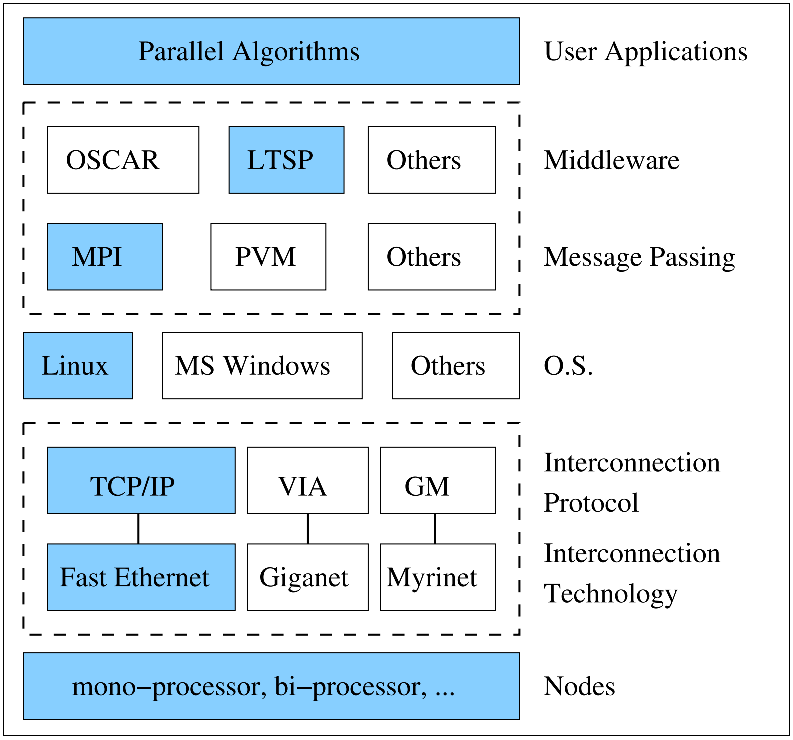


Fig. 1. NOW Architectures. The shaded boxes denote the alternatives used in this work.

tecture which presents high-performance and high scalability, which is reusable for other tasks, and which is cheaper than traditional ones and has a low maintenance cost. This kind of architectures are known as *N*etwork *O*f *W*orkstations (NOW), and there exist several hardware-software alternatives to build them.

In this work we have used Fast Ethernet, Linux as operating system, MPI [12] as message passing layer and Linux Terminal Server Project [4] as middleware (see Figure 1).

The main advantage of our selection is that it is no necessary to install any software in the computation nodes (they do not need hard disk) and, therefore, any available computer can be use without modifying its configuration.

The disadvantage is that the whole software is installed in a special computer (the server), that must have fast and robust hard disks and very good network connection. But, in general, these restrictions also apply to the rest of solutions.

3.2 The IR System

Once that the hardware-software architecture is settled up, we study the architecture of the IR system. First, the documents are transformed from their

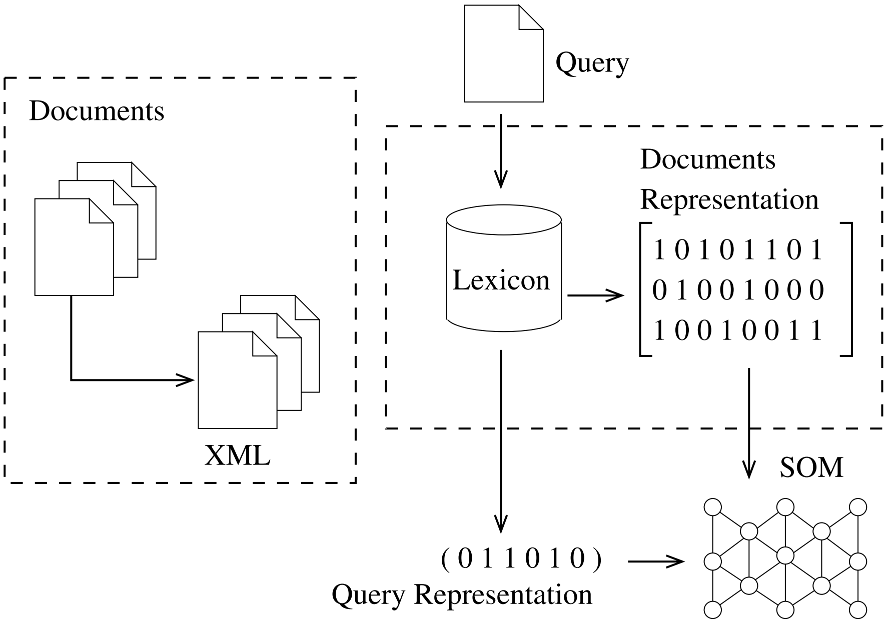


Fig. 2. The IR System

original format into XML. Then, the documents are represented using the *bag-of-words* model. After that, the networks are trained. All these phases have to be carried out before the system can be used. The last stage is the search process itself, in which the user query is matched against the neurons of the network and the most similar documents are returned.

These stages are described in more detail in the following sections and are graphically represented in Figure 2. In each section we study the possible parallelization of the corresponding stage.

Conversion to XML. First of all, the documents in the collection are converted to XML (see Figure 2) from whichever format they are in (appropriate translators are needed for each specific format such as pdf, html or plain text).

The conversion of each document is independent from the others, so this task is parallelized in a straightforward way.

Documents Representation. The use of the clustering algorithms based on neural networks requires that the documents are represented as numerical vectors. This is a common practice in IR, where the *bag of words* [1] representation is widely adopted.

With this representation each document is identified with a vector whose components measure the importance of a word in the document. Usually, this importance is measured by means of its *tfidf* [1] defined by

$$tfidf = tf \log(\frac{N}{df})$$

where *tf* is the number of times that the word appears in the collection, N is the number of documents in the corpus and *df* is the number of different documents in which the word appears. In this paper we adopt this procedure.

Then, previous to document representation, it is necessary to obtain the lexicon or set of words that appear in the collection (see Figure 2). Among them, some words will be too frequent to be informative (for instance adverbs, pronouns, ...) and are eliminated. These words are known as *stop words.*

Also, there will be different words with the same root or *stem* (for instance, *house* and *houses*) which share a common meaning. The usual practice consists of using the *stems* instead of the words themselves. To perform this *stemming* we use the algorithm proposed by Porter [13].

To parallelize this stage we have to tackle the construction of the lexicon and of the matrix representation. The strategy used is based on a master/slave model. Depending on the resources available (number of processors) and the characteristics of the collection (number of documents, ratio attributes/documents, ...) the algorithm is central (just one master) or distributed (each node is master). In this last case, the nodes are arranged in a tree-like hierarchy for the lexicon to be assembled. The number of documents used as threshold for updating the lexicon can also be adjusted.

Kohonen's Self-organizing Maps. The algorithm that will be used for clustering documents produces topological networks of neurons or units that can be connected among them. Each neuron has associated to it a vector of real numbers. The process of obtaining (or *training*) a network whose neurons represent clusters of documents involves several stages and the choice of some parameters.

Kohonen's Self-Organizing Maps are characterized by the fact that they have a rectangular shape and a fixed number of neurons. Also, the way the neurons are interconnected (the topology) is predetermined and does not change during the training process.

Before the training of the network begins, the size (and topology) of the network must be fixed. Then, the values of the vectors of the neurons are randomly initialized.

After the initialization, these values are gradually adapted to represent the documents in the following way:

1. A new document is presented to the network
2. The closest neuron to the document is computed
3. The values of that neuron and of some of its neighbors are adjusted to be closer to the document

This process is repeated a predefined number of times. Also, the way in which the closest neuron is selected (that is, which metric is used to compute the distances) and the degree to which the values of the neurons are modified have to be selected in advance. These values are adjusted using the following formula

$$v_i(t+1) := v_i(t) + h(t) \cdot (x_i - v_i(t))$$

where $v_i(t)$ is the value of the i-th component of the vector at iteration t, $v_i(t+1)$ is the new value for this component at iteration $t+1$, x_i is the value of the i-th component of the document and h is a function which usually depends on a value α called the *learning rate*. The bigger α, the closer $v_i(t+1)$ will be to x_i. A usual choice for h is the *Gaussian function* [5].

The neurons whose values are adapted in each iteration depend on the topology of the network (which can be *rectangular* or *hexagonal*) and on another parameter r called the radius. The bigger r, the more neurons are modified with each document.

After the SOM is trained, each document of the corpus is associated to the neuron which is closest to it. In this way, each neuron is the representative of a group of similar documents.

There exist several alternatives for the parallelization. In the most straightforward, the SOM is splitted into different parts assigned to the different processors. This implies a high flow of communications among the different nodes and it can become inefficient. However, if we adopt Batch SOM's [14] it is possible to process the documents independently since the modification of the vectors of the neurons is delayed until all the documents are presented to the map. Thus, we assign a different group of documents to each computer in the cluster and the adjusting weights are computed independently for each of them. When all the documents are processed the weights are summed up and the SOM is updated. We have selected this latter approach. This is the critical step in the system parallelization and its efficiency will be studied in deep in Section 4.

The choice of the parameters of the training phase, specially the size of the network and the number of documents presented to it, can affect the overall performance of the retrieval. In this work we will use the values of parameters which have shown the best results in previous experiments [15].

Search Process. We adopt the method presented in [3,9,10] to search the network for the most relevant documents. Given a query, this involves the following steps:

1. The query is represented in the same way that the documents are
2. While there are not enough documents:
   - The next closest neuron to the query is computed
   - All the documents associated to the neuron are regarded as relevant
3. The documents selected are ordered taking into account their distance to the query
4. A number of the closest selected documents is returned to the user

Notice that there are two parameters that will influence the performance of the method: the number of documents selected from the network and the number of documents returned to the user. We will study their behavior when analyzing the results of the experiments (Section 5).

The parallelization in this step consists of processing each query independently from the others.

4 The Experiments

In this section we describe the collections of documents that have been used in our experiments, the range of system parameters and the measures adopted to quantify the performance of the systems.

4.1 The Corpora

For the experiments we use 6 different corpora which are well-known to the IR community. These are *adi*, *cacm*, *cisi*, *cran*, *med* and *time*. They are distributed with the SMART system [16] and are widely used in the literature.

All these corpora are publicly available for research purposes[1] and are distributed with a set of queries and the corresponding relevance assessments.

The main properties of these document collections are presented in Table 1.

Table 1. Properties of the corpora

Corpus	adi	cacm	cisi	cran	med	time
Number of documents	117	1587	1460	1398	1033	425
Number of different words	1015	4845	5683	4849	9287	13620
Average length of documents	51.69	93.95	119.18	163.66	153.01	622.748
Number of queries	35	64	111	225	30	83
Av. number of relevant documents	4.86	15.31	40.97	8.18	23.20	3.90

4.2 Settings of the Experiments

For the training of the networks we use the values of parameters which have shown the best results in previous experiments [15].

Since there exist random elements in the network training (the initialization of the neurons), we have repeated the process 30 times. For VSM only one repetition is performed for each collection, since the system is deterministic.

4.3 The Evaluation

The measure considered to quantify the effectiveness of the system is the *R-precision* [17] or simply RP. Given a query, RP is equal to the precision (i.e. the percentage of relevant documents) in the first R documents returned by the system, where R is the total number of documents relevant to the query.

In our experiments this is calculated for each query and the values are averaged over all the queries for each corpus.

To compare the performance of the parallel and sequential versions of the algorithms we use the *efficiency* defined by

$$E = \frac{t_s}{kt_p}$$

where t_s is the sequential time, t_p is the parallel time and k is the number of processors used by the parallel algorithm.

---

[1] The collections can be found at ftp://ftp.cs.cornell.edu/pub/smart/

5 The Results

In Table 2 the results of VSM are compared to the best results obtained with networks. Together with the average value of RP in the 30 repetitions of each experiment, we present the amplitude of the confidence interval (at a confidence level of 95%) for it.

With the exception of corpus *time*, the results obtained by using networks are better than those obtained with VSM, either with the best possible choice of parameters or with the proposed choice. In fact, in all these cases the RP obtained with VSM is below the lower confidence limit for the average RP obtained with networks. In the case of *time* corpus, the results of VSM and of the networks are not significantly different.

The efficiency of the parallelization of the SOM training can be seen in Figure 3.

Table 2. Comparison of VSM with Neural Networks

Corpus	VSM	Best Network
adi	0.3827	0.3876 ± 0.0048
cacm	0.3013	0.3057 ± 0.0015
cisi	0.2363	0.2486 ± 0.0024
cran	0.3286	0.3356 ± 0.0011
med	0.4962	0.5508 ± 0.0025
time	0.5907	0.5903 ± 0.0007

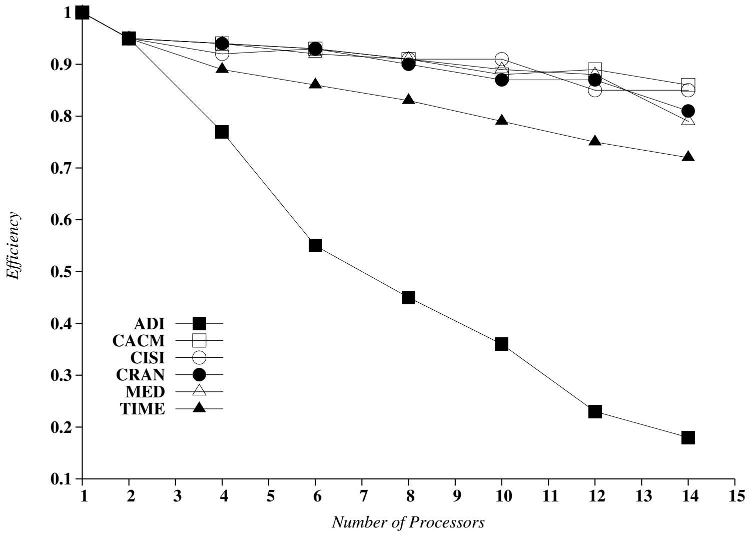


Fig. 3. Efficiency

With the exception of *adi*, which is a collection of extremely small size, the efficiency of the parallel algorithm is very good. In fact, it is above 0.8 for up to 10 processors and even for up to 15 processors if we exclude the collection *time* (which is the second in size after *adi*).

6 Conclusions and Future Work

We have presented a parallel version using NOW architectures of an IR system based on neural networks for clustering similar documents together. The results of the experiments show that the efficiency of the parallelization is good (excepting those cases in which the size of the collection is too small). Also, the clustering can lead to an improvement in the effectiveness of the retrieval.

In future research, we plan to tackle the parallelization of other clustering algorithms based on neural networks (as, for instance, Growing Cell Structures [18] and Growing Neural Gas [19]) using *NOW* architectures.

References

1. Salton, G., McGill, M.J.: An introduction to modern information retrieval. McGraw-Hill (1983)
2. Salton, G., Wong, A., Yang, C.: A vector space model for automatic indexing. Communications of the ACM **18** (1975) 613–620
3. Lagus, K.: Text retrieval using self-organized document maps. Neural Processing Letters **15** (2002) 21–29
4. LTSP: Linux Terminal Server Project. (`http://www.ltsp.org`)
5. Kohonen, T.: Self-Organizing Maps. Volume 30 of Springer Series in Information Science. Springer Verlag (2001)
6. Kaski, S., Kangas, J., Kohonen, T.: Bibliography of self-organizing map (SOM) papers: 1981-1997. Neural Computing Surveys **1** (1998) 1–176
7. Lagus, K., Honkela, T., Kaski, S., Kohonen, T.: Self-organizing maps of document collections: A new approach to interactive exploration. In Simoudis, E., Han, J., Fayyad, U., eds.: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. AAAI Press, Menlo Park, California (1996) 238–243
8. Kohonen, T., Kaski, S., Lagus, K., Honkela, T.: Very large two-level SOM for the browsing of newsgroups. In von der Malsburg, C., von Seelen, W., Vorbrüggen, J.C., Sendhoff, B., eds.: Proceedings of ICANN96, International Conference on Artificial Neural Networks, Bochum, Germany, July 16-19, 1996. Lecture Notes in Computer Science, vol. 1112. Springer, Berlin (1996) 269–274
9. Fernández, J., Mones, R., Díaz, I., Ranilla, J., Combarro, E.F.: Clustering and retrieval of spanish news documents using self organizing maps. In Peters, C., ed.: Working Notes for the CLEF 2003 Workshop. (2003)
10. Fernández, J., Mones, R., Díaz, I., Ranilla, J., Combarro, E.F.: Experiments with self organizing maps in clef 2003. In: Comparative Evaluation of Multilingual Information Access Systems, 4th Workshop of the Cross-Language Evaluation Forum, CLEF 2003. Number 3237 in Lecture Notes in Computer Science, Springer-Verlag (2004) 358–366

11. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic indexing. Journal of the American Society for Information Science **41** (1990) 391–407
12. MPICH: A portable implementation of MPI. (`http://www-unix.mcs.anl.gov/mpi`)
13. Porter, M.F.: An algorithm for suffix stripping. Program (Automated Library and Information Systems) **14** (1980) 130–137
14. Lawrence, R.D., Almasi, G.S., Rushmeier, H.E.: A scalable parallel algorithm for self-organizing maps with applications to sparse data mining problems. Data Min. Knowl. Discov. **3** (1999) 171–195
15. Mones, R., Vázquez, N., Ranilla, J., Combarro, E.F., Díaz, I., Montañés, E.: Self-organizing maps for clustering in Information Retrieval. (Submitted)
16. Salton, G.: The SMART retrieval system. Experiments in automatic document proccesing. Prentice Hall (1971)
17. Voorhess, E.M., Harman, D.K.: Appendix: Evaluation techniques and measures. In: Proceedings of the Eighth Text Retrieval Conference (TREC 8), NIST (2000)
18. Fritzke, B.: Growing cell structures—a self-organizing network in *k* dimensions. In Aleksander, I., Taylor, J., eds.: Artificial Neural Networks, 2. Volume II., Amsterdam, Netherlands, North-Holland (1992) 1051–1056
19. Fritzke, B.: A growing neural gas network learns topologies. In Tesauro, G., Touretzky, D.S., Leen, T.K., eds.: Advances in Neural Information Processing Systems 7. MIT Press, Cambridge MA (1995) 625–632

An Iterative Method for Mining Frequent Temporal Patterns

Francisco Guil[1], Antonio Bailón[2], Alfonso Bosch[1], and Roque Marín[3]

[1] Departamento de Lenguajes y Computación,
Universidad de Almería, 04120 Almería
{fguil, abosch}@ual.es
[2] Dept. Ciencias de la Computación e Inteligencia Artificial,
Universidad de Granada, 18071 Granada
bailon@decsai.ugr.es
[3] Dept. Ingeniería de la Información y las Comunicaciones,
Universidad de Murcia, 30071 Espinardo (Murcia)
roque@dif.um.es

Abstract. The incorporation of temporal semantic into the traditional data mining techniques has caused the creation of a new area called Temporal Data Mining. This incorporation is especially necessary if we want to extract useful knowledge from dynamic domains, which are time-varying in nature. However, this process is computationally complex, and therefore it poses more challenges on efficient processing that non-temporal techniques. Based in the inter-transactional framework, in [11] we proposed an algorithm named $TSET$ for mining temporal patterns (sequences) from datasets which uses a unique tree-based structure for storing all frequent patterns discovered in the mining process. However, in each data mining process, the algorithm must generate the whole structure from scratch. In this work, we propose an extension which consists in the reusing of structures generated in previous data mining process in order to reduce the execution time of the algorithm.

1 Introduction

Data mining is an essential step in the process of knowledge discovery in databases that consists of applying data analysis and discovery algorithms that produce a particular enumeration of structures over the data [9]. There are two types of structures: models and patterns. So, we can talk about local and global methods in data mining [17]. In the case of local methods, the simplest case of pattern discovery is finding *association rules* [2]. The initial motivation for association rules was to aid in the analysis of large transactional databases. The discovery of association rules can potentially aid decision making within organizations. Another approach is integrating the data mining process into the development of Knowledge Based Systems [19].

Since the problem of mining association rules was introduced by *Agrawal* in [2], a large amount of work has been done in several directions, including

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 189–198, 2005.
© Springer-Verlag Berlin Heidelberg 2005

improvement of the *Apriori* algorithm, mining generalized, multi-level, or quantitative association rules, mining weighted association rules, fuzzy association rules mining, constraint-based rule mining, efficient long patterns mining, maintenance of the discovered association rules, etc. We want to point out the work in which a new type of association rules was introduced, the *inter-transaction association rules* [16,15]. Temporal data mining can be viewed as an extension of this work.

Temporal data mining can be defined as the activity of looking for interesting correlations or patterns in large sets of temporal data accumulated for other purposes [7]. It has the capability of mining activity, inferring associations of contextual and temporal proximity, some of which may also indicate a cause-effect association. This important kind of knowledge can be overlooked when the temporal component is ignored or treated as a simple numeric attribute [22].

Data mining is an interdisciplinary area which has received contributions from a lot or disciplines, mainly from databases, machine learning and statistic. In [25] we found a review of three books, each one written from a different perspective. Although each perspective make strong emphasis on different aspects of data mining (efficiency, effectiveness, and validity), only when we simultaneously take these three aspects into account we may get successful data mining results. However, in the case of temporal data mining techniques, the most influential area is artificial intelligence because its work in temporal reasoning have guided the development of many of this techniques. In non-temporal data mining techniques, there are usually two different tasks, the description of the characteristics of the database (or analysis of the data) and the prediction of the evolution of the population. However, in temporal data mining this distinction is less appropriate, because the evolution of the population is already incorporated in the temporal properties of the data being analyzed.

We can found in the literature a large quantity of temporal data mining techniques. We want to highlight some of the most representative ones. So, we can talk about sequential pattern mining [4], episodes in event sequences [18], temporal association rules mining [5,12,13], discovering calendar-based temporal association rules [14], patterns with multiple granularities mining [7], and cyclic association rules mining [20]. However, there is an important form of temporal associations which are useful but could not be discovered with this techniques. These are the inter-transaction associations presented in [15,16]. The introduction of this type of associations was motivated by the observation that many real-world associations happen under certain context, such as time, place, etc. In the case of temporal context, inter-transactional associations represents associations amongst items along the dimension of time. Due to the number of potential association becomes extremely large, the mining of inter-transaction association poses more challenges on efficient processing than classical approaches. In order to make the mining of inter-transaction associations practical and computationally tractable, several methods have been proposed in [24,23,10].

Working in the same direction, in [11] we presented an algorithm named $TSET$ based on the inter-transactional framework for mining frequent sequences (also called frequent temporal patterns or frequent temporal associations) from several kind of datasets. The improvement of the proposed solution was the use of a unique structure to store all frequent sequences. The data structure used is the well-known set-enumeration tree, commonly used in the data mining area [6,1,8], in which the temporal semantic is incorporated. Although the use of a unique data structure implies that the algorithm requires less resources than other approaches, mining this sort of associations is still a computationally intensive problem. So, it is necessary to devise new optimization techniques to speed up the global knowledge discovery process.

The aim of this paper is to propose an extension of the $TSET$ algorithm in order to reduce the time execution. It consists in the incorporation of a mechanism for reusing structures generated by previous data mining process. In cases where an initial data structure is presented, instead of generating the whole structure starting from scratch, the algorithm, named $TSET^I$, iterates over this initial structure looking only for new frequent temporal patterns.

The remainder of this paper is organized as follows. Section 2 gives a formal description of the problem of mining frequent temporal patterns (sequences) from datasets. Section 3 introduces the algorithm named $TSET^I$. Section 4 presents the preliminary performance results obtained. Conclusions and future works are finally drawn in Section 5.

2 The Frequent Sequence Mining Problem

Let us describe the notation, some basic definition, and the goals of the frequent sequences mining.

Definition 1. *A* dataset D *is an ordered sequence of records* $D[0]$*,* $D[1]$*,...,* *where each* $D[i]$ *can have col attributes, c[0],...,c[col-1]. The 0-attribute will be the dimensional attribute, the temporal data associated with the record, expressed in temporal units. The rest of attributes can be quantitative or categorical.*

We assume that the domain of each attribute is a finite subset of non-negative integers, and we also assume that the structure of time is discrete and linear. Due to every event registered has its absolute date identified, we represent the time for events with an absolute dating system [21]. In order to simplify the calculations, we transform the original dataset subtracting the date of each record from the date of the first record, the time origin.

With this generic definition of dataset, and with minimal modifications, the algorithm that we propose works with different types of sources, that is, relational databases, transactional databases and data streams.

Definition 2. *An* event *e is a 3-tuple* $(c[i], v, t)$*, where* $0 < i < col$*,* $v \in dom\{c[i]\}$*, and* $t \in dom\{c[0]\}$*, that is,* $t \in \mathbb{N}$*. Events are "things that happen", and they usually represent the dynamic aspect of the world [21].*

In our case, an event is related to the fact that a value v *is assigned to a certain attribute* $c[i]$ *with the occurrence time* t*. We will use the notation* $e.c$, $e.v$, *and* $e.t$ *to set and get the attribute, value, and time variables related to the event* e*.*

Definition 3. *Given two events* e_1 *and* e_2*, we define the* $\leq$ *relation as follows:*

1. $e_1 = e_2$ *iff* $(e_1.t = e_2.t) \wedge (e_1.c = e_2.c) \wedge (e_1.v = e_2.v)$
2. $e_1 < e_2$ *iff* $(e_1.t < e_2.t) \vee ((e_1.t = e_2.t) \wedge (e_1.c < e_2.c))$

We assume that a lexicographic ordering exists among the pairs (attribute, value) in the dataset.

Definition 4. *A* sequence (or event sequence) *is an ordered set of events* $S = \{e_0, e_1, ..., e_k\}$*, where* $e_i < e_{i+1}$*, for all* $i = 0,...,k-1$*.*

Obviously, $|S| = k + 1$*. Note that different events with the same temporal unit can belong to the same sequence. Also, the same events with different temporal unit associated can belong to the same sequence. But nevertheless, in any sequence there will exist two or more pairs (attribute, value) associated to the same temporal unit. In other words, an attribute can not take two different values in the same instant.*

Definition 5. *Let* U_{tmin} *be the minimal dimensional value associated to the sequence* S*. In other words,* $U_{tmin} = min\{e_i.t\}$*, for* $e_i \in S$*. If* $U_{tmin} = 0$*, we say that* S *is a* normalized sequence*.*

Note that any non-normalized sequence can be transformed into a normalized one through a normalization function.

Example 1. Let $S_1 = \{(0,0,0),(1,0,0),(3,0,2)\}$*, and* $S_2 = \{(0,0,3),(1,0,3),$- $(3,0,5)\}$ *be two sequences.* S_1 *is a normalized sequenced, since it has the minimal value equal to 0 for the temporal dimension. But* S_2 *is not a normalized sequence, because its minimal value is not equal to zero. However, we can normalize* S_2 *by subtracting its minimal value (*$U_{tmin} = 3$*) from the temporal values as follows:* $S_2' = \{(0,0,3-3),(1,0,3-3),(3,0,5-3)\}$*, resulting in the normalized sequence* $S_2' = \{(0,0,0),(1,0,0),(3,0,2)\}$*.*

Let U_{tmax} *be the maximal dimensional value associated to the sequence* S*. This value indicates the maximum distance amongst the events belonging to the normalized sequence* S*. In other words,* $U_{tmax} = e_{k+1}.t$*, where* $|S| = k + 1$*. From both, confidence and complexity points of view [15], this value will be always less or equal than a user-defined parameter called maxspan, denoted by* w*.*

Definition 6. *The* support *(frequency) of a sequence is defined as:*

$$support(S) = \frac{|D_S|}{|D|},$$

where $|D_S|$ *denotes the number of occurrences of the sequence* S *in the dataset, and* $|D|$ *is the number of records in the dataset* D*.*

Definition 7. *A* frequent sequence *is a normalized sequence whose support is greater or equal than a user-specified threshold called minimum support minsup.*

Given a dataset D, and the user-defined parameters $maxspan$ and $minsup$, the goal of temporal pattern (or sequence) mining is to determine in the dataset the set $\mathcal{S}^{D,w,ms}$, where $w = maxspan$ and $ms = minsup$, formed by all the frequent sequences whose support are greater than or equal to $minsup$.

3 The $TSET^I$ Algorithm

Knowledge Discovery and therefore data mining is a human-centered process. After setting the desired values for the user-defined parameters of the selected algorithm, the user analyzes the result of the data mining step in order to extract useful knowledge. In particular, $TSET$ has two user-defined parameters, minimum support ($minsup$) and the length of the temporal windows, ($maxspan$). Generally, after setting the value for $maxspan$, the user carries out an incremental knowledge discovery process modifying the value of $minsup$ in a gradual way. The incremental process consist in the reduction of the minimum support and the study of the new discovered patterns. However, as we can see in [11], with the reduction of this value the number of discovered temporal associations and the execution time of the algorithm increases exponentially. In order to reduce this complexity, we propose an iterative process that consist in the reusing of the structure generated by previous data mining processes (with higher $minsup$ values).

We will use an example to illustrate the basic idea. Suppose that Figure 1 shows the data structure generated by $TSET$ from a dataset D, with $maxspan = 1$ and $minsup = 2$.

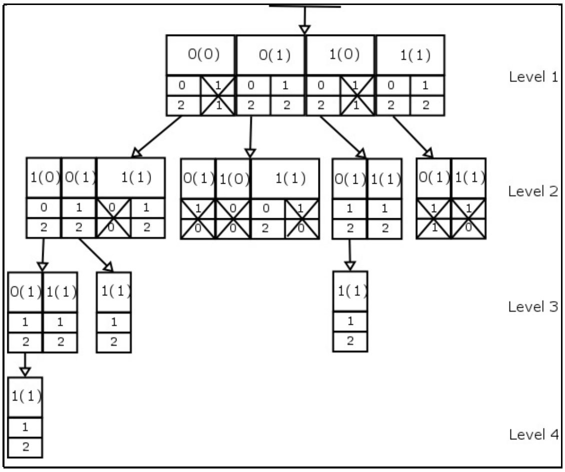


Fig. 1. The initial extended set-enumeration tree structure

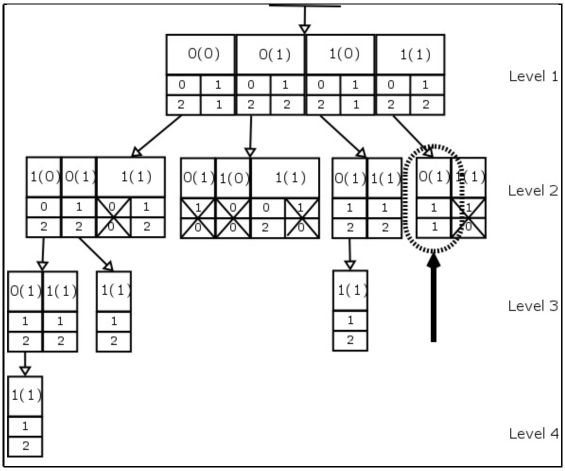


Fig. 2. The second extended set-enumeration tree structure

If the user sets the *minsup* value to 1, instead of generating the whole structure starting from scratch, the algorithm iterates over the structure looking for nodes with support less than or equal to the chosen value. In Figure 2 we can see the generated structure.

$TSET^I$ is an extension of $TSET$, and therefore, it follows the same basic principles as most apriori-based algorithms [2]. Frequent sequence mining is an iterative process, and the focus is on a *level-wise* pattern generation. At the beginning, all frequent 1-sequences (frequent events) are found, these are used to generate frequent 2-sequences, then 3-sequences are found using frequent 2-sequences, and so on. In other words, (k+1)-sequences are generated only after all k-sequences have been generated. On each cycle, the *downward closure* property is used to prune the search space. This property, also called anti-monotonicity property, indicates that if a sequence is infrequent, then all super-sequence must also be infrequent. Figure 3 outlines a generalized frequent sequences mining algorithm. We want to highlight that the main difference between $TSET$ and $TSET^I$ is the getSequences() method which ignores the sequences discovered in previous process.

```
algorithm $TSET^I$(dataset $D$, minsup $m$, maxspan $w$)
   begin
      if tree.isEmpty()
           tree.init($D$, $m$, $w$ );
      tree.getSequences($D$, $m$, $w$);
      Output(tree);
   end;
```

Fig. 3. $TSET^I$ Algorithm

```
procedure getSequences(tree, dataset $D$, minsup $m$, maxspan $w$)
  begin
    Queue $Q = \emptyset$;
    $Q$.push(tree.root);
    while ($Q \neq \emptyset$)
      begin
        act = Q.pop();
        foreach node $n$ in act
          if (n.support < minsup) continue;
          if(n.child <> NULL)
            begin
              $Q$.push(n.child);
              continue;
            end;
          newNode = n.getCandidates();
          newNode.evaluateSupport($D$, $w$);
          newNode.pruningInFrequent($m$);
          if(newNode $\neq \emptyset$)
            begin
              n.child = newNode;
              $Q$.push(newNode);
            end;
        end;
    end;
```

Fig. 4. The code for the method getSequences

4 Empirical Evaluation and Results

The experiments were carried out with modified datasets generated by the IBM test data generator used in [3]. Datasets generated from this tool have been commonly used for evaluating frequent items mining algorithms. By setting up the parameters (see Table 1) of the program, we can generate datasets of transactions as benchmarks to evaluate the improvement of our approach. After that, we add the temporal dimension into the dataset. We have implemented the algorithms $TSET$ and $TSET^I$ algorithm in C++ language and all the experiments were conducted on a PC with a 3GHz CPU and 512MB main memory. For the set of experiments we set T = 5, I = 5, L = 1k, N = 500, and D = 10k, varying the minimum support value from 1% to 0.1%, and the maxspan from 0 to 4 temporal units. The running time against the maxspan and support level are

Table 1. The meaning of all parameters

D	Number of transactions
T	Average size of transactions
I	Average size of the maximal potentially large itemsets
L	Number of potentially large itemsets
N	Number of items

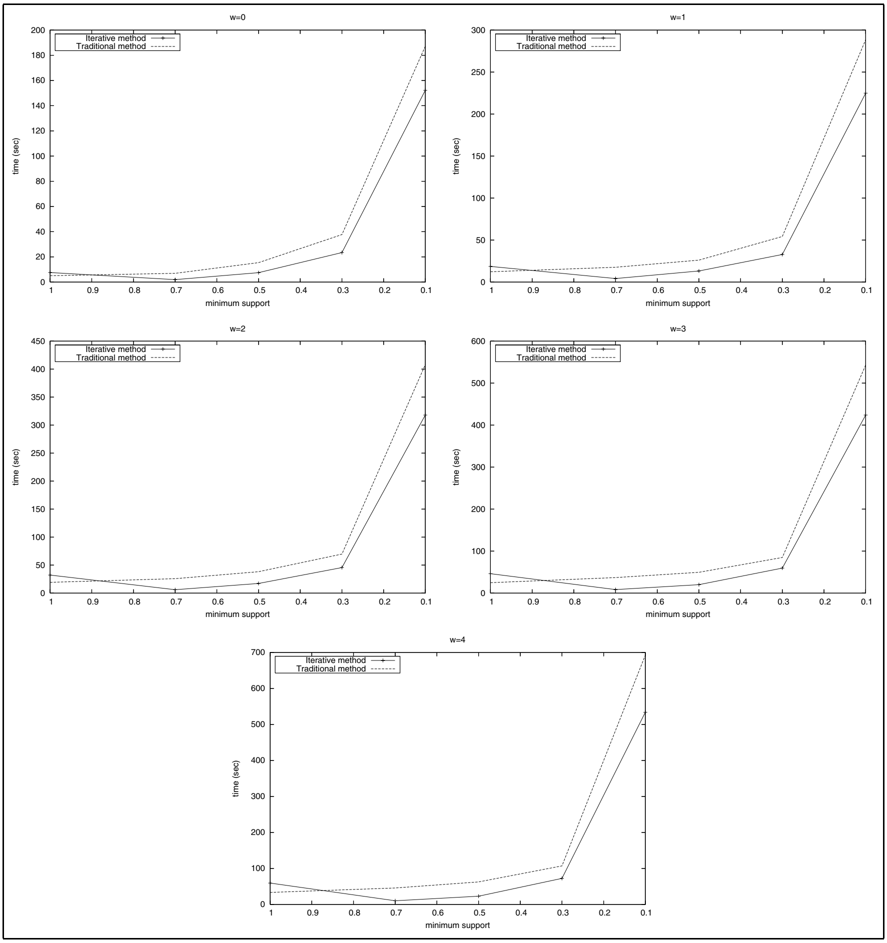


Fig. 5. Execution time versus Maxspan and Minimum Support

shown in Figure 5. We want to highlight the case in which the support value is equal to 1%. The execution time of the iterative method is greater than the non-iterative one because the initial node contains both frequent and non-frequent 1-sequences. In the rest of cases, we can see the improvement in the execution time of the iterative method.

5 Conclusions and Future Works

In this paper we have presented an extension of an algorithm which extracts temporal patterns from datasets. The extension is based on the incorporation of the possibility that the algorithm can reuse tree-based data structures generated by previous data mining processes, with the aim of reducing the time spent in the data mining process.

Several optimizations techniques have been devised to speed up the discovery of sequences. In particular, we are working in the adaptation of the techniques proposed in the literature to reduce the number of database passes, and therefore, to develop an algorithm which can deals with large and dense synthetic and real datasets in an efficient way.

References

1. C. C. Aggarwal. Towards long pattern generation in dense databases. *SIGKDD Explorations*, 3(1):20–26, 2001.
2. R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In P. Buneman and S. Jajodia, editors, *Proc. of the ACM SIGMOD Int. Conf. on Management of Data, Washington, D.C., May 26-28, 1993*, pages 207–216. ACM Press, 1993.
3. R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In J. B. Bocca M. Jarke and C. Zaniolo, editors, *Proc. of 20th Int. Conf. on Very Large Data Bases (VLDB'94), September 12-15, 1994, Santiago de Chile, Chile*, pages 487–499. Morgan Kaufmann, 1994.
4. R. Agrawal and R. Srikant. Mining sequential patterns. In P. S. Yu and A. L. P. Chen, editors, *Proc. of the 11th Int. Conf. on Data Engineering, March 6-10, 1995, Taipei, Taiwan*, pages 3–14. IEEE Computer Society, 1995.
5. Juan M. Ale and G. H. Rossi. An approach to discovering temporal association rules. In *Proc. of the 2000 ACM Symposium on Applied Computing, Villa Olmo, Via Cantoni 1, 22100 Como, Italy, March 19-21, 2000*, pages 294–300. ACM, 2000.
6. R. J. Bayardo. Efficiently mining long patterns from databases. In L. M. Haas and A. Tiwary, editors, *Proc. of the ACM SIGMOD Int. Conf. on Management of Data (SIGMOD 1998), June 2-4, 1998, Seattle, Washington, USA*, pages 85–93. ACM Press, 1998.
7. C. Bettini, X. S. Wang, and S. Jajodia. Testing complex temporal relationships involving multiple granularities and its application to data mining. In *Proc. of the 15th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 3-5, 1996, Montreal, Canada*, pages 68–78. ACM Press, 1996.
8. F. Coenen, G. Goulbourne, and P. Leng. Tree structures for mining association rules. *Data Mining and Knowledge Discovery*, 8:25–51, 2004.
9. U. Fayyad, G. Piatetky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *AIMagazine*, 17(3):37–54, 1996.
10. L. Feng, J. X. Yu, H. Lu, and J. Han. A template model for multidimensional inter-transactional association rules. *The VLDB Journal*, 11:153–175, 2002.
11. F. Guil, A. Bosch, and R. Marín. TSET: An algorithm for mining frequent temporal patterns. In *Proc. of the First Int. Workshop on Knowledge Discovery in Data Streams, in conjunction with ECML/PKDD 2004*, pages 65–74, 2004.
12. C. H. Lee, C. R. Lin, and M. S. Chen. On mining general temporal association rules in a publication database. In N. Cercone, T. Y. Lin, and X. Wu, editors, *Proc. of the 2001 IEEE Int. Conf. on Data Mining, 29 November - 2 December 2001, San Jose, California, USA*, pages 337–344. IEEE Computer Society, 2001.
13. J. W. Lee, Y. J. Lee, H. K. Kim, B. H. Hwang, and K. H. Ryu. Discovering temporal relation rules mining from interval data. In *Proc. of the 1st EurAsian Conf. on Information and Communication Technology (Eurasia-ICT 2002), Shiraz, Iran, October 29-31, 2002*, volume 2510 of *Lecture Notes in Computer Science*, pages 57–66. Springer, 2002.

14. Y. Li, P. Ning, X. S. Wang, and S. Jajodia. Discovering calendar-based temporal association rules. *Data & Knowledge Engineering*, 44:193–218, 2003.
15. H. Lu, L. Feng, and J. Han. Beyond intra-transaction association analysis: Mining multi-dimensional inter-transaction association rules. *ACM Transactions on Information Systems (TOIS)*, 18(4):423–454, 2000.
16. H. Lu, J. Han, and L. Feng. Stock movement and n-dimensional inter-transaction association rules. In *Proc. of the Workshop on Research Issues on Data Mining and Knowledge Discovery (DMKD'98), Seattle, Washington, June 1998*, pages 12:1–12:7, 1998.
17. H. Mannila. Local and global methods in data mining: Basic techniques and open problems. In P. Widmayer, F. Triguero, R. Morales, M. Hennessey, S. Eidenbenz, and R. Conejo, editors, *In Proc. of the 29th Int. Colloquium on Automata, Languages and Programming (ICALP 2002), Malaga, Spain, July 8-13, 2002*, volume 2380 of *Lecture Notes in Computer Science*, pages 57–68. Springer, 2002.
18. H. Mannila, H. Toivonen, , and A. I. Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289, 1997.
19. C. Ordonez, C. A. Santana, and L. de Braal. Discovering interesting association rules in medical data. In D. Gunopulos and R. Rastogi, editors, *Proc. of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, Dallas, Texas, USA, May 14, 2000*, pages 78–85, 2000.
20. B. Özden, S. Ramaswamy, and A. Silberschatz. Cyclic association rules. In *Proc. of the 14th Int. Conf. on Data Engineering, February 23-27, 1998, Orlando, Florida, USA*, pages 412–421. IEEE Computer Society, 1998.
21. A. K. Pani. Temporal representation and reasoning in artificial intelligence: A review. *Mathematical and Computer Modelling*, 34:55–80, 2001.
22. J. F. Roddick and M. Spiliopoulou. A survey of temporal knowledge discovery paradigms and methods. *IEEE Transactions on Knowledge and Data Engineering*, 14(4):750–767, 2002.
23. A. K. H. Tung, H. Lu, J. Han, and L. Feng. Breaking the barrier of transactions: Mining inter-transaction association rules. In *Proc. of the 5th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, August 15-18, 1999, San Diego, CA, USA*, pages 297–301. ACM Press, 1999.
24. A. K. H. Tung, H. Lu, J. Han, and L. Feng. Efficient mining of intertransaction association rules. *IEEE Transactions on Knowledge and Data Engineering*, 15(1):43–56, 2003.
25. Zhi Hua Zhou. Three perspectives of data mining (book review). *Artificial Intelligence*, 143:139–146, 2003.

Data Mining with Scatter Search*

I.J. García del Amo, M. García Torres**, B. Melián Batista,
J.A. Moreno Pérez, J.M. Moreno Vega, and Raquel Rivero Martín

Dep. de Estadística, I.O. y Computacin,
Universidad de La Laguna, 38271 La Laguna, Spain

Abstract. Most Data Mining tasks are performed by the application of Machine Learning techniques. Metaheuristic approaches are becoming very useful for designing efficient tools in Machine Learning. Metaheuristics are general strategies to design efficient heuristic procedures. Scatter Search is a recent metaheuristic that has been successfully applied to solve standard problems in three central paradigms of Machine Learning: Clustering, Classification and Feature Selection. We describe the main components of the Scatter Search metaheuristic and the characteristics of the specific designs to be applied to solve standard problems in these tasks.

1 Introduction

Processing Intelligent Information requires efficient tools to extract the useful information stored in databases. Data Mining and Knowledge Discovery are powerful techniques for the extraction of information from large databases. Heuristic approaches are already quite relevant in Data Mining [1]. Most of the data mining tasks are solved by the application of Machine Learning techniques. Three central paradigms for the application of Machine Learning [9] in Data Mining are Clustering, Instance-Based Classification and Feature Selection. The Scatter Search metaheuristic has been tested for the kind of problems that appear in these tasks ([3], [2]). We describe the main components of this metaheuristic and their specific designs to solve standard problems in these contexts.

Given a set of instances characterized by several features, the clustering or grouping problem consists in grouping similar instances in the same cluster and dissimilar instances in different clusters. If in addition to the description of the objects, their classes in a training set are given, the classification problem consists of obtaining the optimal classification rule to assign the class to the new examples based on their description. Finally, the feature selection problem consisting in selecting a subset of features in order to best perform the classification task.

Scatter Search [8] is a population-based metaheuristic that constructs solutions by combining others in an evolving set of solutions, named reference set

---

[*] This research has been partially supported by the Ministerio de Ciencia y Teconología through the project TIC2002-04242-C03-01; 70% of which are FEDER funds.

[**] This author has been partially supported by the Cajacanarias grant.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 199–204, 2005.
© Springer-Verlag Berlin Heidelberg 2005

(*RefSet*). The procedure combines solutions of the reference set and runs a local search procedure to reach a local optimum that would be used to update the reference set depending on the results of the improvements. The two main differences between Scatter Search and other classical population-based procedures in Data Mining [7] like Genetic Algorithms [6] are the size of the evolving set of solutions and the way the method combines the existing solutions to provide new ones. The evolving set *RefSet* in scatter search has a relatively small or moderate size (typical sizes are 10 or 15, see [8]). Scatter Search combines good solutions to construct others exploiting the knowledge of the problem at hand in an intelligent way. Genetic Algorithms are also evolutionary algorithms in which a population of solutions evolves by using the mutation and crossover operators, which have a significant reliance on randomization to create new solutions.

2 Scatter Search Metehauristic

The principles of the Scatter Search metaheuristic were first introduced in the 1970s as an extension of formulations for combining decision rules and problem constraints. This initial proposal generates solutions taking account of characteristics in several parts of the solution space [4].

In a Scatter Search algorithm [8], a moderate-sized set of solutions, the reference set *RefSet*, evolves due to mechanisms of intelligent combination of solutions. Unlike other strategies of combination of existing rules like genetic algorithms, the search for a local optimum is a guided task. In order to perform this strategy the set of reference solutions, (*RefSet*), is selected from a population of solutions. The *RefSet* is generated and then iteratively updated attempting to intensify and diversify the search. After intelligently combining the solutions in the reference set, a local search procedure is applied to improve the resulting solution, and the *RefSet* is updated to incorporate both good and disperse solutions. These steps are repeated until a stopping condition is met. The method provides not only a single heuristic solutions, like other metaheuristics, but a reduced set of disperse high quality solutions.

The Scatter Search metaheuristic includes five main methods or component processes:

1. Diversification Generation Method This method is used to generate a wide set of diverse solutions.
2. Improvement Method This component process improves the solutions to reach better ones; usually local optima.
3. Reference Set Update Method This is the method that builds and updates the reference set, which consists of a reduced set of good and disperse solutions.
4. Subset Generation Method This is the method applied to select the subsets of solutions from the reference set to be combined.
5. Solution Combination Method This process combines the solutions in the selected subsets to produce new solutions

A comprehensive description of the fundamentals of Scatter Search can be found in [5].

A simple implementation of the basic Scatter Search algorithm based in these methods is shown in Figure 1.

```
procedure Scatter Search

begin
    Diversification Generation Method;
    Improvement Method;
    repeat
        Reference Set Update Method;
        Subset Generation Method;
        Solution Combination Method;
        Improvement Method;
    until (StoppingCriterion);
end.
```

Fig. 1. A Scatter Search Metaheuristic Pseudocode

The algorithm starts generating a population of solutions by running the *Diversification Generation Method.* This procedure creates a large set of disperse solutions that are improved by the *Improvement method.* A representative set of $RefSetSize$ good solutions are chosen to be included in the reference set ($RefSet$). These solutions are not limited to those with the best objective function values; the reference set must also include diverse solutions. The reference set is initially generated by selecting the $RefSetSize_1$ best solutions according to the objective function values that are chosen to be in $RefSet$. Then $RefSetSize_2$ times, the most disperse solution with respect to $RefSet$ is found and added to $RefSet$ (the final size of $RefSet$ is $RefSetSize = RefSetSize_1 + RefSetSize_2$). Several subsets of solutions from the $RefSet$ are then systematically selected by the *Subset Generation Method.* The *Solution Combination Method* combines the solutions in each subset taking account their good features without reliance on randomization. Then, the *Improvement Method* is applied to the result of the combination to get an improved solution. Finally, the *Reference Set Update Method* uses the obtained solution to update the reference set following both intensification and diversification criteria.

3 Application of Scatter Search in Data Mining

Metaheuristic searches are becoming very important for Machine Learning applications in Data Mining, Medical Record, Software Engineering, Autonomous

Driving, Speech Recognition and Self Customizing programs. Three main paradigms in Machine Learning applications in Data Mining are: clustering, instance-based learning and feature selection. We describe the specific design of the main components of the Scatter Search to solve standard problems in these three tasks.

Clustering is the main paradigm of unsupervised learning. The objective of clustering is to find groups of instances constituted by similar instances. Given a set of instances described by a series of features, the problem is to find a partition of the whole set in subsets or classes in such a way that instances in the same class are very similar and instances in different classes are very dissimilar. The distance based approach considers a distance between the instance descriptions to evaluate the similarity and dissimilarity among them. A wide set of distance functions appropriated for different kinds of instance descriptions have been proposed and analyzed in the literature (see [10]). Then, a very usual way to define the partition consists of finding some representative instances for the classes (in the simplest case only one instance is chosen for each class). Then each instance is assigned to the class of the nearest of these representative instances. Scatter Search has been successfully applied to the p-median location problem that is very similar to the Clustering Problem [3]. The p-median problem consists in choosing the p points that minimize the sum of distances to the remainder instances.

In instance-based supervised learning, in addition to the features that describe the instances, an additional variable that represents the class of the instance that is to be predicted from its description is considered. From a training set of instances with known classes, we want to get a classification rule to obtain the unknown classes of a set of test instances or examples. The distance-based classification approach also selects a set of representative instances from the training set and classifies the test instances taking into account the class of the nearest selected instance. A wide set of possible distance functions among descriptions can also be applied for this tasks. This problem is also similar to the p-median problem since it also consists of selecting a number of instances with a different optimization function. They belong to the wide set of the named p-selection problems, for which most of the heuristic procedures are based on swaps in the solutions. The scatter search approach for a p-selection problem based on interchange moves can be easily adapted to other problem in this class.

However, the use of the whole set of features is not useful for being considered in this or other classification paradigms. The feature selection problem tries to get the best subset of features to perform the classification task. The appropriated selection of features has not only the advantage of taking the relevant information in the description of the instances, but also avoiding redundant information and making the classification algorithms and rules more efficient to obtain and to use. Scatter Search has been tested for this problem in [2] using a distance between solutions (now sets of features) to evaluate the diversity among a set of solutions.

The distance function between solutions plays a central role in the Scatter Search to modulate the diversification and intensification. Given a distance between the items that constitute the solutions (instances for clustering and classification and variables for feature selection), the distance between two solutions is the sum of the distances between the items in one solution and the other solution. Similarly, the most diverse solution with respect to a set of solutions is defined in a similar way.

The most important parameter in the population creation method is the size of the population. The usual sizes are a quadratic or linear function of the number of classes for clustering and classification and the number of features to be chosen for feature selection problem. Usual procedures consist of randomly generating solutions from which a good population is obtained by quality and diversity criteria. The *Reference Set Update Method* generates and updates the reference set by following both quality and diversity criteria. Solutions for the reference set are first chosen by quality; e.g. the $RefSetSize/2$ best solutions. Then new solutions are iteratively included in the reference set by following a diversity criterium until the whole reference set is obtained. A usual procedure is described as follows. Let C be the set of items that belong to any solution already in the reference set. The diversity of each possible new solution S is given by a distance between S and C using a corresponding distance measure between items. Then the most diverse solution is chosen $RefSetSize/2$ times until the reference set with $RefSetSize = RefSetSize_1 + RefSetSize_2$ solutions is obtained.

The usual *Subset Generation Method* in the applications of Scatter Search consists of considering all the subsets of a fixed size (usually two) of solutions in the current reference set of solutions. The solutions in the subsets are then combined to construct other solutions avoiding repetitions if the subset have been previously used in a combination. The *Solution Combination Method* combines good characteristics of the selected solutions to get new current solutions.

The possible combination methods for these problems are random/greedy strategies. They start with a partial solution consisting of the items common to the solutions to be combined. Then, at each iteration, one of the remaining items in some of the combined solutions is added. The criteria applied to select the items are between the pure random and greedy criteria and consist of selecting at random one of the most improving item.

The *Improvement Method* applied to the solutions of the population and those generated by the combination method are typical local searches. They are mostly based on the basic exchange method that consists in replacing an item in the solution by an item out of the solution. The solutions obtained by improving the combined solutions are used to update $RefSet$ by the Reference Set Update Method.

The *Reference Set Update Method* also applies intensity and diversity criteria to update the reference set using the improved solutions. Using the strategy called *Static Update*, the improved solutions obtained after combination and improvement are recorded in a pool of solutions, $ImpSolSet$. The method selects the $RefSetSize$ best solutions from $RefSet \cup ImpSolSet$. If a *Dynamic Update* strategy is used, the combination method would be applied to new solution faster than in the static strategy. That is, instead of waiting until all the combinations have been performed to update the reference set, if a new solution is to be added to the reference set because it is better than the worst, this set is updated before the next subset of solutions combination is carried out.

4 Conclusions

The Scatter Search metaheuristic has been proved to be useful for the main standard tasks in Machine Learning for Data Mining: Clustering, Classification and Feature Selection. Future research will be oriented to use scatter search also for the instance pruning problem.

References

1. H.A. Abbass, C.S. Newton , R. Sarker. *Data Mining: A heuristic Approach.* Idea Group (2002).
2. F. García López, M. García Torres, B. Melián Batista, J.A. Moreno Pérez and J.M. Moreno Vega. Solving Feature Subset Selection Problem by a Parallel Scatter Search, *European Journal of Operational Research*, 2005, to appear.
3. F. García López, B. Melián Batista, J.A. Moreno Pérez and J.M. Moreno Vega. Parallelization of the Scatter Search for the p-median problem, *Parallel Computing*, 29 (2003) 575-589.
4. Glover, F., Heuristics for Integer Programming using Surrogate Constraints, *Decision Sciences* 8, (1977) 156–166
5. Glover, F., Laguna, M., Martí, R. Fundamentals of Scatter Search and Path Relinking *Control and Cybernetics*, 39, (2000) 653-684
6. Goldberg, D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, (1989)
7. Ghosh, A. and Jain, L.C. (Eds.) *Evolutionary Computation in Data Mining* Studies in Fuzziness and Soft Computing, 163. Springer (2005)
8. Laguna, M. and R. Martí, *Scatter Search: Metodology and Implementations in C*, Kluwer Academic Press, (2003).
9. Mitchell, T. *Machine Learning*, Series in Computer Science, McGraw-Hill, (1997).
10. D. R. Wilson, T. R. Matinez, Improved heterogeneous distance functions, Journal of Artificial Intelligence Research 6 (1997) 1–34.

Web Usage Mining Project for Improving Web-Based Learning Sites

M.E. Zorrilla[1], E. Menasalvas[2], D. Marín[1], E. Mora[1], and J. Segovia[2]

[1] Department of Applied Mathematics and Computer Sciences, University of Cantabria,
Avda. de los Castros s/n 39005 Santander. Spain
{zorrilm, morae}@unican.es, dmoujo@yahoo.es
[2] Dpto de Lenguajes y Sistemas Informáticos. Facultad de Informática,
Universidad Politécnica de Madrid. Boadilla del Monte. Spain
{fsegovia, emenasalvas}@fi.upm.es

Abstract. Despite the great success of data mining being applied for personalization in web environments, it has not yet been massively applied in the e-learning domains. In this paper, we outline a web usage mining project which has been initiated in University of Cantabria. The aim of this project is to develop tools which let us improve its Web-based learning environment in two main aspects: the first that the teacher obtains information which allows him to evaluate the learning process and the second that the student feels supported in this task.

1 Introduction and Background

It can be said that, the World Wide Web is today the most important media for collecting, sharing and distributing information. Higher Education (HE) is one of the fields where web-based technology has been quickly and successfully adopted. The great proposal of online courses that, nowadays, is offered by universities is one proof of that. Even more, completely virtual universities are appearing.

Managing and tracking students, designing courses, making evaluations, etc. requires specific systems which are called Learning Management Systems (LMS). These systems can be organized in 3 subsets according to Jackson [5]: Course Management Systems (CMS), Enterprise Learning Management (ELMS) and Learning Content Management Systems (LCMS).

CMS facilitate web delivery and management for instructor-led topics and include conferencing systems, polling and quiz modules, virtual workspaces and other tools for measuring outcomes and reporting progress for individual or groups of students. They tend to be very textual and template oriented to provide ease of use, but limiting flexibility. These systems are the most popular in HE institutions (85% according to Gartner). Examples include Blackboard, Virtual-U or WebCT. ELMS and LCMS are more expensive and require significant customization. They typically add strong integrated authoring tools and components to connect to database systems.

Many CMS have been developed and are in use around the world. However they do not support tools which allow educators to thoroughly track and assess all the

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 205 – 210, 2005.
© Springer-Verlag Berlin Heidelberg 2005

activities performed by all learners, nor to evaluate the structure of the course content and its effectiveness on the learning process. In fact, these environments provide the educator with access summary information such as more visited pages, favourite communication method, and other statistics. Nevertheless this information is not enough to analyze the behaviour of each student and his evolution.

The problem is that E-learning environments lack a closer student-educator relationship. The lack of this relation is manifested in facts such as: teacher does not really control the evolution of his students, and students cannot express their problems and deficiencies in a natural way.

This problem has yet been tackled in marketing environments using web mining techniques. In [1], an architecture that successfully integrated data mining with an e-commerce system is shown. In [6] a methodology for evaluating and improving the "success" of a commercial web site based on the exploitation of navigation pattern discovery is proposed. In [3] an overview of approaches for incorporating semantic knowledge into Web Usage Mining and personalization processes is provided.

Results of the application of Web Mining in e-commerce have not been massively applied in e-learning environments while web-based learning systems can profit from them [4]. In this direction, our project tries to solve the presented problems by pre-processing and analysing the web log files, which provide a raw trace of the learners' navigation and activities on the site, using OLAP and data mining techniques [8] in order to extract valuable patterns that will be used to enhance the learning system and help in the learning evaluation.

Thus in this paper we present advances of an e-learning project in which OLAP techniques are applied to obtain data that later will be used to improve the relationship between professor and student.

The rest of this paper is organized as follows: Section 2 briefly presents the objectives and main tasks of the E-learning project which is being developed in University of Cantabria (UC) with the collaboration of Universidad Politécnica de Madrid (UPM). Section 3 presents obtained results in the OLAP analysis. Finally, Section 4 provides conclusions and future work.

2 E-Learning Project

This project initially springs up with the aim to give concrete answers to professors who compromised with these new methods of learning based on new technologies do not get the appropriate feedback compared to the feedback you get from students with traditional teaching methods. Besides, we have in mind other goals which will help administrators and academic responsible to do better their task.

2.1 Objectives

On the one hand, professors will have information that provides them with tracking information to assess the learning process of their students. The system will also provide professor with the most common navigation patterns in their courses that will help them to evaluate their courses structure effectiveness.

On the other hand, learners will obtain a personalized environment that in near future, will recommend them activities and/or resources that would favour and improve their learning

An added value of these tools will be that the site administrator will have parameters to improve the site efficiency and adapt it to the behaviour of their users.

Academic responsible will have information which allows them to know their student profile. It will provide them with measures to better organize their resources, both human and material, and their educational offer.

2.2 Scheduled Tasks

We briefly describe the main five tasks we have considered:

1. Data pre-processing: clean and prepare the web server log file and load the clean data into a relational database.
2. OLAP analysis: design a multidimensional structure in which the main factors under analysis: sessions, courses, pages, time, demographical user data, user behaviour (content or navigational) will be taken as dimensions and later build OLAP cubes in order to analyze the recorded data.
3. Pattern discovery: application of data mining algorithms. Firstly descriptive algorithms (clustering and association) will be applied to obtain typologies of users. Then classification techniques will be used in a later step to classify behaviours according to historical patterns.
4. Pattern evaluation: All the patterns obtained will be valuated to distinguish the patterns that really help to better achieve the site goals.
5. Recommendation engine: integrate the discovered patterns in the online component to provide personalized environment to learners.

In the next section results obtained with the OLAP components are summarized.

3 OLAP Analysis

Although the project main aim is not only to generate OLAP reports but also data mining analysis, in this paper we focus on the OLAP analysis. Consequently results obtained so far are presented in what follows.

3.1 Pre-processing

Web server logs are the primary source of data in which the activities of Web users are captured, although often can be enriched with external information obtained from corporative database systems. These files, due to their huge size and their lack of structure, require to be processed, this means, to be read and recorded in a relational database to be easily managed.

Pre-processing [2] is, in fact, the first part of Web Usage Mining (WUM) which includes the domain dependent tasks of data cleaning, user identification, page identification, session identification and path completion. Although there are free tools such us [7] which allow cleaning and making sessions, they have not been suitable for us because they work with general logs (commercial environments) and they do not allow us to configure specific e-learning characteristics, for this reason we have developed our own tool.

The example we use for illustration in this paper comes from a WebCT web log with records of students' on-line activities in all courses from October 1st, 2004 to December 31st, 2004. There are near 10.000.000 entries in this web log file of a size of 1.1 Gigabytes. After cleaning process, the number of entries was reduced to 2.206.024. Next, 3.235 learners and 322 courses were identified and finally, 48.691 sessions were built. In our case, a new session was considered when a change in a user-course happened or when the time interval between two successive inter-transaction clicks upped 30 minutes. In this last step, the duration of the visit and the number of visited pages were calculated. Besides, the visited pages during each session were registered to be used in the incoming navigational behaviour analysis.

3.2 OLAP Analysis

The multi-dimensional structure of the data cube provides remarkable flexibility to manipulate the data and view it from different perspectives. Building a web log data cube allows the application of OLAP (On-Line Analytical Processing) operations to view and analyze the web log data from different angles, derive ratios (average stay-session time, etc) and compute measures across many dimensions.

The first step towards analysing user behaviour in an e-learning system is to be able to answer questions as the ones that follow:

- How long is our learner connected (by course, degree, month, etc)?
- What is the connected learner distribution over time (hour of the day, day of the week, month and year)?
- How many learning sessions do our learners establish over time?
- Which courses are the most frequent acceded?
- Which is the percentage of connections done inside university campus?
- What is the distribution of network traffic over time?

Consequently after the data was cleaned and transformed, a multi-dimensional structure was designed. From this, a MOLAP data cube was built to aggregate the number of visited pages, the number of error pages and the visit duration according to the following five dimensions: date, time, courses, sessions and learners.

Not only the questions above but any question involving the dimensions of the OLAP structure will be answered. Results of some of these questions are shown.

Example 1. Usage pattern analysis

The session analysis lets us understand how the system was used, how the course structure was designed and how the learners' behaviour evolved over time.

Fig. 1a and Fig. 1b show sessions per learner, average session time and pages per session measures. The first one shows the results taking into account all courses in this period (in weeks) and the second one, the same values but for one of the most frequent acceded and well-designed course.

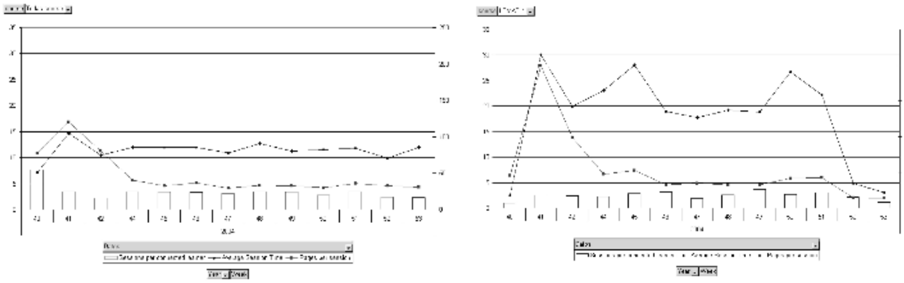


Fig. 1. (a) Usage pattern analysis of the full system, and (b) for a well-designed course

One phenomenon we have discovered is that at the beginning of a course the students tended to explore many different system features (more visited pages per session). However, they became more and more focused over time.

In Fig. 1a, it can be observed how the average connection time by student in this term is low, less than 15 minutes, nevertheless in Figure 1b, this time ups to 20 or 30 minutes. This suggests us that most of the courses have been designed as repositories of contents, i.e., professors have designed html pages with several links to PDF or zip files, so that, students only need connect to the system to download these files. On the other hand, this reduced time indicates us that students do not use very often collaborative tools (chats, mail …) because the interaction requires longer sessions.

Example 2. Learner distribution over time

Another interesting information consists on analysing the distribution over time of the number of connected students depending on the week day and time of the day. It is easy to see in Fig. 2 how, in this study, students were highly connected from outside UC (value IN) in working days (value 1) and from 12 to 20 hours.

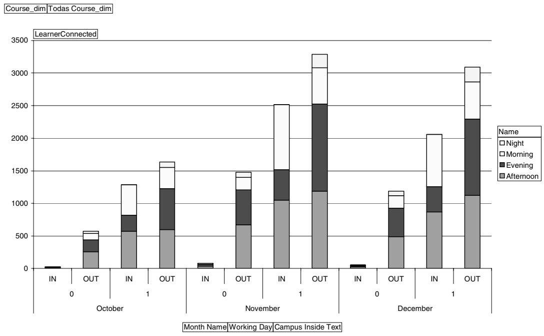


Fig. 2. Learner distribution over time

In relation to these results we can also say that analysed courses are chiefly exploited as a complementary tool as the main activity begins around November. This shows that students need have minimum knowledge about the subject before accessing to the WebCT platform for further knowledge.

Besides, we have observed that the number of connected students versus registered students is around 50% in the case of not completely virtual courses. This can be interpreted as students not connecting to the system unless the activity is completed integrated in the course syllabus.

4 Conclusions and Future Work

In this paper we have presented an e-Learning WUM project which is being developed in UC using its own data. This project is divided in five stages, where two of them have been yet done. Likewise, the first relevant results have been shown.

Our experience shows us that pre-processing step although being time consuming is crucial for the success of the discovery process. A good data cleaning and filtered process needs metadata provided by web site designers and a good knowledge about how the LMS works because generally are environments based on scripts.

Also we can say that, the multi-dimensional structure of the data cube provides remarkable flexibility to manipulate the data and view it from different perspectives. Besides, as it can be managed with MS Excel, teachers and system administrators can evaluate easily the system use.

Our next step will be to select a course whose design allow us to analyse learner navigational behaviour and compare it with professor intention. For that rule association and sequential algorithms will be used. Also its effectiveness in the learning process will be evaluated. On the other hand, we will try to complete learner information and, applying descriptive algorithms, obtain typologies of students. Further development and experiments will be reported in the future.

References

1. Ansari, S., Kohavi, R., Mason, L., Zheng, Z. Integrating E-Commerce and Data Mining: Architecture and Challenges. Workshop on Web Mining for E-Commerce-Challenges and Opportunities Working Notes (KDD 2000), Boston, MA.
2. Cooley, R., Mobasher, B., Srivastava, J. Data Preparation for Mining World Wide Web Browsing Patterns. Jounal of Knowledge an Information Systems, 1(1). 1999.
3. Dai, H., Mobasher, B. A road map to more effective web personalization: Integrating domain knowledge with web usage mining. In Proc. of the International Conference on Internet Computing 2003, Las Vegas, Nevada, June 2003.
4. Hanna, M. Data Mining in the e-learning domain. Campus Wide Information Systems. 16 Jaunary 2004. vol. 21 no. 1 pp. 29-34(6). Research article ISSN: 1065-0741
5. Jackson, R. An Overview of Web-Based Learning. 2004. http://www.knowledgeability.biz/weblearning/
6. Spiliopoulou, M., Pohle, C. Data mining for measuring and improving the success of web sites. Journal of Data Mining and Knowledge Discovery, Special Issue on E-commerce, volume 5(1-2), pages 85-114. Kluwer Academic Publishers, Jan.-Apr. 2001.
7. Web Utilization Miner WUM 7.0 Beta.
8. Zaïane, O. Web Usage Mining for a Better Web-Based Learning Environment. Proc. of Conference on Advantage Technology for Education. Alberta, Canada. 2001.

Similarity Queries in Data Bases Using Metric Distances - from Modeling Semantics to Its Maintenance

Josef Küng and Roland Wagner

Johannes Kepler University of Linz,
Institute for Applied Knowledge Processing (FAW),
Altenbergerstraße 69, 4040 Linz, Austria
`{jkueng, rrwagner}@faw.uni-linz.ac.at`
`http://www.faw.uni-linz.ac.at`

Abstract. Similarity queries in traditional databases work directly on attribute values. But, often similar attribute values do not indicate similar meanings. Semantic background information is needed to enhance similarity query performance. In this paper a method will be addressed which follows the idea to map attribute values to multidimensional points and then interpret the distances between that points as similarity. The second part brings the questions "How to arrange these points that they correspond to real world?" and "Can that be done automatically?" into focus and comes to the following result: For the case that all similarities are known in advance a good solution is given otherwise it turns to a complex optimization problem.

1 Introduction

Similarity search has emerged and become a fundamental requirement, even for data base applications and data base management systems. Unfortunately similarity search based on the attribute values itself does not fit all user needs. For example in the case of a tourism information system the attribute values "San Augustin" and "Playa del Ingles" should be identified as similar because both are villages in the southern part of Gran Canaria, even if their attribute values are anything but similar. Additional concepts to model and maintain semantic background information are needed to support such enhanced similarity queries.

1.1 Flexible Query Answering Systems

Query processing in conventional data base systems is not flexible enough to support advanced similarity search capabilities directly. That means when the available data does not match a users query precisely, the system will only return an empty result set to the user. This limits its applicability to domains where only crisp answers are meaningful. In many other application domains, however, the users expect not only crisp results returned but also some other results close to the query in a sense. Systems that can solve this problem are called flexible query answering systems (FQAS).

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 211 – 216, 2005.
© Springer-Verlag Berlin Heidelberg 2005

1.2 The Need of Semantic Background Information

As mentioned above attribute values do not hold enough information to find similar data objects from a user point of view. The semantic of the objects is needed. Then we can determine semantically near objects which will produce much more adequate results for similarity queries.

2 Modeling Semantics Using Metric Spaces

Nowadays most common approaches for modeling semantic background are techniques which can be categorized by semantic networks or ontologies such as the Resource Description Framework (RDF) or Topic Maps.

This paper will concentrate in a different approach coming from pattern recognition or information retrieval. There feature vectors and corresponding feature spaces are used. The distance between two points in these metric feature spaces is equivalent to the similarity between the corresponding objects.

2.1 Numeric Coordinate Representation

Thinking about similarity queries in databases we see two approaches: Modeling similarity on object level or attribute level. Concerning typical database queries where the query conditions address attribute values our institute focused on attribute level.

The basic idea was that values of non-numerical attributes are mapped to points in a multidimensional feature space. Each attribute domain can have its own feature space. Thus similarity between two attribute values – and based on that even similarity between whole database objects - can be derived from the metric distances in these spaces. The database tables which hold this mapping of attribute values to multidimensional points are called NCR-Tables (Numeric Coordinate Representation Tables). They can be seen as a semantic background information pool that supports an application domain appropriate similarity search.

2.2 A particular Vague Query System

At our institute a prototype called Vague Query System (VQS) was implemented which follows that approach [1] [2]. It works quite well and supports also a standard metric for attributes which have no NCR-Table in the background, multiple similarity query conditions and, vague joins.

3 Maintaining Numeric Coordinate Representations

In the classical case this is the work of a knowledge engineer or a domain expert. He designs the similarity measure between the possible values and enters this information into an NCR-Table. Sometimes existing data can be used. For instance the similarity of location names can be defined as the geographical distance and an eventually existing database table containing geographical coordinates of the locations can be used.

Anyway, autonomous similarity learning by the system itself would be the optimal case. In this chapter we concentrate on the challenges and possible solutions for realizing such an intelligent system.

3.1 Euclidean Distance Matrix

Regardless of the manner how it has been built similarity measure can be placed in a natural way in a matrix. However, this similarity matrix must have special properties in order to be possible to build a sound representation. A matrix which has these properties is called *Euclidean Distance Matrix*:

Definition 1. A matrix $D \in \Re^{n \times n}$ is called Euclidean Distance Matrix (EDM) if and only if the following four properties are satisfied:

1. $\forall i\, j: d_{ij} \geq 0$ *(non-negative)*
2. $\forall i: d_{ii} = 0$ *(zero diagonal)*
3. $\forall i\, j: d_{ij} = d_{ji}$ *(symmetry)*
4. $\forall i\, j\, k: d_{ij} + d_{ik} \geq d_{ik}$ *(triangular inequality)*

3.2 Reinforcement Learning

Two main paradigms of machine-learning are known: learning with a teacher, which is called *supervised learning*, and *learning without a teacher*. The paradigm of learning without a teacher is subdivided into *self-organized (unsupervised)* and *reinforcement learning*. Supervised learning is a "cognitive" learning method performed under tutelage of a teacher: this requires the availability of an adequate set of input-output examples. In contrary, reinforcement learning is a "behavioral" learning method, which is performed through interactions between the learning system and its environment. The operation of a reinforcement learning system can be characterized as follows [3]:

1. The environment stochastically occupies a finite set of discrete states.
2. For each state there is a finite set of possible actions that may be taken.
3. Every time the learning system takes an action, a certain reward is incurred.
4. States ($s_t \in S$) are observed, actions ($a_t \in A$) are taken, and rewards ($r_t \in \Re$) are incurred at discrete time steps ($t \in N$).

The goal of the learning system is to maximize its cumulative reward. This does not mean maximizing immediate gains, but the profit in the long run. In our automated FQAS the states of the reinforcement learning are the searched records (the queries of the users), the actions are the selection of similar records which will be offered to the user and the rewards are computed from the reaction of the user or from his marks.

One should be cautious when learning a similarity matrix, because when the matrix or its approximation is updated then one should check that the new matrix is still an EDM. Testing the first three properties after a cell has changed is straight forward, however verifying the triangular inequality could need a lot of computation. We have another choice namely we do not learn the similarity matrix but we learn directly the NCR-Table instead. In this way we do not need to check the EDM properties. However this approach has a major drawback, viz. in that case changing the dimensions of the representation is hard, or even realizing that we need a higher dimensional NCR-Table during the learning is non-trivial. Moreover, we need to have an advance estimation of the NCR-Table dimension, but if we learn the distance matrix then we do not need such an estimation. Both ways have their own advantages/disadvantages, we will continue with the case when we have learnt the distance matrix and we need to build an NCR-Table from it.

3.3 Building a Numerical Coordinate Representation

If we have an Euclidean Distance Matrix (EDM), for example we have learnt it from the interaction with the user or a knowledge engineer designed it, then we need to find a "good" representation of it. We have already mentioned that Numeric Coordinate Representation Tables provide an efficient way of representing similarity information. The problem of finding a representation is to build such an NCR-Table from, a possibly sparse, EDM which means that we have to find a mapping Φ that projects our points to a sufficiently high dimensional Euclidean space in a way that the distances of the points in that space equals to the desired distances in the given EDM. Naturally, our aim is to minimize the number of needed dimensions, as well.

The problem given above was stated as the fundamental problem of distance geometry and formulated by Blumenthal [4] in 1953. Distance geometry has received much attention in recent years because of its many applications in fields such as chemistry, statistics, archaeology, genetics and geography. A detailed discussion on distance geometry can be found in [5]. We will discuss the representation problem in two steps. First, an easier case will be presented, when all the distances are exactly given. Then, we present a harder problem, when only a few distances are given (so the distance matrix is sparse).

Complete Distance Information. If all the distances are known then we can find the minimal dimension that we need with the help of the following theorem [6]:

Theorem 1. If $x_1, x_2, \ldots, x_n \in \Re^r$ and $\forall i\, j \in \{1, \ldots, n\} : d_{ij} = \|x_i - x_j\|_2$ then the rank of a corresponding distance matrix is at most $r + 2$.

With this theorem we can compute exactly the minimal dimension of the target Euclidean space. We can use the eigenvectors of a specially designed matrix to find the coordinates of the points immediately [4]. Regarding the complexity, this problem is tractable, because both computing the rank of a matrix and finding the eigenvectors can be done in polynomial time, more precisely in $O(n^3)$ if the matrix is $n \times n$. However there is another approach which is much faster than building the representation by eigenvectors. This algorithm is called fastmap and it can place n points in a k dimensional space in O(nk) time [10].

Incomplete Distance Information. When we have given only a sparse set of distances then the distance geometry problem becomes very hard to solve. Easy to see that this problem is equivalent to the graph embedding problem, which we present here:

Definition 2. Let $G = (V, E, \omega)$ be an undirected edge-weighted graph with vertex set $V = (v_1, v_2, \ldots, v_n)$, edge set $E \subseteq V \times V$ and a non-negative ω_{ij} for each $(v_i, v_j) \in E$. G is said to be r-embeddable if there exists a mapping $\Phi : V \rightarrow \Re^r$ such that for every edge $(v_i, v_j) \in E$, the Euclidean distance $\|(v_i) - (v_j)\|_2 = \omega_{ij}$.

As we saw, the distance geometry problem is tractable if all the distances are exactly given. Unfortunately, the following theorem shows that if we have only a sparse set of distances then the problem becomes intractable:

Theorem 2. $\forall r \in N$: the r-embeddability problem of an integer-weighted graph $G = (V, E, \omega)$ is NP-Hard.

Proof. Saxe showed that the one dimensional distance geometry problem with incomplete distance information is equivalent to a set-partition problem which is known to be NP-hard. He extended his proof to higher dimensions, as well. The complete proof can be found in [7].

Concerning this major drawback we just see an application of some optimization techniques as a possible solution. Indeed, an estimation of the number of dimension needed in the corresponding NCR-Table is possible. Then the distance geometry problem can be transformed to a least square optimization problem for which one of the available multidimensional minimization methods can be used. Later on a reduction of the number of dimensions can be performed (e.g. by applying a Principal Component Analysis or an Independent Component Analysis). In [8] some of such solution variants are addressed.

4 Conclusion

This paper addresses the application of metric spaces and metric distances to support similarity queries. After an introduction of a still existing prototype system we investigated questions like: How can we obtain a similarity measure automatically? What is the minimal number of dimensions we need for a multidimensional semantic background representation and what methods can be used to let the mapping done autonomously by the system? We could show that a representation can be built in polynomial time when all distances between the objects are known. However, this is mostly not the case in practice. Normally, it is not possible to gain all the distances. For that case only an upper bound of the minimal number of dimensions can be computed and the mapping can be handled as an optimization problem. Unfortunately, the problem with incomplete distance information is NP-hard. Only heuristic application domain specific methods can decrease that complexity.

References

1. J. Küng, J. Palkoska: VQS-A Vague Query System Prototype. DEXA97, IEEE Computer Society Press (1997)
2. J. Küng, J. Palkoska: Vague Joins-An Extension of the Vaque Query System VQS. DEXA98, IEEE Computer Society Press (1998)

3. S. Haykin: Neural Networks. A Comprehensive Foundation. 2nd Edition, Prentice Hall (1999)
4. L. M. Blumenthal: Theory and Application of Distance Geometry. Chelsea, Bronx, New York (1970)
5. J. Yoon, Y. Gad, Z. Wu: Mathematical Modeling of Protein Structure Using Distance Geometry. Technical Report TR00-24 for the Computational and Applied Mathematics Department of Rice University (2000)
6. C. Borcea, I. Streinu: On the Number of Embeddings in Minimally Rigid Graphs. SoCG'02, Barcelona, Spain (2002)
7. J. B. Saxe: Embeddability ofWeighted Graph in k-space is Strongly NP-hard. Proc. 17th Allerton Conf. in Communications, Control and Computing (1979) 480-489
8. B. C. Csáji, J. Küng, J. Palkoska, R. Wagner: On the Automation of Similarity Information Maintenance in Flexible Query Answering Systems. DEXA2004, Springer, Berlin Heidelberg (2004)

A WEB-CASE Tool Prototype for Hybrid Software Development

F.J. Orellana, F. Guil, I.M. del Aguila, and S. Túnez

Departamento de Lenguajes y Computación,
Universidad de Almería, 04120 Almería
{fjorella, fguil, imaguila, stunez}@ual.es

Abstract. In this paper we present a Web-based CASE tool for hybrid software projects that supports first stages of InSCo, an extended methodology based on CommonKADS and RUP. The tool InSCo Requisite will guide the development of a sort of software where knowledge-based components are integrated with traditional information systems. Furthermore, this tool will allow to manage several development projects at the same time, and determine the users which will take part in each development team.

1 Introduction

When we are developing software, based or not based on knowledge, developers choose a methodology depending on the problem, the prior knowledge, the available tools or other factors. These methodologies lead us to the production process determining the documents and artifacts we must generate, the activities to do and the order that we must follow.

The main artifact generated during the software development process, is a group of models [1]. The models of first stages in the process are often specified using natural language. However, this technique has problems like ambiguity, uncertainty or imprecision. Several alternatives have proposed a solution by means of the use of a structured natural language, description language, description languages of design, graphic notation or formal specifications, that are similar to the successfully used in the final stages of software development [5].

We propose to use these techniques in the organization modelling and requirements level scope, combining the forms and diagrams to model complex systems. In most cases, the models and documentation that we generate and maintain is very complex. For this reason, we need software tools that guide us in the generation of these artifacts, its maintenance, and the control of the integrity of changes. Nowadays, we can find tools which help us in the development of traditional software. However, our main objective is the extension of this kind of tools.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 217–222, 2005.
© Springer-Verlag Berlin Heidelberg 2005

2 InSCo Methodology

In many application domains, there are problems that we need to solve by building a software system that uses knowledge engineering methods. These systems are known as hybrid systems. [2]

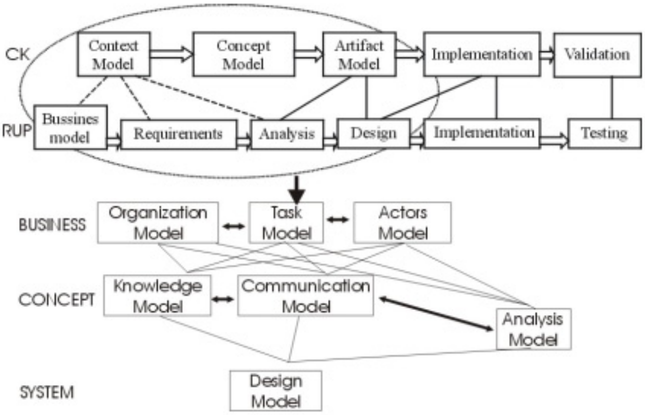


Fig. 1. Models of InSCo Methodology

Hybrid systems are needed in domains and organizations that must solve, on one hand, less structured problems with subjective requirements, uncertainty and imprecision. To deal with these problems, the incorporation of a Knowledge Based System is essential. On the other hand, it is necessary to add not-based on knowledge functionality (a traditional information system [2]) in order to reach the success of the organization and the KBS. For this reason, the methodologies that expect to support the development of this kind of software must create an unique solution, integrating both classes of software systems [4] [5].

The InSCo methodology [3] was designed from a requirements engineering perspective to deal with the hybrid software development . As we can see in Figure 1, InSCo proposes a total of seven main activities for developing hybrid software: Organization, Requirements , Knowledge, Analysis, Design, Implementation and Maintenance or Evolution. It is necessary to emphasize that the prototype we have developed covers the two first stages: Organization and Requirements.

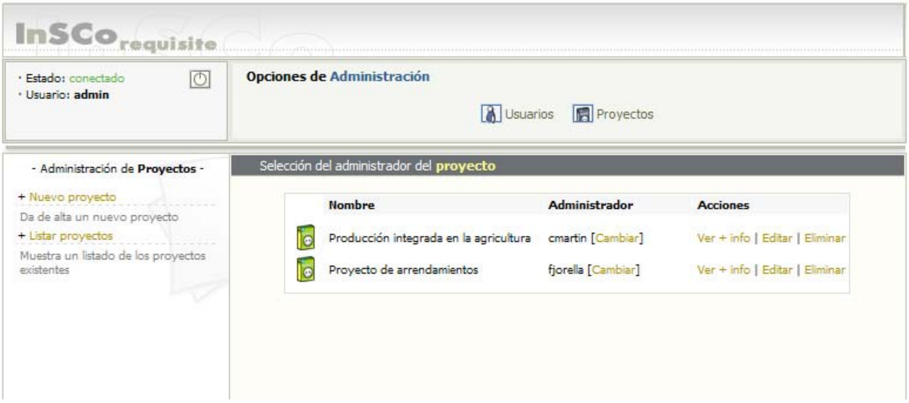
The results of these two activities constitute the business and services models. The business model covers the organizational environment, the roles that take part in the project, the objectives of the organization, the information resources and finally, a list of items that allow us to make the decisions about the feasibility of the project. On the other hand, the services model represents a more detailed specification of the information that we obtain from the business model. This specification is related to functional, non-functional and information requirements of the project. The InSCo methodology proposes the use of templates and graphic notation in order to represent all these models.

3 The InSCo Requisite Tool

A competitive methodology must have a set of associated tools for making the development of software based on it easier. In this paper we present InSCo Requisite, a tool designed for guiding us in the whole development process of the hybrid software. This tool helps the users to carry out the different tasks involved in the development, the forms construction or the administration of the diagrams associated.

One of the strong points of the application is the possibility of accessing the application via the Internet. Most of CASE tools designed for developing traditional or hybrid software need the installation of client programs to work with the application. InSCo Requisite is Web-based, so the only requirements we need will be an Internet connection and a browser. Nowadays, the use of Internet is very extended so, it is easy to find a fast and cheap Internet Connection. On the other hand, getting a browser is very easy because browsers are incorporated by default in most operative systems. Furthermore, this feature will allow the cooperation of users situated in different geographic areas, this way, the distance among the members of a project will not be a problem in order to carry out the development.

It is necessary to emphasize that InSCo Requisite can manage several hybrid software projects with independent development teams working at the same time. All the information related to these projects and the users that take part in them must be managed. For this reason, InSCo Requisite provides an administration area (Figure 2) which is integrated in the application, and that will allow the system administrator to control all these aspects of the application.


InSCo requisite
Opciones de Administración
Usuarios
Proyectos
- Administración de Proyectos -
+ Nuevo proyecto
Da de alta un nuevo proyecto
+ Listar proyectos
Muestra un listado de los proyectos existentes
Selección del administrador del proyecto

· Estado: conectado
· Usuario: admin

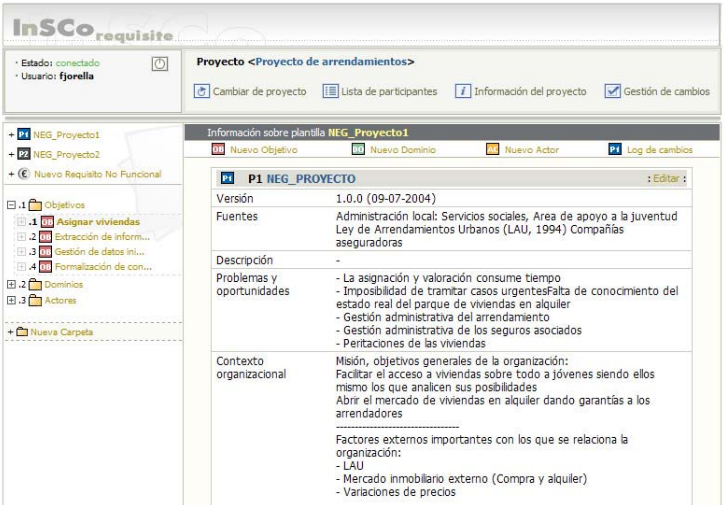
Nombre	Administrador	Acciones
Producción integrada en la agricultura	cmartin [Cambiar]	Ver + info | Editar | Eliminar
Proyecto de arrendamientos	fjorella [Cambiar]	Ver + info | Editar | Eliminar

Fig. 2. Administration Area

The main options we can find in the administration area are:

- Management of the projects. The administrator can insert new hybrid software projects, modify the data of the existing projects or delete them.
- Management of the users of the system. The main tasks that can be performed by the administrator are: registration of new users,the establishment of their access data (username and password) and update the information (personal details, access data)
- Appointment of responsible persons for each project. Each project must be managed individually by a person who will configure the main aspects of the project.

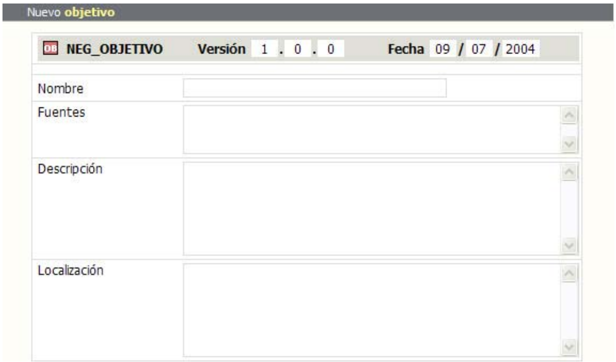
Getting more details about the management of the requirements in a hybrid software project, as we commented before, each project is formed by a group of users which will work with the whole group of templates that belong to the first stages of InSCo methodology. Their main task is the management of the documents and artifacts associated with the project they belong to. Furthermore, each project has a responsible person or administrator defined by the system administrator. This person can access to a group of options related to the project, which allow him to decide what users are going to participate in the project and configure some other aspects.



Versión	1.0.0 (09-07-2004)
Fuentes	Administración local: Servicios sociales, Area de apoyo a la juventud Ley de Arrendamientos Urbanos (LAU, 1994) Compañías aseguradoras
Descripción	-
Problemas y oportunidades	- La asignación y valoración consume tiempo - Imposibilidad de tramitar casos urgentesFalta de conocimiento del estado real del parque de viviendas en alquiler - Gestión administrativa del arrendamiento - Gestión administrativa de los seguros asociados - Peritaciones de las viviendas
Contexto organizacional	Misión, objetivos generales de la organización: Facilitar el acceso a viviendas sobre todo a jóvenes siendo ellos mismo los que analicen sus posibilidades Abrir el mercado de viviendas en alquiler dando garantías a los arrendadores -------------------------------- Factores externos importantes con los que se relaciona la organización: - LAU - Mercado inmobiliario externo (Compra y alquiler) - Variaciones de precios

Fig. 3. Screenshot of InSCo Requisite

Each member can participate in a project interacting with the different templates proposed by the InSCo methodology. The possible operations can be carried using web-based forms (Figure 4) and hyperlinks. This way, each user can access to the whole group of templates, updating their information or establishing connections with other templates by using an intuitive and easy web interface.



Nuevo **objetivo**

OB **NEG_OBJETIVO** **Versión** 1 . 0 . 0 **Fecha** 09 / 07 / 2004

Nombre	
Fuentes	
Descripción	
Localización	

Fig. 4. Forms

One of the most interesting features of the application is a tree-based menu (Figure 5), which represents the hierarchy of the templates that belong to a specified project. This tree allows the users to have a general view of the project and provides a easier method to classify and access to each element of the project. Due to the huge quantity of templates and information that take part in a project, one of the main objectives of InSCo Requisite has been to make the work lighter. For this reason, the application features this tree-based menu, icons associated to each template that allow us to identify them or texts that help us to perform the different processes.

We have proposed several improvements in order to increase the functionality of the application. As we commented, it is possible to represent the models of first stages using graphic notation like diagrams. Up to date, we can include this kind of notation in our project attaching a graphic file to a particular template. We propose the incorporation of a graphic editor integrated with the web application that allow us to create the diagrams without using an external application. Another improvement we have considered is the incorporation of a glossary of terms related to a specified project. The documentation of each hybrid software

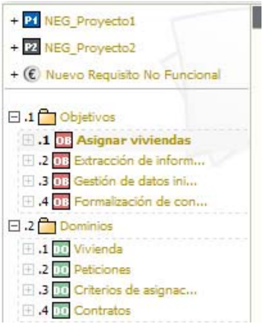

+ P1 NEG_Proyecto1
+ P2 NEG_Proyecto2
+ € Nuevo Requisito No Funcional
⊟ .1 Objetivos
⊞ .1 OB Asignar viviendas
⊞ .2 OB Extracción de inform...
⊞ .3 OB Gestión de datos ini...
⊞ .4 OB Formalización de con...
⊟ .2 Dominios
⊞ .1 DO Vivienda
⊞ .2 DO Peticiones
⊞ .3 DO Criterios de asignac...
⊞ .4 DO Contratos

Fig. 5. Tree-based Menu

project contains a lot of words and technical terms which are related to the project domain. Sometimes, the people who participate in the project are not familiarized with that domain terms. It would be a good idea that the members of a project could access to a glossary to look up the meaning of those words.

References

1. Object management group. In *MDA Guide Version 1.0.1.* OMG Document: omg/2003-06-01, 2003.
2. J. J. Cañadas, A. Bosch I. M. del Águila, and S. Túnez. An intelligent system for therapy control in a distributed organization. *In Proceedings of Information and Communication Technology (EurAsia-ICT 2002), Springer-Verlag. Lecture Notes in Computer Science*, 2510:19–26, 2002.
3. I. M. del Águila, J. J. Cañadas, J. T. Palma, and S. Túnez. Integration of development methodologies for the building of knowledge intensive multiagent systems. In *In Proc. of the First Int. Conf. on Integration of Knowledge Intensive Multi-Agents Systems, KIMAS'03*, pages 203–208. IEEE Press, 2003.
4. I. Jacobson, G. Booch, and J. Rumbaugh. *El Proceso Unificado de Desarrollo del Software.* Addison Wesley, 2000.
5. G. Schreiber, H. Akkermans, A. Anjewierden, R. deHoog, N. Shadbolt, W. VandeVelde, and B. Wielinga. *Knowledge Engineering and Management: The CommonKADS Methodology.* The MIT Press, 1999.

An Augmentative Communication System Based on Adaptive Evolutionary Hypermedia Systems

M.V. Hurtado[1], N. Medina[1], L. García-Cabrera[2], and M.L. Rodríguez[1]

[1] Dept. Lenguajes y Sistemas Informáticos. University of Granada (Spain)
{mhurtado, nmedina, mlra}@ugr.es
[2] Dept. Informática. University of Jaén (Spain)
lina@ujaen.es

Abstract. This paper focuses on improving ACS (Augmentive Communication Systems) by means of an adaptive evolutionary hypermedia. One of the most important features of our approach is the separation of the different aspects involved in the development, use and maintenance of the communication system. Concerning the knowledge representation aspect we use an ontology model that permits specifying the semantic of the represented reality. Regarding the presentation aspect, in order to generate the hypermedia structures we provide a mechanism that allows the creation of different views of the global knowledge model. About the navigation aspect it is important to emphasise its multimodal facet: at the technology level (PC or PDA) and at the interaction level (depending on the access restrictions and the selection units). The user adaptation aspect permits to analyse and to personalise the user navigation using his user profile, his user model and a set of appropriate adaptive methods. In addition, during the whole process we apply an evolutionary mechanism to evolve these aspects in an integral form.

1 Introduction

Augmentive Communication Systems [1] (sign languages, pictorial languages, sign templates and communicators) are part of a technology developed to improve the social integration of people with temporal o permanent communicative difficulties providing useful tools for their rehabilitation.

Each person requires a specific attention and adaptation to fit his profile, which could evolve in time. However these systems don't support mechanisms for their necessary adaptation to each user and situation taking in account his capabilities, skills and progresses at run-time. Therefore, an efficient and suitable approach to design these systems is needed, based on software evolution and adaptability [2,3].

In particular this paper proposes an adaptive evolutionary hypermedia (from a knowledge and an interaction models) that allows to represent control and to adapt the context communication.

2 Features of Adaptive Evolutionary Communication Systems

The elements that this approach manages in order to design adaptive evolutionary communication systems are:

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 223 – 228, 2005.
© Springer-Verlag Berlin Heidelberg 2005

- **Ontology model**, which represents each conceptual world and describes a specific knowledge domain by means of concepts and relationships between them (a semantic net) [4]. The ontology, in this case, provides the link between the symbols used in the communicator (image, sound or text adapted to each person's knowledge) and the entities of the real world they represent. In addition, ontologies both allow a global vision of each conceptual world based on the concepts employed by the users, and allow to standardize into a single model the knowledge about the people with the communicative difficulties, the knowledge about the people around them (i.e. relatives, tutors, rehabilitators, etc.), and the media used for communication (i.e. templates, agendas, etc.).
- **User model**, which provides knowledge about the people with communication disabilities and consists on the following components:
    - **Knowledge domain** represented by means of ontology. In addition, the educators can define different semantic views of the whole knowledge domain taking into account the user profile. For example, figure 1 shows a partial view of ontology that describes a scenario of shopping.

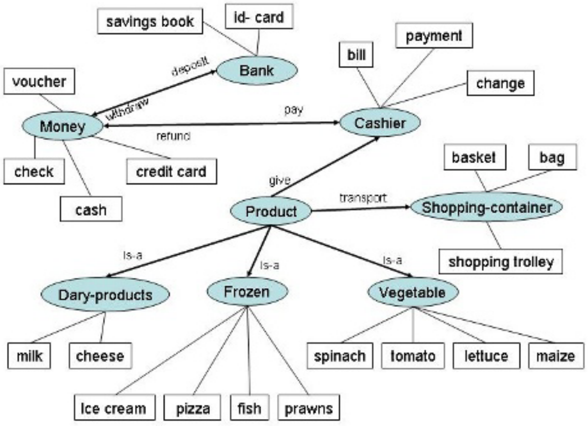


Fig. 1. Shopping scenario

The process followed by the tutor to create a new view consists in hiding in the semantic net the concepts, items and conceptual relations he considers not related to the current view. The system automatically keeps the integrity of the view, for example, if after hiding a relation some concepts get disconnected, they are also automatically hidden. In the same way, when a concept is hidden all the items associated to it and all the relations starting or arriving to the concept are also hidden.

Once the semantic net is built, the system will be able to automatically generate the interaction templates from it. These templates constitute one of the possible media used to establish the communication with the user and are represented in hypermedia format.

  - **User profile** containing the particular characteristics of users (communication habits, training or therapy objectives, and interaction preferences). To construct and select it the user (and relatives or educators) must provide this information. This artefact allows the adaptation and selection of the semantic view that best fits to a particular user.
  - **User interaction** determines how the person must interact with the communicator in order to communicate. We use a format approach to represent the person's interaction. It is cased on direct manipulation style in order to highlight relevant system features and properties [5].
- **Evolution and Adaptation methods.** The evolutionary mechanisms allow the hypermedia model to incorporate the needed changes in an easy, flexible and consistent way. The process of adaptation to the user can be seen as a particular case of evolution where the system changes its behaviour depending on the user utilizing it.

3 Architecture of Adaptive Evolutionary Communication System

The architecture of our adaptive model has two tiers: the system (communicator) and the meta-system [6]. This division allows us to separate on the one hand the interaction, communication and user adaptation and on the other hand its evolution process.

The communicator is used by the user and the meta-system by the educators. The cognitive, interaction, design and adaptation aspects have to be differentiated to avoid the coupling. Thereby, evolution/adaptability can be done more easily and safely because the architecture components are independent.

The system is itself divided in four parts, which allows us to tackle separately the aspects of knowledge representation (memorization system), construction of partial views focused to one o several concrete knowledge subdomains (presentation system), navigation of the built views (navigation system), and personalization of the navigation process (learning system) [2].

3.1 Types of Navigations and Adaptations

The structure offered to navigate is different depending on the used platform. If a personal computer is used the interface has two frames: at the left the semantic net and at the right some information about the element selected in the net (Figure 2). It provides a semantic and contextual mode of navigation.

If it has not been predefined by the educator, the user can choose the navigation mode that he desires. Depending of the chosen mode (free navigation, conceptual navigation, navigation by conceptual relation or navigation by prerequisites), some options in the semantic net are disabled according to the user features and his interests. So, the adaptation is carried out depending on the user, but also on the navigation mode used in that moment.

In the **free navigation**, the conceptual structure is chosen according to the user profile and the user browses the structure without any restriction. While the user visits items, the system gathers information about the navigation strategy. By means of a transition matrix, the system captures the conceptual relationships followed by the person during navigation. The matrix has one row and one column for every concept represented in the knowledge domain. Based on the analysis of the matrix, the system can identify new conceptual relationships that are needed or existing conceptual relations that should be removed in the knowledge domain. Take into account these navigation patterns, the system can suggest changes in the structure and the educator can identify progress or problems in the mental knowledge of the patients (adaptation by feedback).

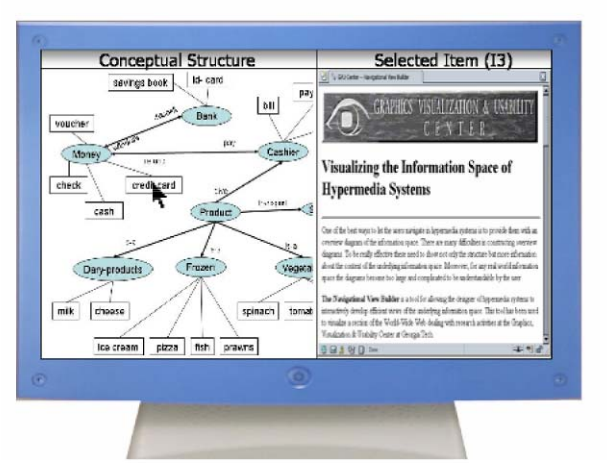


Fig. 2. Shopping with the PC

Conceptual navigation only shows the conceptual structure so, the navigation structure is smaller. When a user visits a concept, the system presents the whole domain information associated to it; a composition of the items order according to a compositional structure specified by the educator.

In the **navigation by conceptual relation** the user must follow the navigational links established among the concepts in the navigation system. Therefore, the user must visit the items in an order that is consistent with the semantic of the relation among them. This type of navigation is suitable for people we need to learn/remember routine tasks (autistic, Alzheimer's disease).

Navigation by prerequisites restricts the items the user can visit depending on some educational or the therapeutic prerequisites. To determinate the accessible items the system checks if the user reaches some achievements. This kind of navigation is the one that permits more adaptation possibilities. For instance, the educator or therapist wants the user works in a particular direction. In this case, the system can personalize the structure and show only those concepts that suit a set of interests or a specify knowledge goal. In addition, the system can provide the optimal route that best matches the preferences of the educator. With this adaptation technique (guides routes) the user loses freedom but in return it increasing the efficiency and quality of the navigational process.

In other case, when a PDA or other embedded system is used, the interface is based in templates. Every template only shows the available options in the current moment, without additional information. Each option is represented by means of an icon which can also have associated text and audio.

Figure 3 shows a possible use of the PDA in a scenario based on figure 1. As it can be seen, in some templates all the options must be selected in a particular sequence, while in others it is enough that the user selects one or several options in any order.

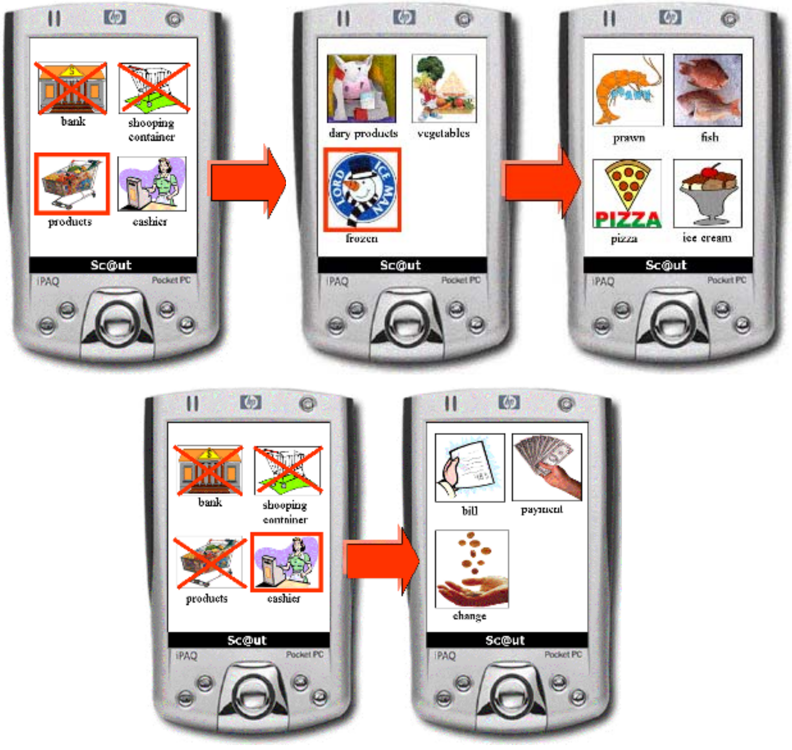


Fig. 3. Shopping with the PDA

3.2 The Evolution Process

The meta-system is in charge of the evolution of the system. It includes the evolution of the complete knowledge domain, of the partial views built from it, and of the navigation and adaptation rules used during the user interaction.

This level of abstraction includes tree evolutionary mechanisms: evolutionary actions, restrictions and change propagation. To perform a change, the educator chooses the appropriate action and run it. The action is only executed if it satisfies a set of restrictions imposed by the model and by the educator. Finally, this change could involve modifications in order to guarantee the consistency of same system and the other systems. For example, when a concept is removed in the knowledge domain, the meta-system removes this concept in all the views that show it. In addition, if after the change, some concept is disconnected in a semantic net, the meta-system performs new modifications in order to guarantee the consistency of the net.

4 Conclusions and Further Work

This work proposes a new way of conceiving Augmentive Communication Systems development based on adaptive evolutionary hypermedia systems. This approach reaches to design adaptive evolutionary communication systems: using an ontological model to define conceptual world, modeling the user's interaction by means of the knowledge model and the user model, and supporting ad hoc communications by an adaptive evolutionary hypermedia system.

This new conception allows: the representation and control of communication with the environment, the generation of different templates and navigation models which are available using a PDA or PC, the selection of the best semantic view take into account the user profile and, finally, the adaptation and evolution of the hypermedia model according to the progress obtained and to the new demands of each user.

Now we are working on creating a tool for educators to allow the design and adaptation of the communicator and to monitor the user behaviour. In next future, we will extend our experience with autistic children (Sc@ut project) to other communities such as people with speech or memory disabilities.

References

1. Schlosser, R.W. Braun, U. Efficacy of AAC Interventions: Methodological Issues in Evaluating Behaviour Change, Generalization and Effects. ACC Augmentative and Alternative Communication, 10, 207-233. 1994.
2. Medina, N. Molina, F. Rodríguez, M.J. García-Cabrera, L. An Architecture for the Development, Evolution and Adaptation of Hypermedia Systems. Proceedings of the International Conference on Software Engineering Research and Practice. SERP'03. Pp: 112-118. Las Vegas, Nevada, USA. June 23-26, 2003.
3. Rodríguez, M.J. Paderewski, P. Rodríguez, M..L. Gea, M. An approach for evolving an adaptive and unanticipated system: a communicator for children with autism. FUSE 2004. Workshop on Foundations of Unanticiped Software Evolution, pp. 40-47, 2004.
4. Hurtado, M.V. García-Cabrera, L Parets, J. Semantic Views over Heterogeneous and Distributed Data Repositories: Integration of Information System based on Ontologies. Advantages In Knowledge Organization. Vol. 8, pp 332-347. Ergon Verlag , 2002.
5. Rodríguez, M.L. Rodríguez, M.J. Gea, M. A Framework for Modelling the User Interaction with a Complex System. Computer Aided Systems Theory-EUROCAST 2003 LNCS 2809. pp: 50-61. Springer Verlag, 2003.
6. García-Cabrera, L. Rodríguez, M.J. Parets, J. Evolving Hypermedia Systems: a Layered Software Architecture. Journal of Software Maintenance and Evolution: Research and Practice. Wiley. Vol 14. pp: 389-406. 2002.

The Gaps of the Thesaurus Wordnet Used in Information Retrieval

Javier de la Mata, Jose A. Olivas, and Jesús Serrano-Guerrero

University of Castilla-La Mancha
Paseo de la universidad 4,
13071, Ciudad Real, Spain
{JavierL.Mata, Jesus.Serrano3}@alu.uclm.es,
Joseangel.olivas@uclm.es

Abstract. Due to the exponential growth of Internet it is very important to have good knowledge structures that let to obtain good results in Web search. The aim of this work is to discover the user tendencies when they use the search engines and to know the limitations of the knowledge structures that GUMSe[1] uses. With this information is possible to design a more efficient system. For this reason, it is analyzed the set of keywords and queries more frequently used in the search engines and how WordNet manage it. This information is very useful to avoid bad situations in our meta-search engine.

1 Introduction

Different users may differ about the relevance of several documents obtained using the same query. Relevance is a subjective notion. Standard search engines try to solve the main problems that affect the quality of the results with the aim of obtaining a relevant collection of documents. The main sources of these problems are the ambiguity and the vocabulary. But the search engine needs to know some kind of semantic information that let it to improve the results.

The main sources that are usually used to discover the semantic information and relations are the dictionaries, thesaurus or ontologies. WordNet [1] is one of the main tools used in information retrieval processes, mainly for disambiguation tasks. These tools have several problems such as for example the granularity of the senses [2] or the lack of recent terms. For example, WordNet 2.0 doesn't recognize the terms "XML" or "CORBA". Other times, it recognizes the common senses of one word, like in the case of "SOAP", but does not recognize the new sense (Simple Object Access Protocol for the previous case).

This work is focused on the study of the weak points of WordNet. Our objective is to use its information to improve our knowledge structures. For this reason, the study of the user's query is an important aspect that helps us to know the terms not recognized by WordNet. This information can be later on used to get better results.

There are many works focused on the study of user queries. Most of them offer statistics about the number of keywords, the number of queries per session, and other

---

[1] GUMSe: Gum Search, meta search engine Developer in the framework of SMILe-ORETO-UCLM (Soft Management of Internet e-Laboratory) research group.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 229 – 234, 2005.
© Springer-Verlag Berlin Heidelberg 2005

statistical measures [3]. For example, Jansen & Spink [4] studied the queries of the users for the Excite Search Engine. But this study was concentrate on users' sessions, queries and terms. Other interesting study is the comparison of three different search mechanisms: query-based search, directory-based search and phrase-based query reformulation assisted search [5]. This study concludes that query reformulation can significantly improve the relevance of the documents but with an increase in the search time and the cognitive load.

Usually one keyword is considered a unique word. It is a serious problem because in many situations it is not possible to use an isolated word to describe a text. For this reason, in this work, it is defined a 'keyword' as a word or combination of words that describe a remarkable characteristic or item of one topic. But, usually user queries have several words, and now another problem appears: how to distinguish keywords within a query?

GUMSe [6] have been developed like a platform that allows us to test new ideas in Web search processes. Using the classic technique of query expansion, GUMSe semantically obtain a collection of additional queries related to the original one. New queries are generated replacing or introducing new related terms to previous ones by means of synonymy, hyponymy or hyperonymy relations.

2 Methodology

The first step was to obtain a collection of user queries. Nowadays, the system is in test phase and we do not have enough queries to make an exhaustive study. Nevertheless, there are many web sites that make available the most popular queries that users submit, or even all the queries. For the object of this study, the main source of our collection of keywords was Hitbrain[2] and MetaSpy. The Hitbrain Web site offers a collection of 10.000 keywords and information about each one such as the frequency of use, the position in the monthly ranking and the last positions. This site assumes that a keyword can consist of several words. In addition, we used other sets of queries from MetaSpy. At this point we have to distinguish between keywords and queries. The difference between both is a little bit unclear because one keyword is also a query, but a query is not a keyword. That is to say, a query can be formed by one or more keywords.

The following step was to adapt the data from different sources to the same format for its later processing and study. Once it was completed, we made three different experiments:

1. **Study of the terms recognized by WordNet**: counting the number of terms that WordNet recognizes.
2. **Study of the terms recognized by WordNet with bad sense**: it is analyzeed if WordNet recognizes the terms in a wrong way. Frequently WordNet recognizes one keyword in a wrong way due to the polisemy of the terms.
3. **Study of the topics of the terms**: Finally, it is analyzed what topics are most frequently used by users.

---

[2] http:/www.hitbrain.com

3 Results

3.1 Terms Recognized by WordNet

Our first experiment was to count the number of keywords that WordNet recognizes. In this experiment, three different situations appear: WordNet recognizes the keyword, WordNet does not recognize the keyword and WordNet recognizes the keyword in a wrong way. In this analysis, the third case is not considered. We assume that WordNet recognized the items in a right way.

For this study two test collections of terms were used: Col1 and Col2. The first collection (Col1) from Hitbrain was compound by 10,007 keywords. The second (Col2) was a collection of queries obtained from Metaspy with 123,809 queries.

Table 1. Characteristics of the two collections

	Items	*Type*	*Source*
Col1	10,007	Keywords	HitBrain
Col2	123,809	Queries	MetaSpy
Col2	307,286	Keywords	MetaSpy

The results of the analysis of Col1 are shown in table 2. This collection has keywords formed by one to five words. The experiment reveals that WordNet recognizes the 66% of the keywords formed by one word, and this percentage decrease when the number of words that compound each keyword increase. WordNet only recognizes the 4,53% of the keywords formed by 2 words (see Table 2), and practically the 0% of the keyword with more than 2 words. Total: the 45,45% of the keywords are recognized by WordNet, where the 66,6% of them are keywords with only one word.

Table 2. Results of the analysis for Col1. The table shows the number of words that form one keyword, the number of keywords recognized and not recognized by WordNet, and the percentage of keywords recognized.

Nº WORDS	*RECOGNIZED*	*NOT RECOGNIZED*	*Nº KEYWORDS*	*PERCENTAGE*
1 word	4420	2250	6670	66,26%
2 words	127	2676	2803	4,53%%
3 words	0	451	451	0%
4 words	1	61	62	1,61%
5 words	0	13	13	0%
Others	0	8	8	0%
TOTAL	4548	5459	10007	45,45%

The analysis of Col2 was different because it is formed by queries of the users, and not only keywords. One query can be formed by one or more keywords. In this experiment we use two criteria of evaluation of the query: the first one was to consider the query like a keyword (formed by one or more words) and the second was to consider one query formed by one or more keywords (each word is assumed like a keyword).

Table 3. Results of the analysis of Col2

Case	*Items*	*Rec.*	*Not Rec.*	*Per.*
Queries	123,809	31,637	92,172	25,55%
Keywords	307,286	152,679	154,607	49,69%

The results in both cases are very different. For the first case the number of queries was 123,809. The number of queries recognised was very low (only the 25,5%) and it was only the queries formed by one keyword. In the second case each word is considered like a keyword. The number of words processed was 307,286 (the average number of terms by query was 2,48) and WordNet recognized the 49,68% of the words. This result is very similar to the average number of the first experiment.

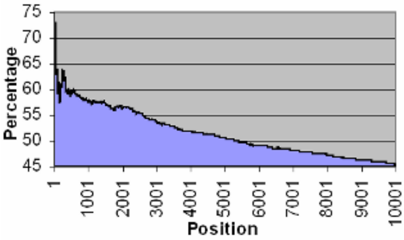


Fig. 1. Average of the terms recognized by WordNet for Col1 related to the frequency of use

Another important aspect is that WordNet recognized around the 60% for the 600 first keywords (see Figure 1). But the average is decreasing in the keywords with lower position (or frequency of use). This means that if the number of different keywords is very small, then the behaviour of WordNet is better than if there are many keywords. But if the frequency of the use of the first keywords is very high, then the behaviour of WordNet improves because correct cases are more frequents. It implies that it could be better to improve only the behaviour for the keywords more frequently used because they are the more probable situations.

3.2 Terms Recognized by WordNet with Bad Sense

Next work is a preliminary study of the precision of WordNet. This experiment uses the 250 first keywords of the collection Col1, where WordNet recognized only 180. The process consists of verifying what keywords recognized by WordNet are wrong. For the accomplishment of this study, the meanings of each keyword recognized by WordNet were observed. If WordNet has the correct sense then the keyword has been recognized right. On the contrary, the sense has been selected incorrectly. Evidently this test is subjective, since the criterion to decide if one keyword is wrong depends of the person that makes the test. For this reason this aspect of our investigation can be improved in the future. The results of this experiment are showed in table 4.

Table 4. Results of the second experiment

N° Keywords	*180*	
OK	157	87,2%
WRONG	23	12,8%

This experiment shows that around the 12% of the keywords recognized by WordNet are not in the correct sense. This is the case, for example, of the keyword "amazon" that in WordNet can be: "a large strong and aggressive woman", "one of a nation of women warriors of Scythia", "a major South American river" or "mainly green tropical American parrot". The previous meanings are correct and there are people that looking for these topics, but in Internet, the usual case (we think) is to use this keyword to search a web site that sells books.

3.3 Study of the Topics of the Terms

The last experiment classifies the keywords with the objective of knowing what domains are more demanded for the users. This information is useful to know why the WordNet thesaurus fails. In this experiment we analyze the first 250 keywords of Col2 and each keyword was assigned to one or more pre-established categories. Table 5 shows the 10 categories. The "other" category includes the keywords that are not in the previous nine ones. This experiment uses, such as the previous one, a subjective criterion. The results show that many queries in Internet are about Internet. This causes that the queries about Internet are not recognized in some cases in a correct way by WordNet. An exhaustive analysis can help us to know the weakness of WordNet and what aspects are necessary to improve if we want to get a more efficient meta-search engine.

Table 5. Distribution of the keywords by categories and wrong senses in each one

Category	*Hits*	*Errors*	*% OK*	*% ERROR*
Web	75	10	30%	13,3%
Computer	55	7	22%	12,7%
Location	25	4	10%	16%
Games	21	3	8,4%	14,3%
Music	19	2	7,6%	10,5%
Movies	15	1	6%	6,6%
People	13	0	5,2%	0%
Sport	10	0	4%	0%
Health	6	0	2,4%	0%
Others	99	0	39,6%	0%

It is also studied the relation between the categories and the recognition of bad senses in WordNet. This experiment reveals that the two main categories that causes fails are *Web* and *Computer* categories (the category *Others* is not considered).There are also other categories such as *Music* or *Games* that are in continuous change. This situation make difficult to update a knowledge structure and it is so easy that it fails when terms about these categories are used.

4 Conclusions

In this study, WordNet recognizes the 45,45% of more frequently keywords used in user queries. The study used 10.007 keywords. The keywords have 1 to 5 words. Other important aspect is that there are situations where WordNet recognizes keywords, but in a wrong way. These cases are very infrequent but they can produce mistakes in search processes.

It is necessary to have good knowledge structures that could be used for information retrieval purposes to focus the search and to obtain better results. For this reason, a subsystem specialized in the improvement of the knowledge structures could be a good support for search engines.

References

1. Barto, A. G., Sutton, R. S., Anderson, C. W.: Neuronlike adaptive elements that can solve difficult learning control problems. IEEE Trans. Systems, Man, and Cybernetics, Vol. SMC-13, (1983) 834–846.
2. Agirre, E, Lopez de Lacalle, O.: Clustering Wordnet word senses. In: Proceedings Recent Advances on Natural Language Processing (RANLP'03) (2003)
3. Werbos, P. J.: Neural networks & the human mind: New mathematics fits humanistic insight. In: Proceedings of the 1992 IEEE Trans. Systems, Man, and Cybernetics, Vol. 1, (1992) 78–83.
4. Jansen, B. J., Spink, A., Saracevic, T.: Real life, real users, and real needs: A study and analysis of user queries on the web. Information Processing & Management, Vol. 36, No. 2, (2000) 207-227.
5. Bruza, P., McArthur, R., Dennis, S.: Interactive internet search: Keyword, directory and query reformulation mechanisms compared. In: Proc. 23rd annual Int. ACM SIGIR conf. on Res. and Devel. in Inform. Retrieval. Athens, ACM Press: New York. (2000) 280-287.
6. Olivas, J. A., de la Mata, J., Serrano-Guerrero, J.: Ontology Constructor Agent for improving Web Search with GUMSe. In: Proc. of Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'04), Vol. 2, Perugia, Italy, (2004). 1341-1348.

Fuzzy Adaptive Objects (Logic of Monitors)

Germano Resconi[1], Javier Alonso[2], and Raul Izquierdo[2]

[1] Catholic University, via Trieste 17, Brescia, Italy
resconi@numerica.it
[2] Object Oriented Technology Laboratory, Department of Computer Science,
University of Oviedo, Calvo Sotero s/n 33005 Oviedo, Spain
{javier, raul}@a4devis.com

Abstract. The active semantic in Adaptive Object-Model (AOM) is only one example of the more complex active semantics at different orders. When we violate the integrity of the system, uncertainty grows up in the system. When monitors are conflicting worlds in the modal logic, we can study uncertainty with the logic of the monitors that is comparable with the logic in the fuzzy set theory. Fuzzy values (integration degree) of a concatenation of interactive objects can be computed by fuzzy AND, OR and NOT operators presented in this paper.

1 Introduction

An Adaptive Object-Model (AOM) is a system that represents classes, attributes and relationships as metadata. Users change the metadata (object model) to reflect changes in the model. These changes modify the system's behaviour.

The relation among objects or interactive object generates constrains that we control by the monitors and propagators. To enforce constrains requires that the related objects were updated with information describing the trigger when the propagator is instantiated Active semantics uses monitors for a local object. For a far object to actively restore the integrity of the interactive object we use the propagator that propagates through objects the message of the monitors. The active semantics in AOM is only one example of the more complex active semantics at different orders. When we violate the integrity, the monitors give the information how and where the integrity is violated. When we assume that any monitor is a world in the modal logic, the monitors are a set of conflicting worlds. With the logic of the monitors we can study how uncertainty can be computed by the AND, OR and NOT elementary logic operators. The logic of the monitors can be compared with the logic of the fuzzy sets. We create logic expressions with the monitors and we compute the degree of integrity in complex logic situations.

2 Entity and Relationship

Following the Specialization Principle we design Entity –Relationship structure as in the following figure and table.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 235 – 240, 2005.
© Springer-Verlag Berlin Heidelberg 2005

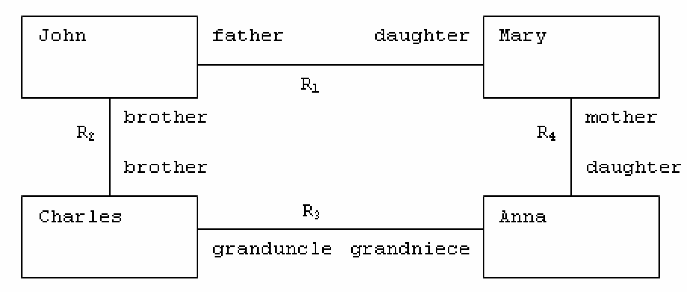


Fig. 1. Graphic image of a particular coherent case of roles and relations

Table 1. Coherent table of role and relations

Roles	Relation_1	Relation_2	Relation_3	Relation_4
father	daughter			
daughter	father			
...	...	...	....	....
daughter				mother

Any entity (E1, E2, E3 & E4) has two roles that must be coherent with the entity itself.

E1 = "John" , E2 = "Mary" , E3 = "Charles" and E4 = "Anna"

So the Entity – Relation can be modelled in this way

$$M = < ROLE , RELATION , ENTITY , F , G > \quad (1)$$

Where ROLE is the set of roles, RELATION is the set of relations, ENTITY is the set of entities and

$$F : ROLE \times RELATION \rightarrow ROLE \quad (2)$$

Is the transition rule for which given the "relation" we can obtain a role from another role. The function G

$$G : ROLE \times RELATION \rightarrow ENTITY \quad (3)$$

Is the reply rule for which we can associate any role and relation with an entity.

Because any entity is not a simple element but is an abstract element with different instances, we can describe the internal structure of the entity by the rule

$$H : INSTANCE \times ATTRIBUTE \rightarrow VALUE \quad (4)$$

Where instances are samples of the Entity class.

3 Active Semantics

Relationships capture the semantics of the interactive objects and become the means by which active semantics is used to express behavioural composition.

Adaptive Object Modelling (AOM) is a model based on instances rather than classes. When we define the object type class, any instance of the class is an instance of the object. With the introduction of the object type, we can have active objects and passive objects. In passive objects we have only the reaction of the object to a message. Inside the object type we have the methods, states, monitor and participant structure.

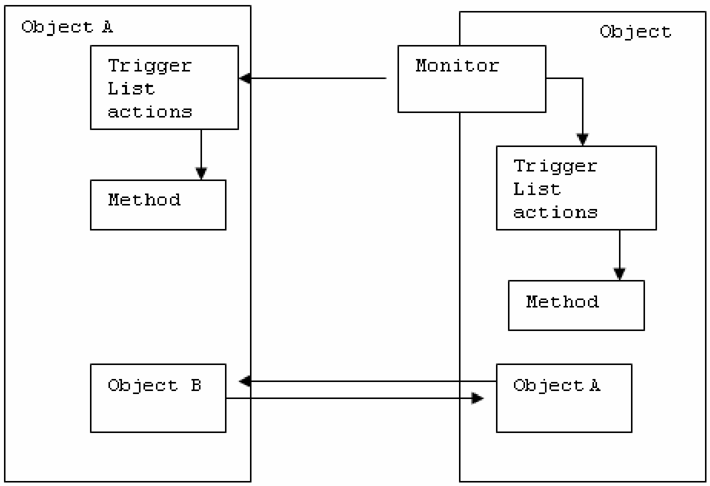


Fig. 2. Internal structure of two interactive objects with monitor and participants in the relationship of the mutual control of the methods

4 Monitors as Worlds in the Meta-theory of Uncertainty

We know that in modal logic the world is the entity by which we can know if a proposition or assertion is TRUE of FALSE. When the enforcement rule (assertion) is violated the assertion is FALSE and the integrity of interactive objects is not valid.

In the Adaptive Object Modelling the implied action of an assertion is to reject any action which would lead to violation of the constraint. But in general we assume that the implied action cannot always eliminate the violation of the constraint. In this case we break the integrity and the system has uncertainty condition. When the assertions are given as TRUE in all the monitors the associated objects are completely integrate and the constrain condition is valid. When all the monitors give the value FALSE then we have a complete violation of the integrity. But, when for a part of the monitors the assertions are true and for other parts the assertions are false we have a partial integrity of the objects. In conclusion we assume that:

- We associate to a monitor a possible world
- The relation among the monitors is a relation among the worlds
- We associate to any assertion in a monitor (world) a logic value TRUE or FALSE

With the monitors (worlds) we define the *Kripke model* i

$$M = < W, R, V > \tag{5}$$

where W is a non-empty set of possible worlds (monitors), R is any type of relation among the worlds. $R \subseteq W \times W$ is an accessibility relation on W, and V is the function that assigns a logic value TRUE or FALSE to any monitor.

$$V\text{: Propositions} \times W \rightarrow \{T, F\} \tag{6}$$

A proposition p is necessarily true when in all the accessible worlds (monitors) p is true. The proposition p is necessarily false when there is at least one accessible world (monitor) where p is false. The proposition p is possibly true when there is at least one accessible world (monitor) where p is true.

Resconi, *et al.* (1992-1996) suggested to adjoin a function

$$\Psi : W \rightarrow R \tag{7}$$

where R is the set of real numbers assigned to worlds W in order to obtain the new model

$$S1 = < W, R, V, \Psi > \tag{8}$$

That is for every world, there is an associated real number that is assigned to it. With the model S1, we can build the *hierarchical meta-theory* where we can calculate the expression for the membership function in the fuzzy set theory .

Imprecision means that an “entity” (temperature , velocity …) cannot have a crisp logic evaluation. The meaning of a word in a proposition may usually be evaluated in different ways for different assessments of an entity by different monitors, i.e. worlds. In this case a world is associated to a monitor in the active object.

The violation of the integrity in an interactive object is the principal source of the imprecision in the meaning representation of the interactive object.

When we write for short:

$$\mu_{p_A}(\mathrm{x}) = \frac{(\mathit{set\ of}\ \text{monitors where } \mathrm{p_A}(x) \text{ is true})}{|W(x)|} \tag{9}$$

where pA(x) is the integrity attribute for the interactive object x in the set A of interactive objects, the variable μA(x) is the membership function in the fuzzy set A of the interactive objects and for a particular interactive object x.

The membership expression is computed as the value of Ψ in S1 stated in (1) above. It is computed by the expression

$$\Psi = \frac{1}{|W|} \tag{10}$$

5 Logic of the Monitors and Fuzzy Set Theory

Because any monitor as a world gives us information where we violate the integrity of the system, we are interested in the development of logic operations by which we can create logic expressions in the logic of the monitors.

5.1 Operation AND

Starting from two propositions, such as ' "John is tall" is true' AND '"John is heavy" is true', given that we know the membership functions of $\mu_{p_1}(x)$ for P_1: '"John is tall" is true', and $\mu_{p_2}(x)$ for P_2: '"John is heavy" is true'. It should be clear that if we know the set of possible worlds $W_1=\{Wi\}$ where p_1 is true and the set of possible worlds $W_2=\{Wj\}$ where p_2 is true, then we can compute

$$\mu_{p_1}(x)=|W_1|/|W| \text{ and } \mu_{p_2}(x)=|W_2|/|W|. \tag{11}$$

Because $|W_i| = N_i$ is the number of possible worlds where the proposition p_i is true. We generate a new event p such that "p_1 and p_2" is "true" by the expression

$$p = p_1 \wedge p_2 \tag{12}$$

where “and” is interpreted to be equivalent to “$\wedge$” operation. So we have

$$\mu_p(x)=\mu_{p_1\wedge p_2}(x)=\frac{|W_1\cap W_2|}{|W|} \tag{13}$$

Remark 1. When we know the value of p_1 *and* p_2 *for every world we evaluate the expression* $p = p_1 \wedge p_2$*. Because for* p_2 *I can choose any type of sentence, we can choose* $p_2 = \neg p_1$*. When the set of worlds where* p_1 *is true and the set of worlds where* p_2 *is true have intersection different from zero, irrational worlds (monitors) can grow up: obtaining worlds where* $p = p_1 \wedge \neg p_1$ *is true.*

We can easily show that for W_1^C , the complement of W_1 we have

$$\mu_{p_1\wedge p_2}(x)=\frac{|W_1\cap W_2|}{|W|}=\frac{|W_2|}{|W|}-\frac{|W_1^C\cap W_2|}{|W|}=\min[\mu_{p_1}(x),\mu_{p_2}(x)]-\frac{|W_1^C\cap W_2|}{|W|} \tag{14}$$

In this case we have $W1\cap W2 = W2$ that is the set with the minimum value of cardinality.

When $p2 = \neg p1$ we have that all the worlds in W2 are irrational. We can prove that

$$0\le\mu_{p_1\wedge p_2}\le\min(\mu_{p_1},\mu_{p_2}) \tag{15}$$

Between a zero irrationality to the maximum of the irrationality in the monitors.

5.2 Operation OR

For the OR combination, following similar steps as in the paragraph above, we have:

$$\max(\mu_{p_1},\mu_{\neg p_1})\le\mu_{p_1\vee\neg p_1}\le 1 \tag{16}$$

In this case, the set where $\neg$ p1 is true is included in the set where p1 is true, we break the classical property for which the set of worlds where $\neg$ p1 is true is the complement set of the worlds where p1 is true.

5.3 Operation NOT

In the fuzzy calculus we break the classical symmetry for which:

$$\mu_{\neg p} = 1 - \mu_p \tag{17}$$

The set where p is true and the set where ¬p is true are separate sets without any connection one with the other as we have in the classical modal logic.

6 Conclusion

This paper introduces the partial coherence or integrity. When we have defect in knowledge, we violate the integrity of the system of interacting objects. The monitors give us the position in the system where the integrity is violated. Meta-theory of uncertainty by modal logic, where the world is a monitor, can generate a special logic or logic of the monitors by which we can compose the monitors results with the AND, OR and NOT logic operations.

References

1. G.Resconi,T.Murai, Field Theory and Modal Logic by Semantic Field to Make Uncertainty Emerge from Information, Int.J.General System 2000.
2. G.Resconi and I.B. Türkşen, Canonical forms of fuzzy truthfulnesss by meta-theory based upon modal logic, Information Sciences 131 /2001) 157 – 194
3. G. Resconi, G.J. Klir and U. St. Clair, Hierarchical Uncertainty Metatheory Based Upon Modal Logic, Int. J. of General Systems, Vol. 21, pp. 23-50, 1992.

A Model-Based Architecture for Fuzzy Temporal Diagnosis*

J.M. Juarez, J. Palma, M. Campos, J. Salort, A. Morales, and R. Marin

Artificial Intelligence and Knowledge Engineering Group, University of Murcia, Spain
`jmjuarez@um.es`

Abstract. The application of Model Based Diagnosis (MBD) techniques together with Knowledge Base Systems, points out the need of general modelling frameworks in domains where time management is important. The development of this kind of systems, far from becoming simpler, reveals the additional complexity inherent in each specific domain. This paper describes a general architecture for this purpose using a model based approach for developping temporal knowledge based systems in different domains. This work also relates our experience in applying this architecture in the concrete domain of Intensive Care Units.

1 Introduction

Since its beginnings, Artificial Intelligence has paid special attention to knowledge-based systems and its design. In particular, the application of model-based reasoning has obtained important results in the last decades [1,2].

However, the experience in developing these systems reveals the complexity of the design, the development, and its maintenance. Furthermore, in most of cases, the systems are highly dependent on the domain and the problem to be solved. In our opinion, there are some particular situations where these difficulties are critical, for instance the analysis of knowledge-based systems in order to solve temporal diagnosis problems ([3,4]).

This work presents a general model-based framework for intensive-knowledge domains, where the temporal dimension is considered to deal with the diagnosis problem. Nevertheless, the task of designing a generic architecture is a complex issue. One of the most difficult steps is to establish the set of goals to be reached. Our general framework tries to deal with the following key topics: domain independence, the management of temporal uncertainty, the knowledge acquisition bottleneck, model-based representation, and the diagnosis task. This paper also relates our practical experience in the application of this architecture.

The structure of this paper is organised as follows: firstly there is a general overview of the proposed architecture. After that, the two main modules (temporal/domain and

---

* This work was supported by the Spanish MEC under project MEDICI (TIC2003-09400-C04-01), the Regional Government of Murcia under project PB/46/FS/02, and the Spanish MEC under the FPU national plan (grant ref. AP2003-4476).

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 241–246, 2005.
© Springer-Verlag Berlin Heidelberg 2005

decision support module) are explained in detail.Finally, we describe our practical experience in the application of this architecture in the particular domain of the Intensive Care Units.

2 Model-Based Architecture Overview

The architecture proposed is based on a behavioural model in order to solve the temporal diagnosis problem from a Model-Based Reasoning approach (MBR). This section shows the main structure of the architecture, describing each part in detail afterwards. This structure defines two modules, the Temporal/Domain Module and the Decision Support Module.

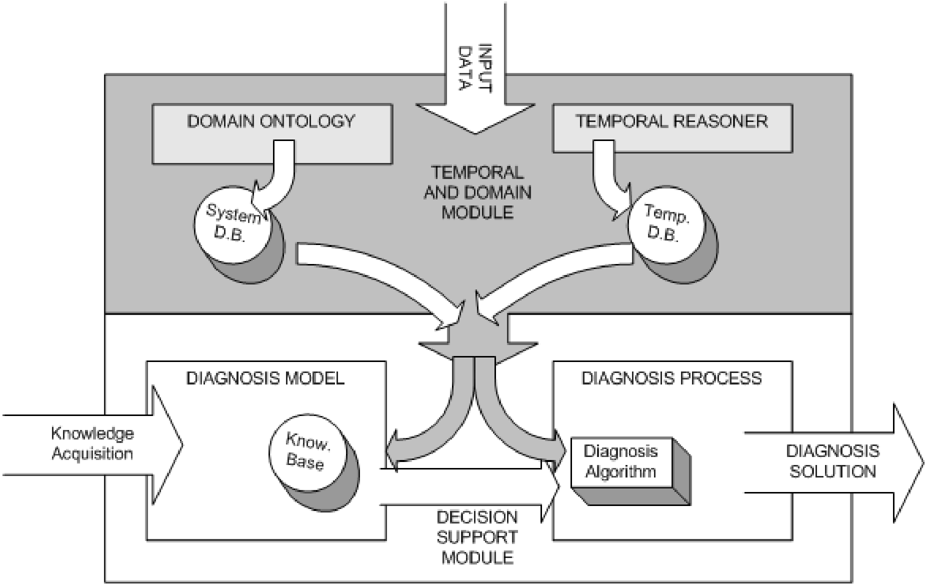


Fig. 1. The General Model-Based Architecture Overview

The Temporal/Domain Module is composed by all those elements that allows the architecture to be independent of the domain, and the management of the temporal dimension. **The Decision Support Module** includes all the elements needed to obtain the diagnosis solution.

3 The Temporal / Domain Module

The design of any system architecture for model-based diagnosis becomes highly difficult if the importance of the knowledge domain is considered, as well as its impact on the design of systems. This work proposes an architecture able to be used in the development of fuzzy temporal diagnosis systems for different domains. To this end, we suggest the use of ontologies and general purposes temporal reasoners.

The temporal/domain module provides the general architecture with two main advantages. Firstly, this module deals with the consistency task: both domain and temporal consistency. Secondly, the use of ontologies to represent part of the domain knowledge gives the architecture flexibility, in order to apply it on different domains.

The complexity inherent to some domain knowledge increases the problems associated to generic architectures. The use of a domain ontology server allows to solve this situation partially by keeping the semantic consistency of the concepts used.

The temporal database stores all the system information as well as some temporal data. However, those tags are not enough to manage the temporal dimension. Thus, the use of temporal reasoning techniques provides the architecture with important advantages. For instance, it guarantees the temporal consistency, and it infers new temporal relations [5]. This architecture relies on FuzzyTIME [6], a general purpose fuzzy temporal reasoner.

4 The Decision Support Module

The main elements of the Decision Support Module are the formal model, the knowledge acquisition tool and the diagnosis process. These elements conform the set of fundamental components of the architecture in order to build a diagnosis solution.

This architecture is supported by a formal model which is a temporal and causal model of failures named TBM (Temporal Behavioural Model). The TBM describes the underlying structure stored in the KB. This model is structured as a causal network in which each failure is connected to the abnormal manifestations and to other caused failures. Time dimension is an important factor to be considered. Therefore, each failure description is extended to include temporal knowledge as a set of temporal constraints among elements. The application of TBM in the medical domain is the result of previous works and can be read in [7].

The model (TBM) that supports the architecture is used and stored in the knowledge base of the architecture. This knowledge base is supported by some knowledge acquisition tools, which must access both the formal model and the domain ontology.

The diagnosis process is the main element of the module. The diagnosis process, is a fuzzy temporal model-based diagnosis. In this kind of process, the role of the diagnosis is to find a diagnosis solution from the temporal data (from the temporal database), from the domain ontology (from the ontology server), from the knowledge base, and with the help of the temporal reasoner. This process is based on an algorithm that obtains a diagnosis solution. This solution must be temporally consistent, semantically consistent, fulfill the temporal and causal constraints of the formal model, and explaining the findings of the temporal database information. Essentially, our diagnosis proposal is based on a causal network (temporally consistent) and a temporal constraint network that are built using and abductive strategy to explain the set of events.

5 Practical Experience. An Application for Intensive Care Unit

This section describes the practical application of the architecture by explaining our implementation for the medical domain, in particular the Intensive Care Unit (ICU).

Physicians at intensive care units have to deal with an overwhelming amount of data provided not only by on-line monitoring but also collected from patients' records (e.g., laboratory results), which are, in most cases, collected manually at different time instants. In order to provide efficient decision support systems and medical research tools in the ICU domain, it is necessary to integrate and analyze the information provided from these different sources. The result is ACUDES , the Architecture for intensive Care Unit Decision Support.

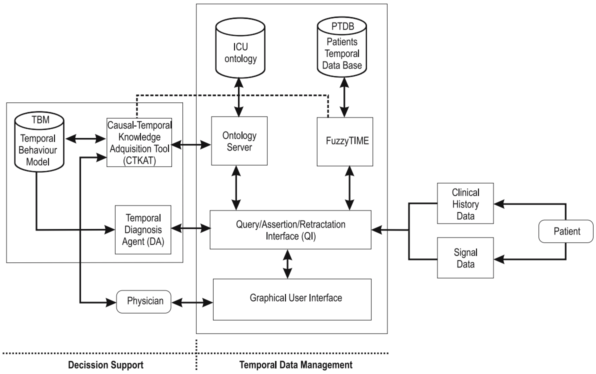


Fig. 2. ACUDES: Architecture for intensive Care Unit Decision Support

ACUDES is an specific system that follows the structure of the architecture proposed. Thus, it includes a temporal Data management Module, where the FuzzyTime temporal reasoner and the Ontology Server are allocated; in this module you can find also the Temporal Data Base (TDB), the ICU Ontology and the interface defined for accessing the module. On the other hand, the Decision Support Module (DSM) contains the diagnosis agent which is in charge of the diagnosis process. This process is the core of ACUDES and it is described by a causal and temporal behaviour model-based algorithm. It is worth mentioning that we have designed and implemented CATEKAT, a web-based tool to acquire the domain knowledge for the diagnosis model.

5.1 Knowledge Acquisition Tool:CATEKAT

The medical domain, in particular ICUs, is hard to classify and its terminology is hard to interpret. In some cases, the definition of terms may leave space for wrong interpretation. Hence, the knowledge acquisition lies on an ontology which allows a consistent use of the medical terms. Thus, using ontologies, KB acquisition could be viewed as a process for enlarging a domain ontology with specific knowledge of a particular domain.

We suggest the use of dictionaries of synonyms and thesaurus for solving cases of equivalent terms that are not considered in the ontology.

Another point that we dealt with was the incompleteness of the knowledge acquisition. For example, the physician could describe some related clinical signs. However , the physician does not specify how this signs must be interpreted when they are present in the patient. This insufficient information must be acquired from some other experts.

The main module in CATEKAT is the web based KA user interface. There are several requirements that characterise its design and implementation:

- a multi-user environment.
- role-aware, i.e. the management of different roles such as expert or knowledge engineer
- avoiding inconsistencies in temporal pattern edition by locking a pattern when a user is working on it
- providing an effective cooperative work platform
- allowing the definition of projects related to different domains
- browsing and querying capabilities.

5.2 Graphical Diagnosis Tool

In our experience at the ICU domain, we had dealt with two problems of the clinical decision support: the gathering of temporal medical information (the input of the diagnosis process); and the representation of the diagnosis outcome.

The aim of the **Evidence Acquisition Tool** is to provide physicians with the data gathering of clinical histories, allowing the system to return the diagnosis outcome. This tool has been designed to facilitate a visual and interactive acquisition of patients' findings, taking into account the temporal dimension of this information, which is critical in ICUs domains.

In decision support systems, the representation of the diagnostis solution is a key question, not only from a practical point of view, but also to provide an understable diagnosis output. To this end, the **Diagnosis Navigator** shows an understable output of the diagnosis process, as well as an explanation of the diagnosis solution. This explanation is made by describing each decision that the diagnosis algorithm has reached through the process of causal network construction within the reasoning process. This functionality allows the physician to understand the behaviour of the diagnosis process. Thus, the expert could criticize it and suggest new modifications by means of the CATEKAT tool.

6 Conclusions

This paper proposes a general architecture based on a causal and temporal behavioural model [7] (TBM) for treating the fuzzy temporal diagnosis problem. This architecture defines two modules, the Temporal/Domain Module, and the Decision Support Module in order to deal with: domain independence, the management of temporal uncertainty, model-based representation, and decision support.

In the design of architectures based on models, the selection of a knowledge model is a critical factor. At this point, an important question to be considered is the degree of dependency of the model and the domain which is modeled. On the one hand, a weak dependence facilitates the design of a generic model that can be reused in other domains. On the other hand, the approach of a highly dependent model on the domain provides an easy design of knowledge acquisition tools. The proposed architecture is based on our research in the representation of temporal behavioural models TBM. The model used in the architecture deals with both, generic model and easy acquisition. However, the agreement of the TBM with the domain provides an easy KA, but also a complete description of the domain behaviour.

As we mentioned in Section 5, this work also describes our experiences in the implementation of this architecture for a Decision Support system in a ICU. One of the problems in this domain is the high difficulty of the knowledge acquisition process, where the temporal dimension plays an essential role in the knowledge domain. Thus, in this work we present ACUDES, an implemented system for decision support in ICU to deal with these problems. We also propose the knowledge acquisition tool CATEKAT, which uses TBM to built the knowledge base.

References

1. Console, L., Torraso, P.: A spectrum of logical definitions of model-based diagnosis. In Hamscher, W., Console, L., de Kleer, J., eds.: Readings in Model-Based Diagnosis. Morgan Kauffmann Publisher, Inc. (1992) 78–88
2. Console, L., Dupré, D.T., Torraso, P. In: Abductive Diagnosis with Abstraction Axioms. Volume LNCS 810 of Foundations of Knowledge Representation and Reasoning. Springer, Berlin (1994) 98–112
3. Brusoni, V., Console, L., Terenziani, P., Dupré, D.T.: A spectrum of definitions for temporal model-based diagnosis. Artificial Intelligence **102** (1998) 39–79
4. Chantler, M.J., Coghill, G.M., Shen, Q., Leitch, R.R.: Selecting tools and techniques for model-based diagnosis. Artificial Intelligence in Engineering **12** (1998) 81–98
5. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM **26** (1983) 832–843
6. Campos, M., A.Cárceles, Palma, J., R.Marín: A general purpose fuzzy temporal information management engine. In: Proceedings of the EurAsia-ICT 2002. (2002) 93–97
7. J.Palma, J.M.Juarez, M.Campos, R.Marin: A fuzzy approach to temporal model-based systems for intensive care unit. In: European Congress of Artificial Intelligence ECAI'04. (2004) 868–872

Extension of Ontologies Assisted by Automated Reasoning Systems*

Joaquín Borrego-Díaz and Antonia M. Chávez-González

Departamento de Ciencias de la Computación e Inteligencia Artificial,
E.T.S. Ingeniería Informática-Universidad de Sevilla,
Avda. Reina Mercedes s.n. 41012-Sevilla. Spain

Abstract. A method to extend ontologies with the assistance of automated reasoning systems and preserving a kind of completeness with respect to their associate conceptualizations is presented. The use of such systems makes feasible the *ontological insertion* of new concepts, but it is necessary to re-interpret the older ones with respect to new ontological commitments. We illustrate the method extending a well-known ontology about spatial relationships, the called *Region Connection Calculus.*

1 Introduction

Ontology Management has becomed in a critical issue in fields related with Knowledge Representation and intelligent information processing as the Semantic Web. One of the involved tasks, the most important, is the need of extending or revising ontologies. This task may be, from the point of view of companies, dangerous and expensive: every change in the ontology can affect to the overall knowledge organization of the company. Moreover it is also known that the self process of extension is hard to automatize: the tools are designed to facilitate the syntactic extension or ontology mapping. But the effect of ontology mapping on the logical reasoning may be, in general, unknown, and specially on the use of automated reasoning systems [2].

The aim of this paper is to propose a formal semantics for ontology extension (following the foundational principles given in [2] and suggested by the computer-assisted cleaning of Knowledge Databases [1]) as well as a feasible method, assisted by Automated Reasoning Systems (ARS), to extend ontologies preserving a certain type of *robustness.*

2 Lattice Categorical Extensions

We assume throughout that the conceptualization associated to the ontology is endowed of lattice structure. Actually it is not a constraint: there are methods

---

* Partially supported by the project TIN2004-03884 of Spanish Ministry of Education and Science, cofinanced by FEDER funds.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 247–253, 2005.
© Springer-Verlag Berlin Heidelberg 2005

to extract lattices of concepts from data (such as the Formal Concepts Analysis, see e.g. [8]), and an ontology is easy to be extended by definition to satisfy it. Although we think about Description Logics as a language (a logical basis for ontology languages like OWL, see `http://www.w3.org/TR/owl-features/`), the framework is useful for First Order Logic (FOL).

A *lattice categorical* theory is a theory that proves the lattice structure of its basic concepts. It is a reasonable requirement: the theory must certify the basic relationships among them. We aim to replace *completeness* by *lattice categoricity* to make feasible the extension of formal ontologies.

Fixed a language, let $\mathcal{C} = \{C_1, \ldots, C_n\}$ be a set of concept symbols and let T be a theory (in the general case, definable concepts in T can be assumed) and let us consider the language $L_{\mathcal{C}} = \{\top, \bot, \leq\} \cup \{c \ : \ c \in \mathcal{C}\}$. Given $M \models T$, we consider the $L_{\mathcal{C}}$-structure $L(M, \mathcal{C})$, whose universe is the set of the interpretations in M of the concepts ($\top$ is M, $\bot$ is $\emptyset$), and $\leq$ is the subset relation.

The relationship between $L(M, \mathcal{C})$ and the self M is based in two facts. The first one, that the lattice can be characterized by a finite set of equations E, plus a set of formulas $\Theta_{\mathcal{C}}$ categorizing the lattice under completion, that is, $\Theta_{\mathcal{C}}$ includes the domain closure axiom, the unique names axioms and aditionally the axioms of lattice theory. Secondly, there is a natural translation Π of lattice equations into FOL formulas such that if E is a set of equations characterizing $L(M, \mathcal{C})$, then $M \models \Pi(E)$.

Definition 1. *We say that a $L_{\mathcal{C}}$-theory E is a* **lattice skeleton** *(l.s.) for T if*

- *There is $M \models T$ such that $L(M, \mathcal{C}) \models E + \Theta_{\mathcal{C}}$, and*
- *$E + \Theta_{\mathcal{C}}$ has an only model (modulo isomorphism).*

Every consistent theory T has a lattice skeleton (it is sufficent to categorically axiomatize the lattice associated to some model of T). Intuitively, the existence of essentially different lattice skeletons difficults the reasoning with the conceptualization associate to T.

Definition 2. *T is called a* **lattice categorical (l.c.) theory** *if whatever two lattice skeletons for T are equivalent modulo $\Theta_{\mathcal{C}}$.*

It is easy to see that every T consistent has a lattice categorical extension: it is sufficent to consider a model $M \models T$, and next to find a set E of equations such that $\Theta_{\mathcal{C}} + E$ has $L(M, \mathcal{C})$ as only model. The theory $T + \Pi(E)$ (and any consistent extension of it) is l.c.

To simplify, we deal with a pair (T, E) -where T is lattice categorical and E is a lattice sekeleton for T- that we call a *lattice categorical core* (l.c.c.). Thus, (T, E) is a l.c.c. if $T + \Pi(E)$ is a l.c. theory.

Definition 3. *Given two l.c.c. $(T_1, E_1), (T_2, E_2)$ with respect to the sets of concepts $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively, we say that (T_2, E_2) is a* **lattice categorical extension** *of (T_1, E_1) if $L(T_1, \mathcal{C}_1) \subseteq L(T_2, \mathcal{C}_2)$ and $L(T_2, \mathcal{C}_2) \models E_1$.*

3 Extending Ontologies

In order to obtain a practical method, some of the basic (theoretical) logical principles required by the *definitional methodologies* of building of formal ontologies must be weakened [3]. Such principles, in their original forms, are:

1. *Ontologies should be based upon a small number of primitive concepts.*
2. *These primitives should be given definite model theoretic semantics.*
3. *Axioms should only be given for the primitive concepts.*
4. *Categorical axiom sets should be sought.*
5. *The remaining vocabulary of the ontology (which may be very large), should be introduced purely by means of definitions.*

The three first principles are assumed, but, in order to a feasible management, the last two ones (two strong logical constraints) are weakened. The fourth one will be replaced by *lattice categoricity*, more manageable than logical categoricity or completeness. With respect to the last one, if we start with a basic theory, it can be hard to define any new concept/relation by means of the basic elements of the ontology. Thus, we must consider that there are *ontological insertions*, that is, additions of new concepts/relations not ontologically defined on the former ontology. This may produce a deep readdress of the domain analysis.

The method consists of four steps, assisted by an automated theorem prover (in our case, OTTER, `http://www-unix.mcs.anl.gov/AR/otter/`), a model finder (MACE4, `www-unix.mcs.anl.gov/AR/mace4/`), and a last stage for ontological reconsideration. Starting from a lattice categorical theory:

1. First, one extends the lattice of the basic concepts of the ontology by extending the selected skeleton.
2. Next, one applies MACE4 on a possible axiomatization of the new lattice in order to obtain the new lattices. In general, the characterization of the lattice is a theory weaker than the initial ontology.
3. The third step consists of the refinement of the skeleton in order to MACE4 exhibits one only model (that is, the theory is lattice categorical).
4. Finally, it is necessary to certify (by means OTTER or hand-made) the unicity of above model.

The final stage of the method is not algorithmical. It consists of an ontological interpretation of the new element, by re-interpreting (generally by refining) if necessary, the older ones. This task, nonalgorithmical in essence, is responsability of experts in the domain represented by the ontology. In fact, such re-interpretation can force us to reconsider the initial ontological commitments.

4 An Example in Qualitative Spatial Reasoning

We shall apply the method for extending an ontology on Qualitative Spatial Reasoning by means of the insertion of relations on imperfect spatial information, concretely the well-known *Region Connection Calculus* (RCC) [6]. The

$DC(x,y) \leftrightarrow \neg C(x,y)$	(x is disconnected from y)
$P(x,y) \leftrightarrow \forall z[C(z,x) \rightarrow C(z,y)]$	(x is part of y)
$PP(x,y) \leftrightarrow P(x,y) \wedge \neg P(y,x)$	(x is proper part of y)
$EQ(x,y) \leftrightarrow P(x,y) \wedge P(y,x)$	(x is identical with y)
$O(x,y) \leftrightarrow \exists z[P(z,x) \wedge P(z,y)]$	(x overlaps y)
$DR(x,y) \leftrightarrow \neg O(x,y)$	(x is discrete from y)
$PO(x,y) \leftrightarrow O(x,y) \wedge \neg P(x,y) \wedge \neg P(y,x)$	(x partially overlaps y)
$EC(x,y) \leftrightarrow C(x,y) \wedge \neg O(x,y)$	(x is externally connected to y)
$TPP(x,y) \leftrightarrow PP(x,y) \wedge \exists z[EC(z,x) \wedge EC(z,y)]$	(x is a tangential prop. part of y)
$NTPP(x,y) \leftrightarrow PP(x,y) \wedge \neg\exists z[EC(z,x) \wedge EC(z,y)]$	(x is a non-tang. prop. part of y)

Fig. 1. Axioms of RCC

$$\begin{array}{ccc}
\top \equiv C \sqcup DR & PO \sqsubseteq \neg P \sqcap \neg Pi \sqcap \neg DR & DR \equiv EC \sqcup DC \\
NTPP \sqsubseteq \neg TPP \sqcap \neg Pi \sqcap \neg DR & C \equiv O \sqcup EC & TPP \sqsubseteq \neg Pi \sqcap \neg DR \\
O \equiv PO \sqcup P \sqcup Pi & EQ \sqsubseteq \neg PPi \sqcap \neg DR & Pi \equiv EQ \sqcup PPi \\
TPPi \sqsubseteq \neg NTPPi \sqcap \neg DR & P \equiv EQ \sqcup PP & NTPPi \sqsubseteq \neg DR \\
PPi \equiv TPPi \sqcup NTPPi & EC \sqsubseteq \neg DC & PP \equiv TPP \sqcup NTPP
\end{array}$$

Fig. 2. The skeleton E for the lattice of concepts of RCC

need of such extension arose, for example, when we applied RCC as a meta-ontological tool for analysing and repairing anomalies in ontologies [1] [5]. RCC is a mereotopological approach to spatial reasoning; the *spatial entities* are non-empty regular sets. The primary relation is the *connection*, $C(x,y)$, with intended meaning: *"the topological closures of* x *and* y *intersect"* and basic axioms $\forall x[C(x,x)]$ and $\forall x,y[C(x,y) \rightarrow C(y,x)]$ jointly with a set of definitions on the main spatial relations (fig. 1). Actually the theory has other axioms (see [6]), but these are not necessary to prove the lattice structure of the set of relations (shown in fig. 3). Thus, RCC is lattice categorical.

4.1 Isolating a Skeleton for RCC

In order to isolate a skeleton without redundant formulas, we start with the lattice equations induced by the Hasse diagram of the RCC-relations. Next we sequentially remove equations of this set when such elimination does not produce other new lattices modelling the final set. The set of equations E, see the figure 2, we obtain has an only model (is a skeleton). categorizes under completion the lattice of the RCC-spatial relationships (given in fig. 3) [5].

4.2 Inserting New Elements

The Jointly Exhaustive and Pairwise Disjoint set (JEPD) of atomic relations of the lattice (fig. 3) is denoted by RCC8. It represents the set of the most specific spatial relations in RCC8. Our aim is to insert a new relation representing undefinition, such relation must be disjoint with RCC8.

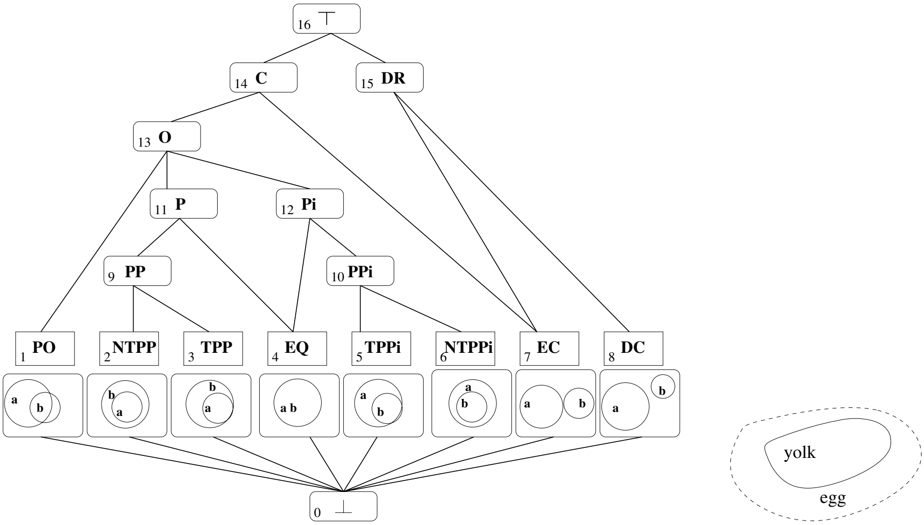


Fig. 3. The lattice of RCC-relations (left) and the egg-yolk representation of vague regions (right)

Theorem 1. *There are only eight E-conservative extensions of the lattice of RCC by insertion of a new relation D such that $RCC8 \cup \{D\}$ is a JEPD set.*

In the proof of the theorem we use MACE4 for listing the lattice extensions, taking as input the lattice axioms, the skeleton, the unique names principles and the closure domain axiom. The system outputs eight extensions. Since MACE4 has not been formally verified to work correctly, it is necessary to certify that such models are correct, and, by finding OTTER's proofs, to show that the list of models is exhaustive. The analysis of the extensions (fig. 4) suggests us that the new relations represent *undefinition up to a degree.*

4.3 Refining the Skeleton for the New Extension

We are not specially interested here in a determinated extension, although there exist situations where it is necessary to select one of them (by example, when we intend to classify unaccurate data [4]). However, the refinement of the skeleton is easy: once an extension is selected. For instance, with respect to the first extension, it is sufficent to substitute the formula $PP \equiv TPP \sqcup NTPP$ by the new formula $PP \equiv TPP \sqcup I_1 \sqcup NTPP$ to obtain a skeleton E' for the extension. Every extension of (RCC, E') is a l.c. extension of (RCC, E).

4.4 Final Stage: Ontological Interpretation

Finally, we need to mereotopologically interpret the new relations. In [5] four different interpretations are offered, we tried to use some of them for supporting

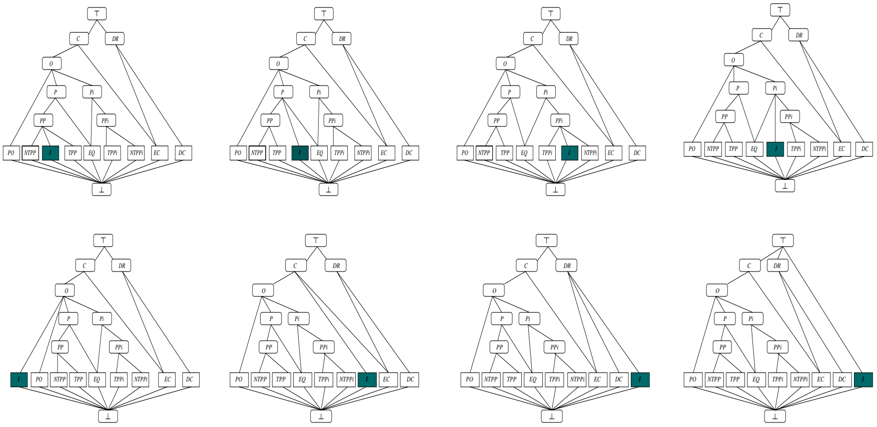


Fig. 4. The eight lattice describing the l.c. extensions of RCC by undefinition

ontological cleaning tasks. In order to show how they can be interpreted in each case, we consider the classical *egg-yolk* interpretation of spatial vague regions [7]. Intuitively, a spatial region a compounded by two subregions, as figure 3 shows, the first, $y(a)$ (*the yolk*) which represents accurate locations in a, and the second one, $e(a)$ bounding the unaccurate locations of a. In [7] the 48 possible spatial relations between two vague regions are shown. If we want to work with I_1, for example, its vague interpretation is $I_1(a, b) \equiv PP(e(a), e(b))$, while RCC relations are interpreted by the natural way.

5 Final Remarks

The method described here is a logical basis for extending ontologies. Since there is a lack of formal notions -feasible in practice- describing features about completeness in the evolution of formal ontologies, we think that our proposal can be useful to add formal semantics to several ontological transformations [5] [1], achieving in this way the logical trust. The feasibility of the method depends of two factors: the use of efficent ARS and the simplicity of the *completeness* notion, related with the conceptualization of the ontology. Future research lines are addressed to embrace the use of roles on spatial reasoning.

References

1. J. A. Alonso-Jiménez, J. Borrego-Díaz and A. M. Chávez-González, Ontology Cleaning by Mereotopological Reasoning Proc. of DEXA Workshop on Web Semantics (WebS'04), (2004). IEEE Press, pp. 132-137.
2. J. A. Alonso-Jiménez, J. Borrego-Díaz, A. M. Chávez-González and F.J. Martín-Mateos, Foundational challenges in Automated and Ontology Cleaning in the Semantic Web. To appear in IEEE Intelligent Systems.

3. B. Bennett, The Role of Definitions in Construction and Analysis of Formal Ontologies, in: P. Doherty, J. McCarthy, and M. Williams (eds.) Logical Formalization of Commonsense Reasoning (2003 AAAI Symposium), pp. 27-35, AAAI Press (2003).
4. J. Borrego-Díaz and A. M. Chávez-González, Management Undefinability in Data Through Automated Ontology Extensions, submitted (2005).
5. A. Chávez-González, Mereotopological Automated Reasoning for Ontology Cleaning, forthcoming Ph.D. thesis.
6. A. G. Cohn, B. Bennett, J. M. Gooday and N. M. Gotts. Representing and Reasoning with Qualitative Spatial Relations about Regions. Chapter 4 in O. Stock (ed.), Spatial and Temporal Reasoning, Kluwer, Dordrecth (1997).
7. A.G. Cohn and N.M. Gotts, The 'Egg-Yolk' Representation of Regions with Indeterminate Boundaries in P. Burrough and A. M. Frank (eds), Proc. GISDATA Specialist Meeting on Geographical Objects with Undetermined Boundaries, GISDATA Series, vol. 3, Taylor and Francis, pp. 171-187 (1996).
8. G. Stumme, R. Taouil, Y. Bastide, N. Pasquier and L. Lakhal, Computing Iceberg Concept Lattices with TITANIC. *Data Knowl. Eng.* 42(2) pp. 189-222 (2002).

A Software Architecture for Effective Document Identifier Reassignment

Roi Blanco and Álvaro Barreiro

Computer Science Department, University of Corunna, Spain
{rblanco, barreiro}@udc.es

Abstract. This works presents a software solution for enhancing inverted file compression based on the reassignment of document identifiers. We introduce different techniques recently presented in the Information Retrieval forums to address this problem. We give further details on how it is possible to perform the reassignment efficiently by applying a dimensionality reduction to the original inverted file and on the evaluation results obtained with this technique. This paper is devoted to the software architecture and design practises taken into account for this particular task. Here, we show that making use of design patterns and reusing software components leads to better research applications for Information Retrieval.

1 Introduction

Indexing mechanisms are a critical part of Information Retrieval systems, as they provide fast access to term information needed for query evaluation [7]. Inverted files are by far the most used indexing structure in Information Retrieval (and even in large-scale database systems), specially when dealing with very large sets of data. Indexing structures store different information related to each term appearing in the collection, depending on the granularity specified. In this paper we will focus on document-level inverted files, this is, we only take into account the occurrences of terms in documents. Figure 1 is an illustrative example, where each line stands for a different document, and the structure stores term document frequency (number of different documents that contain the given term) and document identifiers.

This indexing structure is organised in posting lists, where each one holds the information for a different term. Therefore, an inverted file can be expressed as:

$$\{< t_i; f_{t_i}; d_1, d_2, \ldots, d_{ft_i} >, d_i < d_j \forall i < j\} \forall t_i \in T, \tag{1}$$

where T is the set of terms, f_{t_i} stands for the document frequency of the term t_i, and d_i is the document identifier. As the notation implies, the document identifiers are ordered.

Usually, posting lists are compressed with a suitable coding for the data involved. Compression methods need a suitable model of the data to perform

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 254–262, 2005.
© Springer-Verlag Berlin Heidelberg 2005

Doc.Id	Input Text:
1:	Rain on the green grass
2:	and rain on the tree
3:	And rain on the housetop
4:	but not on me
5:	Rain, rain, go away

term	Documents
rain	$\langle 4; 1, 2, 3, 5 \rangle$
on	$\langle 4; 1, 2, 3, 4 \rangle$
the	$\langle 3; 1, 2, 3 \rangle$
green	$\langle 1; 1 \rangle$
grass	$\langle 1; 1 \rangle$
and	$\langle 2; 2, 3 \rangle$
tree	$\langle 1; 2 \rangle$
housetop	$\langle 1; 3 \rangle$
but	$\langle 1; 4 \rangle$
not	$\langle 1; 4 \rangle$
me	$\langle 1; 4 \rangle$
go	$\langle 1; 5 \rangle$
away	$\langle 1; 5 \rangle$

Fig. 1. Inverted File Example

the coding and decoding operations. Nevertheless there exists a family of methods, known as *static codes* [6] that work without having to store any information about the data. These coding methods are useful for posting lists, as they only contain integers and a model can become larger than the real compressed data.

Actually, not every integer corresponding to a document identifier is coded, but the difference between two consecutive ones, also known as *d-gaps* [7]. Static codes use the fact that small integers occurs often, and give shorter codes to them (measured in bits). So the shorter the differences between consecutive documents, the higher gain in compression, which translates into a smaller inverted file.

The reassignment of document identifiers is a very recent technique for enhancing compression with static codes [2]. The main idea is to map each original document identifier appearing in the collection into a new one, trying to reduce the distances between occurrences in the posting lists. Some papers proved that reordering can lead into gains in compression ratio for static inverted files of medium size collections [2,8,9,10]. These works approached the problem from different perspectives, using other well-known problems for approximating the original one. In this paper, we will show how to build a software architecture for the technique described in [10], which tries to solve the problem in an efficient way by reducing the dimensionality of the input data through matrix transformations. The set of programs to be used must be able to index, perform matrix operations with large sets of data and recompress inverted files using a previously calculated order, everything in an independent manner and using reusable and modular data structures. We focus on an object oriented programming paradigm and make extensive use of design patterns [12]. Section 2 describes the known approaches to the document identifier reassignment problem. Section 3 shows the main technique implemented in the software architecture we are describing. Sections 4 and 5 presents the architecture developed for the referred solution and implementation details. Finally, section 6 summarises the work and shows some future research lines.

2 Previous Work

Next, we describe the state of the art concerning the reassignment of document identifiers. Different approaches to the problem consider different data structures. Most works build a *weighted similarity graph* G, where the nodes v_i, v_j represent the document identifiers i, j and an edge (v_i, v_j) represents the similarity between documents i and j [2,8,10]. On the other hand, the work in [9] uses document clusters for reordering the identifiers.

Blandford and Blelloch (B&B) [2] developed a technique that improved the compression ratio about fourteen percent in TREC text collections. The technique employs a similarity graph as described before, and operates in three different phases. The first phase constructs the document-document similarity graph from the original inverted file. The second part of the algorithm calls to a graph partitioning package which implements the Metis [11] algorithm for splitting recursively the similarity graphs produced by the first part. Finally, the algorithm applies rotations to the clustered graph outputted by the second part for optimising the obtained order. The final assignment for the document identifiers is obtained by simply depth-first traversing the resulting graph. As is stated in [2], constructing a full similarity graph is $O(n^2)$, so the raw technique may not suitable for very large collections. Nevertheless, the efficiency of the algorithm can be controlled by two parameters: τ and ρ. The first parameter, τ, acts as a threshold for discarding high-frequency terms, i.e., if a term t_i has size $|t_i| > \tau$ it is discarded for the construction of the similarity graph. Actually the algorithm works with a sample of the full similarity graph. The parameter ρ stands for how aggressively the algorithm sub-samples the data: if the index size is n it extracts one element out of $\lfloor n^\rho \rfloor$. Tuning τ and ρ the technique may lead to a tradeoff between efficiency and time and memory usage.

Shie et al. [8] proposed a different graph-based approach, based on the well known Travelling Salesman Problem. The TSP is stated as follows: given a weighted graph $G = (V, E)$ where $e(v_i, v_j)$ is the weight for the edge from v_i to v_j, find a minimal path $P = \{v_1, v_2, \ldots, v_n\}$ containing all the vertexes in V, such as if $P' = \{v'_1, v'_2, \ldots, v'_n\}$ is another path in G, $\sum_{i=2}^{n} e(v_i, v_{i-1}) \leq \sum_{i=2}^{n} e(v'_i, v'_{i-1})$. Considering Sim a weighted adjacency matrix, it is possible to build a Document Similarity Graph (DSG). The idea is to assign close document identifiers to similar documents as this will likely reduce the d-gaps in common terms postings. This traversing problem can be transformed into a TSP just by considering the complement of the similarity as the weights in the edges of the DSG. The solution found by the TSP is the path that minimises the sum of the distances between documents, therefore the algorithm is an appropriate strategy to the document reassignment problem.

Silvestri et at [9] proposed a method which aimed at enhancing the clustering property of the index. Prior to reassigning, the technique computes a so called *transactional* representation for the documents, which consists in storing a 4-bytes truncated MD5 [13] digest for the terms appearing in them. Starting from that, the authors follow into two different assigning schemes, differing in the starting point of the algorithms: *top-down*, considering the whole collection and recursively splitting it, and *bottom-up* assignment, starting from a flat set of documents and extracting disjoint sequences containing similar documents, grouping them.

The previously described techniques are only approximations for solving the real problem, and have some efficiency drawbacks, as stated in [10], like computing and storing the full similarity graph or in [9] the linear dependence of time and memory usage respect to the document size.

3 Document Identifier Reassignment Through Dimensionality Reduction

The architecture to be presented here, solves the document identifier reassignment problem as a TSP like in [8] but avoiding some efficiency problems. The main idea is to reduce the input similarity matrix data via Singular Value Decomposition. That leads to the development of a new software framework, in which reordering and recompressing techniques can operate after carrying out a dimensionality reduction. This way, the memory usage by such algorithms is controlled and as it is possible to determine the total amount of memory used in each step. Moreover, by assigning document identifiers through dimensionality reduction, results are consistent between different collections and are comparable with those obtained by working with the full dimension schema, as stated by [10].

Next, the main method for effectively reducing the dimensionality by SVD is described. Section 5 gives further details on the algorithm used.

3.1 Singular Value Decomposition

Singular Value Decomposition (SVD) is a well known mathematical technique used in a wide variety of fields. It is used to decompose an arbitrary rectangular matrix into three matrices containing singular vectors and singular values. This matrices show a breakdown of the original relationships into linearly independent factors. The SVD technique is used as the mathematical base of the Latent Semantic Indexing (LSI) IR model [14].

Analytically, we start with X, a $t \times d$ matrix of terms and documents. Then, applying the SVD X is decomposed into three matrices:

$$X = T_0 S_0 D_0' \tag{2}$$

T_0 and D_0 have orthonormal columns, and S_0 is diagonal and, by convention, $s_{ii} \geq 0$ and $s_{ii} \geq s_{jj} \forall i \geq j$. T_0 is a $t \times m$ matrix, S_0 is $m \times m$ and D_0', the transposed matrix of D_0, is $m \times d$ where m is the rank of X. However it is possible to obtain a k-ranked approximation of the X original matrix by keeping the k largest values in S_0 and setting the remaining ones to zero obtaining the matrix S with $k \times k$ dimensions. As S is a diagonal matrix with k non-zero values, the corresponding columns of T_0 and rows D_0 can be deleted to obtain T, sized $t \times k$, and D', sized $k \times d$, respectively.

This way we can obtain $\hat{X}$ which is a reduced rank k approximation of X:

$$X \approx \hat{X} = TSD' \tag{3}$$

$\hat{X}$ is the closest rank k approximation of X in terms of the Euclidean or Frobenious norms, i.e. the matrix which minimises $||X - \hat{X}||_N^2$ where $|| \cdot ||_N^2$ is the involved norm.

The i-th row of DS gives the representation of the document i in the reduced k-space and the similarity matrix $\Theta(X)$ is k-approximated by $\Theta(\hat{X})$:

$$\Theta(X) \approx \Theta(\hat{X}) = \hat{X}'\hat{X} = DS^2D', \tag{4}$$

where $\hat{X}'$ is the transposed matrix of $\hat{X}$ and D' is the transposed of D.

If $D_{d\times k} = \{z_{ij}\}$ and $\{s_i\}$ is the set of diagonal elements of S, it is easy to prove that

$$\Theta(\hat{X})_{ij} = \sum_{\gamma=0}^{k-1} z_{i\gamma} z_{j\gamma} s_\gamma^2 \tag{5}$$

Therefore it is possible to calculate $\Theta(\hat{X})_{ij}$ only storing the set of k elements $\{s_i\}$ and the $d \times k$ matrix D instead of computing and writing the full rank matrix $\Theta(X)_{d\times d}$.

The output of the SVD of X, $\hat{X}$ has been used in the computation of $\Theta(\hat{X})$ (equation 4). The same result could be obtained by calculating the SVD of $\Theta(X)$ due to the uniqueness property of SVD [1]. Since SVD computes the best rank k approximation, it is proved that the best rank k approximation of $\Theta(X)$ is obtained starting from X and without the need of computing $\Theta(X)$.

3.2 Results

Applying this technique and tackling the TSP with a Greedy Nearest Neighbour algorithm (Greedy-NN), we obtain good values in compression ratios, measured in bits per document gap used. For a detailed reference of this results see [10]. Tests were driven in the LATimes and FBIS collections, which form the TREC-5 disk, and with different values of the k parameter (which reflects the matrix dimension in the reduced space).

With k=200, for the LATimes collection (FIBS collection) we achieved a 13.65% (8.02%) gain in compression ratio respect to the original document identifier order with the gamma encoding, 13.2%(8.7%) for the delta encoding, and 11.32% (5.15%) for the interpolative coding. These values are 17.67% (21.92%), 17.8%(21.1%) and 13.66% (14.58%) repectively for both collections and the three encoding schemes, respect to a random reassignment. Computing the Greedy-NN TSP with the reduced space approximation $\Theta(\hat{X})$ gives worthy compression ratios in every case. The gains in the FBIS collection are worse than the ones in the LATimes, although starting from a randomized order the result is inverted. This is the expected behaviour if the FBIS collection exhibits a better original document order. One point to remark is that even in the case of interpolative coding, where the starting point is much better, the method is able to produce gains in bits per document gap.

4 Architecture

Figure 2 describes the system built for testing this approach. Nodes represent the data and components represent the different modules deployed. A solid arrows means direct dependency between data and a module (either input or output), and dashed arrows drive the flow of the program through the different components.

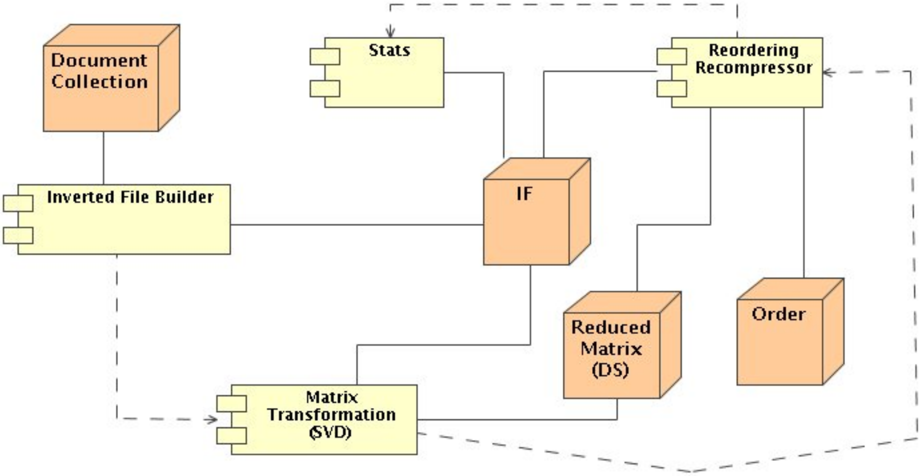


Fig. 2. Main component diagram for the software architecture

The full process is driven by a Mediator [12], which controls the interaction between the different components and reduces the dependency between the processing steps. The mediator serves as an intermediary and keeps modules from referring to each other explicitly, thereby reducing the number of interconnections. This has a purpose of generality for the architectural design, as it is a feature that facilitates the development and extension of the current software by allowing the construction of program pieces completely independent between each other. This is interesting for having different indexing, compression, dimensionality reduction, statistical and reordering modules. Extensions could be, for example, a component for building Direct Files, which are convenient for doing query expansion.

The process starts with the inversion of the text collection. The inverted file builder mechanism produces a complete version of the inverted file, considering the document identifiers in natural order, this is, as they appear originally in the collection. Also, this module outputs the X data matrix to a SVD module (see 3 for more details on this). This module produces the matrices $D_{d \times k}$ and $S_{k \times k}$ that allow the computation of $\Theta(\hat{X})$, therefore there is no longer needed to store the similarity matrix $\Theta(X)_{d \times d}$. The reassignment module uses the SVD output matrix to compute the TSP approach fully described in [10], however the open design allows the interchangeability with the other techniques introduced in 2.

The output of the TSP reassignment module is used by an inverted file recoding program which exploits the new locality of the documents to enhance d-gap compression. Finally, some statical information is taken to make suitable comparisons between compression ratios achieved by the original encoding and those obtained after reassignment.

Respect to the complexity involved, as k is a constant factor, we can conclude that the space usage of the algorithm now is $O(d)$, i.e., linear in collection size and not dependant on document size. The main difference with respect to previous implementations for the TSP technique for the document identifiers reassignment problem, is that computing the similarity between two documents d_i and d_j involves k operations $(\sum_{\gamma=0}^{k-1}(DS)_{i\gamma}(DS)_{j\gamma})$ and storing k real pointers per document, making a total of $k \times d$ for the full matrix. This representation can fit smoothly into memory by adjusting the parameter k and uses considerably less space than the original $d \times d$ matrix. Even more, the space usage can be precalculated so suitable scalable algorithms can be easily developed. Considering 32 bits per float (real number), our implementation uses $4 \times k \times d$ bytes of main memory.

5 Implementation

In this section, it is explained how the architecture was developed. The modules implemented make extensive use of design patterns: descriptions of problems that appears repeatedly and solutions to them, in such way that the solution can be applied in different situations. Design patterns are an extended technique for developing Object Oriented software providing a good balance between space and time [12].

For indexing and compressing tasks we used the MG4J [4] from the University of Milan, a free Java implementation of the indexing and compression techniques described in [7] and originally implemented in the MG software [3]. The indexing is made in three different passes to the original text of the collection, using a technique called in-memory text-based partitioning inversion [7]. As an overview, the technique builds the index entirely in main memory and avoids random accesses to disk by controlling the amount of storage needed in each step of the algorithm and using compression techniques. This way it is possible to perform an efficient map from memory to disk, avoiding seeking and swapping times. The first pass collects statistics from the text, like term document frequency and in-document term frequency, and builds a dictionary file, also known as lexicon. The second pass reads the lexicon and calculates appropriate values for compressing efficiently the data structures used in the inversion process, and allocates disk space for guarantying an optimum usage of the resources. After that, the program builds a random access in-memory inverted file for chunks of the collection, and the inverted file in disk for the whole collection. Finally, a third an optional pass over the text is done for recompressing and producing a new final inverted file.

For the SVD module we used the SVDLIBC [5], a C library based on the SVDPACKC library. It should be pointed that we needed to modify the MG4J software to output data directly to the SVDLIBC module. This module computes the singular values associated to a matrix, which can be inputted in different formats, by the algorithm *las2*, standing for Single-Vector Lanczos Method [15].

The rest of the program code (reassignment, recoding and statistical software) is written in Java, taking into account that some routines and methods are shareable between different components. Some excerpts of the written code may include modifying coding routines and generating generic libraries for graph manipulation, matrix transformations and standard data formats for matrices. As a brief example, in the figure 3 we present an usage of the strategy pattern [12] for coding routines, which was adapted for including interpolative coding [6] for document pointers.

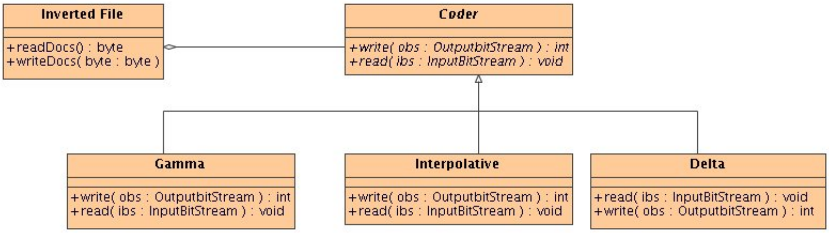

Inverted File
+readDocs() : byte
+writeDocs(byte : byte)
Coder
+write(obs : OutputbitStream) : int
+read(ibs : InputBitStream) : void
Gamma
+write(obs : OutputbitStream) : int
+read(ibs : InputBitStream) : void
Interpolative
+write(obs : OutputbitStream) : int
+read(ibs : InputBitStream) : void
Delta
+read(ibs : InputBitStream) : void
+write(obs : OutputbitStream) : int

Fig. 3. Main component diagram for the software architecture

The inverted file acts as the context for the pattern, and the individual strategies implement an interface with the coding and decoding methods. This way the coding scheme is a behaviour of the different subclasses involved in the pattern, thus, adding new coding methods is just a matter of implementing the interface with different classes and behaviours.

6 Conclusions

Previous works established that reassigning the identifiers of the documents appearing in a collection improves compression ratios of inverted files, when coding the differences between consecutive identifiers with static codes. Moreover, the work in [10] shows how the reassignment can be done effectively by making a prior dimensionality reduction of the term-document matrix. In this paper we have focused on the software architecture and the concrete design and implementation issues. We underlined the importance of reusing software pieces and making use of good design practises. These methodologies must have an impact not only in a corporate environment but also for developing software tools with research purposes, where software lies above well-founded mathematical concepts. Therefore, working with a reusable framework for Information Retrieval research avoids a lot of unnecessary work when reusing software pieces, and facilitates the deployment and development of more robust software kits that can be used by the IR community.

Acknowledgements

The work reported here was co-founded by the "Secretaría de Estado de Universidades e Investigación" and FEDER funds under research projects TIC2002-00947 and Xunta de Galicia under project PGIDT03PXIC10501PN.

References

1. B. T. Bartell, G. W. Cottrel and R. K. Belew. Latent Semantic Indexing is an optimal special case of Multidimensional Scaling. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 161-167, 1992.
2. D. Blandford and G.Blelloch. Index compression through document reordering. In *Proceedings of the IEEE Data Compression Conference (DCC'02)*, pp. 342-351, 2002.
3. `http://www.cs.mu.oz.au/mg/` Managing Gigabytes.
4. `http://mg4j.dsi.unimi.it/` MG4J (Managing Gigabytes for Java).
5. `http://tedlab.mit.edu/~dr/SVDLIBC/` SVDLIBC.
6. A. Moffat, A. Turpin. *Compression and Coding Algorithms*, Kluwer 2002.
7. I. H. Witten, A. Moffat and T. C. Bell. *Managing Gigabytes - Compressing and Indexing Documents and Images*, 2nd edition. Morgan Kaufmann Publishing, San Francisco, 1999.
8. W.-Y. Shieh, T.-F. Chen, J. J.-J. Shann and C.-P. Chung. Inverted file compression through document identifier reassignment. *Information Processing and Management*, 39(1):117-131, January 2003.
9. F. Silvestri, S. Orlando and R. Perego. Assigning identifiers to documents to enhance the clustering property of fulltext indexes. In *Proceeding of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 305-312, 2004.
10. R. Blanco, A. Barreiro. Document identifier reassignment through dimensionality reduction. In *Proceeding of the 27th European Conference on IR Research, ECIR 2005*, LNCS 3408, pp. 375-387, 2005.
11. G. Karypis and V. Kumar. A fast and high quality multilevel scheme for patitioning irregular graphs. Technical Report TR 95-035, 1995.
12. E. Gamma, R. Heml, R. Johnson, J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software* Addison Wesley 1995.
13. R. Rivest, RFC 1321: The md5 algorithm.
14. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer and R. Harshman. Indexing by Latent Semantic Analysis. In *Journal of the American Society for Information Science*, 41(6):391-407, 1990.
15. M. Berry. Large Scale Singular Value Computations. In *International Journal of Supercomputer Applications*. 6:1, (1992), pp. 13-49

An Ontology for Reusing Synthetic Tasks

Abraham Rodríguez-Rodríguez and Francisca Quintana-Domínguez

Dept. of Computer Science,
University of Las Palmas de Gran Canaria,
Edificio de Informática, Campus de Tafira,
35017 Las Palmas de Gran Canaria,
Canary Islands- Spain
{arodriguez, fquintana}@dis.ulpgc.es

Abstract. Problem Solving Methods are valuable structures that facilitate reusing components for building Knowledge-Based Systems. We explain the vocabulary used to describe a synthetic PSM which was successfully applied to four different domains. The main classes of the ontology will be described, in particular those required to map the concepts of the domain with the PSM method.

1 Context

Knowledge-Based Systems (KBS) development must overcome multiple difficulties, making it a high cost technology with a large number of development failures. Current development methodologies, such as VITAL [6] , MIKE [1] or CommonKADS [15], stress the use of generic components to facilitate knowledge reusing, Problem Solving Methods (PSM) and code. Ontologies also play an important role in this process because they normalize the concepts and relations involved in the task and/or the domain that they describe.

Among the different definitions of the ontology term, the one given by Gruber [8] in 1993, and later completed by Studer et al. [16] is the most accepted: *An Ontology is a formal, explicit specification of a shared conceptualization.* An ontology is usually represented by a taxonomy of hierarchically organised terms, together with transversal relations among them. Its objective is to explicit an abstract model of some phenomenon in the world *(conceptualization)*. '*Explicit specification'* means that the different elements must be clearly represented, avoiding unclear, incomplete or inconsistent definitions. It is likely that the Ontologies area will take advantage of the increasing interest that is generating the semantic web, because one of the main elements that will support the Semantic Web success is the availability of consensual ontologies to facilitate the inter-operation of automated services and processes.

PSMs are high-level structures that indicate us how to solve a generic task such as diagnosis, assessment, planning, etc. They provide us with a set of inferences and knowledge descriptions that allow us to build a solution without any reference to specific domain elements. Thus, PSMs work at the knowledge-level as defined by Newell in [12]. Clancey [5], McDermott [10] with his work on Role-Limiting Methods and Chandrasekaran's Generic Tasks [4] enriched and evolved the notion of PSM so these ideas were assumed by methodologies such as CommonKADS [15] or MIKE

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 263 – 268, 2005.
© Springer-Verlag Berlin Heidelberg 2005

[1]. These methodologies consider the PSMs essential structures as they are useful for controlling the methodological activities that must be carried out to build the expertise model. PSMs claim to be useful pieces to build KBS because they enforce the reutilization of proven problem solving strategies, and they also facilitate organising and structuring a knowledge base through the use of knowledge structures (sometimes referred as knowledge-roles).

We adhere to the CommonKADS point of view regarding the PSMs because we intent to build several applications, and this methodology gives us support for all the development stages, enforcing, at the same time, the reusability of the components. Moreover, the model-oriented policy of CommonKADS facilitates us focusing on the knowledge modelling, setting aside other development issues. CommonKADS identifies three types of knowledge: domain, task and inference knowledge. While the first one is application specific, the other two types are easily re-usable among applications. Task knowledge is based on a generic-task classification, where tasks are hierarchically organised and only have high level descriptions. Task-method knowledge relates these task types to specific algorithms (PSMs) that specify how to solve them. Although CommonKADS suggests specific templates (PSMs) for each task type, we have developed and applied the same PSM on several types of synthetic tasks over different domains.

2 Synthetic Tasks and Constraint Satisfaction Techniques

CommonKADS and authors like [11] or [7] suggest a shared characterisation of all synthetic tasks by using some terms such as requirements, constraints, preference, etc. These terms are similar to those used by the Constraint Satisfaction (CS) techniques, where a problem consists of [2]:

1. a set of variables,
2. for each variable, a finite set of possible values,
3. a set of constraints restricting the values that the variables can simultaneously take.

Synthetic tasks are usually NP-hard (computationally intractable) problems, and the CS field provides us with a large amount of techniques to solve them. However, selecting the technique that best fits the characteristics of the problem that is being modelled is a hard problem itself. Programmers usually implement multiple techniques and adopt the one with the best results, balancing the trade off between the computational cost of the algorithm and the overall gain in terms of execution time or quality of the solutions generated. In all cases, CS techniques are highly influenced by the codification of good domain dependent constraints that limit the solution space to be explored.

We have adopted the technique known as Forward Checking, because it provides us with the needed simplicity to transform it into a PSM and it is easily extensible with some other techniques such as consistency techniques and variable/value ordering. Moreover, some authors (see i.e. Bartak excellent online tutorial) recognises that Forward Checking can obtain better results than some others more elaborated (and computationally costly) algorithms, like full-look-ahead, if it is tuned properly.

3 Test Applications

The purpose of the developed applications is not to build efficient code but to determine the actual benefits of the reusability of components. These domains (see Table 1) correspond to different task types according to the definition made in the CommonKADS methodology. Their main characteristics are:

- Educational Centre timetable. It relates courses, subjects, teachers, classrooms and laboratories to configure a coherent timetable, considering restrictions on the teachers workload, availability of equipment in labs, size of classrooms, etc.
- Employee timetable. It has been implemented to schedule the working hours of employees for a well-known fast-food restaurant. It considers multiple contract types and limitations, expected work demand, specialisation of employees on specific tasks, etc.
- PCs configuring system. It helps users without computer knowledge to build up a PC starting from requirements such as intended use, cost, availability of components, etc.

Airlines line programming. It was the hardest application. It assigns airplanes to flights for a small-sized airline company. It must consider facts such as flight data (origin, destination, flight time, etc), number of passengers, etc. A solution is valid when all the flights are operated.

Table 1. We use the task types as they are defined in CommonKADS

Application	Synthetic Task Type
Educational centre timetable	Scheduling
Employee timetable	Scheduling
PC configuring system	Configuration design
Airline lines programming	Planning

4 Mappings Between the PSM Ontology and the Test Applications

Figure 1 shows the PSM for the Forward Checking algorithm. The algorithm works building solutions incrementally, assigning values to variables and forward checking inconsistencies of the current partial solution with those variables that have not been assigned any value yet. For an extended description of this PSM see [14].

Mapping the classes of the application domain to the ones defined by the PSM is one of the main difficulties for reusing components. We have opted to reuse as much as possible the structure given in the task model of CommonKADS, adding some details to facilitate the link between the domain and the PSM vocabularies. Figure 2 shows the graphical representation of some classes of the PSM ontology together with some of its attributes and relations. Table 2 shows part of this information including some remarking relations using OWL functions. The main classes of the ontology are those of the task model of CommonKADS: task, task-method, and role. The first one consist on the high-level description of the class. The task-method class comprises the methods used to solve a certain task through the realizes attribute.

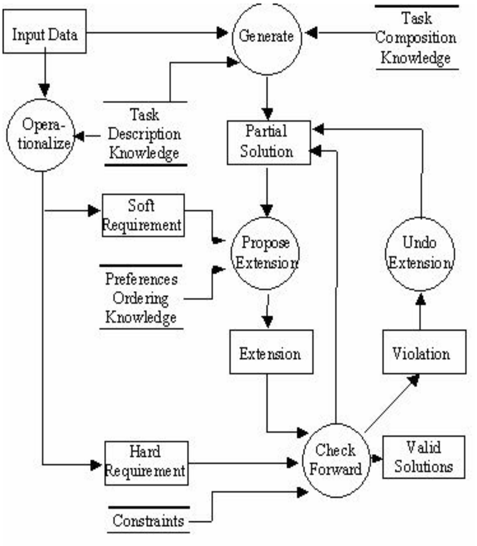


Fig. 1. PSM for the Forward Check method

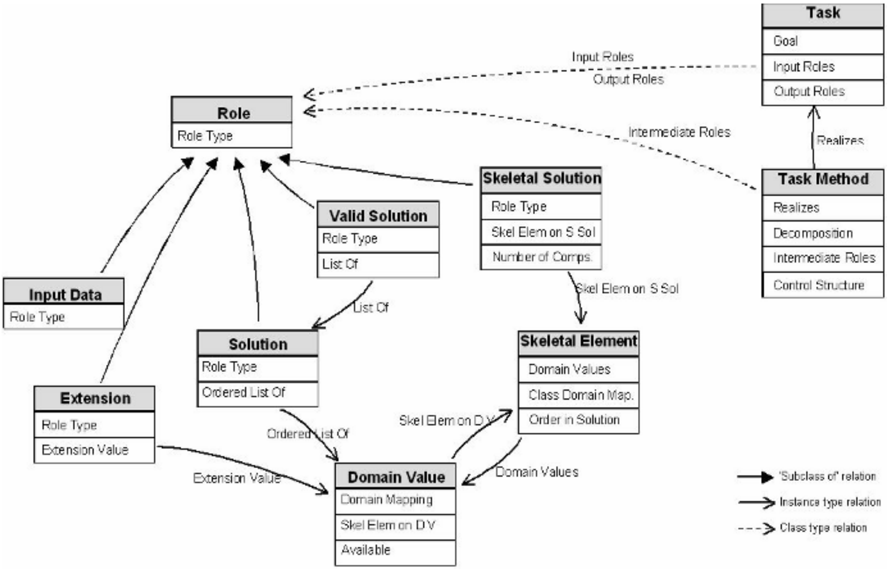


Fig. 2. Graphical representation of relevant classes of the PSM ontology

Roles shown in Figure 2 correspond to the dynamic data used in the PSM. The mapping between the domain and PSM is made explicit using the Skeletal_solution, Skeletal_element and Domain_value classes. Skeletal_solution is an aggregation of Skeletal_elements, where an instance of Skeletal_element will be created for each of the variables that take part on the solution structure. There will also be an instance of the Domain_value class for each domain value of each Skeletal_element (for each possible value of each variable in the solution structure).

Table 2 summarizes the ontological description of the classes of Figure 2. We have used the Protegé ontology editor [13] to validate it using the OWL [3] plugin. It shows the relevant characteristics for each modelled class. We can observe that the definition of the domain_values attribute as inverse_functional ensures that each value for every domain will be unique, and, at the same time, they will be accessible through the inverse relation included on the definition of the Domain_value class. However, it is worth mentioning that the range definition for attributes such as Output_role implies that the ontology is written in the OWL-full sublanguage, making it difficult to be processed by reasoners, as stated in [9].

Table 2. Classes from the PSM Forward Checking ontology

Class	Slot	Range	Restrictions
Task			
	Goal	string	
	Input roles	Class: role	
	Output roles	Class: role	
Task Method			
	Realizes	instance: task	
	Intermediate roles	Class: role	
	Decomposition	Instance: task Union Instance: inference	
	Control_structure	String	
Skeletal_ Solution			subclassof Role
	Role_Type		owl:hasvalue dynamic
	Number_of_Compnents	int	
	Skeletal_elements _on_solution	instance: Skeletal_ element	min cardinality 2
Domain_ value			
	Available	int	owl:one of (0,1)
	Domain_mapping	instance:(domain class)	owl: functional
	Skeletal_element on_Domain	instance:Skeletal_element	owl: functional Inverse: Domain_values
Skeletal_ element	Class_domain_mapping	Class: (domain class)	owl: cardinality 1
	Domain_values	instance: domain_value	owl: inverse functional Inverse: Skeletal_element on_Domain
	Order in solution	int	owl:functional

5 Conclusions

In this paper we have described our experience on developing several synthetic applications trying to reuse as much components of knowledge as possible. We have derived a PSM that was successfully applied over four quite different applications. This process is partly supported on the standardization of the vocabulary used to describe the PSM, which is partly described in the CommonKADS methodology. We have included some basic classes that facilitate the mapping with the entities of the application domain.

According to the people participating in this initiative, the reuse of the PSM and its associated executable code means a significant improvement in the development effort (only for the expert module). The profit is even larger depending on some other factors such as the complexity of the application being modelled. The PSM Ontology allows us to prepare structured interviews with the human experts in order to acquire specific knowledge for that role. The experts easily understood the PSM diagram and collaborated actively providing valid rules that were later included in the system.

References

1. Angele, J., Fensel, D., Studer, R. Domain and Task Modelling in MIKE. In Suthliffe, A et al. (eds). Domain Knowledge for interactive system design. Chapman & Hall. 1996
2. Baptiste, P., Le Pape, C., and Nuijten, W., Constraint-Based Scheduling. Applying constraint programming to scheduling problems. Kluwer Academic Publishers. 2001
3. Bechhofer, S., Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D., Patel-Schneider, P., Stein, L.. OWL Web Ontolgoy Language Reference. W3C Recomentation. Editors Mike Dean and Guus Schreiber. http://www.w3.org/TR/2004/REC-owl-ref-20040210/, 2004
4. Chandrasekaran, B. Generic Tasks in Knowledge-based reasoning: High level building blocks for system design. IEEE expert 1986; 1(3): 23-30.
5. Clancey, WJ. Heuristic Classification. Artificial Intelligence 1985; 27: 289-350
6. Domingue, J., Motta, E., Watt, S. The emerging VITAL workbench. In Aussenac et al. editor, EKAW'93 Knowledge Acquisition for Knowledge-Based Systems. Lecture Notes in Artificial Intelligence, LNCE723, Berlin Germany, 1993, Springer-Verlag.
7. Fensel, D., Motta, E. Decker, S., and Zdrahal, Z. Using Ontologies for Defining Tasks, Problem-Solving Methods and Their Mappings. Proceedings of European Knowledge Acquisition Workshop. Lecture Notes in Artificial Intelligence (LNAI), Springer-Verlag, 1997.
8. Gruber, T. A translation Approach to portable ontology specifications. Knowledge Acquisition. Vol. 5. 1993. 199-220
9. 9.Horridge, M. A Practical Guide to building OWL Ontologies with the Proté gé -OWL Plugin. http://protege.stanford.edu. 2004
10. McDermott, J. Preliminary Steps towards a taxonomy of problem-solving methods. In S. Marcus (ed.). Automating Knowledge Acquisition for Expert Systems. 1988; 225-256
11. Motta, E., Rajpathak, D., Zdrahal Z., Roy, R. The Epistemology of Scheduling Problems. Proceedings of the 15th European Conference on Artificial Intelligence, F. van Harmelen (editor), IOS Press, Amsterdam, 2002.
12. Newell, A. The Knowledge Level. Artifificial Intelligence 1982; 18: 27-127
13. PROTÉGÉ ontology editor. developed by Stanford Medical Informatics at the Stanford University School of Medicine. http://protege.stanford.edu.
14. Rodríguez-Rodríguez, A., Quintana-Domí nguez, F., Alemá n-Flores, M. Reusing Problem Solving Methods in Synthetic Tasks. In proceedings of the International Conference on Information and Knowledge Engineering. IKE 2003.
15. Schreiber, Akkermans, Anjewierden, de Hoog, Shadbolt, Van de Velde, Wielinga. Knowledge Engineering and Management. The CommonKADS Methodology. The MIT Press. 1999.
16. Studer, Benjamins, Fensel. Knowledge Engineering: Principles and Methods. Data and Knowledge Engineering. 25 (1998) 161-197

A Tractable Subclass of Fuzzy Constraint Networks

Alfonso Bosch[1], Francisco Guil[1], and Roque Marin[2]

[1] Dpto. de Lenguajes y Computación, Universidad de Almeria, 04120 Almeria (Spain)
`{abosch, fguil}@ual.es`
[2] Dpto. de Ingenieria de la Informacion y las Comunicaciones,
Universidad de Murcia, Campus de Espinardo, 30071 Espinardo (Murcia, Spain)
`roque@dif.um.es`

Abstract. The Fuzzy Constraint Networks model, a generalization of the Disjunctive Temporal Fuzzy Constraint Networks, is a framework that allows representing and reasoning with fuzzy qualitative and quantitative complex constraints. However, its general complexity is exponential, and we need to find tractable subclasses. In this paper we propose two algorithms to deal with a tractable subclass named Series-Parallel Fuzzy Constraint Networks.

1 Introduction

Fuzzy Temporal Constraint Networks model (FTCN), introduced in [6], allows expressing simple constraints, representing them by means of a convex and normalized possibility distribution. Fuzzy temporal constraints allow combining precise, imprecise, qualitative and quantitative information. Then, this model is suitable for temporal reasoning in domains where the combination of such constraint types is required. A fuzzy model allows intermediate consistency degrees, and to quantify the possibility and necessity of a relationship or query. In addition, constraint propagation reduces one of the drawbacks associated with fuzzy reasoning, the degradation of distributions when chaining fuzzy rules (reduction of the core, and enlargement of the support of the possibility distributions).

In certain tasks, such as planning or scheduling, a more general model with disjunctions is needed. Then, the FTCN model is enhanced, defining a constraint with a finite set of possibility distributions, normalized and convex, obtaining the Disjunctive Fuzzy Temporal Constraint Networks (DFTCN) [1]. This model extends the TSCP framework proposed by Dechter [3], allowing constraints such as "Irrigation is much before or a little after than Treatment", and subsumes the Vilain & Kautz Point Algebra [9]. This framework allows representing all the possible relationships between time points, between intervals and between time points and intervals, and their disjunctions.

The main drawback of DFTCN is its computational inefficiency, because generally these networks are non-decomposable, needing backtracking to find a solution [1]. Determining the consistency and computing the minimal network are also exponential. With small problems, this is not a drawback, but in order to generalize the use of the model in a general scope, it would be interesting to find tractable subclasses, as series-parallel networks [2].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 269 – 274, 2005.
© Springer-Verlag Berlin Heidelberg 2005

In this paper we use a generalization of DFTCN, named Fuzzy Constraint Networks (FCN) [2], and we present two efficient algorithms, SP^+ and SPR, to decide if a FCN is consistent, series-parallel and obtain its equivalent path-consistent (and minimal) network, using the information of the original problem constraints.

The remainder of this paper is organized as follows. Section 2 describes the FCN model; Section 3 describes a tractable subclass named Series-Parallel Networks; Section 4 presents two efficient algorithms to deal with this Series-Parallel Networks and Section 5 summarizes the conclusions and presents the future work.

2 Fuzzy Constraint Networks

A Fuzzy Constraint Network (FCN) L^d consists of a finite set of n+1 variables $X_0, \ldots, X_n$ (X_0 as origin for problem variables), whose domain is the set of real numbers $\boldsymbol{R}$, and a finite set of disjunctive binary constraints L^d_{ij} among these variables. X_0 is a variable added to use only binary constraints, and it can be assigned to an arbitrary value (for simplicity's sake, this value is usually 0).

A disjunctive binary constraint L^d_{ij} among variables X_i, X_j is defined with a finite set of possibility distributions, $\{\pi^1_{ij}, \pi^2_{ij}, \ldots, \pi^k_{ij}\}$ normalized and convex [5], defined over the set of real numbers $\boldsymbol{R}$; for $x \in \boldsymbol{R}$, $\pi_m(x) \in [0,1]$ represents the possibility that a quantity m can be precisely x.

A value assignation for variables X_i, X_j, $X_i=a$; $X_j=b$, $a, b \in \boldsymbol{R}$, satisfies the constraint L^d_{ij} iff it satisfies one of its individual constraints:

$$\exists \pi^p_{ij} \in L^d_{ij} \,/\, \pi^p_{ij}(b-a) > 0 \tag{1}$$

The maximum possibility degree of satisfaction of a constraint L^d_{ij} for an assignment $X_i = a$, $X_j = b$ is

$$\sigma^{\max}_{ij}(a,b) = \max_{1 \le p \le k} \pi^p_{ij}(b-a) \tag{2}$$

A constraint L^d_{ij} among variables X_i, X_j defines a symmetric constraint L^d_{ji} among X_j, X_i, and the lack of a constraint is equivalent to the universal constraint π_U. A FCN can be represented with a directed graph, where each node corresponds to a variable and each arc corresponds to a constraint between the connected variables, omitting symmetric and universal constraints. The set of possible solutions of a FCN L^d is defined as the fuzzy subset from $\boldsymbol{R}^n$ associated to the possibility distribution given as:

$$\pi_S(v_1, \ldots, v_n) = \min_{\substack{0 \le i \le n \\ 0 \le j \le n}} (\sigma^{\max}_{ij}(v_i, v_j)) \tag{3}$$

An n-tuple $V = (v_1, \ldots v_n) \in \boldsymbol{R}^n$ of precise values is an σ-possible solution of a FCN L^d if $\pi_S(V) = \sigma$. We say that a FCN L^d is consistent if it is 1-consistent, and it is inconsistent if it does not have any solution.

Given a FCN L^d, it is possible to find out several networks which are equivalent to L^d. We can obtain this networks using the composition and intersection operations, defined in [1] for temporal reasoning. Among all the equivalent networks, there is always a network M^d that is minimal. This network contains the minimal constraints. If

M^d contains an empty constraint, L^d is inconsistent. If p if the maximum of possibility distributions in each constraint, and the network has q disjunctive constrains and n variables, then the minimal network M^d of a FCN L^d can be obtained with a complexity $O(p^q n^3)$, where n^3 is the cost of solving each case non disjunctive FCN [7]. Due to this exponential complexity, we need to find a more practical approach.

3 Series-Parallel Fuzzy Constraint Networks

A network is series-parallel [8] in respect to a pair of nodes *i,j* if it can be reduced to arc (*i,j*) applying iteratively this reduction operation: a) select a node with a degree of two or less; b) remove it from the network; c) connect its neighbours. A network is series-parallel if it is series-parallel in respect to every pair of nodes.

The basic algorithm for checking if a network is series-parallel has an $O(n^3)$ complexity, and there is a more efficient algorithm that checks this property with an $O(n)$ complexity [10], applied to fault-tolerant networks (IFI networks). In [2], we introduced the SP algorithm, a variant of the later approach for constraint networks.

Series-parallel networks present some interesting properties. If a network is series-parallel, the path-consistent network is the minimal network, although the intersection and composition operations are non-distributive [8]. As a subproduct of checking whether a network is series-parallel, a variable ordering is obtained when deleting the nodes. Applying directional path-consistency (DPC) algorithm [4] in the reverse order, a backtrack-free network is obtained and the minimal constraint between the first two variables of the ordering too. This can be interesting when we need only to compute a minimal constraint for two variables, and not the minimal network. In addition, if the network is series-parallel, we can decide absolutely whether the network is consistent, by applying DPC algorithm in the reverse order. Using all these properties, we can say that series-parallel Fuzzy Constraint Networks are a tractable subclass of Fuzzy Constraint Networks, because we can manage it without backtracking.

4 Algorithms for Series-Parallel Fuzzy Constraint Networks

The SP algorithm decides whether a network is series-parallel using the graph topology. After the SP test, we need to use another algorithms to manage the network. But if we enhance SP algorithm to use constraint information, we can process the constraints, deciding simultaneously whether the network is series-parallel or not.

Our initial proposal is to include a constraint relaxation operation when a node or variable of degree two is eliminated. This operation updates a constraint, deleting the values disallowed by a path of length two, ensuring that the constraint will be path-consistent in respect to this path.

Working in this direction, we propose the algorithm SP^+, which can decide with an $O(n)$ complexity whether a network is inconsistent if it detects an empty constraint. If an empty constraint is not detected and the algorithm decides that the network is series-parallel, the network is consistent. As a subproduct, the final obtained constraint is the minimal constraint among the two lasting variables, and the equivalent network obtained with the computed constraints is decomposable in the reverse variable reduction ordering. This is the code of SP^+ algorithm:

```
SP+ Algorithm
begin
  for each i=0..n Compute-degree (i)
  NodeQueue = {nodes with degree 1 and 2}
  while (NodeQueue <> ∅ and |V| > 2)
    begin
    node=Get(NodeQueue)
      if Degree(node) = 2 then    $L_{ij} \leftarrow L_{ij} \dot{\cap} L_{ik} \dot{\oplus} L_{kj}$
      if L_ij = ∅ then exit
        $V \leftarrow V - \{node\}$
        if Degree(node)=1 then Degree(Neighbour(node)) --
        if Degree(Neighbour(node)) = 2 then
              Put(NodeQueue,Neighbour(node))
            else if Connected(Neighbours(node)) then
              Degree(Neighbour(node)) --
        if Degree(Neighbours(node)) = 2 then
              Put(NodeQueue, Neighbours(node))
            else  $E \leftarrow E +$ {ArcNeighbours(node)}
   end
 if (NodeQueue = ∅ and |V| > 2) then
            exit ("Network is not series-parallel")
end
```

Next, we will enumerate the properties of the algorithm.

Lemma 1.- If the series-parallel reduction algorithm stops because it computes an empty constraint, the network is inconsistent.

Lemma 2.- If the series-parallel reduction algorithm stops without detecting an empty constraint, and it decides that the network is series-parallel, then the network is consistent.

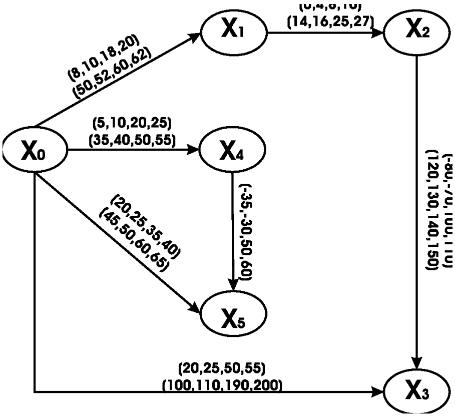


Fig. 1. Example of network to be reduced with SP^+

Lemma 3.- If series-parallel reduction algorithm terminates without detecting an empty constraint and it decides that the network is series-parallel, then the final constraint obtained is the minimal constraint among the two lasting variables.

Figure 1 shows a FCN network to be reduced with SP^+. Figure 2 shows the reduction process and refined constraints obtained.

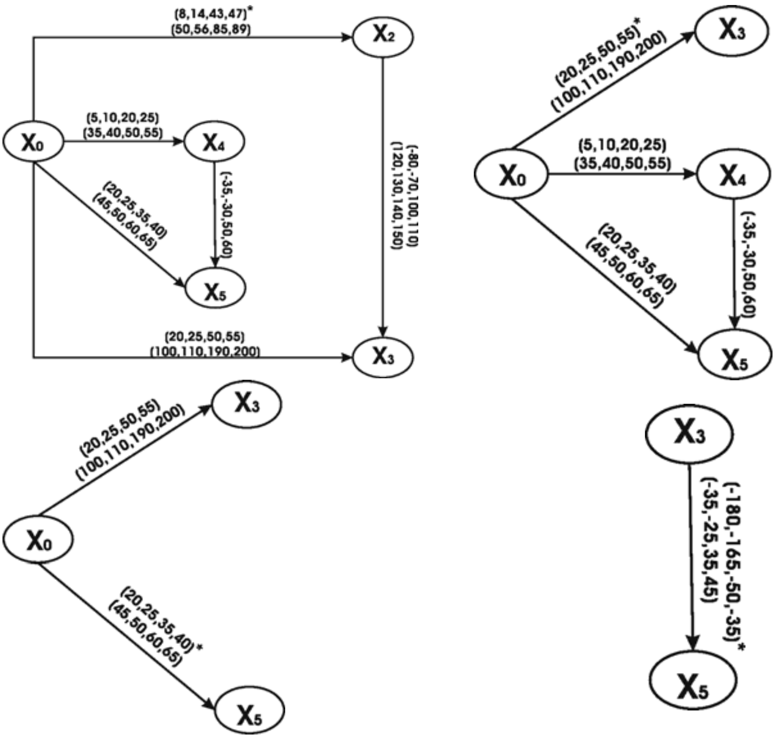


Fig. 2. Example of reduction process

Based on the results of SP^+, we can propose a linear rebuilding algorithm SPR, which obtains a path-consistent network respecting the original network topology. The obtained network is not the full path-consistent or minimal network, because it is not a complete graph.

5 Conclusions and Future Work

In this paper, we have defined Series-Parallel Fuzzy Constraint Networks as a tractable subclass of Fuzzy Constraint Networks. We have presented the SP^+ linear algorithm for checking simultaneously if a constraint network is series-parallel and it has a solution, obtaining a minimal constraint among two variables too.

Using the results of SP$^+$, we can write a linear rebuiding algorithm for obtaining a path-consistent network, with the original network topology.

As future work, we can highlight the evaluation of these techniques and its comparation versus other approaches for series-parallel networks, the implementation of a linear algorithm for obtaining the minimal network, and the search of another FCN tractable subclasses and decomposition methods.

References

1. A. Bosch, M. Torres, and R. Marín. Reasoning with Disjunctive Fuzzy Temporal Constraint Networks. In *Proc. of 9th Int. Symposium on Temporal Representation and Reasoning (TIME'02).* IEEE Computer Society Press, pages 36-43, 2002.
2. A. Bosch, F. Guil, C. Martinez, and R. Marín. Series-Parallel and Tree-Decomposition Approaches for Fuzzy Constraint Networks. In *Proc. of the 8th Ibero-American Conference on Artificial Intelligence (IBERAMIA'02).* LNAI, 2527:275-284, 2002.
3. R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49:61-95, 1991.
4. R. Dechter, and J. Pearl, "Network-based heuristics for constraint-satisfaction problems", *Artificial Intelligence*, 34, Elsevier, 1987, pp. 1-38
5. D. Dubois, H. Prade, *Possibility Theory: An approach to computerized processing of uncertainty*, Plenum Press, New York, 1988.
6. R. Marin, S. Barro, A. Bosch, and J. Mira. Modeling time representation from a fuzzy perspective. *Cybernetic & Systems*, 25(2):207-215, 1994.
7. R. Marín, M. Cardenas, M. Balsa, and J. Sanchez, "Obtaining solutions in fuzzy constraint networks", *Int. Journal of Approximate Reasoning*, 16, Elsevier, 1997, pp. 261-288.
8. U. Montanari, "Networks of constraints: fundamental properties and applications to picture processing", *Information Science*, 7, 1974, pp. 95-132.
9. E. Schwalb, and L. Vila, "Temporal Constraints: A Survey", *Constraints* 3 (2/3), 1998, pp. 129-149.
10. J.A. Wald, and C.J. Colburn, "Steiner Trees, Partial 2-Trees and Minimum IFI Networks", *Networks,* 13, 1983, pp. 159-167.

Parallel State Space Generation and Exploration on Shared-Memory Architectures

Milan Češka, Bohuslav Křena, and Tomáš Vojnar

Faculty of Information Technology, Brno University of Technology,
Božetěchova 2, 612 66 Brno, Czech Republic
`{ceska, krena, vojnar}@fit.vutbr.cz`

1 Introduction

Model checking is a technique for an automated formal verification of correctness of various classes of systems. Compared to the more traditional approaches to this problem based on simulation and testing, the main advantage of model checking is that the desirable behavioural properties of the considered system are checked in such a way that all the reachable states which may affect the given property are guaranteed to be covered. The disadvantage of model checking is that it may have very high computational time and space requirements. That is why various ways of dealing with the computational complexity of model checking are sought.

The systems to be verified may be described using various languages. Here, we consider the systems to be described by the PNtalk language based on Object-Oriented Petri Nets (OOPNs). PNtalk [6] has been developed at the Brno University of Technology as a tool suitable for modelling, prototyping and rapid development of concurrent and distributed software systems.

In the paper, we discuss possibilities of parallel OOPN state space generation and exploration on shared-memory architectures. The goal is to combat the high time complexity of state spaces-based verification methods.

2 Object-Oriented Petri Nets

The OOPN formalism is characterized by a Smalltalk-like object orientation enriched with concurrency and polymorphic transition execution, which allows for message sending, waiting for and accepting responses, creating new objects, and performing computations.

OOPNs are based on active objects whose internal activity is described by high-level Petri nets that are called object nets. The objects communicate via message sending. Asynchronous reactions of objects to incoming message are described by method nets. Method nets are also high-level Petri nets and each of them has a set of parameter places and a return place. Method nets can access places of the corresponding object net. Both object nets and method nets are dynamically instantiable. Objects can also communicate synchronously via synchronous ports that resemble special transitions of objects nets, which can be fired only when they are activated from some classical transition. A simple OOPN model in Fig. 1 demonstrates modelling by OOPNs on a simple example of dealing with the stack data structure.

The latest research in the area of OOPNs shows that due to their features, OOPNs are very suitable for describing open and reflective systems [7].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 275–280, 2005.
© Springer-Verlag Berlin Heidelberg 2005

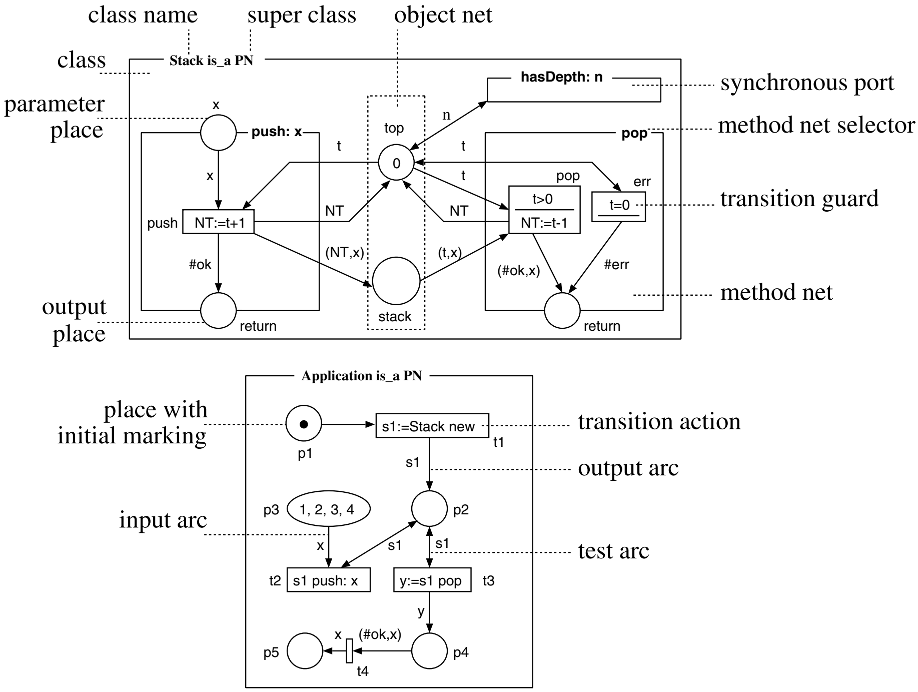


Fig. 1. A simple OOPN model

3 A Parallel OOPN State Space Generator

The main problem of the model checking methods is the so-called *state space explosion*—the number of states grows exponentially with the size of the model. Thus, it is difficult or practically impossible to apply these methods directly to large systems. We can identify two main approaches for dealing with the problem. The first class of these methods consists in sophisticated generation and exploration of the state spaces while the second one tries to exploit more powerful computer architectures. In the context of OOPNs, we have explored the first approach, e.g., in [4,9,5]. Here, we consider the second possibility. Namely, we discuss the techniques behind a parallel state space generator of OOPNs which we have developed for architectures with shared memory using the Java programming language and the JOMP tool [2].

The main task in programming parallel applications for shared-memory architectures is to achieve a good load balance among multiple computation threads taking into account their need to synchronize when accessing shared data. In our case, we need to deal with three shared data structures—(1) a hash table, which is used for storing the state space such that fast searching of states is possible when we need to know whether a certain state has already been found or not, (2) a list of non-explored states that contains states whose successors have not been explored yet, and (3) a marking pool which is a special pool for storing decomposed markings.

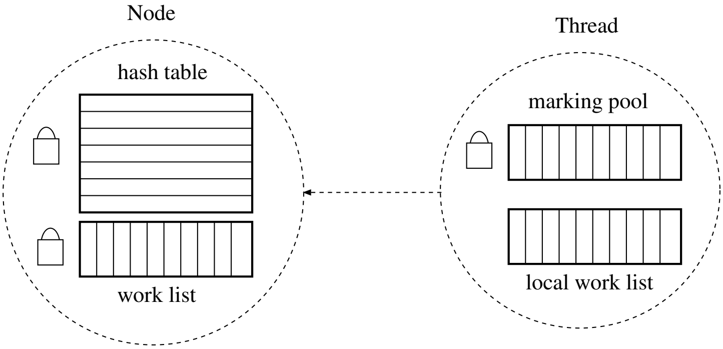


Fig. 2. Data Distribution

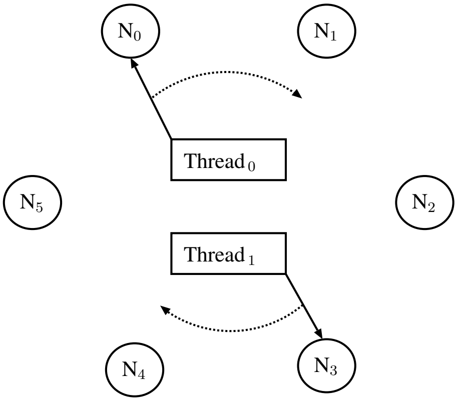


Fig. 3. Work Seeking

In order to minimize conflicts among threads trying to access the hash table, we have divided the table into several parts. In addition, we have divided also the list of non-explored states (also called as the work list) in such a way that a newly generated state is put to the unique part of the work list associated with the part of the hash table to which the state is stored. Moreover, we let the thread that is responsible for handling a certain part of the hash table use a private work list when processing the states to be explored. Finally, to further improve the performance, we create a copy of the marking pool for each thread. The distribution of data we thus obtain is shown in Fig. 2.

As we have already said, one of the crucial issues for efficiency of parallel applications is the quality of load balance among threads. We have started with one data node (i.e. a fraction of the hash table) for each thread. However, this did not work well. At first, we are not able to divide the hash table to parts with the same number of states in advance. Moreover, for a good load balance, it is necessary that each thread has some work (i.e. states to explore) during the whole production run, which is impossible to predict and guarantee in advance too.

Therefore, we have divided the hash table and the work list to more parts than we have threads. We find by experiments that the best performance is reached when three times more nodes than threads are created. Then, threads switch among nodes and try to find some work to do as it is shown in Fig. 3 for the case of two threads.

4 Experimental Results

In this section, we show parallel speedups achieved using all the optimizations discussed in the previous section. For our experiments and measurements, we used two servers, namely, a Sun Fire 15k server (52 processors UltraSPARC III, heartbeat 900 MHz, and 52 GB of memory, provided by the Edinburgh Parallel Computing Centre) and a Sun Enterprise 450 server (4 processors Sun UltraSPARC-II, heartbeat 400 MHz, and 4 GB of memory, provided by the Brno University of Technology).

In the following tables, we show average values for three successive production runs in order to statistically minimize the measuring error. The simple OOPN model called *life model* [8] was used in all the measurements discussed here.

The average execution times for the server Sun Fire 15k are listed in Tab. 1 while the corresponding speedups can be found in Tab. 2. The values in the left columns were obtained using the sequential garbage collector (SGC) while the values in the right ones were obtained using the parallel garbage collector (PGC). The average execution times for the server Sun Enterprise 450 are listed in Tab. 3 while the corresponding speedups can be found in Tab. 4.

Table 1. Average execution times for the Sun Fire 15k server

Threads	**Number of Unique States**			
	102 340		**400 995**	
	SGC	**PGC**	**SGC**	**PGC**
1	26,50 s		105,68 s	
2	20,08 s	20,85 s	78,73 s	90,07 s
4	13,61 s	13,50 s	59,53 s	58,04 s
8	9,82 s	9,20 s	49,50 s	38,88 s
12	9,21 s	8,80 s	42,76 s	32,52 s
16	8,77 s	8,31 s	44,57 s	32,30 s

Table 2. Average speedups for the Sun Fire 15k server

Threads	**Number of Unique States**			
	102 340		**400 995**	
	SGC	**PGC**	**SGC**	**PGC**
1	1,00		1,00	
2	1,32	1,27	1,34	1,17
4	1,95	1,96	1,78	1,82
8	2,70	2,88	2,14	2,72
12	2,88	3,01	2,47	3,25
16	3,02	3,19	2,37	3,27

Table 3. Average execution times for the Sun Enterprise 450 server

Threads	Number of Unique States				
	23 426	**176 851**	**585 276**	**1 373 701**	**2 667 126**
1	12,5 s	81,7 s	264,6 s	656,8 s	2 571 s (42 min 51 s)
2	8,1 s	51,0 s	180,0 s	436,8 s	1 874 s (31 min 14 s)
3	6,7 s	39,6 s	140,3 s	340,7 s	1 642 s (27 min 22 s)
4	6,6 s	35,8 s	126,3 s	304,5 s	1 517 s (25 min 17 s)

Table 4. Average speedups for the Sun Enterprise 450 server

Threads	Number of Unique States				
	23 426	**176 851**	**585 276**	**1 373 701**	**2 667 126**
1	1,00	1,00	1,00	1,00	1,00
2	1,54	1,60	1,47	1,50	1,37
3	1,87	2,06	1,89	1,92	1,57
4	1,89	2,28	2,10	2,16	1,69

Table 5. Load balance among threads (in number of states)

Threads	Thread 0	Thread 1	Thread 2	Thread 3
1	23 426	0	0	0
	88 401	0	0	0
2	11 871	11 555	0	0
	43 881	44 520	0	0
3	8 011	7 573	7 742	0
	29 685	28 702	30 014	0
4	5 946	6 501	5 078	5 901
	22 516	23 428	20 735	21 722

The parallel speedups that we have currently achieved are satisfactory despite we had expected a little bit better results (especially efficiency). However, the load balance among particular threads is nearly ideal. This is promising for the future development because the good load balance is a basic condition for achieving a good parallel speedup. We illustrate the quality of our algorithm regarding the load balance in Tab. 5 and Tab. 6 where the values for the Sun Enterprise 450 server and a small number of states are showed. An upper number in each cell corresponds with a number of unique states while the lower one corresponds with a number of generated states. We show here the values for the model with small number of states due to the load balance is even better when models with bigger number of states are considered.

5 Conclusions and Future Work

We have discussed possibilities of exploiting parallel architectures with shared memory for combating the high computational complexity of model checking. We have used Java and JOMP [2] as programming tools and considered OOPNs as a modelling formalism. Proposing a parallel algorithm with good load balance among threads is an important result of our work.

Table 6. Load balance among threads (in percentage)

Threads	Thread 0	Thread 1	Thread 2	Thread 3
1	100,0	0	0	0
	100,0	0	0	0
2	50,7	49,3	0	0
	49,6	50,4	0	0
3	34,2	32,3	33,5	0
	33,6	32,5	34,0	0
4	25,4	27,8	21,7	25,2
	25,5	26,5	23,5	24,6

We have also presented experimental results that we have achieved. To sum up them, we have reached parallel speedups up to 3.3 using 16 processors on the Sun Fire 15k server and speedups up to 2.3 using 4 processors on the Sun Enterprise 450 server.

The main problem that prevents us to achieve an even better parallel speedup is the memory management system (especially garbage collecting) due to the memory operations being performed sequentially. We have tried to exploit parallel garbage collectors too—the results are, however, not satisfactory. Another possible approach to this problem is to implement a user-specific memory management system like in [1] to reduce the number of dynamically created objects.

Acknowledgement. This work was supported by the Czech Grant Agency under the contracts 102/04/0780 and 102/03/D211 and it was also supported by the European Community within the “Access to Research Infrastructure Action of the Improving Human Potential Programme” under the contract HPRI-CT-1999-00026.

References

1. S. C. Allmaier and G. Horton. Parallel Shared-Memory State-Space Exploration in Stochastic Modelling. In Proc. of IRREGULAR ’97, LNCS 1253, 1997. Springer.
2. M. Bull and M. E. Kambites. JOMP—an OpenMP-like Interface for Java. In Proc. of the ACM 2000 conference on Java Grande, 2000. ACM Press.
3. E. M. Clarke, O. Grumberg, and D. A. Peled. Model Checking. The MIT Press, Cambridge, Massachusetts, London, England, 2000.
4. M. Češka, V. Janoušek, and T. Vojnar. Towards Verifying Distributed Systems Using Object-Oriented Petri Nets. In Proc. of EUROCAST’99, LNCS 1798, 2000. Springer.
5. M. Češka, L. Haša, and T. Vojnar. Partial Order Reduction in Model Checking of Object-Oriented Petri Nets. In Proc. of EUROCAST’03, LNCS 2809, 2003. Springer.
6. V. Janoušek. Modelling Objects by Petri Nets. PhD. thesis, Brno University of Technology, Brno, Czech Republic, 1998.
7. B. Kočí. Methods and Tools for Implementing Open Simulation Systems. PhD. thesis, Brno University of Technology, Brno, Czech Republic, 2004.
8. B. Křena. Analysis Methods of Object Oriented Petri Nets. PhD. thesis, Brno University of Technology, Brno, Czech Republic, 2004.
9. T. Vojnar. Towards Formal Analysis and Verification over State Spaces of Object-Oriented Petri Nets. PhD. thesis, Brno University of Technology, Brno, Czech Republic, 2001.

Towards Automated Controlling of Human Projectworking Based on Multiagent Systems

Manfred Mauerkirchner and Gerhard Hoefer

Partner of University of Derby in Austria, A-4060 Leonding, Austria
m.mauerkirchner@eduhi.at
g.hoefer@htl-leonding.ac.at

Abstract. Calculating optimized project plans, consisting of an arbitrary number of activities of different types and within a dynamic available pool of human resources characterized by specific profiles can be a most challenging task. We focus mainly on the project control process based on such project plans, which also provide the possibility to integrate different types of external disturbances that normally influence a project workflow. Observed deviations of precalculated time intervals of activities immediately lead to the activation of the so-called self learning components, which automatically tune the personal parameters of each involved workgroup member. At least there are two processes to observe and to synchronize permanently: Real project workflow (the so-called Real System) and its realtime emulation (the so-called Simulation System). Due to all these dynamic requirements and to gain scalability of the system, the implementation selected is based on the usage of different intelligent software agents.

1 Introduction

Due to unpredictable changes of previously planned project resources the control of running projects can be a very demanding task. To automate external generated feedback and corresponding system adaptation we propose a flexible and scalable model which can be implemented best by means of multiagent methodology. At least two processes have to be observed and synchronized simultaneously: real project workflow (Real System) and its real-time emulation (Simulation System), both using a common, permanent adapted and optimized project plan. Such a mechanism could be a suitable means to integrate specific events like allocation, modifying or deleting of project resources in real-time without corrupting the comprehensive system.

As we focus mainly on individuals as project resources we are forced to add several characteristic properties which are typical concerning human resources: Specific skills and capacities and costs to perform certain activities, heuristic relationships between arbitrary persons to describe their abilities to permit groupworking, etc. Finally, the usage of appropriate self learning components should enable the storage of actual knowledge about the involved persons of the resource pool in a transparent way, in order to achieve at least our overall goal: Automated controlling of project working.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 281–290, 2005.
© Springer-Verlag Berlin Heidelberg 2005

1.1 Planning Subsystem

A simple way to start with is a directed graph of so-called *project subtasks* (without specific types) to define predecessor-succesor relations between subtasks with estimated execution durations and an arbitrary set of non specific human resources. Because of the very high degree of complexity the use of exact solution methods (i.e. based on the Branch-and-Bound principle) are of minor practical relevance today. Over the years various techniques have been developed that attempt to solve this classic NP problem, known as *Resource Constrained Project Scheduling Problem* (RCPSP). The implementation of heuristic models has been preferred because they generate solutions iteratively by checking so-called *priority rules* and, in addition, weighted by adaptive random sampling methods [6]. Our proposed model is characterized by an arbitrary directed graph of so-called *project activities* (of any type), a pool of human resources with different skills/capacities/costs and a matrix schema to describe heuristic relationships between all members of the pool [7].

Our planning subsystem consists of two parts, each using another methodology: *Local Optimizer* searches optimal project groups within a dynamic environment for activity-specific requests and using appropriate genetic algorithms [4], *Global Optimizer* simulates any project workflow and produces project plans, which are optimized by a numeric non-gradient method by Rosenbrock [11]. As a result all project plans contain a complete workflow of project activities and schedules of all involved persons [8]. The most important requirements of such a planning system are:

- calculation of complete project plans very efficiently and whenever needed
- re-planning of partly executed plans (from any moment on)
- re-planning of partly executed plans with regard to an addional restriction: existing group memberships remain unchanged
- delivery of schedules/plans between two arbitrary dates

1.2 Controlling Subsystem

Based on a common actual project plan, both systems (Real System, Simulation System) are active and running synchronously. Real project workflow becomes emulated and visualized by the so-called *Controlling Subsystem*, used by a responsible project manager. Using a so-called *Controlling Interface* a great part of different external disturbances (category-1 disturbances) that influence real project working will be captured explicitly, i.e. allocation and/or deleting of human resources, deviations from pre-calculated time intervals of already finished project activities, etc.[9] The category-2 type of disturbances concerns personal skills of project members and their relationships within project groups whose integration is completely automated by the methodology described in chapter 2. (Optionally) accompanied by strategic decisions (i.e. modification of logical project flow and/or project structure, etc.) and based on all modified previously stated project parameters, the project manager starts the (re)planning

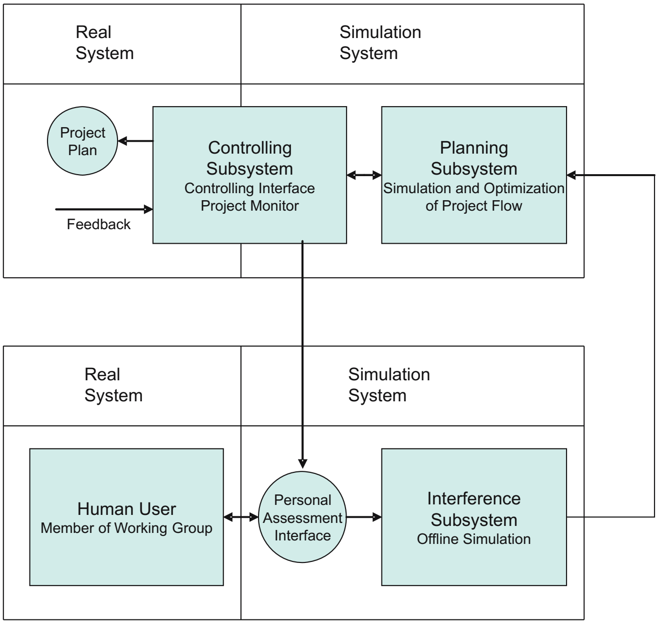


Fig. 1. System View

subsystem asynchronously while both main systems keep running without interruption (see Fig.1).

1.3 Synchronization

Synchronization in our context means that a newly generated project plan (part of a former active plan) becomes effective as actual base for the next future. Activation of new plans may happen automatically on the occasion of a project milestone (begin or end of project activity) or enforced explicitly by a decision of the project manager. Both cases ensure a fully synchronized project system based on an optimized and modified project plan with regard to the actual project conditions. It should be clear that such a system provides many possibilities to inform interested stakeholders about potential risks which eventually could hinder achieving planned results.

2 Self Learning Components

The components described in this section enhance the performance of the overall system by adding the capability of self learning based on personal feedback of all participating group members. The ability of self learning is achieved by adding two components described in this section and helps to increase the problem specific adaptation of the overall system. This kind of adaptation deals with a set of system inherent parameters that represent the characteristics of all human resources participating in the various activities necessary to achieve global goals. This characteristics are classified in two distinct groups : The first group is made up of so-called *skill parameters* whereas the second one contains so-called *interaction parameters*. Both classes are crucial for the synchronization of the real and the simulated project flow. Only realistic assumptions for their values keep both flows in track permanently. Determining the entirety of these values is a demanding task and usually only rough estimations are available in advance. Therefore, tuning all parameter values in a proper way is an inevitable requirement for long time synchronicity of the real and the simulated project flow. This tuning is realized as an offline adaptation process which is executed in two successive steps : The *personal assessment interface* collects suggestions for improving of both skill and group interaction parameter values from all members of the concerned groups. The *interference subsystem* tries to gradually improve the tuning of system inherent parameters by iteratively incorporating the collected suggestions into the actual parameter values (see also Fig.1). Self learning is regarded as a steady process of adaptation that maintains the synchronicity of both project flows by tuning all system relevant parameters.

2.1 Personal Assessment Interface

The tuning process is triggered by an apparent disruption of the synchronicity of the real and the simulated project flow. However, a relevant difference in calculated and real existing execution time of any project activity must not be neglected. As an immediate response the personal assessment interface requests all members of the affected working groups to notify which parameter values may not be chosen properly. Therefore, the feedback of each group member consists of suggestions for the revision of the potentially incorrect parameter values.The resulting suggestions are formalized comments on parameter values of problem relevant user skills and of critical member interactions in each working group. This formalized inquiry is based on standardized questionnaires with regard to all the relevant parameters mentioned above. Questionnaires should have the following structure : One scale bar per each relevant parameter displays the actual parameter value. The displayed values may be adjustable by all individual group members according to their latest group related experiences. Therefore, the scale should represent all possible values of judging skills and interaction that are proposed by modern social psychology [5]. Questionnaires may be realized as electronic documents that are transportable over the intranet. Furthermore, the personal assessment interface has to transfer the potentially conflicting statements of this inquiry to the central interference subsystem and store them in its database.

2.2 Interference Subsystem

The interference subsystem generates context specific updates of the parameter values used by the central planning subsystem in order to re-synchronize the real project flow and its emulation. The collection of stored statements for improval forms the basis for the generation of these updates which is carried out in the following distinct steps.

Step 1 : Create a population of sets that encode one statement for each characteristic parameter respectively.

Step 2 : Determine the fitness of each set by repeating the latest simulation of the planning subsystem offline.

Step 3 : Choose sets that are able to improve the synchronization of the latest activities in the real system and the simulated one.

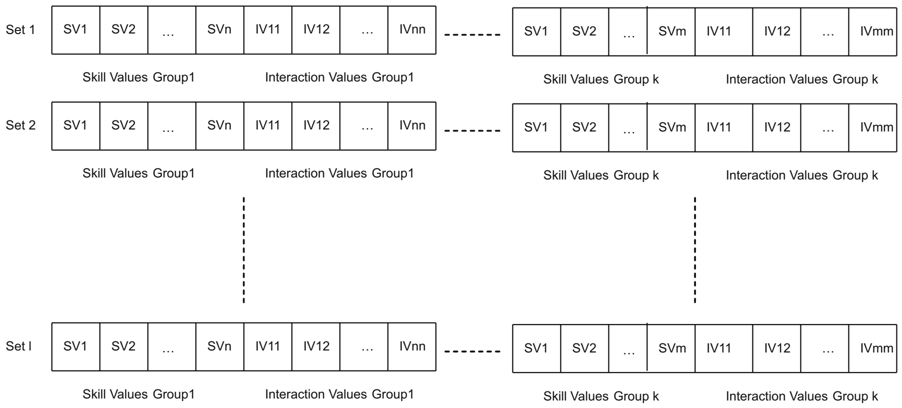
Step 4 : Generate new populations of bests by performing genetic operations.

Repeat steps 2 to 4 until some improved steady state level is reached.

The updated parameter values are checked by the controlling subsystem for plausibility and then transferred to the planning subsystem for future simulations of the project flow.

Each of these steps needs to be elaborated more in depth :

Step 1 guaranties that the retrieved information about user and group specific parameters is encoded in a proper way for further processing especially for the employment of heuristic optimization strategies. Therefore, it is of great importance that potentially conflicting suggestions of the various group members are distributed over the entire population. This means that in each set there is exactly one group specific gene encoding all group relevant parameters. Group relevant parameters form a set of parameter values containing the skill and interaction values of all group members (see Fig.2).



- Set_1 to Set_l consist out of k different groups representing all different work groups participating in the current project
- Each group consists out of member specific skill values SV and group specific interaction values IV
- If one group contains g members there exist g individual skill values SV_i and g x g interaction values IV_{ij}

Fig. 2. Population Structure

Step 2 assesses the power of all individual sets of this population to simulate the actual situation in each group of the real project flow. Therefore, an offline rerun of the latest simulation is carried out for all these sets. The potential of each simulation run to re-establish the synchronicity of the real and the simulated project flow might be a convincing measure of fitness. This measure may be obtained by sampling the deviations in real and simulated execution times of all actual group relevant activities. The onset of the sampling interval and its length are of high relevance. The onset of re-running the simulation with the population of manipulated parameter values should be the point in time when activities that cause that deviations in execution times have been started. The length of the sampling interval may be the time interval until all real and simulated activities are finished. The measure is sampled as the sum of all deviation times of the latest activities of the real project flow and their offline simulation. In conclusion, the potentially fittest set of the population yields the shortest sample time.

Step 3 selects the sets that constitute the bases of the population for the next adaptation cycle. Selection may be executed by employing one of the various well established strategies like roulette wheel or tournament selection depending on the individual fitness measures determined in step 2.

Step 4 deals with the organization of the next population. One convenient strategy might be to perform complete or partial crossover of genes that encode for the same group activity and to induce mutations at a low rate for all parameter values. As a result this restriction in crossover operations causes the fact that the parameter specific information retrieved by the personal assessment interface are not spread out on other genes or mixed up in identical genes. Crossover operations are employed to find combinations of manipulated parameter values that may be regarded as being a close representation of all internal affairs of each group. Additionally, mutations are also introduced to check whether some parameter values are sensitive to some further tuning. Mutations should only slightly alter the encoding of one parameter value [10].

2.3 Integration

The components that enable self learning capabilities are extensions of the overall system and realized as the above described personal assessment interface and the interference subsystem. Both extensions are integrated in the overall system in a straight forward way since the interference subsystem is connected to all human users via the personal assessment interface. On the other hand, the interference subsystem acesses the central planning subsystem for its offline test runs which also does not need any change in the overall system architecture.

3 Multiagent Architecture

Considering Fig.1 depicting all features of a project controlling mechanism from a system viewpoint we have to reflect next which underlying software architecture could meet our requirements best: We propose the implementation of

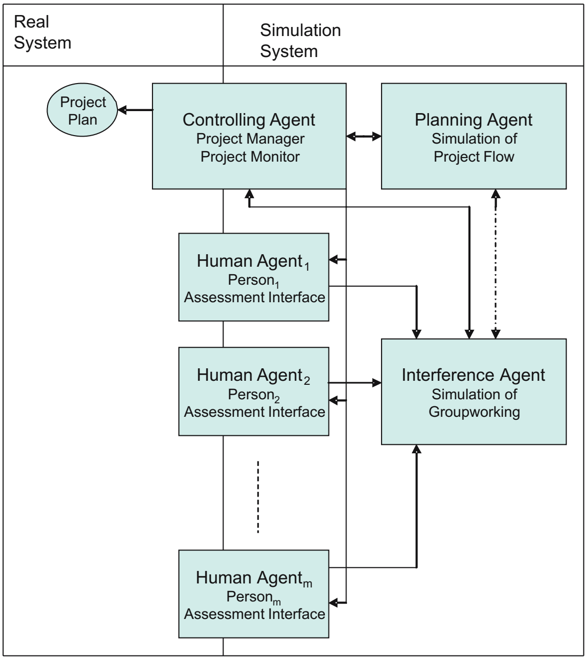


Fig. 3. Multiagent Structure

a so-called *multiagent system*, more exactly the usage of different (intelligent) *software agents*. Fig.3 illustrates the transformation of our previously designed model in a diagram showing all involved issues using agent technology. The next paragraphs show the reasons why the selected technology is appropriate and promising concerning our problem in more detail.

To achieve a basic understanding of the tasks and functioning of intelligent agents we start with characteristic properties that differentiate an intelligent agent from traditional software programs [1]. It should be mentioned that not every agent must provide all stated properties, it rather depends completely on its complexity. The characteristics of intelligent agents can be grouped into two main categories: *internal* and *external properties.* Internal properties define the internal being of an agent determining the actions within the agent. They include the ability to learn, reactivity, autonomy and goal orientations. The external properties comprehend all those characteristics that affect the interaction of several agents or human agent communication. Not all characteristics can be assigned to just one group, rather parts can belong to both groups. The property *character* of an agent is a well known example for the issue which significant parts determine the internal behavior and which parts also play an important role in the external communication [12].

The principle property of agents is *reactivity* which means that agents are able to realize their environment and to react in an acceptable time. All types of agents active in the system described above are basically realized as reactive agents since all activies are initiated primarily by the project manager and individual agents react in a hierarchical order.

Proactivity is regarded as a property grouped a level above reactivity. It does not only react to the changes of its environment but takes itself the initiative under specific circumstances. This ability normally requires that the agent has well defined goals or even a complex goal system in addition. This property is named *goal-orientation*. The control subsystem may be realized as a proactive agent provided that strict rules for initiating a synchronisation are laid down.

Because the complexity of an agent can range widely also areas like rule-based-systems, knowledge-based-systems or neural networks from classical artificial intelligence can be suitable. Reasoning capability and the ability to recognize the environment gives the agent the chance to make specific decisions as a result of changing environment. This *reasoning/learning* is a property of the interference subsystem which is realized using adaptive evolutionary strategies.

The ability of an agent to follow its goal to a certain user defined degree and without user intervention is named *autonomy*. It is one of the key characteristics of intelligent agents. To act autonomously an agent must have both control over its actions and internal states. Autonomy is a property of all agents active in this system since all activities of individual agents are carried out without any monitoring. Only the project manager is allowed to monitor and verify the final outcome of all simulations and learning steps.

The property *mobility* can be described by the ability to move from one computer to another in electronic communications networks. This results in high demands on the network environment, security, data privacy and data management. Every involved computer must be able to pack a mobile agent and send it to another computer as well as to receive new agents. This property is essential for the presented system since the personal assessment interface is realized as a set of mobile agents. All individual human user communication interfaces are made up by the corresponding personal agent.

The previously mentioned property *character* collects all features to interact with human beings over a virtual person, i.e. using an avatar.

The types of agents described above have to interact in a concerted way. The intended way of interaction is determined by three main elements of interaction [2]. These elements are the aim of the agent, its individual abilities, and the resources it demands. The aims of interacting agents may be compatible or imcompatible, whereas their abilities and demanded resources may be sufficient or insufficient to achieve certain tasks. Compatible aims mean that reaching the aim of one agent does not prevent any other agent from reaching its individual aim. An agent with sufficient abilities is able to solve the complete problem on its own. Sufficient resources make sure that interacting agents do not have to compete for the use of certain resources. In the system described above the individual aims of all agents are compatible since the complete system is designed in a modular and hierarchical way. So agents interact in a well defined way that lacks the potential of conflicting aims. The abilities of each participating agent is sufficient to deal with its individuak task but insufficient to take charge of any other task. The resources demanded by each agent may be available without problematic restrictions posed by the overall system since each agent is realized as a distributed process running in its own context. The overall type of this interaction scheme is classified as simple cooperation and typical for communication systems like the one introduced above.

4 Summary and Preview

The overall intention on our proposed way towards automating the project control process is to integrate (nearly) all kinds of external disturbances that influence the precalculated project flow. We distinguish between

– disturbances of category-1, which enforce an immediate decision of the project manager, to start the (re)planning subsystem and furthermore, to synchronize the real project work and its emulation process using the same new calculated project plan.
– disturbances of category-2, which are not neglectable deviations of durations of project activities, normally caused by not corresponding values of personal parameters of the involved human resources (skill parameters and group interaction parameters). In such cases the so-called self learning components will be executed automatically trying to tune the set of personal parameters by requesting suggestions of all group members and optimizing them by using a specific evolutionary methodology. Optimal solutions are recommendations for the project manager, followed by the same procedure like described with category-1 disturbances.

It should be clear that such a permanent modified and adapted system can be used as a resource of information for interested stakeholders about potential risks of the running project at any time. Another obvious benefit is that a permanent updated database of parameters of all involved human resources can be reused within any new projects - a possible way to store specific knowledge.

The transformation of our proposed system in a system of different software agents is a matter of current work. In addition, the implementation of the so-called interference subsystem with the aspect of satisfying all performance requirements, has to be mastered in the next future. Finally it should be mentioned that the base system (planning subsystem, controlling subsystem, integration of category-1 disturbances) has been implemented as a running prototype since 1998.

References

1. W.Brenner, R.Zarnekow, H.Wittig, (1998): Intelligent Software Agents, Foundations and Applications, in *Springer-Verlag Berlin Heidelberg*
2. J.Ferber, (1999): Multi-Agent Systems, An Introduction to Distributed Artificial Intelligence, in *Addison-Wesley Publishing Company*

3. http://www.fipa.org, visited 15-12-2004
4. D.E.Goldberg, (1989): Genetic Algorithms in Search, Optimization and Machine Learning, *Addison-Wesley Publishing Company*
5. K.Holm, (1991): Die Befragung I, in *UNI-TB, Stuttgart*
6. R.Kolisch, A.Drexl, A.Sprecher, (1997): Neuere Entwicklungen in der Projektplanung, in *Zeitschrift für betriebswirtschaftliche Forschung, Jg.49*
7. M.Mauerkirchner, (1997): Event Based Modelling and Control of Software Processes, in *Engineering of Computer-Based Systems - Proceedings ECBS'97 Monterey, California*
8. M.Mauerkirchner, (2001): A General Planning Method for Allocation of Human Resource Groups, in *Lecture Notes in Computer Science - Proceedings EUROCAST'01 Las Palmas de Gran Canaria, Spain*
9. M.Mauerkirchner, (1999): Decision Based Adaptive Model for Managing Software Development Projects, in *Lecture Notes in Computer Science - Proceedings EUROCAST'99 Vienna, Austria*
10. H.Pohlheim, (2000): Evolutionäre Algorithmen, in *Springer-Verlag Berlin Heidelberg*
11. H.H.Rosenbrock, (1960): An Automatic Method for Finding the Greatest or Least Value of a Function, in *Computer Journal Vol.4*
12. M.Wooldridge, N.R.Jennings, (1995): Intelligent Agents: Theory and Practice, in *The Knowledge Engineering Review, 10(2)*

Tree-Structured Legendre Multi-wavelets

Ekaterina Pogossova, Karen Egiazarian, Atanas Gotchev, and Jaakko Astola

Tampere University of Technology, Institute of Signal Processing,
P. O. Box 553, FIN-33101 Tampere, Finland

Abstract. We address the problem of constructing multi-wavelets, that is, wavelets with more than one scaling and wavelet function. We generalize the algorithm, proposed by Alpert [1] for generating discrete Legendre multi-wavelets to the case of arbitrary, non-dyadic time interval splitting.

1 Introduction

In the past two decades there has been a considerable interest to the wavelet analysis, a tool that emerged from mathematics and was quickly adopted by diverse fields of science and engineering [7]. Wavelets are being applied to a wide and growing range of applications such as signal processing, data and image compression, solution of partial differential equations, and statistics.

Recently, multi-wavelets, that is, wavelets with more than one scaling and wavelet function have gained a considerable interest in the area of signal processing. Promising results have been obtained with multi-wavelets in signal denoising and compression [8].

The first multi-wavelets were introduced by Alpert [1], who constructed Legendre type of wavelets on the interval [0,1) with several scaling functions $\phi_0, \phi_1, \ldots, \phi_{N-1}$, instead of just one scaling function ϕ_0. This difference enabled high-order approximation with basis functions supported on non-overlapping intervals. In the particular case of $N = 1$, Alpert's multi-wavelets coincide with the Haar basis.

The discrete analogue of continuous Legendre multi-wavelets was also introduced by Alpert [1]. The structure of this analogue is essentially similar to that of continuous multi-wavelet bases, but the discrete construction is more convenient when the representation of a function (and its related operations) is based on its values at a finite set of points [1].

In [3], a new basis, called the tree-structured Haar (TSH) basis was introduced. It is a generalization of the classical Haar basis to arbitrary time and scale splitting. The idea behind such a construction is to adapt the basis to the signal on hand. Discrete orthogonal TSH transform has been successfully applied to the problem of de-noising signals [5].

In [4], the TSH structure has been extended to the multi-wavelet bases by an analog to Alpert's construction. The construction of the basis functions assured their orthogonality in the continuous space. However, the question regarding the construction of discrete orthogonal bases has been left unexplored. In this

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 291–300, 2005.
© Springer-Verlag Berlin Heidelberg 2005

paper we address the problem of constructing the discrete counterpart of tree-structured multi-wavelets, so called, tree-structured Legendre (TSL) transform, by generalizing the construction procedure, that has been introduced in [1].

2 Wavelet Fundamentals

The key concept behind the wavelet theory is *multiresolution analysis* (MRA). A multiresolution analysis in $L^2(R)$ is given by a nested sequence of subspaces:

$$\begin{array}{c} \ldots \subset V_{-1} \subset V_0 \subset V_1 \subset \ldots, \\ \longleftarrow Coarser \quad Finer \longrightarrow \end{array} \tag{1}$$

such that $clos_{L^2}(\bigcup_{j\in Z} V_j) = L^2(R)$ (completeness), $\bigcap_{j\in Z} V_j = \{0\}$ (uniqueness), and $f(t) \in V_j \leftrightarrow f(2t) \in V_{j+1}, j \in Z$ (scaling property).

It can be shown that the following relations hold for MRA. There exists a function $\phi(t)$:

$$\phi(t) = \sum_{k=-\infty}^{\infty} p_k \phi(2t-k), \tag{2}$$

called *scaling function*, such that $V_j =$ linear span $\{\phi_{j,k}\};\ j,k \in Z$, where

$$\phi_{j,k}(t) = \sqrt{2^j}\phi(2^j t - k);\ j,k \in Z \tag{3}$$

are dilated and translated versions of $\phi(t)$ (we refer to Mallat [6] for detailed explanation).

Given a set of nested subspaces V_j, there exists a set of subspaces W_j, such that

$$V_{j+1} = V_j \bigoplus W_j,\, W_{j+1} \perp W_j\ ,\ \text{ if } j \neq j';\, j,j' \in Z. \tag{4}$$

These subspaces give an orthogonal decomposition of $L^2(R)$, namely

$$L^2(R) = \bigoplus_{j\in Z} W_j. \tag{5}$$

Moreover, W_j's inherit the scaling property from Vj: $f(t) \in W_j \leftrightarrow f(2t) \in W_{j+1};\ j \in Z$. For a scaling function ϕ in V_0, there exists its counterpart ψ in W_0, called *wavelet*, such that $\{\psi_{j,k}\}$ generate W_j, where

$$\psi_{j,k}(t) = \sqrt{2^j}\psi(2^j t - k); j,k \in Z. \tag{6}$$

Since $\psi \in W_0 \subset V_1$, it can be written in terms of $\phi(2t-k)$. A pair

$$\begin{cases} \phi(t) = \sum_{k=-\infty}^{\infty} p_k \phi(2t-k), \\ \psi(t) = \sum_{k=-\infty}^{\infty} q_k \phi(2t-k) \end{cases} \tag{7}$$

is called *two-scale relations.*

3 Continuous Legendre Multi-wavelets

We now describe briefly the Alpert's design of Legendre type of multi-wavelets on the interval [0,1). The two-scale relations for N *Legendre scaling functions* of order $(N-1)$, $\phi_0, \phi_1, \ldots, \phi_{N-1}$, are defined as

$$\phi_i(t) = \textstyle\sum_{j=0}^{N-1} p_{i,j}\phi_j(2t) + \sum_{j=0}^{m} p_{i,N+j}\phi_j(2t-1);\, i \in \{0,1,\ldots,N-1\}. \quad (8)$$

Here as well as below we assume $t \in [0,1)$. The $i-th$ scaling function, ϕ_i, is an $i-th$ order polynomial, and all ϕ_i's form an orthonormal basis, that is,

$$\phi_i(t) = \textstyle\sum_{k=0}^{i} a_{i,k}x^k;\, i \in \{0,1,\ldots,N-1\}, \quad (9)$$

where

$$\textstyle\int_{-\infty}^{\infty} \phi_i(t)\phi_k(t)dt = \delta_{i,k};\, i,k \in \{0,1,\ldots,N-1\}. \quad (10)$$

The coefficients $p_{i,j}$ in (8) are determined uniquely from the conditions (9),(10).

Remarks:

1. Since $\phi_i(t)$ is the $i-th$ order polynomial, if follows that $p_{i,j} = p_{i,N+j} = 0$, for $i < j$.
2. The two-scale relations for the Legendre scaling function of order $n < N-1$ is a subset of the first n two-scale relations for ϕ_i for $i \in \{0,1,\ldots,n\}$ from the $(N-1)-th$ order two-scale relations.

The two-scale relations for the $(N-1)-th$ order Legendre wavelets are in the form:

$$\psi_i(t) = \textstyle\sum_{j=0}^{N-1} q_{i,j}\phi_j(2t) + \sum_{j=0}^{N-1} q_{i,N+j}\phi_j(2t-1);\, i \in \{0,1,\ldots,N-1\}. \quad (11)$$

Since there are $2N^2$ unknown coefficients $q_{i,j}$ in (11), we need $2N^2$ independent conditions to determine the two-scale relations. Among many possible choices for these conditions that would determine different wavelets, the orthonormality and vanishing moment conditions were selected by Alpert [2]:

$$\textstyle\int_{-\infty}^{\infty} \psi_i(t)\psi_k(t)dt = \delta_{i,k};\, i,k \in \{0,1,\ldots,N-1\}, \quad (12)$$

$$\textstyle\int_{-\infty}^{\infty} \psi_i(t)t^j dt = 0,\, i \in \{0,1,\ldots,N-1\}; j \in \{0,1,\ldots,i+N-1\}. \quad (13)$$

4 Continuous Legendre Multi-wavelets: Non-dyadic Interval Splitting

In [4], the concept of the dyadic two-scale relations between Legenrde scaling functions and wavelets has been generalized to two-scale relations having an arbitrary interval-splitting point, namely, α. The two-scale relations for the $(N-1)$-th order non-dyadic Legendre scaling functions and wavelets are:

$$\phi_i(t) = \textstyle\sum_{j=0}^{N-1} p_{i,j}\phi_j(\alpha t) + \sum_{j=0}^{N-1} p_{i,N+j}\phi_j(\frac{\alpha t-1}{\alpha-1}), \quad (14)$$

$$\psi_i(t) = \sum_{j=0}^{N-1} q_{i,j}\psi_j(\alpha t) + \sum_{j=0}^{N-1} q_{i,N+j}\phi_j(\tfrac{\alpha t-1}{\alpha-1}), \tag{15}$$

for $i \in \{0, 1, \ldots, N-1\}$, $t \in [0, 1)$. The unknown coefficients $p_{i,j}$ and $q_{i,j}$ are uniquely defined based on the same considerations, as for the dyadic Legendre multi-wavelets.

It has been mentioned in [4], that by the construction (14), (15), there is no more scale invariance. No integer shifts at the same scale (like in the dyadic case) form the subspaces V_j. In other words, the nested subspaces cannot anymore be indexed by their scale. However, the functions $\phi_j(\alpha t)$ and $\phi_j(\frac{\alpha t-1}{\alpha-1})$ are orthogonal and hence form a basis for V_0 and for its orthogonal complement, W_0 [4].

In the particular case of $\alpha = 2$, the new construction coincides with dyadic Legendre multi-wavelets.

Example 1. Linear non-dyadic Legendre multi-wavelets. If $N = 2$, the two-scale relations for scaling functions and wavelets take the form:

$$\begin{pmatrix} \phi_0(t) \\ \phi_1(t) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ -\frac{\sqrt{3}(\alpha-1)}{\alpha} & \frac{1}{\alpha} & \frac{\sqrt{3}}{\alpha} & \frac{\alpha-1}{\alpha} \end{pmatrix} \begin{pmatrix} \phi_0(\alpha t) \\ \phi_1(\alpha t) \\ \phi_0(\frac{\alpha t-1}{\alpha-1}) \\ \phi_1(\frac{\alpha t-1}{\alpha-1}) \end{pmatrix}, \tag{16}$$

$$\begin{pmatrix} \psi_0(t) \\ \psi_1(t) \end{pmatrix} = \begin{pmatrix} \frac{(\alpha-2)\sqrt{\alpha-1}}{\sqrt{\alpha^2-\alpha+1}} & \frac{-\sqrt{3}\sqrt{\alpha-1}}{\sqrt{\alpha^2-\alpha+1}} & \frac{(\alpha-2)\sqrt{\alpha-1}}{(\alpha-1)\sqrt{\alpha^2-\alpha+1}} & \frac{\sqrt{3}\sqrt{\alpha-1}}{\sqrt{\alpha^2-\alpha+1}} \\ \frac{\sqrt{3}(\alpha-1)\sqrt{\alpha-1}}{\alpha\sqrt{\alpha^2-\alpha+1}} & \frac{(\alpha+1)(\alpha-1)\sqrt{\alpha-1}}{\alpha\sqrt{\alpha^2-\alpha+1}} & \frac{-\sqrt{3}\sqrt{\alpha-1}}{\alpha\sqrt{\alpha^2-\alpha+1}} & \frac{(2\alpha-2)\sqrt{\alpha-1}}{\alpha(\alpha-1)\sqrt{\alpha^2-\alpha+1}} \end{pmatrix} \times$$
$$\times \begin{pmatrix} \phi_0(\alpha t) \\ \phi_1(\alpha t) \\ \phi_0(\frac{\alpha t-1}{\alpha-1}) \\ \phi_1(\frac{\alpha t-1}{\alpha-1}) \end{pmatrix}. \tag{17}$$

If $\alpha = 2$, (16), (17) will be identical to the linear two-scale relations for dyadic Alpert's multi-wavelets.

The wavelets ψ_0 and ψ_1 for the case of $\alpha = 3, N = 2$ are presented in Fig. 1.

5 Splitting Point Selection

For general non-dyadic splitting, the scaling parameter, α, is selected according to so called *binary interval splitting tree* (BIST), that has been introduced in [3]. BIST is a binary tree, where each nonterminal node has either two children (splitting node), or a single child (non-splitting node). The root of the tree is associated to the entire interval $[0, 1)$. The outedges are labelled as below:

1. if the node is splitting, its left outedge is labelled by $\lambda = 0$, and its right outedge is labelled by $\lambda = 1$;
2. if the node is non-splitting, its only outedge is labelled by $\lambda = 2$.

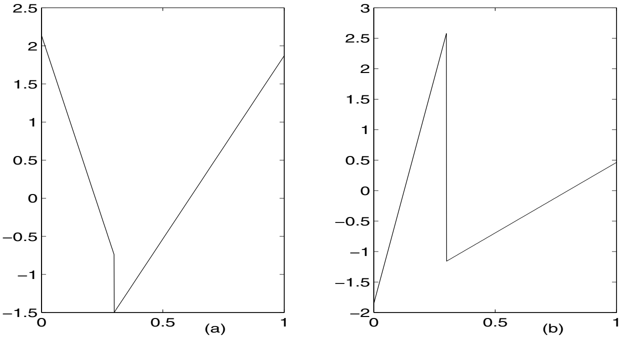


Fig. 1. Non-symmetric Legendre multi-wavelets for $\alpha = 3$: a) wavelet $\psi_0(t)$, b) wavelet $\psi_1(t)$

Each node, a, is indexed by a ternary vector $(\lambda_1(a), \lambda_2(a), \ldots, \lambda_k(a))$, where λ_j are the outedge labels from the root to this node, and $depth(a) = k$. Additionally, the node a is labelled by the number l of leaves of the subtree rooted in that node (all leaves are labelled with 1). The described tree determines splitting of $[0, 1)$ interval into subintervals, each defined by a node of the tree:

1. assign $I_{root} = [0, 1)$;
2. if $I_{\lambda_1,\lambda_2,\ldots,\lambda_k} = [0, 1)$ for a node $(\lambda_1, \lambda_2, \ldots, \lambda_k)$, then

$$\begin{cases} I_{\lambda_1,\lambda_2,\ldots,\lambda_k,0} = [c, c + \frac{l_{\lambda_1,\lambda_2,\ldots,\lambda_k}}{l_{\lambda_1,\lambda_2,\ldots,\lambda_k,0}}(d-c)), \\ I_{\lambda_1,\lambda_2,\ldots,\lambda_k,1} = [c + \frac{l_{\lambda_1,\lambda_2,\ldots,\lambda_k}}{l_{\lambda_1,\lambda_2,\ldots,\lambda_k,0}}(d-c), d), \end{cases} \tag{18}$$

   if $(\lambda_1, \lambda_2, \ldots, \lambda_k)$ is a splitting node; and

$$I_{\lambda_1,\lambda_2,\ldots,\lambda_k,2} = I_{\lambda_1,\lambda_2,\ldots,\lambda_k}, \tag{19}$$

   if $(\lambda_1, \lambda_2, \ldots, \lambda_k)$ is a non-splitting node.

A BIST is illustrated in Fig. 2. The parameter α determines the interval splitting ratio:

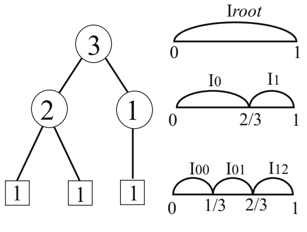


Fig. 2. Binary interval splitting tree

$$\begin{cases} \frac{|I_{\lambda_1,\lambda_2,\ldots,\lambda_k}|}{|I_{\lambda_1,\lambda_2,\ldots,\lambda_k,0}|} = \frac{l_{\lambda_1,\lambda_2,\ldots,\lambda_k}}{l_{\lambda_1,\lambda_2,\ldots,\lambda_k,0}} = \alpha, \\ \frac{|I_{\lambda_1,\lambda_2,\ldots,\lambda_k}|}{|I_{\lambda_1,\lambda_2,\ldots,\lambda_k,1}|} = \frac{l_{\lambda_1,\lambda_2,\ldots,\lambda_k}}{l_{\lambda_1,\lambda_2,\ldots,\lambda_k,1}} = \frac{\alpha}{\alpha-1}. \end{cases} \tag{20}$$

Given such a rule for interval splitting, a multi-wavelet basis with non-dyadic splitting points can be generated. For each splitting node, two sets of functions, $\phi_0, \phi_1, \ldots, \phi_{N-1}$ and $\psi_0, \psi_1, \ldots, \psi_{N-1}$ can be assigned according to α, defined from (20).

Construction of the basis functions, described by (16), (17) assures their orthogonality.

6 Discrete Construction

In [1], Alpert proposed a discrete bases construction for Legendre multi-wavelets. The structure of this analogue is essentially similar to that of the continuous bases, but the discrete construction is more convinient when representation of the function (and its related operators) is based on its values at a finite set of points [1]. The price is the loss of complete scale invariance: V_n's are no longer the dilates of a single space V_0, rather only nearly so.

In the more general case of non-dyadic spitting, the continuous multi-wavelet basis can be also transformed into a discrete basis. Below we present an algorithm for constructing such bases for non-dyadic Legendre multi-wavelets that includes Alpert's dyadic contruction as a special case. We will refer to these beses as to *tree-structured Legendre* (TSL) multi-wavelets.

Given a set of n discrete points, $\{x_1, x_2, \ldots, x_n\} \subset R$, our goal is to define an orthogonal basis for the n-dimensional space of functions, defined on $\{x_1, x_2, \ldots, x_n\}$. We assume, that $x_1 < x_2 < \ldots < x_n$, and $n = 2^m N$, where N is the required order of approximation, and m is a positive integer. The basis will have two fundamental properties:

1. all but N basis vectors will have N vanishing moments;
2. the basis vectors will be nonzero on different scales.

We start with constructing a binary tree, similar to BIST, though slightly different. Now a label at each node of the tree must be a multiple of N, where $N \geq 1$. Each label is equal the sum of labels of its children, however, all terminal nodes will now be labelled with N, instead of 1 as in the case of BIST. The root of the tree has label n. An example of such a tree with $N = 2$ is presented in Figure 3. By d we denote the depth of the tree, and let μ_j, $j \in \{0, 1, \ldots, d\}$ be the number of nodes on level j ($j = d$ corresponds to the leaves). The above tree is of depth $d = 4$, with $\mu_0 = 1, \mu_1 = 2, \mu_2 = 3, \mu_3 = 4$, and $\mu_4 = 6$. Let also $c(j, i)$, $i \in \{1, 2, \ldots, \mu_j\}$ be the nodes on level j, counted from the left to the right, and $\nu(c(j, i))$ be the label at node $c(j, i)$.

Let us fix the following notations. For a $(2k \times 2k)$ matrix S we let S^U and S^L denote the two $(k \times 2k)$ matrices: S^U, consisting of the upper k rows and S^L, consisting of the lower k rows of S. Suppose that the columns $s_1, s_2, \ldots, s_{2k}$ of S are linearly independent. We define $T = Orth(S)$ to be the matrix that

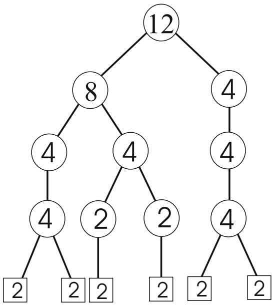


Fig. 3. TSL tree

results from the column-by-column Gram-Shmidt orthogonalization of S. Then, denoting the columns of T by $t_1, t_2, \ldots, t_{2k}$, we have:

$$\left.\begin{array}{r} \text{linear span}\{t_1, t_2, \ldots, t_i\} = \text{linear span}\{s_1, s_2, \ldots, s_i\} \\ t_i^T t_i = \delta_{ij} \end{array}\right\} i, j \in \{1, 2, \ldots, 2k\}. \tag{21}$$

Now we proceed to the definition of basis matrices. We construct so called *moment matrices*, $M_{1,i}$ for $i \in \{1, 2, \ldots, \mu_{d-1}\}$ as follows:

$$M_{1,i} = \begin{pmatrix} 1 & x_{u_i+1} & x_{u_i+1}^2 & \cdots & x_{u_i+1}^{2N-1} \\ 1 & x_{u_i+2} & x_{u_i+2}^2 & \cdots & x_{u_i+2}^{2N-1} \\ \vdots & & & & \vdots \\ 1 & x_{u_i+2N} & x_{u_i+2N}^2 & \cdots & x_{u_i+2N}^{2N-1} \end{pmatrix}, \tag{22}$$

if $c(d-1, i)$ is a splitting node, and

$$M_{1,i} = \begin{pmatrix} 1 & x_{u_i+1} & x_{u_i+1}^2 & \cdots & x_{u_i+1}^{2N-1} \\ 1 & x_{u_i+2} & x_{u_i+2}^2 & \cdots & x_{u_i+2}^{2N-1} \\ \vdots & & & & \vdots \\ 1 & x_{u_i+N} & x_{u_i+N}^2 & \cdots & x_{u_i+N}^{2N-1} \end{pmatrix}, \tag{23}$$

if $c(d-1, i)$ is a non-splitting node. Here $u_i = \sum_{j=1}^{i-1} \nu(c(d-1, j))$ for $i > 1$, and $u_1 = 0$. Additionally, we define matrices $U_{1,i}$, $i \in \{1, 2, \ldots, \mu_{d-1}\}$:

$$U_{1,i} = \begin{cases} Orth(M_{1,i})^T, \text{ if } c(d-1, i) \text{ is a splitting node,} \\ \begin{pmatrix} I_N \\ 0_N \end{pmatrix}, \quad \text{otherwise.} \end{cases} \tag{24}$$

Here I_N and 0_N denote $(N \times N)$ identity and zero matrices, respectively. In this way, the lower N rows of $U_{1,i}$ will have at least N vanishing moments (since they are orthogonal to the first N columns of $M_{1,i}$). These N last rows of $U_{1,i}$ will be included into the final basis, while the first N rows will remain for further processing.

The first $(n \times n)$ basis matrix, U_1, is constructed by deleting all zero rows from the following auxiliary matrix $\tilde{U_1}$:

$$\tilde{U_1} = \begin{pmatrix} U^U_{1,1} & & & \\ & U^U_{1,2} & & \\ & & \ddots & \\ & & & U^U_{1,\mu_{d\;1}} \\ U^L_{1,1} & & & \\ & U^L_{1,2} & & \\ & & \ddots & \\ & & & U^L_{1,\mu_{d\;1}} \end{pmatrix}. \tag{25}$$

The second basis matrix is $U_2 \times U_1$, with an $(n \times n)$ matrix U_2, defined by the formula

$$U_2 = \begin{pmatrix} U_2' & \\ & I_{n-l_2} \end{pmatrix}, \tag{26}$$

where l_2 is the size of matrix U_2'. The latter is obtained by deleting zero rows from $\tilde{U_2'}$,

$$\tilde{U_2'} = \begin{pmatrix} \tilde{U}^U_{2,1} & & & \\ & \tilde{U}^U_{2,2} & & \\ & & \ddots & \\ & & & \tilde{U}^U_{2,\mu_{d\;2}} \\ \tilde{U}^L_{2,1} & & & \\ & \tilde{U}^L_{2,2} & & \\ & & \ddots & \\ & & & \tilde{U}^L_{2,\mu_{d\;2}} \end{pmatrix}, \tag{27}$$

with

$$\tilde{U}_{2,i} = \begin{cases} U_{2,i}, & \text{if } c(d-2,i) \text{ is a splitting node,} \\ \begin{pmatrix} I_N \\ 0_N \end{pmatrix}, & \text{otherwise.} \end{cases} \tag{28}$$

If $c(d-2,i)$ is a splitting node with two children $c(d-1,k)$ and $c(d-1,k+1)$, then $U_{2,i} = Orth(M_{2,i})^T$, where $M_{2,i}$ is given by

$$M_{2,i} = \begin{pmatrix} U^U_{1,k} \times M_{1,k} \\ U^U_{1,k+1} \times M_{1,k+1} \end{pmatrix}. \tag{29}$$

Otherwise, if the node $c(d-2,i)$ is non-splitting, and $c(d-1,k)$ is its (only) child, then $U_{2,i} = U_{1,k}$ and $M_{2,i} = M_{1,k}$.

In general, the $m-th$ basis matrix, for $m \in \{2, \dots, d\}$, is $U_m \times U_{m-1} \times \dots \times U_1$ with U_j, $j \in \{2, \dots, m\}$, defined by the formula

$$U_j = \begin{pmatrix} U_j' & \\ & I_{n-l_j} \end{pmatrix}, \tag{30}$$

where l_j is the size of U'_j, and U'_j are obtained by deleting zero rows from the following matrices $\tilde{U}'_j$:

$$\tilde{U}'_j = \begin{pmatrix} \tilde{U}^U_{j,1} & & \\ & \ddots & \\ & & \tilde{U}^U_{j,\mu_{d\ \ j}} \\ \tilde{U}^L_{j,1} & & \\ & \ddots & \\ & & \tilde{U}^L_{j,\mu_{d\ \ j}} \end{pmatrix}. \tag{31}$$

Here again

$$\tilde{U}_{j,i} = \begin{cases} U_{j,i}, & \text{if } c(d-j,i) \text{ is a splitting node,} \\ \begin{pmatrix} I_N \\ 0_N \end{pmatrix}, & \text{otherwise.} \end{cases} \tag{32}$$

$U_{j,i} = Orth(M_{j,i})^T$, and $M_{j,i}$ is

$$M_{j,i} = \begin{pmatrix} U^U_{j-1,k} \times M_{j-1,k} \\ U^U_{j-1,k+1} \times M_{j-1,k+1} \end{pmatrix}, \tag{33}$$

if $c(d-j,i)$ is a splitting node, and $c(d-j+1,k)$, $c(d-j+1,k+1)$ are its left and right children, respectively. $U_{j,i} = U_{j-1,k}$, $M_{j,i} = M_{j-1,k}$ if the node $c(d-j,i)$ is non-splitting, and $c(d-j+1,k)$ is its only child.

The final matrix, $U = U_d \times U_{d-1} \times \cdots \times U_1$, represents the discrete wavelet-like basis of parameter N on $x_1, x_2 \ldots, x_n$. As it has been mentioned in [1], some adjustments must be made to this formula to ensure numerical stability. These issues are discussed in [2] and we will not cover them here.

Example 2. For the TSL tree, illustrated in Fig. 3, the schematic final basis matrix showing the irregular structure of the construction is given below. Nonzero elements of the matrix are denoted by "$\star$".

$$U = \begin{pmatrix}
\star & \star & \star & \star & \star & \star & \star & \star & \star & \star & \star & \star \\
\star & \star & \star & \star & \star & \star & \star & \star & \star & \star & \star & \star \\
\star & \star & \star & \star & \star & \star & \star & \star & \star & \star & \star & \star \\
\star & \star & \star & \star & \star & \star & \star & \star & \star & \star & \star & \star \\
\star & \star & \star & \star & \star & \star & \star & \star & 0 & 0 & 0 & 0 \\
\star & \star & \star & \star & \star & \star & \star & \star & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \star & \star & \star & \star & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \star & \star & \star & \star & 0 & 0 & 0 & 0 \\
\star & \star & \star & \star & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\star & \star & \star & \star & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \star & \star & \star & \star \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \star & \star & \star & \star
\end{pmatrix}.$$

7 Fast Implementation

Since the basis matrix U is represented as a product of sparse matrices, $U = U_d \times U_{d-1} \times \cdots \times U_1$, it is apparent, that for an arbitrary vector of length n, the application of matrix U can be accomplished in order $O(n)$ arithmetic operations. Thus, the described transform is equally efficient with Haar or TSH transofms. The inverse transform matrix, U^{-1} is factorized as $U^{-1} = U^T = U_1^T \times U_2^T \times U_d^T$ (since U is orthogonal), and thus the inverse transform is of the same complexity.

8 Conclusion

We have considered multi-wavelet bases, that is, bases with a finite set of scaling functions and wavelets. In particular, we have focused on Legendre type of multiwavelets, introduced by Alpert in [1] and generalized later in [4] to the case of arbitrary (non-dyadic) time splitting. The latter can be appropriate for signals with some irregular structure. Knowing this structure, an adaptive wavelet-like basis can be constructed that can lead to a more efficient transform-domain expansion of the signal. Here we have proposed an algorithm for constructing discrete counterparts of such non-dyadic Legendre multi-wavelets. The discrete construction is more convenient, when representation of a function (and its related operators) is based on its values at a finite set of points.

References

1. Alpert, B.: A class of bases in L2 for the sparse representation of integral operators. SIAM J. Math. Anal. **24** (1993) 246–262
2. Alpert, B., Beylkin, G., Coifman, R., Rochlin, V.: Wavelet bases for the fast solution of second-kind integral equations. SIAM Journal on Scientific Computing **14, 1** (1993) 159–184
3. Egiazarian K., Astola, J.: Tree-structured Haar transforms. Journal of Mathematical and Imaging Vision **16** (2002) 267–277
4. Gotchev, A., Egiazarian, K., Astola, J.: On tree-structured legendre multi-wavelet bases. Proc. Int. Workshop on Spectral Methods and Multirate Signal Processing (SMMSP'01) (2001) 51–56
5. Pogossova , E., Egiazarian, K., Astola, J.: Signal de-noising in tree-structured Haar basis. Proc. 3rd International Symposium on Image and Signal Processing and Analysis (ISPA'03) **2** (2003) 736–739
6. Mallat, S.: A Wavelet Tour of Signal Processing. San Diego, Academic Press (1999)
7. Resnikoff, H.L., Wells, R.O.Jr.: Wavelet Analysis. New York, Springer-Verlag (1998)
8. Strela, V., Heller, P.N., Strang, G., Topiwala, P., Heil, C.: The application of mutiwavelet filter banks to signal and image processing. IEEE Trans. on Image Proc. **8,4** (1999) 548–563

Remarks on Calculation of Autocorrelation on Finite Dyadic Groups by Local Transformations of Decision Diagrams

Radomir S. Stanković[1] and Mark G. Karpovsky[2]

[1] Dept. of Computer Science, Faculty of Electronics,
Beogradska 14, 18 000 Niš, Serbia
[2] Dept. of Electrical and Computer Engineering, Boston University,
8 Saint Marry's Street, Boston Ma 02215, USA

Abstract. The paper considers calculation of autocorrelation functions on finite dyadic groups over decision diagrams. The methods exploit recursive structure of both autocorrelation matrices and decision diagrams. First, it is discussed calculation of the autocorrelation through the Wiener-Khinchin theorem implemented over decision diagrams. Then, it is proposed a method for calculation of separate autocorrelation coefficients over decision diagrams with permuted labels at the edges. For the case of restricted memory resources, a procedure with in-place calculations over the decision diagram for the function processed has been defined.

1 Introduction

Autocorrelation is an important operation in signal processing and systems theory [1], [2]. In particular, the autocorrelation on finite dyadic groups, denoted as dyadic autocorrelation B_f, (see Definition 1) is useful in switching theory and design of systems whose inputs and outputs are represented by functions defined in 2^n, $n \in N$ points, including switching functions as an example [6], [8], [7], [11], [14], [16], [17], [19], [20], [25]. Recently, some new applications of dyadic autocorrelation in spectral methods for switching functions, [5], testing of logic networks [9], and optimization of decision diagrams (DDs) for representation of discrete functions have been reported [21].

In this paper, we define and discuss a method for calculation of the dyadic autocorrelation through decision diagrams, the use of which permits processing of functions of a large number of variables. Then, we discussed calculation of separate autocorrelation coefficients over decision diagrams with permuted labels of the edges. In case of restricted memory resources, these calculations can be performed by traversing in a suitable way the decision diagram for the function whose autocorrelation coefficients are required.

2 Background Theory

Denote by C_2^n the finite dyadic group, where $C_2^n = \times_{i=1}^n C_2$, and $C_2 = (\{0,1\}, \oplus)$, where $\oplus$ denotes multiplication modulo 2 (EXOR).

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 301–310, 2005.
© Springer-Verlag Berlin Heidelberg 2005

Definition 1 *For a function* $f : C_2^n \rightarrow R$*, where* R *is the field of real numbers, the autocorrelation* B_f *is defined by* $B_f(\tau) = \sum_{x=0}^{2^n-1} f(x)f(x \oplus \tau)$*, where* $\tau = 0, \ldots, 2^n - 1$*. In binary notation,* $x = (x_1, \ldots, x_n)$ *and* $\tau = (\tau_1, \ldots, \tau_n)$*, where* $x_i, \tau_i \in \{0, 1\}$*.*

In matrix notation, if a given function f and the corresponding autocorrelation function B_f for f are represented by vectors $\mathbf{F} = [f(0), \ldots, f(2^n - 1)]^T$ and $\mathbf{B}_f = [B_f(0), \ldots B_f(2^n - 1)]^T$, respectively, then,

$$\mathbf{B}_f = \mathbf{B}(n)\mathbf{F},$$

where $\mathbf{B}(n)$ is the dyadic autocorrelation matrix for f. The recursive structure of the autocorrelation matrix will be exploited in calculation of the autocorrelation coefficients.

The Walsh transform for functions on C_2^n is defined by the Walsh matrix

$$\mathbf{W}(n) = \bigotimes_{i=1}^{n} \mathbf{W}(1),$$

where $\otimes$ denotes the Kronecker product, and $\mathbf{W}(1) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ is the the basic Walsh matrix [6].

The relationship between the autocorrelation function and Walsh coefficients can be expresses as [6]

$$B_f = 2^n W^{-1}(Wf)^2,$$

where W denotes the Walsh transform operator.

This theorem we are using is called the Wiener-Khinchin theorem in classical Fourier analysis, and by this analogy the same term is used also in this paper. However, it seems that this theorem for the Walsh transform, was formulated for the first time by Franz Pichler in the paper [12], and also, in a mathematically more satisfying paper based on *sal* and *cal* functions in [13].

3 Decision Diagrams

Decision diagrams are data structures providing compact representations of discrete functions defined in a large number of points [22]. In this paper, we assume that a given function f with binary-valued variables is represented by a Multiterminal Binary DD (MTBDD(f)) [4], [22]. A MTBDD is a directed acyclic graph consisting of non-terminal nodes and constant nodes connected by edges. Each node has two outgoing edges labeled by the negative and positive literals $\overline{x}_i$ and x_i of the decision variable assigned to the node. Nodes to which the same variable is assigned form a level in the MTBDD.

If f is a switching binary-valued function, instead of MTBDDs [4], Binary decision diagrams (BDDs) [3] are used, since there are two possible values for constant nodes. MTBDDs and BDDs are derived by the reduction of the Multi-terminal binary decision trees (MTBDTs) and Binary decision trees (BDTs), respectively. The reduction is performed by deleting the redundant information and sharing isomorphic subtrees in the MTBDT, respectively BDT, for a given function f [22]. Notice that in calculations over decision diagrams, the impact of the deleted nodes should be taken into account through the cross points defined as points of intersections of paths from the root node to the constant nodes with the imaginary lines showing levels in decision diagrams, which means lines connecting nodes to which the same decision variable is assigned [24]. Complexity of a decision diagram is usually expressed in terms of the number of non-terminal and constant nodes, called the size of the decision diagram. In this paper, the notion of MTBDTs and MTBDDs will be introduced by the following example.

Example 1. *Fig. 1 shows a MTBDT, the related MTBDD, and the MTBDD of the autocorrelation function* $B_f(\tau)$ *for the function* f *of* $n = 3$ *variables, which is given by the vector* $\mathbf{F} = [0, 0, 1, 2, 3, 3, 3, 3]^T$*. In this figure, we also show the cross points in the MTBDD for* f*.*

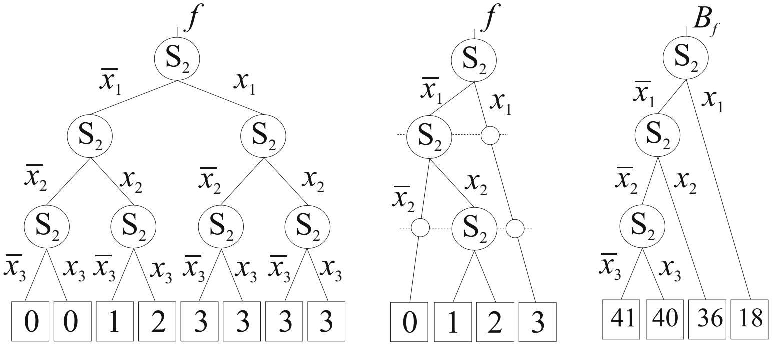


Fig. 1. MTBDT, MTBDD, and the MTBDD for the autocorrelation function for f in Example 1

4 Wiener-Khinchin Theorem over Decision Diagrams

The Walsh spectrum S_f of a given function f represented by a MTBDD is determined by performing at each node and the cross point of the MTBDD(f) the calculations determined by $\mathbf{W}(1)$. For simplicity, we say the nodes and cross points in MTBDD(f) are processed by $\mathbf{W}(1)$. In this way, MTBDD(f) is converted into the MTBDD(S_f). We perform the multiplication of S_f by itself by replacing the values of constant nodes $S_f(i)$ with $S_f^2(i)$ [22]. Then, the MTBDD(B_f) is determined by performing the calculations determined by $\mathbf{W}(1)$ at each node and the cross point of the resulting MTBDD(S_f) followed by the normalization with 2^n, since the Walsh matrix is self-inverse up to the constant 2^{-n}.

4.1 Complexity of the Method

Since in calculation of the Walsh spectrum, we perform an addition and a subtraction at each node and the cross point distributed over n levels, the complexity is $O(2n \cdot size(MTBDD(f)))$. Notice that the number of cross points in a MTBDD is on the average at about 30% of the number of non-terminal nodes [22]. The result of these calculations is the MTBDD(S_f). Then, we perform squaring of the values of constant nodes and perform the inverse transform. Thus, since the Walsh transform is self inverse, the complexity of these calculations is $O(2n \cdot size(MTBDD(S_f)))$. After multiplication with the scaling factor 2^n, the MTBDD(B_f) is derived.

Notice that the size of the MTBDD for the Walsh spectrum is usually greater than that of the MTBDD for functions with a limited number of different values. Since in calculation of the autcorrelation function, MTBDD(f) is converted into a MTBDD(S_f), which is in many cases larger in terms of size than the MTBDD(f), the space complexity of the method is $O(size(MTBDD(S_f)))$.

For an illustration, Table 1 shows the sizes of MTBDDs and Walsh transform decision diagrams (WDDs) [24] for few standard *mcnc* benchmark functions used in logic design. Notice that, due to spectral interpretation of decision diagrams [23], WDDs are actually MTBDDs for the Walsh spectrum, and thus, this table provides a relevant information for these considerations. This table shows the number of inputs (In) of benchmark functions, number of non-terminal nodes (ntn), constant nodes (cn), size (s), and number of paths (paths) in the MTBDDs and WDDs.

Example 2. *For the function f represented by the MTBDD in Fig. 1, the Walsh spectrum is calculated as follows.*

We first process the cross points and the node at the level for x_3. For the left cross point, calculation is trivial since the constant node shows the values 0, the result will be the zero valued vector of order 2. For the completeness of presentation, we show also these calculations

$$\mathbf{W}(1)\begin{bmatrix}0\\0\end{bmatrix} = \begin{bmatrix}0+0\\0-0\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix}.$$

For the node for x_3

$$\mathbf{W}(1)\begin{bmatrix}1\\2\end{bmatrix} = \begin{bmatrix}1+2\\1-2\end{bmatrix} = \begin{bmatrix}3\\-1\end{bmatrix}.$$

Table 1. Characteristics of MTBDDs and WDDs for some benchmark functions

		MTBDD				WDD			
f	In	ntn	cn	s	paths	ntn	cn	s	paths
5xp1	7	127	128	255	128	41	14	55	128
9sym	9	43	3	46	125	101	30	131	224
add4	8	147	31	178	256	36	11	47	37
add5	10	387	63	450	1024	55	13	68	56
apex4	9	446	319	765	450	511	512	1023	512
bw	5	29	24	53	30	31	32	63	32
clip	9	339	33	372	498	449	170	619	464
con1	7	46	5	51	83	83	26	109	96
ex1010	10	899	178	1077	1887	1023	972	1995	1024
mul2	4	13	7	20	14	12	8	20	13
mul3	6	59	26	85	59	30	16	46	31
rd53	5	21	6	27	24	30	13	43	32
rd73	7	57	8	25	96	64	24	88	98
rd84	8	85	9	94	192	118	40	158	193
sao2	10	96	11	107	237	295	70	365	508
sqrt8	8	64	17	81	65	127	54	181	176
xor5	5	15	3	18	22	9	6	15	10
av.	7.55	163.44	49.05	210.27	297.72	167.50	111.72	279.22	201.33

For the right cross point

$$\mathbf{W}(1)\begin{bmatrix}3\\3\end{bmatrix}=\begin{bmatrix}3+3\\3-3\end{bmatrix}=\begin{bmatrix}6\\0\end{bmatrix}.$$

Then, we process the node for x_2

$$\mathbf{W}(1)\circ\begin{bmatrix}\begin{bmatrix}0\\0\end{bmatrix}\\ \\ \begin{bmatrix}3\\-1\end{bmatrix}\end{bmatrix}=\begin{bmatrix}\begin{bmatrix}0\\0\end{bmatrix}+\begin{bmatrix}3\\-1\end{bmatrix}\\ \\ \begin{bmatrix}0\\0\end{bmatrix}-\begin{bmatrix}3\\-1\end{bmatrix}\end{bmatrix}=\begin{bmatrix}3\\-1\\-3\\1\end{bmatrix}$$

where $\circ$ symbolically denotes multiplication of a matrix by a vector consisting of subvectors.

For the cross point at the level for x_2

$$\mathbf{W}(1)\circ\begin{bmatrix}\begin{bmatrix}6\\0\end{bmatrix}\\ \\ \begin{bmatrix}6\\0\end{bmatrix}\end{bmatrix}=\begin{bmatrix}\begin{bmatrix}6\\0\end{bmatrix}+\begin{bmatrix}6\\0\end{bmatrix}\\ \\ \begin{bmatrix}6\\0\end{bmatrix}-\begin{bmatrix}6\\0\end{bmatrix}\end{bmatrix}=\begin{bmatrix}12\\0\\0\\0\end{bmatrix}.$$

For x_1,

$$\mathbf{W}1 \circ \begin{bmatrix} \begin{bmatrix} 3 \\ -1 \\ -3 \\ 1 \end{bmatrix} \\ \\ \begin{bmatrix} 12 \\ 0 \\ 0 \\ 0 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 3 \\ -1 \\ -3 \\ 1 \end{bmatrix} + \begin{bmatrix} 12 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ \\ \begin{bmatrix} 3 \\ -1 \\ -3 \\ 1 \end{bmatrix} - \begin{bmatrix} 12 \\ 0 \\ 0 \\ 0 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 15 \\ -1 \\ -3 \\ 1 \\ -9 \\ -1 \\ -3 \\ 1 \end{bmatrix}.$$

Thus determined vector is multiplied by $1/8$ *to get the Walsh spectrum for* f*.*

Notice that matrix calculations are used for the explanations of the method. In practice, each step of the calculation is represented by a decision diagram which is a subdiagram in a decision diagram representing the Walsh spectrum for the function f*.*

5 In-Place Calculation of Autocorrelation Coefficients

We define a transformation of nodes in MTBDDs that consists of permutation of labels at the outgoing edges as shown in Fig. 2 The i-th row of the autocorrelation matrix is the vector of function values $f(x \oplus i)$, where $\oplus$ denotes the componentwise EXOR over the binary representations for $x = (x_1, \ldots, x_n)$, and $i = (i_1, \ldots, i_n)$. In decision diagrams, this shift of the argument for f implies permutation of labels at the edges of some nodes in the decision diagram for f. Nodes whose edges should be permuted are situated at the levels whose position within the decision diagram corresponds to the position of 1-bits in the binary representation for the row index i.

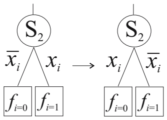


Fig. 2. Transformation of nodes

Example 3. *Fig. 3 shows MTBTDs for the first four rows of the autocorrelation matrix* $\mathbf{B}_f$ *for a function of* $n = 3$ *binary-valued variables.*

The i-th autocorrelation coefficient is calculated by the multiplication of the i-th row of the autocorrelation matrix $\mathbf{B}_f$ by the vector $\mathbf{F}$ of function values for f. When f and rows of $\mathbf{B}_f$ are represented by decision diagrams, it follows that the i-th autocorrelation coefficient is calculated by the multiplication of the

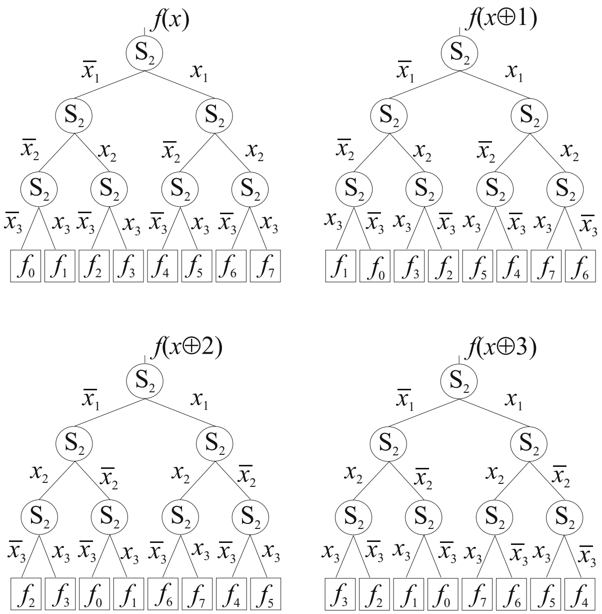

f(x)
S2
$\overline{x}_1$ x_1
$\overline{x}_2$ x_2 $\overline{x}_2$ x_2
$\overline{x}_3$ x_3 $\overline{x}_3$ x_3 $\overline{x}_3$ x_3 $\overline{x}_3$ x_3
f_0 f_1 f_2 f_3 f_4 f_5 f_6 f_7
f(x⊕1)
$\overline{x}_1$ x_1
$\overline{x}_2$ x_2 $\overline{x}_2$ x_2
x_3 $\overline{x}_3$ x_3 $\overline{x}_3$ x_3 $\overline{x}_3$ x_3 $\overline{x}_3$
f_1 f_0 f_3 f_2 f_5 f_4 f_7 f_6
f(x⊕2)
$\overline{x}_1$ x_1
x_2 $\overline{x}_2$ x_2 $\overline{x}_2$
$\overline{x}_3$ x_3 $\overline{x}_3$ x_3 $\overline{x}_3$ x_3 $\overline{x}_3$ x_3
f_2 f_3 f_0 f_1 f_6 f_7 f_4 f_5
f(x⊕3)
$\overline{x}_1$ x_1
x_2 $\overline{x}_2$ x_2 $\overline{x}_2$
x_3 $\overline{x}_3$ x_3 $\overline{x}_3$ x_3 $\overline{x}_3$ x_3 $\overline{x}_3$
f_3 f_2 f_1 f_0 f_7 f_6 f_5 f_4

Fig. 3. MTBDTs for the first four rows of the autocorrelation matrix for $n = 3$

decision diagrams for f and $f(x \oplus i)$. This can be performed by the classical procedure for multiplication of decision diagrams. However, since decision diagrams for $f(x)$ and $f(x \oplus i)$ differ in labels at the edges, in practical programming implementations, calculations can be organized over a single diagram similar as calculations of FFT can be organized in-place [1]. The complexity of calculation is proportional to the number of nodes in the decision diagram for f.

Fig. 4 shows a procedure for in-place calculation of the autocorrelation function through decision diagrams with permuted labels at the edges. In this procedure, f is represented by a MTBDD which is then traversed in such a way to multiply values of constant nodes in the MTBDD for f with the values of constant nodes in the MTBDD for $f(x \oplus \tau)$ and perform the addition of these values. The way of traversing is determined by the binary components τ_i, $i = 0, 1, \ldots, n-1$ of τ. A flag is associate to each non-terminal node, to show if the node was already traversed. In this manner, the coefficient $B_f(\tau)$ is calculated. The procedure has to be repeated for each coefficient.

5.1 Complexity of In-Place Calculations

In-place calculations are performed over the MTBDD(f) and, therefore, the space complexity is $O(size(MTBDD(f)))$. Since for each coefficient we perform a multiplication at each constant node and an addition at each non-terminal node, the number of multiplications is $O(cn)$, and the number of additions is

```
int AUTOCORREL($*node1$, $*node2$, $level$)
{
  $r = level - node \rightarrow level$
      if ($node$ NOT TERMINAL)
    {
        if ($node \rightarrow flag = 0$)
        {
        if($\tau_i = 1$)
        {
        $pom1 = node2 \rightarrow right$
        $pom2 = node2 \rightarrow left$
        }
        else
        {
        $pom1 = node2 \rightarrow left$
        $pom2 = node2 \rightarrow right$
        }
        $i = i + 1$
        $a =$ AUTOCORREL($node1 \rightarrow left, pom1$)
        + AUTOCORREL($node1 \rightarrow right, pom2$)
        $node \rightarrow sub - value$
        $node \rightarrow flag = 1$
        return ($2^{r-1} \cdot a$)
      }
      else
      {
      return $node \rightarrow sub - value$
      }
    }
    else
    $a = node1 \rightarrow value \cdot node2 \rightarrow value$
     return ($2^{r-1} \cdot a$)
}
End of pseudocode.
```

Fig. 4. Calculation of the autocorrelation coefficient $B_f(\tau)$

$O(ntn)$, where *cn* and *ntn* are the number of constant and non-terminal nodes, respectively. Therefore, the total complexity of in-place calculation of an autocorrelation coefficient is $O(size(MTBDD(f)))$. Table 1 shows number of constant nodes and non-terminal nodes in the considered set of benchmark functions. The procedure is performed for each coefficient. Thus, it is suitable for calculation of a single coefficient or a subset of coefficients.

6 Closing Remarks

In this paper, we discussed calculation of autocorrelation functions over decision diagram representations of functions with binary-valued variables. Two approaches are considered, calculation of the autocorrelation functions by performing Wiener-Khinchin theorem over decision diagrams, and in-place calculations by decision diagrams with permuted labels at the edges. In the implementation of the Wiener-Khinchin theorem, the complete autocorrelation function is determined and represented by a decision diagram. The time complexity of calculations is $O(2n \cdot size(MTBDD(f)))$, and since the interim calculations involve determination of the Walsh spectrum, which is also represented by a decision diagram, the space complexity is maximum of $O(size(MTBDD(f))$ and $O(size(MTBDD(S_f)))$.

Calculation over decision diagrams with permuted edges permits determination of a single coefficient with both space and time complexity proportional to the size of the diagram for a given function f.

Acknowledgments

This work was supported by the Academy of Finland, Finnish Center of Excellence Programme, Grant No. 44876.

References

1. Agaian, S., Astola, J., Egiazarian, K., *Binary Polynomial Transforms and Nonlinear Digital Filters*, Marcel Dekker, 1995.
2. Bracewell, R., "The Autocorrelation Function", in *The Fourier Transform and Its Applications*, 3rd ed., McGraw-Hill, New York, 1999, 40-45.
3. Bryant, R.E., “Graph-based algorithms for Boolean functions manipulation,” *IEEE Trans. Comput.*, Vol. C-35, No. 8, 1986, 667-691.
4. Clarke. E, M., M.C., Millan, K.L., Zhao, X., Fujita, M., “Spectral transforms for extremely large Boolean functions”, in Kebschull, U., Schubert, E., Rosenstiel, W., Eds., *Proc. IFIP WG 10.5 Workshop on Applications of the Reed-Muller Expression in circuit Design*, Hamburg, Germany, September 16-17, 1993, 86-90. Workshop Reed-Muller'93, 86-90.
5. Falkowski, B.J., Chang, C.H., "Properties and methods of calculating generalized arithmetic and adding transforms", *IEE Proc. Circuits, Devices and Systems*, Vol. 144, No. 5, 1997, 249-258.
6. Karpovsky, M.G., *Finite Orthogonal Series in the Design of Digital Devices*, John Wiley, 1976.
7. Karpovsky, M.G., (ed.), *Spectral Techniques and Fault Detection*, Academic Press, 1985, 35-90.
8. Karpovsky, M.G., Moskalev, E.S., "Utilization of autocorrelation characteristics for the realization of systems of logical functions", *Avtomatika i Telmekhanika*, No. 2, 1970, 83-90, English translation *Automatic and Remote Control*, Vol. 31, 1970, 342-350.
9. Karpovsky, M.G., Stanković, R.S., Astola, J.T., "Spectral techniques for design and testing of computer hardware", *Proc. Int. Worksop on Spectral Techniques in Logic Design, SPECLOG-2000*, Tampere, Finland, June 2-3, 2000, 1-34.
10. Meinel, Ch., Somenzi, F., Tehobald, T., "Linear shifting of decision diagrams and its application in synthesis", *IEEE Trans. CAD*, Vol. 19, No. 5, 2000, 521-533.

11. Pichler, F., “Walsh functions and linear system theory”, Proc. Applic. Walsh Functions, Washington, D.C., 1970, 175-182.
12. Pichler, F., *Some Aspects of a Theory of Correlation with respect to Walsh Harmonic Analysis*, Techn. Report. Dept. of Electrical Engineering, University of Maryland, Washington, DC, Report R-70-11, August 1970.
13. Franz Pichler, "Walsh-Fourier Synthese optimaler Filter", *AEÜ*, Band 24, 1970, Heft 7/8, 350-360.
14. Pichler, F., “Realizations of Prigogine’s Λ-transform by dyadic convolution”, in Trappl, R., Horn, W., (eds.), Austrian Society for Cybernetic Studies, ISBN 385206127X, 1992.
15. Rice, J.E., *Autocorrelation Ceffıcients in Representation and Classification of Switching Functions*, Ph.D. Disertation, University of Victoria, Victoria, B.C., Canada, 2003.
16. Rice, J.E., Muzio, J.C., "Methods for calculating autocorrelation coefficients", in *Proc. 4th Int. Workshop on Boolean Problems, (IWSBP2000)*, 2000, 6976.
17. Rice, J.E., Muzio, J.C., "Use of autocorrelation function in the classification of switching functions", *Euromicro Symposium on Digital System Design*, 2002, 244-251.
18. Rice, J.E., Muzio, J.C., "Properties of autocorrelation coefficients", *Proc. IEEE Pacific Rim Conf. on Comunications, Computers and Signal Processing*, 2003.
19. Rice, J.E., Muzio, J.C., "On the use of autocorrelation coefficients in the identification of three-level decompositions", *Proc. PAC 2003*, 2003, 577-580.
20. Rice, J.E., Muzio, J.C., "On the use of autocorrelation coefficients in the identification of three-level decompositions", *Proc. Int. Workshop on Logic Synthesis, (IWLS 2003)*, 2003.
21. Rice, J., Serra, M., Muzio, J.C., "The use of autocorrelation coefficients for variable ordering for ROBDDs", *Proc. 4th Int. Workshop on Applications of Reed-Muller Expansion in Circuit Design*, Victoria, Canada, August 20-21, 1999, 185-196.
22. Sasao, T., Fujita, M., (eds.), *Representations of Discrete Functions*, Kluwer, 1996.
23. Stanković, R.S., Astola, J.T., *Spectral Interpretation of Decision Diagrams*, Springer, 2003.
24. Stanković, R.S., Sasao, T., Moraga, C., “Spectral transform decision diagrams”, in: T. Sasao, M. Fujita, Ed., *Representations of Discrete Functions*, Kluwer Academic Publishers, 1996, 55-92.
25. Tomczuk, R., *Autocorrelation and Decomposition Methods in Combinational Logic Design*, PhD thesis, University of Victoria, 1996.

A New Pseudo-Random Generator Based on Gollmann Cascades of Baker-Register-Machines

Dominik Jochinger and Franz Pichler

Systems Theory, Johannes Kepler University Linz,
Altenberger Str. 69, A-4040 Linz, Austria

Abstract. In this paper, we present a new pseudo-random sequence generator, constructed by the generalized discrete Baker transformation. This new generator is called Cascaded Baker Register Machine (CBRM), which uses the sensitivity of chaotic behaviour and allows the application of automata- and shift-register theory. It is shown that a CBRM has good properties of randomness, such as large periods and high linear complexity. It can provide high cryptographic security with fast encryption speed, and can be realized effectively by both hardware and software.

1 Introduction

Chaos theory has been established since 1970s by many different research areas, such as physics, mathematics, biology and chemistry, etc. The most well-known characteristics of chaos is the so-called "butterfly-effect" (the sensitivity to the initial conditions), and the pseudo-randomness generated by deterministic equations. Many fundamental characteristics of chaos, such as the mixing property and the sensitivity to initial conditions, can be connected with "confusion" and "diffusion" property in good ciphers. The well-mixing transformations (permutations) used in secrecy systems can be constructed by the basic "stretch-and-fold" mechanism of the Baker transformation, which implies chaos. The first paper about ciphers with dynamical systems was Wolfram's paper published in Crypto'85 [1], in which he introduced a cellular automata based cryptosystem.

In this paper, we propose a new pseudo-random-generator based on the generalized discrete Baker transformation, which is used for creating complex, key-dependent permutations. Most of today's symmetric encryption schemes rely on complex substitution while the important role of permutation is neglected.

We suggest a 256-bit key to define the initial condition of the generator, therefore a brute force attack by key exhaustion seems to be impossible.

2 The Generalized Discrete Baker Transformation

We introduce the generalized discrete Baker transformation following the paper of Pichler and Scharinger [2]:

By N_0^n we denote the subset $N_0^n := \{0,1,2,...,n-1\}$ of integers. With π we denote a list of non-negative integers $\pi = \{n_1, n_2, ..., n_k\}$ with the following properties:

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 311 – 316, 2005.
© Springer-Verlag Berlin Heidelberg 2005

(1) $\sum_{i=1}^{i\le k} n_i = n$ and

(2)each number n_s ($s = 1, 2, \ldots, k$) divides n

The transformation $T_{n,\pi} : N_0^n \times N_0^n \to N_0^n \times N_0^n$ is a discrete finite version of the generalized baker transformation. $T_{n,\pi}$ is defined as follow:

Let the numbers q_s ($s = 1, 2, \ldots, k$) be defined by $q_s = \frac{n}{n_s}$. Then

$$T_{n,\pi}(x, y) = \left(q_s\left(x - N_s\right) + y \bmod q_s, \frac{1}{q_s}\left(y - y \bmod q_s\right) + N_s \right)$$

for $(x, y) \in [N_s, N_s + n_s) \times N_0^n$ where $N_1 := 0$ and $N_s := n_1 + \ldots + n_{s-1}$ for s = 2, …, k.

It is easy to see, that $T_{n,\pi}$ maps from $N_0^n \times N_0^n$ the vertical strips $[N_s, N_s + n_s) \times N_0^n$ ($s = 1, 2, \ldots, k$) into the corresponding horizontal strips $N_0^n \times [N_s, N_s + n_s)$.

3 Structure of Baker-Permutation

The generalized discrete Baker transformation is a permutation of n^2 data items. Figure 1 gives a schematic view of a Baker permutation with partition $\pi = \{2, 2\}$ and $n = 4$.

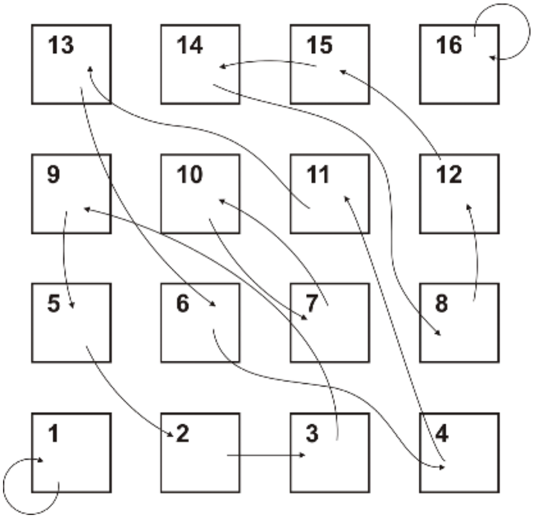


Fig. 1. Baker-permutation

The properties of the permutations defined by the discrete Baker-transformation correspond to a typical random permutation in the sense of Feller [5], [6].

Computer experiments, done for the Baker-permutation with many different partition keys π, demonstrate that the average length of cycles and the average number of different cycles have values similar to those for random permutations.

4 Cascaded Baker Register Machine (CBRM)

The building blocks of a CBRM are given by Baker-Register-Machines (BRM). The blockdiagram of a BRM is shown in figure 2. A BRM consists of n register cells and a binary clock-controlled input x.

The BRM is initialized with the n-bit parameter ($K1$) and the partition-parameter π.

Figure 3 shows a single stage of the CBRM. The input bit a_t clocks the BRM, and then is XORed to the output from the BRM. The *delay* assures that the addition takes place after each clock step.

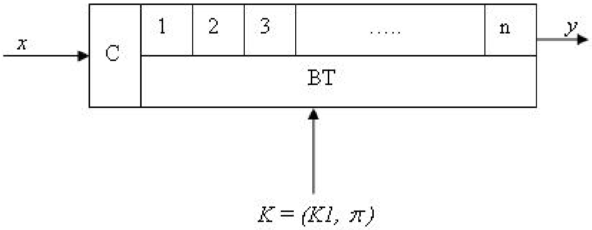


Fig. 2. Blockdiagram of a BRM

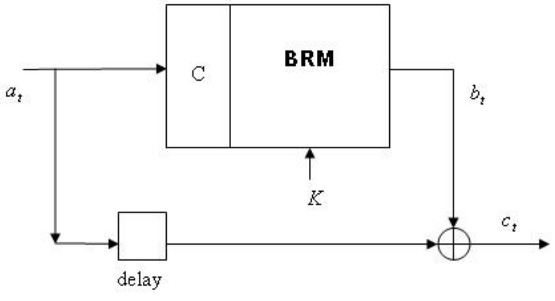


Fig. 3. A stage of the cascade

A nice representation of a BRM can be done by using several feedback-shift-registers. This can be seen as follows.

To determine the cycles of the Baker-transformation n^2 steps are needed. Each cycle, however, can be realized by a feedback-shift-register of cycle length.

To achieve a scalar output from the BRM the feedback-shift-registers of a BRM with exception of the two registers of length 1 (which correspond to the fix points) are output-coupled by a XOR-Operation.

Only one XOR-operation and several shift operations are needed for a single iteration of a Baker-Register-Machine.

The CBRM is defined similar to the well known Gollmann cascade [7], [8] of *k* stages. It consists of a sequence of *k* stages of Baker-Register-Machines (BRM), of same length.

By irregular clocking of the individual BRMs and by the cascaded structure we obtain non-linear effects in the output sequence.

A Cascaded Baker Register Machine (CBRM) with 3 stages is shown in figure 4.

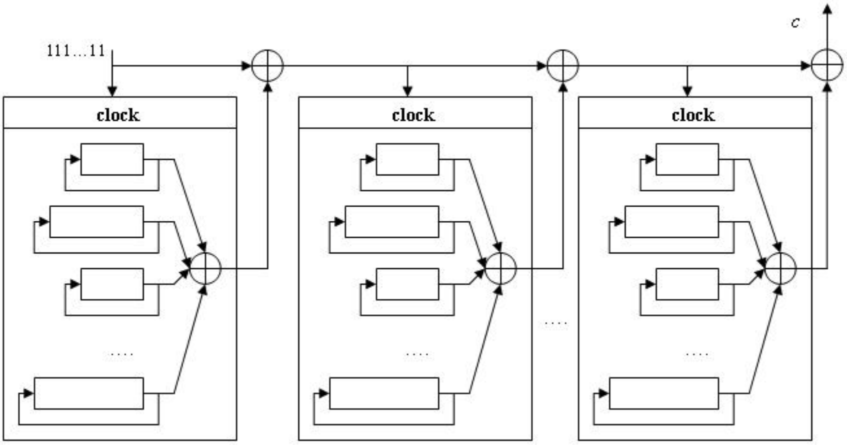


Fig. 4. Cascaded Baker Register Machine with 3 stages

5 Selection of the Partition-Parameter

The partition-parameter π can serve as a customer specific key. It is needed to initialize a Baker-Register-Machine. A selected key π determines the cyclic length of the BRM. Not every possible partition-key leads to a secure cyclic length. The table below shows some acceptable partitions of a 16 × 16 Baker map and their cyclic length.

Table 1. Some acceptable partitions

Partition	Cyclic length
(8,4,4)	622440
(4,4,8)	622440
(4,4,4,2,2)	1680
(4,4,2,2,4)	55440
(4,2,2,4,4)	55440
(2,2,4,2,2,2,2)	55440
(2,2,2,2,4,2,2)	55440

6 Randomness Properties

An analysis of the CBRM shows that it has many good statistical properties. For example n-gram distributions are uniform.

Another nice property of a BRM is that the generated sequences are highly sensitive to the initial key. By statistical analysis, we found that changing one bit in the encryption key *K1* caused 49.4% of the bits to change in the corresponding cipher-image. Theoretically half of the bits should change, in order to hide any information about the key from leaking.

The linear complexity profile of the CBRM algorithm is staying very close to be optimal.

The cycle length (period) of the CBRM depends on the partition key, as mentioned above. It is difficult to give a general formula to compute the exact cycle length of the CBRM. However, for the case that the length of the individual shift registers is known, a formula for this computation can be given as follows:

Let *r* be the number of the feedback-shift-registers that realize the Baker-Register-Machine, σ_i (i =1, 2, …, r) the length of the shift-register *i*, and k the number of cascaded stages. Then the cycle length of the CBRM is given by:

$$L = \mathrm{lcm}\left(\sigma_1, \sigma_2, \ldots., \sigma_r\right)^k$$

As an example L for n = 256, k = 10, π ={4, 4, 8}. We get L = 622440^{10} $\approx 8.7 \cdot 10^{57}$. Such a length is enough for many practical applications.

7 Conclusion

In this paper, we propose a new pseudo random generator based on the generalized discrete Baker transformation, which has good properties with respect to speed and security.

It can be demonstrated that the CBRM has good statistical properties as required in cryptography.

In the future, we will investigate further facts concerning cryptographic security (e.g. resistance against state identification attacks).

CBRM’s have been simulated and tested by VHDL (hardware) and JAVA™ (software) implementation.

However, to evaluate the possibility for the application of the Baker-transformation in stream ciphering further cryptological research is needed.

Note: This paper is a part of the progressing PhD thesis of the first author.

References

1. Stephen Wolfram, “Cryptography with cellular automata”, In Advances in Cryptology – Crypto’85, Lecture Notes in Computer Science vol. 0218, pp. 429-432, Springer-Verlag, Berlin, 1985.
2. Franz Pichler, Josef Scharinger, “Ciphering by Bernoulli-Shifts in Finite Abelian Groups”, Contributions to General Algebra (ed. G. Pilz), Hölder-Pichler-Tempsky, pp. 249-256, Wien, 1995.

3. Josef Scharinger, “Experimentelle harmonische Anaylse von Bäcker-dynamischen 2D Systemen und ihre Anwendung in der Kryptographie“, PhD thesis, Johannes Kepler Universität Linz, 1994.
4. Josef Scharinger, Franz Pichler, “Bernoulli-Chiffren“, Elektrotechnik und Informationstechnik (e&i), 111. Jg, 1994, Heft 11, pp. 576-582.
5. W. Feller, “An Introduction to Probability Theory and Its Applications”, pp. 242-243, John Wiley, New York, 1957.
6. N. J. A. Sloane, ”Encrypting by Random Rotations“, Cryptography (ed. Thomas Beth) , Lecture Notes in Computer Science 149, pp. 71-128.
7. Dieter Gollmann, “Kaskadenschaltungen taktgesteuerter Schieberegister als Pseudozufallszahlengeneratoren“, PhD thesis, Johannes Kepler Universität Linz, VWGÖ-Verlag, Wien, 1986.
8. Bruce Schneier, ”Applied Cryptography, Protocols, Algorithms and Source Code in C, Second Edition“, John Wiley and Sons, pp. 445-446, 1995.

An Excellent Permutation Operator for Cryptographic Applications

Josef Scharinger

Johannes Kepler University,
Institute of Computational Perception, 4040 Linz, Austria
`Josef.Scharinger@jku.at`

Abstract. Permutations are a core component of almost every cipher. No matter if we consider the DES, AES or most of the other encryption algorithms relevant nowadays, we always find permutation operators as essential building blocks inside. In this contribution we will introduce key-dependent permutation operators of provably excellent quality inspired by chaotic Kolmogorov flows.

From chaotic systems theory it is known that the class of Kolmogorov flows exhibits the highest degree of instability among all dynamical systems. As will be derived and proven in detail in this paper, these outstanding properties make them a perfect inspiration for developing a novel class of strong cryptographic permutation operators.

1 Introduction

In recent years chaos theory has definitely been among the hot topics in systems theory. Remarkable progress has been made in the analysis of chaotic systems and up to a certain extent also in their application. Nevertheless, success in applications lags progress in analysis. This may well be related to the fact that many of the promising systems are defined to act on a continuous phase space and it is often quite difficult or even impossible to find discrete counterparts that preserve all the nice features present in the continuous case.

In this contribution we focus on chaotic Kolmogorov systems [6]. In the well-established continuous form they provide a family of highly unstable systems where it has been proven [1,3] that every member of this family provides perfect mixing of the underlying phase space which makes them a tempting choice for realizing excellent permutation operators. However, for practical computer applications such a permutation operator is only useful if it can be applied to mixing elements arranged on a discrete grid where we have to deal with integer grid positions.

It is the main purpose of this paper to show that it is possible to derive adequate discrete counterparts for classical continuous Kolmogorov systems. Additionally we provide a detailed analysis under which criteria these novel discrete Kolmogorov systems offer high-quality permutation operators. Availability of such discrete permutation systems will finally be utilized to sketch several examples of potential applications for important tasks in information security such as symmetric block ciphering, message digest computation, or copyright protection via digital watermarking.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 317–326, 2005.
© Springer-Verlag Berlin Heidelberg 2005

2 Chaotic Kolmogorov Systems

2.1 Continuous Kolmogorov Systems

Continuous Kolmogorov systems [1,3,6] act as permutation operators upon the unit square. Figure 1 is intended to give a notion of the dynamics associated with a specific Kolmogorov system parameterized by the partition $\pi = (\frac{1}{3}, \frac{1}{2}, \frac{1}{6})$. As can be seen, the unit square is first partitioned into vertical strips which are then stretched in the horizontal and squeezed in the vertical direction and finally stacked atop of each other. Just after a few applications (see Fig. 1 from top left to bottom right) this iterated stretching, squeezing and folding achieves perfect mixing of the elements within the state space.

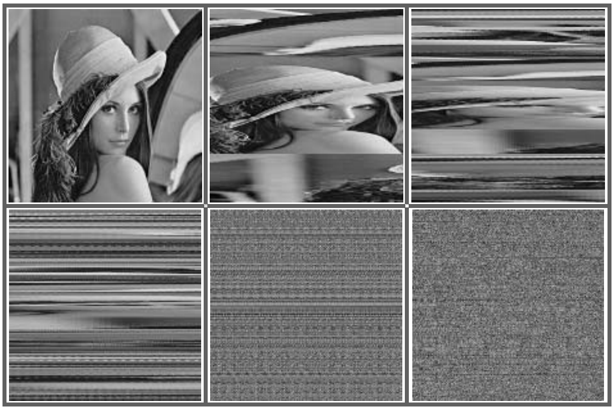


Fig. 1. Illustrating the chaotic and mixing dynamics associated when iterating a Kolmogorov system

Formally this process of stretching, squeezing and folding is specified as follows. Given a partition $\pi = (p_1, p_2, \ldots, p_k)$, $0 < p_i < 1$ and $\sum_{i=1}^{k} p_i = 1$ of the unit interval $\mathbb{U}$ and stretching and squeezing factors defined by $q_i = \frac{1}{p_i}$. Furthermore, let F_i defined by $F_1 = 0$ and $F_i = F_{i-1} + p_{i-1}$ denote the left border of the vertical strip containing the point $(x, y) \in \mathbb{E}$ to transform. Then the continuous Kolmogorov system T_π will move $(x, y) \in [F_i, F_i + p_i) \times [0, 1)$ to the position

$$T_\pi(x, y) = (q_i(x - F_i), \frac{y}{q_i} + F_i). \tag{1}$$

2.2 Discrete Kolmogorov Systems

In our notation a specific discrete Kolmogorov system for permuting a data block of dimensions $n \times n$ is defined by a list $\delta = (n_1, n_2, \ldots, n_k)$, $0 < n_i < n$ and $\sum_{i=1}^{k} n_i = n$ of positive integers that adhere to the restriction that all $n_i \in \delta$ must partition the side length n. Furthermore let the quantities q_i be defined by

$q_i = \frac{n}{n_i}$ and let N_i specified by $N_1 = 0$ and $N_i = N_{i-1} + n_{i-1}$ denote the left border of the vertical strip that contains the point (x, y) to transform. Then the discrete Kolmogorov system $T_{n,\delta}$ will move the point $(x, y) \in [N_i, N_i + n_i) \times [0, n)$ to the position

$$T_{n,\delta}(x, y) = (q_i(x - N_i) + (y \bmod q_i), (y \operatorname{div} q_i) + N_i). \qquad (2)$$

The restriction to integral stretching- and squeezing factors is necessary to keep resultant points at integer positions within the $n \times n$ grid. Use of the div (division of positive integers a and b delivering $\lfloor \frac{a}{b} \rfloor$) and mod (remainder when dividing positive integers) operation ensures that points in $n \times n$ are mapped onto each other in a bijective and reversible manner.

2.3 Important Properties

Kolmogorov systems tend to permute elements of the state space in a chaotic non-linear and apparently random fashion. After a sufficient number of iterations it becomes extremely hard for an observer to deduce the initial state of a Kolmogorov system from its final state. To be more specific, Kolmogorov systems offer very unique properties that are explained in more detail in the sequel.

Ergodicity. Ergodicity is important for a system that is to be applied in cryptography because it stands as a synonym for confusion. Informally speaking and expressed in terms of permutation systems, ergodicity stands for the property that almost any initial point will move to almost any other position in state space with equal probability as the system evolves in time. In other words there is no statistical way to predict the initial from the final position or vice versa.

Ergodicity of continuous Kolmogorov systems has been proven long ago [1]. As for discrete Kolmogorov systems, we have no knowledge that anyone has succeeded in defining them in a way such that ergodicity can be shown. In the sequel we derive necessary and sufficient conditions on the number of iterations necessary to ensure ergodicity of discrete Kolmogorov systems as introduced by equation 2. Note that this way a constructive proof of ergodicity is achieved.

In the following we restrict attention to the practically most relevant case of $n = p^m$ being an integral power of a prime p. The discrete Kolmogorov system T_{n,δ_r} is defined by the list $\delta_r = (n_{1r}, n_{2r}, \ldots, n_{k_r r})$ of length k_r containing the positive integers to be used as key in round r. As mentioned before there are the restrictions $1 \leq i \leq k_r$, $0 < n_{ir} < n$, $\sum_{i=1}^{k_r} n_{ir} = n$ and the constraint that all $n_{ir} \in \delta_r$ must partition the side length n.

Furthermore let the stretching and squeezing factors q_{ir} to use for vertical strip number i in round number r be defined by $q_{ir} = \frac{n}{n_{ir}}$. This results in quantities q_{ir}, $q_{ir} \geq p$ that also have to be integral powers of p because of the divisibility assumption made.

Consider an arbitrary point $(x, y) \in [N_{ir}, N_{ir} + n_{ir}) \times [0, n)$ in vertical strip number i to be transformed in round number r under the influence of the key δ_r (see equation 2 and figure 1). Coordinates x and y can then be expressed by q_{ir}-adic representations of length $t_{ir} = \lceil \log_{q_{ir}} n \rceil$ by $x = \sum_{j=1}^{t_{ir}} x_{jr} (q_{ir})^{t_{ir}-j}$ and $y = \sum_{j=1}^{t_{ir}} y_{jr} (q_{ir})^{t_{ir}-j}$. Similarly N_{ir} can be expanded according to $N_{ir} = \sum_{j=1}^{t_{ir}} Ni_{jr} (q_{ir})^{t_{ir}-j}$ and $x - N_{ir}$ may be expressed as $x - N_{ir} = \sum_{j=1}^{t_{ir}} xm_{jr}$ $(q_{ir})^{t_{ir}-j}$. Obviously x is the sum of $x - N_{ir}$ and N_{ir}.

To clarify these relations, the following illustration should be helpful. Please note that while in the representation of x the most significant position stands on the right side, the most significant position in the q_{ir}-adic representation of y is found on the left side. This arrangement has been made so that the subsequent transformation can essentially be depicted as a cyclic right shift by one position.

x

$x_{t_{ir}r}$	...	x_{3r}	x_{2r}	x_{1r}

$xm_{t_{ir}r}$	...	xm_{3r}	xm_{2r}	0
$Ni_{t_{ir}r}$	...	Ni_{3r}	Ni_{2r}	Ni_{1r}

y

y_{1r}	y_{2r}	y_{3r}	...	$y_{t_{ir}r}$

y_{1r}	y_{2r}	y_{3r}	...	$y_{t_{ir}r}$
0	0	0	...	0

According to equation 2 application of T_{n,δ_r} will move the point (x, y) to a new position $(x', y') = T_{n,\delta_r}(x, y)$ with coordinates $x' = q_{ir}(x - N_{ir}) + (y \bmod q_{ir})$ and $y' = (y \operatorname{div} q_{ir}) + N_{ir}$, as made clear by the subsequent figure.

x'

$y_{t_{ir}r}$	...	xm_{4r}	xm_{3r}	xm_{2r}
0	...	0	0	0

y'

0	y_{1r}	y_{2r}	...	$y_{(t_{ir}-1)r}$
Ni_{1r}	Ni_{2r}	Ni_{3r}	...	$Ni_{t_{ir}r}$

Suppose that lists δ_r are chosen independently and at random[1]. Neglecting the constraint $N_{ir} \leq x$ which follows from the fact that N_{ir} is the left border of the vertical strip containing the point (x, y) for a moment, the proof of ergodicity becomes straightforward. N_{ir} adds random q_{ir}-bits to all the q_{ir}-bits of y' yielding a random value for the new y-coordinate in one step. Cyclically shifting the least significant position of the y-coordinate to the least significant position in the x-coordinate and shifting these random q_{ir}-bits towards more significant positions in the x-coordinate ensures that after at most an additional $\max_{i=1}^{k_r} t_{ir} \leq m$ iterations the transformed point can move to almost any other position in state space with equal probability. Thus ergodicity is achieved after at most $m + 1$ iterations.

---

[1] This is a common assumption whenever proving specific properties of iterated cryptographic schemes. Round keys are generally supposed to be random and independent.

Now let us pay attention to the constraint $N_{ir} \leq x$. A moment of thought reveals that the worst non-trivial point that will need the largest number of rounds until being able to move to any position has a x-coordinate of 0 and a y-coordinate where just y_{1r} is different from zero. Then it takes at most $m+1$ iterations until the second-least significant q_{ir}-bit in the x-coordinate is set and the least significant q_{ir}-bit in N_{ir} (and also in the x-coordinate!) may assume any random value. By shifting q_{ir}-bits towards more significant positions in the x-coordinate every iteration causes one additional position in x to become random and by adding N_{ir} the same applies to the y-coordinate. This way it is guaranteed that after another at most $m-1$ iterations ergodicity is achieved after at most $2m$ steps in total.

Theorem 1. *Let the side-length $n = p^m$ be given as integral power of a prime p. Then the discrete Kolmogorov system T_{n,δ_r} as defined in equation 2 is ergodic provided that at least $2m$ iterations are performed and lists δ_r used in every step r are chosen independently and at random.*

In the discussion above we have noted that the restriction $N_{ir} \leq x$ to observe in every step significantly increases the number of iterations necessary until an initial point can move to any other position. Particularly points with small (zero) x-coordinate need a long time until exhibiting ergodic behaviour. However, a simple trick can help a lot in reducing the number of iterations necessary to achieve ergodicity of the underlying system: after every discrete Kolmogorov permutation round just apply a cyclic shift by $\frac{n}{2} - 1$ to the elements in the $n \times n$ array. This corresponds to adding $\frac{n}{2} - 1$ modulo n to every x-coordinate and helps points with initially small x-coordinates to move to any other position in a reduced number of rounds. Additionally this simple trick also solves the problems associated with the fixed points $(0, 0)$ and $(n-1, n-1)$ so that not just almost all points can move to almost any position but really all of the $n \times n$ points will have ergodic behaviour.

Exponential Divergence. Informally speaking and expressed in terms of permutation systems, exponential divergence implies that neighboring points contained in the same subspace of the state space (e.g. points of the same vertical strip corresponding to the same block of the defining partition) diverge at an exponential rate. This way even highly correlated points in input blocks will quickly loose correlations and structures present in input data will soon disappear.

Proving exponential divergence of specific discrete Kolmogorov systems can proceed using similar arguments as applied in proving ergodicity of discrete Kolmogorv systems. Specifically we derived the the following theorem [12].

Theorem 2. *Let the side-length $n = p^m$ be given as integral power of a prime p. Then the discrete Kolmogorov system T_{n,δ_r} as defined in equation 2 exhibits exponential divergence of points contained in the same blocks defined by partitions δ_r ensuring that after at most $2m-1$ iterations arbitrary non-zero deviations between initial points have propagated at least once to the most significant position in the x-coordinate.*

Mixing Property. Informally speaking and expressed in terms of permutation systems, fulfillment of the mixing property implies that any subspace of the state space will dissipate uniformly over the whole state space. Obviously this is an even stronger requirement than ergodicity because it does not only imply that almost any point will move to almost any position in state space with equal probability but also that distances between neighboring points within certain subspaces will become random as the system evolves in time.

Combining results derived in proving ergodicity and exponential divergence of discrete Kolmogorov systems, we have proven the following theorem [11].

Theorem 3. *Let the side-length* $n = p^m$ *be given as integral power of a prime* p*. Then the discrete Kolmogorov system* T_{n,δ_r} *as defined in equation 2 is mixing provided that at least* $4m$ *iterations are performed and lists* δ_r *used in every step* r *are chosen independently and at random.*

2.4 Analysis Summary

Summarizing the preceding discussion, a simple law on the conditions necessary to ensure that discrete Kolmogorov systems generate high-quality permutations can be stated as follows:

Theorem 4. *Let the side-length* $n = p^m$ *be given as integral power of a prime* p*. Then the discrete Kolmogorov system* T_{n,δ_r} *as defined in equation 2 fulfills the properties of ergodicity, exponential divergence and mixing provided that at least* $4m$ *iterations are performed and lists* δ_r *used in every step* r *are chosen independently and at random.*

Based on this theorem it is well justified to claim that the permutation operator developed in this contribution is indeed an excellent key-dependent permutation operator for cryptographic applications.

3 Applications

As shown in our analysis section, discrete chaotic Kolmogorov systems offer perfect permutation operators. In this section we would like to emphasize relevance of this analysis by giving several examples showing that discrete Kolmogorov systems can successfully be applied to many important problems encountered in communication security. Due to the limited space available, description must be restricted to just outlining some examples in symmetric encryption, secure hashing, password based access control, and digital image watermarking.

3.1 Efficient Block Ciphering

The structure of iterated symmetric product ciphers [13] which perform a blockwise encryption of the plaintext input to the system by repeated intertwined application of r round of permutations and substitutions can be observed from

Fig. 2. Input to the system is a block of plaintext and a pass-phrase. From this key the internal key management derives individual keys and supplies them to the various rounds. Every round applies one permutation and one substitution operation to the output of the previous round (initially the plaintext block). After r rounds, the output of the final round gives the ciphertext output by the r-round product cipher.

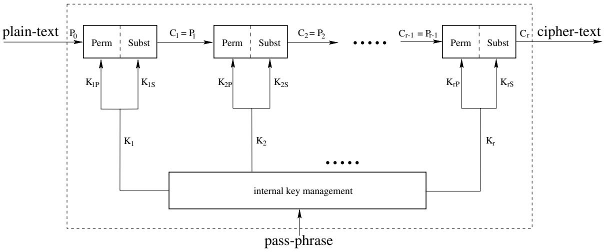


Fig. 2. Structure of an r-round product cipher

The role for discrete Kolmogorov systems within this framework is immediate to see. We use them as high-quality permutation operators for implementing the permutations needed. When complemented with an adequate substitution operator, this approach can deliver very strong and efficient ciphers. More details on that matter can e.g. be found in [10].

3.2 Cryptographic Message Digests

To provide integrity [8] and authenticity [7] in secure communications applications at reasonable computational costs, efficient and strong cryptographic hash functions are needed. Our approach to compute a message digest based on discrete chaotic Kolmogorov systems runs as follows.

First a 16×16 square array of bits is initialized with 256 pseudo-random bits (128 zeros, 128 ones) taken from the after-comma binary expansion of some "magic" constants (π, e, golden ratio ϕ, $\sqrt{2}$, $\sqrt{5}$, etc.) as done in almost any cryptographic hash function. Taken line-by-line or column-by-column, this provides the initial 256 bit message digest MD_0.

After initialization, in every step $t = 1, 2, \ldots$ the message digest MD_{t-1} is updated by processing the message in blocks W_t of 256 bit each. Since message lengths are usually not a multiple of 256, padding the last block with arbitrary constant bits may be necessary.

Now these 256 message bits are XORed with the current 256 bit message digest to obtain $X_t = W_t \oplus MD_{t-1}$. This step ensures that any block contains approximately an equal number of zeros and ones, regardless of the message block (which could be entirely zero etc.).

To maximize input avalanche effects, the 8 32-bit words $X_t(i)$ $(0 \leq i \leq 7)$ are processed according to a linear recurrence relation. First a forward dissipation

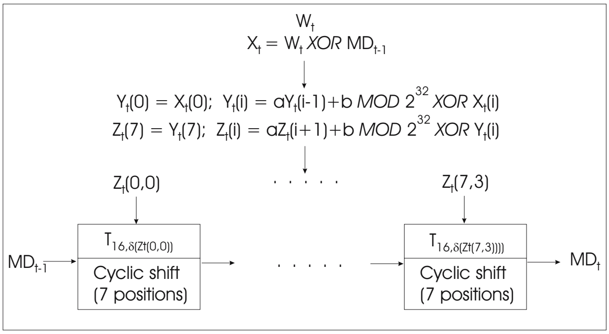

W_t
$X_t = W_t$ XOR MD_{t-1}
$Y_t(0) = X_t(0)$; $Y_t(i) = aY_t(i-1)+b$ MOD 2^{32} XOR $X_t(i)$
$Z_t(7) = Y_t(7)$; $Z_t(i) = aZ_t(i+1)+b$ MOD 2^{32} XOR $Y_t(i)$
$Z_t(0,0)$
$Z_t(7,3)$
$T_{16,\delta(Zt(0,0))}$
Cyclic shift (7 positions)
$T_{16,\delta(Zt(7,3)))}$
Cyclic shift (7 positions)
MD_{t-1}
MD_t

Fig. 3. One step in calculating data dependent chaotic permutation hashes based on discrete Kolmogorov systems

step is done according to $Y_t(0) = X_t(0)$, $Y_t(i) = aY_t(i-1) + b \bmod 2^{32} \oplus X_t(i)$ with parameters a and b set accordingly (see e.g. [9] for a large variety of suitable parameter settings) to give pseudo-random sequences $Y_t(i)$. This is followed by a backward dissipation step (with index i decreasing) according to $Z_t(7) = Y_t(7)$, $Z_t(i) = aZ_t(i+1) + b \; mod \; 2^{32} \oplus Y_t(i)$.

After preprocessing the message block W_t to obtain the block Z_t, the actual hashing step takes place. The 256 bit of Z_t are used to provide 32 key bytes $Z_t(i,j)$ $(0 \leq i \leq 7,\ 0 \leq j \leq 3)$ to permute the message digest MD_{t-1} stored in the 16×16 array of bits using the corresponding discrete Kolmogorov system. Fig. 3 summarizes one round when calculating data dependent chaotic permutation hashes based on chaotic Kolmogorov systems. Iterating this procedure for all blocks of the input message and finally reading the 16×16 2D array line-by-line or column-by-column delivers the 256 bit message digest of the message to hash in a very efficient and elegant manner as pseudo-random message-dependent permutation of the initial message digest MD_0.

3.3 Digital Image Watermarking

Most of the commercially available systems for digital image watermarking [14] are based on ideas known from spread spectrum radio communications [2]. In spread spectrum communications, one transmits a narrowband signal over a much larger bandwidth such that the signal energy present in any single frequency is undetectable. This allows the signal reception even if there is interference on some frequencies.

Although there are many variants of spread-spectrum communications, we will focus on *Direct-Sequence Spread Spectrum* (DSSS) as the method most useful for application in digital image watermarking. A descriptive exposition how this can be achieved is found e.g. in [4] and similarly in [5]; to illustrate the principle we will closely follow along these lines.

Fig. 4 illustrates a simple, straightforward example of spread spectrum watermarking. The watermark bits (`key2`) to be embedded[2] are spread to fill an image of the same size as the image to be watermarked. The spread information bits are then modulated with a cryptographically secure PN signal keyed by watermarking key `key1`, scaled according to perceptual criteria, and added to the image in a pixel-wise fashion.

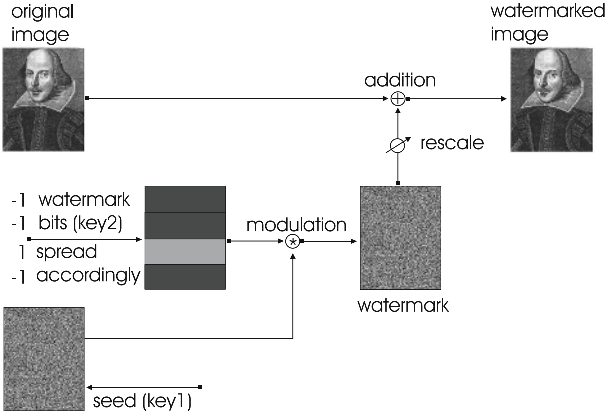


Fig. 4. Spread spectrum watermark embedding

Considering this practically most relevant approach for digital copyright protection via watermarking, an potential role for discrete Kolmogorov systems within the framework of DSSS watermarking becomes obvious. Security of any such DSSS watermarking scheme is heavily based on the cryptographically secure PN signal keyed by watermarking key `key1`. We implement this process as follows. Starting with a balanced initial binary image (might be a corporate logo), this image is permuted by discrete Kolmogorov systems under the influence of a key for as many rounds as are necessary to ensure that a high-quality PN signal is achieved. This PN signal is then used in the watermark embedding (and also detection) phase as depicted in Fig. 4, a fact that stresses the vital role that chaotic permutation operators can play in copyright protection via digital watermarking.

4 Conclusion

In this contribution we have shown that it is possible to derive adequate discrete counterparts for classical continuous Kolmogorov systems. Additionally we provided a detailed analysis under which criteria these novel discrete Kolmogorov systems offer high-quality permutation operators. Availability of such discrete permutation systems was finally utilized to sketch several examples of potential applications for important tasks in information security such as symmetric

---

[2] For simplicity, we just embed 4 bits; in real systems, 128 bit or more are used.

block ciphering, message digest computation, or copyright protection via digital watermarking. Summing up it can be concluded that our analysis performed for discrete Kolmogorov systems proves validity of specific important properties and constitutes a solid basis to apply them in many fields of secure communications.

References

1. V.I. Arnold and A. Avez. *Ergodic Problems of Classical Mechanics.* W.A. Benjamin, New York, 1968.
2. R. Dixon. *Spread Spectrum Systems.* John Wiley and Sons, New York, 1984.
3. S. Goldstein, B. Misra, and M. Courbage. On intrinsic randomness of dynamical systems. *Journal of Statistical Physics*, 25(1):111–126, 1981.
4. F. Hartung, J. Su, and B. Girod. Spread spectrum watermarking: Malicious attacks and counterattacks. In *Security and Watermarking of Multimedia Contents, Proc. SPIE 3657*, January 1999.
5. Martin Kutter. Performance improvement of spread spectrum based image watermarking schemes through m-ary modulation. In *Workshop on Information Hiding, Lecture Notes in Computer Science, volume 1768*, pages 238–250, 1999.
6. Jürgen Moser. *Stable and Random Motions in Dynamical Systems.* Princeton University Press, Princeton, 1973.
7. NIST. Digital signature standard. U. S. Department of Commerce, 1994.
8. NIST. Secure hash standard. FIPS PUB 180-1, April 1995.
9. W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipies in C: The Art of Scientific Computing.* Cambridge University Press, 1988.
10. Josef Scharinger. Fast encryption of image data using chaotic Kolmogorov flows. *Journal of Electronic Imaging*, 7(2):318–325, 1998.
11. Josef Scharinger. Analysis of chaotic systems for communication security applications. In *Proceedings of the 14th International Conference on Systems Science*, volume 3, pages 78–85, 2001.
12. Josef Scharinger. Application of signed Kolmogorov hashes to provide integrity and authenticity in web-based software distribution. In *Formal Methods and Tools for Computer Science*, pages 85–88, 2001.
13. Bruce Schneier. *Applied Cryptography.* Addison-Wesley, 1996.
14. A.Z. Tirkel, G.A. Rankin, R.G. van Schyndel, W.J. Ho, N.R.A. Mee, and C.F. Osborne. Electronic watermark. In *Dicta-93*, pages 666–672, 1993.

Fault Cryptanalysis of ElGamal Signature Scheme

Janusz Biernat and Maciej Nikodem

Wrocław University of Technology,
Institute of Engineering Cybernetics,
11/17 Janiszewskiego Street, 50-372 Wrocław, Poland

Abstract. In this paper we examine the immunity of ElGamal signature scheme and its variants against fault cryptanalysis. Although such schemes have been already widely adopted, their resistance against fault cryptanalysis has not been verified in detail yet. However, at least some of them are not immune to fault cryptanalysis and can be broken without solving discrete logarithm problem. We will show that the selected signature schemes can be broken in $O(nlog2n)$ steps if single bit-flip errors are inducted during computations. We also present two modifications that can be used to improve security of ElGamal scheme.

1 Introduction

In 1996, Boneh, DeMillo and Lipton [2] announced a new type of attack against public cryptosystems implemented in tamper resistant device e.g. smart cards. Soon Biham and Shamir announced and then published article about Differential Fault Analysis (DFA), that applies to secret key cryptosystems such as DES or AES. Since then, many reserchers have been investigating the problem of fault cryptanalysis, in an effort to find methods of improving security of various cryptographic schemes. The researches concentrated mostly on RSA-like and CRT-based (Chinese Reminder Theorem) cryptosystems, and a handful solutions have been proposed to improve their security. Unfortunately less attention has been paid to signature schemes, which security is based on discrete logarithm problem (DLP): e.g. ElGamal, Shnorr or DSA (Digital Signature Algorithm) schemes.

In 1997, Bao et al.[1] presented methods on how to implement fault cryptanalysis against RSA, ElGamal, Shnorr, and DSA signature schemes. They assumed that an attacker can induct single bit-flip errors, and showed how to use this fact to recover secret keys. Nevertheless they have not given any precise information about complexity of this attack. They also proposed some countermeasures that could be implemented to improve security (e.g. repeating calculations, introducing randomness, and result checking before output).

In 2000, Yen and Joye [6] demonstrated that checking the results before output may in some cases leak secret information. In 2003, Yen et al.[7], showed that checking procedure also can be affected by an fault induction so that the device will output erroneous ciphertext or signature.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 327–336, 2005.
© Springer-Verlag Berlin Heidelberg 2005

In 2000, Dottax presented how one can implement fault cryptanalysis to ECDSA and other signature schemes. In 2004, Giraud and Knudsen [3] extended results presented in previous works and analyzed an attack that take advantage of byte errors instead of bit errors. They also gave some information about the attack complexity and the number of faulty signatures that are necessary to restrict possible secret key values to the amount requested. On the other hand they have not presented any countermeasures that can be applied to improve security.

Simultaneously methods that improve security of CRT based algorithms have been presented. In [7] Yen et al. showed novel technique on how to prevent fault cryptanalysis by fault propagation. In this scheme, if the attacker succeeds to induct an error into one part of CRT calculation, then the other part will also be affected. Nevertheless scheme presented in [7] is in fact also susceptible to fault cryptanalysis, error propagation method can be improved. Important fact that comes from this scheme is that splitting cryptographic operations, and introducing error propagation can improve immunity to fault cryptanalysis.

In this paper we focus on six different variants of ElGamal. We will show that, choosing algorithm for a particular application has rost impact to fault cryptanalysis. Our attack assumes that signing purposes are implemented in tamper-proof device and one is able to induct random, temporary and single bit-flip errors into private key a stored in this device. We also assume that each error affects only one bit and the probability that i-th bit of key a is erroneous equals $1/n$ where $n = \log_2 a$.

Apart from analysis of the ElGamal scheme, we also present new technique that allows to improve security of this signature scheme. With our modification, the complexity of recovering single bit of the key a increases up to $O(n2^n)$.

2 ElGamal Signature Scheme

During the key generation for ElGamal scheme, each user acts as follows:

1. selects a large prime p and a generator g in multiplicative group Z_p^*,
2. selects private key - random integer a, $1 < a < p - 2$,
3. computes public key $y = g^a \bmod p$.

Afterwards, when user wants to sign a message m, he does the following signing scheme:

1. selects a random integer $1 < k < p - 2$ with $gcd(k, p - 1) = 1$,
2. computes $r = g^k \bmod p$,
3. computes $s = k^{-1}(h(m) - ar) \bmod (p - 1)$ where $h(m)$ is a hash function, then the signature for m is a pair $\langle k, s \rangle$.

Signature $\langle k, s \rangle$ can be verified using user's public key. The verification procedure is not important for our discussion and thus will be omitted.

There are many variants of basic ElGamal signing scheme and the majority of them alert third step of signing — which is commonly referred to as the

Table 1. Variants of ElGamal signing equation

	u	v	w	Signing equation
1	$h(m)$	r	s	$h(m) = ar + ks$
2	$h(m)$	s	r	$h(m) = as + kr$
3	s	r	$h(m)$	$s = ar + kh(m)$
4	s	$h(m)$	r	$s = ah(m) + kr$
5	r	s	$h(m)$	$r = as + kh(m)$
6	r	$h(m)$	s	$r = ah(m) + ks$

signing equation. After suitable rearrangement, this basic signing equation can be written as $u = av + kw \bmod (p-1)$ where $u = h(m)$, $v = r$ and $w = s$. Other variants of ElGamal can be obtained by permitting u, v and w to take on the values of $h(m)$, r and s in different orders. Thus there are six different variants of ElGamal scheme (see Tab.1) with different properties [5] and, what is more important, they are affected by fault cryptanalysis in different ways.

2.1 Fault Cryptanalysis of Basic ElGamal Scheme

Assume that the attacker has a tamper-resistant device that performs basic ElGamal signature scheme. Moreover the attacker is able to induct single bit-flip errors into secret key a. This error is inducted at random and thus probability that fault is inducted into i-th bit of a equals $1/n$. Then the fault cryptanalysis is performed as follows:

- while the device is computing signature, the attacker inducts random bit-flip error into i-th bit of secret key (this changes a into $\bar{a} = a \pm 2^i$),
- the device computes erroneous signature as follows:
  - selects a random integer $1 < k < p - 2$ with $gcd(k, p-1) = 1$,
  - computes $r = g^k \bmod p$,
  - computes $\bar{s} = k^{-1}(h(m) - \bar{a}r) \bmod (p-1)$,
    erroneous signature for m is a pair $\langle k, \bar{s} \rangle$.
- the attacker uses erroneous signature and computes $r^{\bar{s}}, g^{h(m)}, y^{-r}$ where

$$\begin{aligned} r^{\bar{s}} &= \left(g^k\right)^{k^{-1}(h(m)-\bar{a}r)} = g^{(h(m)-\bar{a}r)} \\ &= g^{h(m)} g^{-(a\pm 2^i)r} = g^{h(m)} y^{-r} g^{\mp 2^i r}, \end{aligned} \tag{1}$$

- the attacker then finds the fault value of $\mp 2^i$ for which following equation holds

$$\frac{r^{\bar{s}}}{g^{h(m)} y^{-r}} = g^{\mp 2^i r}, \tag{2}$$

- when the proper fault value is found, then the attacker knows that i-th bit was flipped to 0 or 1 (e.g. when fault value equals $+2^5$ then 5-th bit was flipped from 1 to 0 and thus $a_5 = 1$),
- attacker repeats this attack while enough bits of a are known — remaining bits can be found by exhaustive search.

In each iteration, the attacker needs to find inducted fault value, which computation complexity is dominated by time needed to perform $2n$ modular exponentiations. The above scheme is repeated until the required number of different faults are inducted into device. Because faults are inducted at random with equal probability, the probability of the event, that in $k > n$ inductions each bit will be erroneous at least once, equals $1 - n\left(1 - 1/n\right)^k$. If $k = n \log 2n$ then this probability equals $1 - n\left(1 - 1/n\right)^{n \log 2n} \geq 1 - n\frac{1}{2n} = 1/2$.

The above analysis shows that the attacker can recover the secret key a with probability $1/2$ after $n \log 2n$ faults were inducted. Computational complexity of this attack is thus dominated by time needed to perform $2n^2 \log 2n$ modular exponentiations.

2.2 Fault Cryptanalysis of Other ElGamal Schemes

The attack presented in last section (and those presented in [1] and [3]) applies to all ElGamal variants except variants 2 and 4 (see Tab.1). Those variants are immune to fault cryptanalysis of the presented type, because attack complexity equals to complexity of Diffie-Hellman problem. We will present this for variant 2 of ElGamal scheme.

Assume that the device, that performs signing according to variant 2 of ElGamal scheme, outputs erroneous signature of the form

$$\langle r, \bar{s} \rangle = \left\langle g^k \bmod p, \overline{a^{-1}}\left(h(m) - kr\right) \right\rangle .$$

Using this signature the attacker can calculate:

$$\begin{aligned} y^{\bar{s}} &= \left(g^a\right)^{\overline{a\;\;{}^{1}}(h(m)-kr)} = \left(g^a\right)^{(a\;\;{}^{1} \pm 2^i)(h(m)-kr)} \\ &= \left(g^a\right)^{a\;\;{}^{1}(h(m)-kr)} \left(g^a\right)^{\pm 2^i (h(m)-kr)} \\ &= g^{(h(m)-kr)} g^{\pm 2^i a h(m)} g^{\mp 2^i a k r} \\ &= g^{h(m)} r^{-r} y^{\pm 2^i h(m)} \left(g^{ak}\right)^{\mp 2^i r} . \end{aligned} \tag{3}$$

According to (3), the attacker looks for the fault value $\pm 2^i$ for which the following equation holds:

$$\frac{y^{\bar{s}}}{g^{h(m)} r^{-r}} = y^{\pm 2^i h(m)} \left(g^{ak}\right)^{\mp 2^i r} . \tag{4}$$

Fortunately there is a factor g^{ak}, in equation (4), which is unknown to the attacker and has to be computed. Computing this factor using only public information ($y = g^a \bmod p$ and $r = g^k \bmod p$) is a Diffie-Hellman problem. Thus the attacker cannot find fault value for which (4) holds, and the only way he can perform cryptanalysis is the exhaustive search.

3 Modifications of ElGamal Scheme

In this section we present two modifications of basic ElGamal scheme. Those modifications use the same technique to improve security against fault cryptanalysis but they differ in computational overhead and resulting security level.

Throughout this section we will assume that attacker can induct bit-flip errors into secret key a at random with equal probability $1/n$.

Note, that our modifications influence only signing procedure, while key generation and signature verification remains the same as for basic ElGamal scheme, and thus they will not be discussed.

Before we present our proposal let us prove following theorem.

Theorem 1. *Let $p_i|p-1$ denote the set of prime factors p_i of $p-1$. Given a prime number p and an integer k_1 with $k_1 < p$ and $gcd(k_1, p-1) = 1$ there are*

$$\vartheta(p-1) = \frac{p-1}{2} \prod_{\substack{p_i|p-1 \\ p_i > 2}} \left(1 - \frac{2}{p_i}\right) \tag{5}$$

pairs of integers $\langle k, k_2 \rangle$ with $k, k_2 < p$ and $gcd(k, p-1) = gcd(k_2, p-1) = 1$ that satisfy the following formula

$$k = 2k_1 + k_2 \bmod (p-1). \tag{6}$$

Proof. According to the theorem we want to find the number of integers k, k_1, k_2 each coprime with $p-1$ that satisfy (6). Every integer $p-1$ can be represented using only its prime factors

$$p - 1 = \prod_{p_i|p-1} p_i^{e_i},$$

where p_i are prime factors of $p-1$ and e_i are positive integers greater then 0. To ensure that k, k_2 are coprime with $p-1$ it is enough to ensure that k, k_2 are coprime with each of those prime factors p_i. In other words k and k_2 have to satisfy following system of equations:

$$\begin{cases} k \bmod p_i \neq 0 \\ k_2 \bmod p_i \neq 0 \end{cases} \text{for each } p_i.$$

But from (6) we have that $k_2 = k - 2k_1$ so we can write that

$$\begin{cases} k \bmod p_i \neq 0 \\ k - 2k_1 \bmod p_i \neq 0 \end{cases} \text{for each } p_i.$$

and this equals to

$$\begin{cases} k \bmod p_i \neq 0 \\ k \bmod p_i \neq 2k_1 \bmod p_i \end{cases} \text{for each } p_i.$$

For $p_i = 2$, which is always a factor of $p-1$, this two equations are identical because $2k_1 \bmod 2 = 0$. For other factors ($p_i > 2$) there are always two equations because we have chosen k_1 coprime with $p-1$ and thus $2k_1 \bmod p_i \neq 0$. Thus

$2k_1 \bmod p_i$ will produce exactly one remainder $r_i \neq 0$ for each $p_i > 2$. It lets us write that

$$\begin{cases} k \bmod p_i \neq 0 & \text{for each } p_i \\ k \bmod p_i \neq r_i \neq 0 & \text{for each } p_i > 2. \end{cases} \tag{7}$$

From equation (7) we can conclude that theorem is satisfied for every k that produces remainders modulo p_i that are not equal to 0 and some — exacly one — other integer r_i. In other words, we allow k to produce $p_i - 2$ remainders q_i modulo each $p_i > 2$ and 1 modulo $p_i = 2$. Thus, if there are l prime factors p_i, the correct k must satisfy following system of equations

$$\begin{cases} k \bmod 2 = 1 & \\ k \bmod p_1 = q_1 & \text{where } q_1 \in Z_{p_1} \setminus \{0, r_1\} \\ k \bmod p_2 = q_2 & \text{where } q_2 \in Z_{p_2} \setminus \{0, r_2\} \\ \vdots & \\ k \bmod p_{l-1} = q_{l-1} & \text{where } q_{l-1} \in Z_{p_{l\ 1}} \setminus \{0, r_{l-1}\}\,. \end{cases} \tag{8}$$

On the other hand, for each combination of possible remainders q_i system (8) has exactly one unique solution k within the range $\left[0, \prod_{p_i|p-1} p_i - 1\right]$ (which can be computed using Chinese Reminder Theorem). Finally we can conclude, that there are

$$\prod_{p_i|p-1, p_i>2} (p_i - 2)$$

integers k with in the range $\left[0, \prod_{p_i|p-1} p_i - 1\right]$ that both k and $k_2 = k + 2k_1 \bmod (p-1)$ are coprime with $p - 1$. Moreover there are

$$\frac{p-1}{\prod_{p_i|p-1} p_i}$$

intervals of the length $\prod_{p_i|p-1} p_i$ in range $[0, p-2]$. So within the whole range there are

$$\begin{aligned} \vartheta(p-1) &= \frac{p-1}{\prod_{p_i|p-1} p_i} \prod_{p_i|p-1, p_i>2} (p_i - 2) = \frac{p-1}{2\prod_{p_i|p-1, p_i>2} p_i} \prod_{p_i|p-1, p_i>2} (p_i - 2) \\ &= \frac{p-1}{2} \prod_{p_i|p-1, p_i>2} \left(1 - \frac{2}{p_i}\right). \end{aligned}$$

integers k, k_1 that satisfies (6). That concludes the proof. □

As an immediate consequence of the theorem we can formulate the following corollary.

Corollary 1. *Let $\varphi(x)$ be an Euler function of x. Then there are exactly $\varphi(p-1) \cdot \vartheta(p-1)$ tuples $\langle k, k_1, k_2 \rangle$, with k, k_1, k_2 in range $[0, p-2]$ and coprime with $p - 1$, that satisfy equation (6).*

The proof is obvious because in the Theorem 1 we choose $k_1 < p-1$, coprime with $p-1$. Because Euler function $\varphi(p-1)$ gives the number of elements that are coprime with $p-1$ thus there are $\varphi(p-1)$ possible values for k_1 in range $[0, p-2]$.

3.1 Modification 1 (M1)

In this modification two random parameters k_1 and k_2 are used to compute separate signatures s_1 and s_2. Then this signatures are being used to construct the ElGamal signature s under message m. The signing procedure goes as follows:

1. select a random integers k_1, k_2 such that $1 < k_j < p-2$ and $gcd(k_j, p-1) = 1$ for $j = 1, 2$,
2. compute $k' = \left(2k_1^{-1} + k_2^{-1}\right) mod(p-1)$,
3. if $gcd(k', p-1) \neq 1$ then choose new k_2 and repeat step 2,
4. compute $k = k'^{-1} \bmod (p-1)$ and $r = g^k \bmod p$,
5. compute

$$
\begin{aligned}
s_1 &= k_1^{-1}(h(m) - ar) \bmod (p-1) \\
s_2 &= k_2^{-1}(h(m) - ar) \bmod (p-1),
\end{aligned}
$$

6. compute signature as a pair $\langle r, s\rangle = \left\langle g^k \bmod p, s = 2s_1 + s_2 \bmod (p-1)\right\rangle$.

There are two drawbacks in this scheme, from now on denoted as M1. First, there is a condition in third step of signing procedure. The condition is to assure that the device can compute multiplicative inverse in fourth step. Such computation can be done if, and only if $gcd(k', p-1) = 1$. Form the Theorem 1 we have that, the probability that uniformly chosen integers k_1, k_2 with $gcd(k_1, p-1) = gdc(k_2, p-1) = 1$ produce such k' is given by following formula:

$$
\begin{aligned}
\phi(p-1) &= \frac{1}{\varphi(p-1)}\vartheta(p-1) = \frac{1}{\varphi(p-1)}\frac{p-1}{2}\prod_{p_i | p-1, p_i > 2}\left(1 - \frac{2}{p_i}\right) \\
&= \frac{p-1}{2 \cdot \varphi(p-1)}\prod_{p_i | p-1, p_i > 2}\left(1 - \frac{2}{p_i}\right),
\end{aligned} \tag{9}
$$

where the product is computed for every prime factor $p_i > 2$ of $p-1$. The probability (9) equals 1 if, and only if $p-1$ is of the form 2^i which yields $p-1$ has only one prime factor 2, and $\varphi(p-1)$ equal to $(p-1)/2$. This is obvious since for $p-1 = 2^i$ each sum $2k_1 + k_2$, where k_1, k_2 are coprime with $p-1$ (and thus odd), is always odd and thus coprime with $p-1$. If $p-1$ has more prime factors p_i then $\phi(p-1)$ is always smaller then 1, and its exact value depends on those factors. Thus, when selecting prime p for ElGamal scheme, we should ensure that $p-1$ has as less as possible different small factors (except 2 which is always a factor). In this way we ensure that the probability (9) almost equals to 1.

The second drawback is the computational complexity wich is double the complexity of basic ElGamal scheme. Moreover in scheme M1 the device must compute three different multiplicative inverses (only one in case of basic ElGamal scheme).

The advantage of scheme M1 is that it increases immunity to fault cryptanalysis using no checking nor compare procedure. Assume that the attacker can induct only random bit-flip errors into the secret key a while the device performs signing according to scheme M1. Inducting an error, the attacker can get erroneous output of the following form

$$\langle r, \bar{s}\rangle = \left\langle g^k \bmod p, 2k_1^{-1}(h(m) - \bar{a}r) + k_2^{-1}(h(m) - \tilde{a}r) \bmod (p-1)\right\rangle .$$

Because errors were inducted randomly, thus $\bar{a} = a \pm 2^i$ and $\tilde{a} = a \pm 2^j$ for $i \neq j$ with probability $1 - \frac{1}{n}$. Given erroneous signature $\langle r, \bar{s}\rangle$ the attacker computes

$$\begin{aligned} r^{\bar{s}} &= r^{2k_1{}^{1}(h(m)-(a\pm 2^i)r)+k_2{}^{1}(h(m)-(a\pm 2^j)r)} \\ &= r^{2k_1{}^{1}(h(m)-ar)\mp 2^i r 2k_1{}^{1}+k_2{}^{1}(h(m)-ar)\mp 2^j r k_2{}^{1}} \\ &= r^{(2k_1{}^{1}+k_2{}^{1})(h(m)-ar)-r(\pm 2^{i+1}k_1{}^{1}+\pm 2^j k_2{}^{1})} \\ &= g^{kk\ {}^{1}(h(m)-ar)} r^{-r(\pm 2^{i+1}k_1{}^{1}+\pm 2^j k_2{}^{1})} \\ &= g^{h(m)-ar} r^{-r(\pm 2^{i+1}k_1{}^{1}+\pm 2^j k_2{}^{1})} = g^{h(m)} y^{-r} r^{-r(\pm 2^{i+1}k_1{}^{1}+\pm 2^j k_2{}^{1})} . \end{aligned} \tag{10}$$

Next the attacker looks for a tuple $T = \left\langle \pm 2^i, \pm 2^j, k_1, k_2 \right\rangle$ for which following equation holds:

$$\frac{r^{\bar{s}}}{g^{h(m)} y^{-r}} = r^{-r(\pm 2^{i+1}k_1{}^{1}+\pm 2^j k_2{}^{1})} . \tag{11}$$

Because there are n possible values for i, j and 2^n possible values for k_1 and k_2 thus there is $n^2 2^{n+1}$ possible tuples T that the attacker has to check.

More probable situation is when the attacker inducts only one error e.g. during the computation of s_1. In such case the erroneous signature has a form

$$\langle r, \bar{s}\rangle = \left\langle g^k \bmod p, 2k_1^{-1}(h(m) - \bar{a}r) + k_2^{-1}(h(m) - ar) \bmod (p-1)\right\rangle .$$

The attacker can use this signature and perform fault cryptanalysis as follows:

$$\begin{aligned} r^{\bar{s}} &= r^{2k_1{}^{1}(h(m)-(a\pm 2^i)r)+k_2{}^{1}(h(m)-ar)} \\ &= r^{2k_1{}^{1}(h(m)-ar)\mp 2^i r 2k_1{}^{1}+k_2{}^{1}(h(m)-ar)} \\ &= r^{(2k_1{}^{1}+k_2{}^{1})(h(m)-ar)-r(\pm 2^{i+1}k_1{}^{1})} \\ &= g^{(h(m)-ar)} r^{\mp 2^{i+1}k_1{}^{1}r} = g^{h(m)} y^{-r} r^{\mp 2^{i+1}k_1{}^{1}r} . \end{aligned} \tag{12}$$

Equation (12) shows that for successful fault cryptanalysis the attacker needs to draw a pair $P = \left\langle \mp 2^i, k_1 \right\rangle$ for which following equation holds

$$\frac{r^{\bar{s}}}{g^{h(m)} y^{-r}} = r^{\mp 2^{i+1}k_1{}^{1}r} \tag{13}$$

There are $n2^n$ possible values of P but only one of them satisfies the above equation. Thus we can state that in scheme M1 the complexity of recovering single bit of the key a equals $O(n2^n)$.

3.2 Modification 2 (M2)

In this modification signing is also divided into two steps but we use a different technique. Now, when a user wants to sign a message m then he does the following:

1. selects a random integers k such that $1 < k < p - 2$ and $gcd(k, p - 1) = 1$,
2. computes $r = g^k \bmod p$,
3. chooses $0 < r_1 < r$ at random and computes $r_2 = r - r_1$,
4. computes
$$\begin{aligned} s_1 &= k^{-1}(h(m) - ar_1) \bmod (p-1) \\ s_2 &= k^{-1}(h(m) - ar_2) \bmod (p-1), \end{aligned}$$
5. computes signature as a pair
$$\langle r, s\rangle = \left\langle g^k \bmod p, s_1 + s_2 - k^{-1}h(m) \bmod (p-1)\right\rangle .$$

In this modification, from now on denoted as M2, signing is also divided into two steps. This is achieved by splitting r into r_1 and r_2. We expect this method to obtain different and better properties when compared to scheme M1. There are three main advantages of scheme M2:

- there is no checking procedure during signing which depends on prime factors of $p - 1$,
- computation of multiplicative inverse is performed only once, as in basic ElGamal scheme,
- less computational overhead to perform signing is needed (only two modular multiplications and three additions more as compared to basic ElGamal scheme).

The modification M2 also improves the security against fault cryptanalysis. Assume that the attacker inducts single bit-flip error into a secret key a during creation of signature s_1. As a result the device outputs erroneous signature of the form

$$\langle r, \bar{s}\rangle = \left\langle g^k \bmod p, k^{-1}(h(m) - \bar{a}r_1) + k^{-1}(h(m) - ar_2) - k^{-1}h(m) \bmod (p-1)\right\rangle .$$

This signature can be then used to calculate

$$\begin{aligned} r^{\bar{s}} &= r^{k\ ^{1}(h(m)-\bar{a}r_1)+k\ ^{1}(h(m)-ar_2)-k\ ^{1}h(m)} \\ &= r^{k\ ^{1}(h(m)-(a\pm 2^i)r_1+h(m)-ar_2-h(m))} \\ &= g^{kk\ ^{1}(h(m)-ar_1-ar_2\mp 2^i r_1)} \\ &= g^{h(m)-ar\mp 2^i r_1} = g^{h(m)}y^{-r}g^{\mp 2^i r_1} . \end{aligned} \tag{14}$$

Now the attacker looks for a fault value, which is a pair $P = \left\langle \mp 2^i, r_1 \right\rangle$ that satisfies the following equation:

$$\frac{r^{\bar{s}}}{g^{h(m)}y^{-r}} = g^{\mp 2^i r_1} . \tag{15}$$

To find the proper fault value the attacker needs to check all possible values for P and there is a total of $n2^n$ values that need to be checked. Moreover there are many different Ps for which (15) holds. More precisely, if (15) is satisfied for $P = \left\langle \mp 2^i, r_1 \right\rangle$ then it is also satisfied for $P = \left\langle \mp 2^{i-j}, r_1 2^j \right\rangle$. Therefore there are n values of P for which (15) holds and the attacker cannot distinguish between them (he does not know how r was spitted into r_1 and r_2).

If two bit-flip errors are inducted into the computation of s_1 and s_2, then the complexity of fault cryptanalysis increases. This is due to fact that the attacker needs to find a correct tuple $T = \left\langle \mp 2^i, \mp 2^j, r_1 \right\rangle$ and there are $n^2 2^n$ possible values for T.

4 Conclusion

In this paper we consider ElGamal signature scheme and its security against fault cryptanalysis. We pointed out attacks that can reveal secret key a, calculated attack complexity and showed that different variants of ElGamal scheme are vulnerable to fault cryptanalysis in different ways. We also presented two modifications that allow to increase ElGamal security by splitting signature computation. Our analysis shows that the modification M2 is both sound and effective and cause fault cryptanalysis difficult to perform. Moreover the modification M2 can be used in all variants of ElGamal scheme and its complexity is slightly larger compareing to the complexity of basic ElGamal scheme.

References

1. Bao F., Deng R., Han Y., Jeng A., Narasimhalu A.D., Ngair T.-H.: Breaking Public Key Cryptosystems an Tamper Resistance Devices in the Presence of Transient Fault, In 5^{th} Security Protocols WorkShop, LNCS, vol.1361, pp.115-124, Springer-Verlag, 1997
2. Boneh D., DeMillo R.A., Lipton R.J.: On the Importance of Checking Cryptographic Protocols for Faults, Advances in Cryptology - EUROCRYPT'97, LNCS, vol.1233, pp.37-51, Springer-Verlag, 1997
3. Giraud C., Knudsen E.: Fault Attacks on Signature Schemes, ACISP 2004, LNCS, vol.3108, pp.478-491, Springer-Verlag, 2004.
4. Koblitz N.: A Course in Number Theory and Cryptography, Springer-Verlag New York, 1994, ISBN 83-204-1836-4
5. Menezes A.J., van Oorschot P.C., Vanstone S.A.: Handbook of Applied Cryptography, CRC Press, 1996, ISBN 0-8493-8523-7.
6. Yen S., Joye M.: Checking Before Output May Not Be Enough against fault-based cryptanalysis, IEEE Transactions on Computers, vol.49, no.9, pp.967-970, September 2000
7. Yen S., Kim S., Lim S., Moon S.: RSA Speedup with Chinese Reminder Theorem Immune Against Hardware Fault Cryptanalysis, IEEE Transactions on Computers, vol.52, no.4, pp.461-472, April 2003

Complexity-Theoretical Approaches to the Design and Analysis of Cryptographical Boolean Functions

Juan David González Cobas and José Antonio López Brugos

Departamento de Informática, Universidad de Oviedo
cobas@epsig.uniovi.es, brugos@epsig.uniovi.es

Abstract. In the theory of symmetric cipher design, criteria for the choice of Boolean functions with good behavior have been thoroughly studied. The character of these criteria is mainly statistiscal. We survey the often conflicting propoerties which are generally acknowledged, which shows the almost universal neglect of complexity-theoretic techniques. Of these, we propose the most prominent complexity measure concerning Boolean functions, to wit, *boolean circuit complexity (BCC)*, as a means to assess Boolean function behavior in the context of symmetric algorithm design. The connection between BCC of the non-linear elements of a design and the pseudorandom stream generated with their help is shown by scrutiny of linear complexity profiles.

1 Boolean Function Design

The design theory (and the art) of symmetric cipher comprises a number of standard techniques whose target is the creation of dependable algorithms for confidentiality preservation (stream and block cipehrs) and integrity checking (hash functions and MACs).

In this area, choice criteria for *Boolean functions* appear once and again. The list of applications for which a proper design theory of Boolean functions has been developed is very long. Just to summarize a few:

- nonlinear combiners for stream ciphers
- nonlinear feedback for special stream cipher applications (self-synchronizing or other FSR-based key sequence generators).
- S-boxes for substitution-permutation networks
- round functions for hashing and MACs

Accordingly, a number of established criteria for acceptability is at our disposal. We first review the most prominent criteria that have been proposed in the literature.

It is apparent that complexity-theoretic approaches, which have been so fruitful in stream-cipher theory, appear to have been neglected in the chore of selecting the best candidates for the non-linear elements of symmetric cipher algorithms. The intuitive fact that the complexity involved in computing a symmetric

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 337–345, 2005.
© Springer-Verlag Berlin Heidelberg 2005

algorithm (be it depending on its number of rounds, or the intrinsic properties of the round function involved) has something to do with its potential weaknesses suggests that complexity measures applied to the elements involved in *computing* a cipher, provide minimal conditions to guarantee that its non-linear elements have been properly chosen.

We propose *boolean circuit complexity (BCC)* as such a complexity measure, giving some plausible technique for evaluating Boolean functions, analyzing the impact of this criterion in the output of a concrete design.

2 Review of Criteria

We can find a wealth of design criteria for boolean functions in the recent literature on symmetric cipher design. To mention a few

- Balance
- (High) nonlinear order
- Correlation-immunity [1,2].
- Bentness [3]
- Distance to linear structures [4]
- Strict avalanche criterion [5]
- Propagation characteristics [6]
- Global avalanche criteria [7]

2.1 Balance, Nonlinear Order and Correlation-Immunity

Let $f : GF(2)^n \to GF(2)$ be a Boolean function of n arguments.

We say f is *balanced* if

$$\#\{x \in GF(2)^n \mid f(x) = 1\} = \#\{x \in GF(2)^n \mid f(x) = 0\} \tag{1}$$

which amounts to say that $\Pr\{f(x) = 1\} = \Pr\{f(x) = 0\} = 1/2$ when the inputs of f are independent uniformly distributed binary random variables.

The function f is said to be *correlation-immune of order* m if, for every set $i_1, i_2, \ldots, i_m$ of its positional arguments, the value of the function is statistically independent of them, assuming that the arguments $(X_{i_1}, \ldots, X_{i_m})$ are independent uniformly distributed binary random variables. This is equivalent to say that the information leaked by f about the values of any m-tuple $X_{i_1}, X_{i_2}, \ldots, X_{i_m}$ of its arguments is zero.

$$I(f; X_{i_1}, X_{i_2}, \ldots, X_{i_m}) = 0 \tag{2}$$

If f is put in *algebraic normal form* (ANF)

$$f(x_1, \ldots, x_n) = a_0 + \sum a_i x_i + \sum a_{ij} x_i x_j + \ldots + a_{1\ldots n} x_1 \ldots x_n \tag{3}$$

its *nonlinear order* is defined as the highest degree of a non-zero monomial in 3.

2.2 Spectral Characterization

These properties are spectral in character [8]. They can be defined through the *Walsh-Hadamard transform* of f [9]. This is a function $Wf : GF(2)^n \rightarrow \mathrm{I\!R}$ defined by the formula

$$(Wf)(\omega) = \sum_{x \in GF(2)^n} f(x)(-1)^{w \cdot x} \qquad \text{for } \omega \in GF(2)^n \tag{4}$$

Note that this makes sense even for f with complex values. We usually translate the truth table of f, a list of $\{0,1\}$ values, into a sequence of $\{\pm 1\}$ values which we call the *sequence of f* and denote by $\hat{f}$:

$$\hat{f}(x) = (-1)^{f(x)} \tag{5}$$

All properties mentioned in Sec. 2.1 can be defined in terms of Walsh transforms or the ANF, which is another kind of linear transform.

Theorem 1. *A Boolean function f is balanced if and only if $(Wf)(0) = 0$*

Theorem 2. *A Boolean function f is correlation-immune of order m if and only if $(Wf)(\omega) = 0$ for every ω of Hamming weight less than or equal to m [8].*

Theorem 3. *If f is correlation immune of order m, with $1 \leq m < n$, then no terms of degree $n - m + 1$ or more can be present in the ANF of f. Moreover, if f is balanced, then no term of degree $n - m$ can be present either, unless $m = n - 1$ [1].*

As we can see, correlation immunity conflicts with high non-linear order as a requirement; we cannot have both.

2.3 Avalanche Criteria

A function $f : GF(2)^n \rightarrow GF(2)$ satisfies the *strict avalanche criterion (SAC)* [5] if

$$\Pr_x\{f(x) + f(x+a) = 1\} = 1/2 \qquad \text{when } hw(a) = 1 \tag{6}$$

A function $f : GF(2)^n \rightarrow GF(2)$ satisfies the *propagation criterion of degree k (PC)* [6] if

$$\Pr_x\{f(x) + f(x+a) = 1\} = 1/2 \qquad \text{when } 1 \leq hw(a) \leq k \tag{7}$$

Perfect nonlinearity amounts to satisfying a PC of degree n.

The broadest generalization of these criteria is the *global avalanche characteristic (GAC)* of a Boolean function [7]. It is defined as the function

$$\Delta_f(a) = \sum_{x \in GF(2)^n} (-1)^{f(x)}(-1)^{f(x+a)} \tag{8}$$

or, more concisely, as the self-correlation function of the sequence $\hat{f}$ of f.

A good GAC means that $|\Delta_f(a)|$ is close to zero for most nonzero values of a. Accordingly, the *sum-of-squares* indicator of GAC is defined as

$$\sigma_f = \sum_{a \in GF(2)^n} \Delta^2(a) \tag{9}$$

2.4 Bentness

Bent functions [3] are the paradigm of *perfect nonlinear functions* [4]: they satisfy a propagation criterion of degree n, which amounts to the balancedness property

$$\Pr_x\{f(x) + f(x+a) = 1\} = 1/2 \tag{10}$$

for every vector $a \neq 0$ in $GF(2)^n$.

Again, this has a nice spectral form:

Theorem 4. *A function $f : GF(2)^n \to \{\pm 1\}$ is* bent *if the modulus of the Walsh transform of its sequence $\hat{f}$ is constant*

$$|(W\hat{f})(\omega)| = 2^{n/2} \tag{11}$$

Note that, by definition, *a bent funcion cannot be balanced.*

2.5 Conflicting Requirements

It is not possible to have everything at the same time. As we have seen, there exist tradeoffs between some of the aforementioned criteria. To mention a few

- Functions *must* be balanced in most cases.
- Bent functions *are never balanced* as they stand.
- No bent functions exist with an odd number of arguments
- High linear order means low correlation-immunity (linear compromise).

In general, one must adjust the choice of non-linear elements of a design to its peculiar demands, prioritizing criteria according to them.

3 The Role of Complexity

Our review of criteria shows that their character is usually *spectral*, *algebraic* or *statistic*. None of them is *complexity-theoretic*.

However, measures of complexity play a role in analysis of symmetric ciphers, and we have good examples at hand

- Linear complexity [2]
- Quadratic complexity
- Maximum order complexity [10]

- Entropy [11]
- Lempel-Ziv complexity [12]

The usual role of these measures is profiling the output of a cryptographic device (in most cases a random sequence generator closely related to a symmetric algorithm) to match it to the expected profile of a “true” random source.

Does the complexity of a Boolean function play a role? After security, the second target in symmetric cipher design is efficiency, so a high computational complexity of the cipher algorithm is *a priori* undesirable. However, lower bounds of it can be established under which no good cryptographic properties can be obtained. This motivates the idea of requiring some minimal complexity to the non-linear elements in consideration, a complexity which need not be strictly correlated to the usual Turing complexity associated with its computation.

4 Boolean Circuit Complexity

A *Boolean circuit* is a directed acyclic graph in which

1. every node with input degree $\mathrm{deg}_{\mathrm{in}} = 0$ is labeled with a variable x_i or a constant $\{1, 0\}$.
2. every node with input degree $\mathrm{deg}_{\mathrm{in}} > 0$ is labeled with a Boolean function of corresponding arity, taken from a finite basis (usually the set of conective symbols $\{\cap, \cup, \neg\}$). We call this a *gate*.
3. there is a special node with out degree $\mathrm{deg}_{\mathrm{out}} = 0$ defined to be the *output* of the circuit

This kind of construction represents in a natural way (a procedure to evaluate) a Boolean function. The smallest number of gates of a circuit representing $f : GF(2)^n \rightarrow GF(2)$ is the *Boolean circuit complexity* of f.

As an example, let us consider the circuits depicted in Fig. 1. The circuit in the right of Fig. 1 evaluates (slightly more costly) the same function evaluated in the left, that is

$$x_1 + x_2 + x_1x_3 + x_2x_4 = (x_1 + x_2 + x_3)x_1 + (x_1 + x_2 + x_4)x_2 \qquad (12)$$

From that we can conclude that the BCC of this polynomial is not higher than 5.

Some facts about BCC are summarized in the following theorems

Theorem 5. *For a function of n arguments, BCC is at most 2^n and at least n*

Theorem 6. *There is a function of n arguments requiring more than $2^n/2n$ gates (in fact,* most *functions require $\Omega(2^n/n)$ gates) [13].*

Computation of BCC is hard in practical cases (concerning lower bounds), however it is not too dificult to estimate tight upper bounds of functions of small number of arguments by a simple exhaustive-search algorithm.

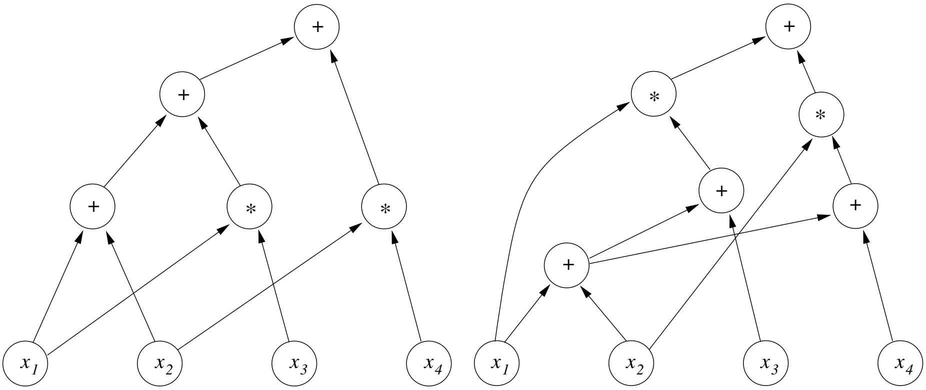


Fig. 1. Boolean circuits for the function $x_1 + x_2 + x_1x_3 + x_2x_4 = (x_1 + x_2 + x_3)x_1 + (x_1 + x_2 + x_4)x_2$

5 Using BCC to Test Quality of a Key Sequence Generator

Let us check a first example of how we can apply this concept to the analysis of a very elementary key sequence generator (KSG).

In Fig. 2 we depict a KSG with a feedback function having two nonlinear elements f and g which we set out to define. Note that this shall be a very weak KSG in any case, the number of arguments of f and g and the size of the cells being four bits.

We will use *linear complexity profiles* to measure the randomness of the output. For determining the BCC of the feedback functions f and g, we take $\{\cap, +, \neg\} = \{+, \cdot, \neg\}$, instead of the standard basis $\{\cup, \cap, \neg\}$, which is more natural in the context of $GF(2)$.

Computing exact BCCs for *a priori* given feedback functions is too costly, but we can use their algebraic form to guess the bounds, and construct functions

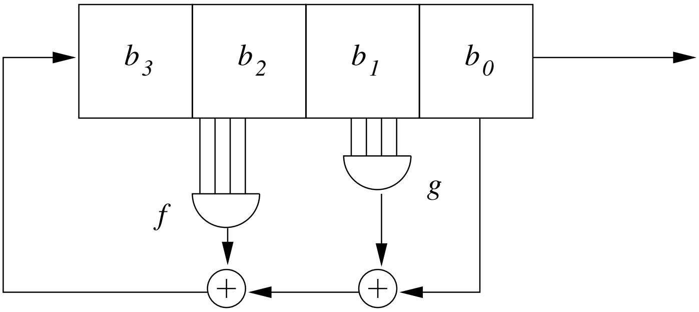


Fig. 2. A simple key sequence generator

```
FOR bc :=  2n TO 2^n
    gatelist := [1..bc]
    WHILE (legal circuits on bc not exhausted) {
        C = next legal circuit
        T = truth table(C)
        IF (T previously constructed) {
           reject
           continue
        }
        IF (balanced(T) AND good PC(T))
        store T
    }
}
```

Fig. 3. Process of selection of f and g

Table 1. Pushing up BCC of f and g

f	g	Var(L^N)	ERV
8	9	49.5111	46.6326
9	9	23.6609	22.2853
10	9	1.4057	1.3240
11	9	1.2070	1.1368
12	9	1.1771	1.1086
13	9	1.2569	1.1838
14	9	1.2367	1.1648
15	9	1.1505	1.0836
16	9	1.3098	1.2337

f	g	Var(L^N)	ERV
8	12	8.6084	14.7032
9	12	5.9395	7.4234
10	12	1.1890	1.1693
11	12	1.1477	1.1195
12	12	1.1924	1.0429
13	12	1.1451	1.1118
14	12	1.1389	1.1250
15	12	1.1321	1.0337
16	12	1.2349	1.1447

with a prescribed upper bound of BCC incrementally (e.g., [4]). A tentative and very simple-minded algorithm of (almost) exhaustive enumeration for this task is given in pseudocode in Fig. 3.

As data in Table 1 show, better profiles were obtained as the complexity of function f was increased (for two fixed complexities of g), as was to be expected. The expected value of a true random sequence (ERV) is given as a means of comparison.

It is obvious that this gives us just hints as to minimal conditions on the feedback functions chosen. Balance and propagation criteria have to be taken into account; BCC in itself is not quite meaningful.

6 An Experiment with a Weakened Serpent

Serpent [14] was one of the candidates for the AES contest. It is a very simple, fast block cipher with 128-bit block size and 256-bit key. Its operation is that of a classical S-P network of 32 rounds.

Each round uses cyclic copies of eight S-boxes $S_k : GF(2)^4 \rightarrow GF(2)^4$ which act as permutations on $GF(2)^4$. They are derived from the S-boxes of DES.

A peculiar characteristic of Serpent is its fast confusion/diffusion effect per round. Choosing 32 rounds was a conservative choice by the authors. To get a weakened version of Serpent, we used a reduced-round version of Serpent with only 4 rounds, replacing its S-boxes by the functions S_1, S_2, S_3, S_4 to be tested. Being by design permutations on $GF(2)^4$, each S_k is a tuple of four Boolean functions of the kind studied in Sec. 5. Moreover, we set a constant key schedule value to minimize the confusion brought in by the key and focus on the properties of nonlinear elements. The block cipher was set up as a KSG in OFB mode of operation.

It is reasonable to expect that the (bad) quality of the sequence reflects somehow the design properties of the chosen S_k, and indeed this is what can be seen in Table 2, where the variance of the linear complexity profile $\mathrm{Var}(L^N)$ is related to the maximum BCC allowed for the S-boxes of the experiment. The expected value for this parameter of a random sequence converges to $86/81 = 1.061$ (see [2]).

Table 2. Typical figures from increasingly complex S_k

$BCC(S_k)$	$\mathrm{Var}(L^N)$
8	8.1171
9	3.5723
10	1.0619
11	1.0691
12	1.0783
13	1.0751
14	1.0820

7 Conclusions

Estimation of Boolean circuit complexity of nonlinear elements appearing in constructions of limited cryptographic strength has a clarifying role in the process of design. This role is complementary to other more established criteria as propagation characteristic, nonlinearity, bentness, etc., all of them providing a test bench for the construction and automatized assesment of simple nonlinear elements.

Estimations of BCC are hard to obtain for functions with high arities; however, it is not that difficult to obtain tight bounds for functions given by nontrivial algebraic expressions. More efficient algorithms for the computation of these bounds would be of grat help in the process of complexity-theoretic assesment of Boolean functions.

References

1. Siegenthaler, T.: Correlation-immunity of nonlinear combining functions for cryptographic applications. IEEE Transactions on Information Theory **IT-30** (1984) 776–780
2. Rueppel, R.A.: Analysis and Design of Stream Ciphers. Springer-Verlag, Berlin (1986)
3. Rothaus, O.S.: On "bent" functions. J. Comb. Theory, Ser. A **20** (1976) 300–305
4. Meier, W., Staffelbach, O.: Nonlinearity criteria for cryptographic functions. In: EUROCRYPT. (1989) 549–562
5. Webster, A.F., Tavares, S.E.: On the design of S-boxes. In Williams, H.C., ed.: Advances in Cryptology — Crypto '85, New York, Springer-Verlag (1986) 523–534
6. Preneel, B., Leekwijck, W.V., Linden, L.V., Govaerts, R., Vandewalle, J.: Propagation characteristics of Boolean functions. In Damgård, I.B., ed.: Advances in Cryptology — Eurocrypt '90, Berlin, Springer-Verlag (1991) 161–173
7. Zhang, K., Zheng, Y.: GAC—the criterion for global avalanche characteristics of cryptographic functions. The Journal of Universal Computer Science **1** (1995) 316–
8. Guo-Zhen, X., Massey, J.L.: A spectral characterization of correlation-immune combining functions. IEEE Transactions on Information Theory **IT-34** (1988) 569–571
9. Golomb, S.W.: On the classification of boolean functions. IEEE Transactions on Information Theory (1959) 176–186
10. Jansen, C.J.A., Boekee, D.E.: The shortest feedback shift register that can generate a given sequence. In Brassard, G., ed.: Advances in Cryptology—CRYPTO '89. Volume 435 of Lecture Notes in Computer Science., Springer-Verlag, 1990 (1989) 90–99
11. Shannon, C.E.: Communication theory of secrecy systems. Bell System Technical Journal **28** (1949)
12. Lempel, A., Ziv, J.: On the complexity of finite sequences. IEEE Trans. Inf. Theory **IT-22** (1976) 783–795
13. Shannon, C.: The synthesis of two–terminal switching circuits. Bell System Technical Journal **28** (1949) 59–98
14. Anderson, R., Biham, E., Knudsen, L.: Serpent: A proposal for the Advanced Encryption Standard. NIST AES proposal, National Institute for Standards and Technology, Gaithersburg, MD, USA (1998)

Algorithm for Proving the Knowledge of an Independent Vertex Set*

Pino Caballero-Gil and Candelaria Hernández-Goya

Dept. Statistics, Operations Research and Computing,
University of La Laguna, 38271 La Laguna, Tenerife, Spain
{pcaballe, mchgoya }@ull.es

Abstract. A new protocol is presented that allows to convince of the knowledge of a solution to the Independent Vertex Set Problem without revealing anything about it. It is constructed from a bit commitment scheme based on the hardness of the Discrete Logarithm Problem, which guarantees its efficient performance and formal security. One of its possible applications is node identification in ad-hoc wireless network because it does not require any authentication servers. Furthermore, recent works on network security has pointed out the importance of the design of efficient Zero Knowledge Proofs of Knowledge for the Independent Vertex Set Problem in broadcast models.

1 Introduction

Since the introduction of the notion of Zero-Knowledge Proof (ZKP) in the seminal paper of Goldwasser, Micali and Rackoff [12], it has proven to be very useful both in Complexity Theory and in Cryptography, playing in this latter field a major role as a building block in the construction of different cryptographic protocols [1]. It is remarkable that most of the different ZKP that have been published so far are related to the same presumably intractable problems on which Public Key Cryptography is based. Such are the cases of the identification scheme based on the discrete logarithm problem [14], and the digital signature based on the computation of square roots [7].

One of the most relevant results regarding ZKP was the demonstration that the existence of perfect zero-knowledge for an $NP-complete$ problem would cause the Polynomial Time Hierarchy to collapse [9]. However, this work deals with a different ZKP known as computational ZKP, whose existence has been proven for any NP-problem under the assumption that a one-way function exists [11]. In the same work, the authors provided a ZKP for the 3-coloring problem and suggested the use of standard reductions to achieve a ZKP for any other NP-problem. The efficiency of the algorithm here proposed comes from a distinct approach based on an specific design of a computational ZKP for a concrete NP-problem. It avoids the use of general reductions by combining tools from Number

---

* Research partly supported by the Spanish Ministry of Education and Science and the European FEDER Fund under Project SEG2004-04352-C04-03.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 346–354, 2005.
© Springer-Verlag Berlin Heidelberg 2005

Theory and Graph Theory. Indeed, since this latter field is a dense source of NP-problems, several ZKP for different graph problems such as isomorphism, non-isomorphism, hamiltonian circuits, clustering and independent vertex set have been previously introduced in the literature [13] but it is remarkable that all of them are based exclusively on Graph Theory problems. Special mention regards the algorithms proposed in [5] since they are described for the same problem, the Independent Vertex Set Problem ($IVSP$), as this present work. However here a number theoretical problem, the Discrete Logarithm Problem (DLP), is also involved in the design of the ZKP in order to improve its efficiency and security.

All the aforementioned bibliographic references include proposals related in some way to the same basic graph problem, the Graph Isomorphism [3]. The major drawback of such an approach is due to the fact that the computational complexity of this problem is not yet known and furthermore the problem seems to be easy for most random graphs [8]. On the contrary, the present work proposes a new Computational ZKP for the $IVSP$ whose security relies on the hardness of a number theoretical problem, the DLP, whose difficulty is generally assumed in Cryptography, [4]. On the other hand, while general ZKP seem to be the most promising identification method in ad-hoc wireless networks, the concrete choice of the $IVSP$ as base of our proposal comes from the necessity of efficient ZKP for such a problem in broadcast models pointed out in [10].

This paper is organized as follows. First we recall the basic requirements for the design of a ZKP. Then, in Section 3, the problems and notations that are used throughout the work are defined. In the following Section, the proposed ZKP is fully described and its security is formally proved. The adequate choice of parameters and the performance of the scheme are analyzed in Section 5. Finally, several conclusions and open questions are drawn in Section 6.

2 ZKP Design

A Zero-Knowledge Proof of Knowledge ($ZKPK$) may be defined as a two-party cryptographic protocol that allows an infinitely powerful prover Alice (A) to convince a probabilistic polynomial time verifier Bob (B), beyond any reasonable doubt, that she knows some verifiable information such as the solution of a given difficult problem, but in a way that does not help him to determine anything about this information.

The three main characteristic properties of $ZKPK$ are completeness (which means that if the claim is valid, then A convinces B of it with very high probability), soundness (if the claim is not valid, then B is convinced of the contrary with very small probability), and zero-knowledge (B does not receive any other information except for the certainty that the claim is valid). This latter property may be checked through the demonstration that the prover A can be replaced by an efficient (expected polynomial time) simulator which generates an interaction indistinguishable from the real one. The main difficulty of this proof, which is usually based on a constructive specification of the way such a simulator proceeds, is to achieve that the simulator convince the verifier about the knowledge of the secret information without actually having it. Generally, this problem is solved thanks to the rewinding capability of the simulator, which may use several tries to answer the verifier without letting him know how many tries the simulator has used.

Two basic variants of zero-knowledge may be distinguished depending on the assumed computing power of possible dishonest parties. Computational zero-knowledge arises when it would take more than polynomial time for a dishonest verifier to obtain some information about the secret, whereas perfect zero-knowledge involves that even an infinitely powerful cheating verifier could not extract any information. Both previous notions can also be characterized through the amount of computational resources necessary to distinguish between the interaction generated by the simulator and the verifier, and the one associated to the prover and the verifier.

Generally, bit commitment and cut-and-choose techniques are basic ingredients for the design of $ZKPK$. In these cases, A 'cuts' her secret solution in several parts, commits to them, and afterwards B chooses at random one of those parts as a challenge. The typical design of $ZKPK$ is also based on the existence of a concrete possibility of fraud: a cheater is usually able to answer to some types of challenges (for which he was prepared in advance) but not for all of them. So, most protocols are designed as interactive challenge-response schemes in such a way that some of A's possible responses prove A's knowledge of the secret solution, whereas the others guarantee against A's possible fraud. More concretely, an answer to one question gives no information (zero-knowledge), while answering all the questions is proved to reveal prover's knowledge (soundness). So, the security is based on the impossibility that the prover can predict verifier's questions. Also typically $ZKPK$ consist of several iterations of the atomic subroutine described below, so that by repeating it an enough number of times the verifier's confidence in the prover's honesty increases because the global fraud probability becomes smaller with the number of iterations. A. Consequently, this number m of iterations should be agreed by A and B according to their different interests.

3 Notations and Definitions

As mentioned before, the two problems that constitute the base of the proposed algorithm are the Independent Vertex Set Problem ($IVSP$) and the Discrete Logarithm Problem (DLP).

On the one hand, the DLP may be described as follows. Let p be a prime, let g be a generator of $\mathbb{Z}_p^*$ (the multiplicative group of integers modulo p) and let x be an integer between 0 and $p-1$. Define $DLP_{p,g}(x)$ to be y such that $0 < y < p$, $g^y = x (mod\ p)$. Such a problem is in NPI class, which means that no probabilistic polynomial algorithm is known for solving it. The intractability assumption of the DLP has been yet used on public-key cryptography and as single base of a $ZKPK$ [4].

On the other hand, the $IVSP$ is an $NP - complete$ problem that may be defined as follows. Given a graph $G = (V, E)$, it consists in finding a size k subset $I \subseteq V$ (independent vertex set) such that no two vertices in I are joined by an edge in E.

A useful method to hide an independent vertex set in a graph such that it is resistant to general heuristic approaches has been described in [2]. This method tries to balance the vertices degree sequence so that there is no difference between those belonging to the independent vertex set and the others. Consequently, due to its robustness it may be used in the instances generation of the proposed algorithm.

The concrete choice of the $IVSP$ as base for our proposal seems to be quite convenient since recent work on network security [10] has pointed as an important contribution to the field of the design of protocols for broadcast channels the definition of efficient $ZKPK$ for the $IVSP$.

Since the $IVSP$ is NP-complete, by the result of [9], we know that this problem cannot have perfect $ZKPK$ unless the polynomial hierarchy collapses, so the $ZKPK$ for the $IVSP$ described in the next section is a computational $ZKPK$.

4 Zero-Knowledge Proof of Knowledge for the Independent Vertex Set Problem

A Computational $ZKPK$ for the $IVSP$ that uses a bit commitment scheme based on the DLP is now presented. In our proposal A's inputs are a graph $G = (V, E)$ and an integer k, and her goal is to convince B that she knows a size k independent vertex set of G.

In a pre-processing stage, A generates at random a graph G with n vertices and an embedded secret independent vertex set I of size k through the method described in [2], and publishes her inputs (G, k). Such a construction allows that the embedded and secret independent vertex set I may be used in practice as A's secret identification because hiding the secret subset I takes polynomial time. During the processing stage of the algorithm, in each iteration A generates a c-coloring of G where the k vertices of I have the same colour that is not used for any other vertex. It must be pointed out that the number of colours c is not restricted to any value, so the computation of such a c-coloring takes polynomial time.

A's secret commitment is then formed by c binary n-dimensional vectors $a_i = (a_i^j), a_i^j \in \{0, 1\}, i = 1, 2, ..., c, j = 1, 2, ..., n$, where each position corresponding to a vertex j colored with colour i contains a one and the rest contains a zero. The cardinality of each vertex subset defined by the c-coloring is given by the Hamming weight of vector a_i, $W_H(a_i)$ (where the Hamming weight of a vector is simply the number of nonzero digits in the vector), which is a value which plays a special role in the algorithm. After an initialization stage where A and B agree on integers m and c, primes p_i $(i = 1, 2, ..., c)$, generators g_i of $\mathbb{Z}_{p_i}^*, (i = 1, 2, ..., c)$

and random integers $r_i \in \mathbb{Z}^*_{p_i}$ $(i = 1, 2, ..., c)$, the $IVSP - ZKPK$ algorithm consists of m iterations of the following four steps:

Atomic Subroutine:

Commitment step: A generates the vectors $a_i = (a_i^1, a_i^2, ..., a_i^n), i = 1, 2, ..., c,$ chooses secret random integers $y_j \in \mathbb{Z}^*_{p-1}$, $j = 1, 2, \ldots, n$, with $p = \min\limits_{i=1,\ldots,c} \{p_i\}$ and commits to such parameters by sending to B the n-dimensional vectors: $V_i = ((r_i^{a_i^j} \cdot g_i^{y_j}) mod\ p_i), (i = 1, 2, \ldots, c,\ j = 1, 2, \ldots, n)$.

Challenge step: B chooses at random and sends to A one bit b, and if $b = 0$, he also sends two random adjacent vertices v and w.

Response step: A sends to B:

- if $b = 0$, the integers y_v and y_w
- else, the integers $y = \sum\limits_{j=1}^{n} y_j$ and $W_H(a_i) = \sum\limits_{j=1}^{n} a_i^j,\ i = 1, 2, \ldots, c.$

Verification step: B checks whether the values provided by A in previous steps are correct, that is to say,

- when $b = 0$, from the elements V_i^v and V_i^w (i=1,2,. . . ,c), B checks that only two different vectors exist where the components associated to v and w have the value 1. $(\exists! h, l \in \{1, 2, \ldots, c\}\ | h \neq l,\ a_h^v = a_l^w = 1)$.
- when $b = 1$, B checks that
  - $\sum\limits_{i=1}^{c} W_H(a_i) = n,$
  - $\exists i \in \{1, 2, ..., c\} | W_H(a_i) = k,$
  - $\forall i \in \{1, 2, ..., c\} : (\prod\limits_{j=1}^{n} r_i^{a_i^j} \cdot g_i^{y_j}) = (r_i^{W_H(a_i)} \cdot g_i^{y}) mod\ p_i.$

Note that in the previous algorithm the verification step is only possible thanks to B's knowledge of g_i, p_i, r_i and V_i and the use of efficient modular exponentiation methods.

In order to prove the security of the protocol, we follow the approach of [6], first proving completeness, then soundness and finally the zero-knowledge property.

Theorem 1. *The $IVSP - ZKPK$ algorithm is a computational zero-knowledge proof of knowledge for the independent vertex set problem.*

Sketch of Proof. In order to prove that completeness is met there should be shown that if A knows a k size independent vertex set in G and both participants follow correctly the protocol, then the verifier B always accepts the proof.

If challenge $b = 0$ is requested by B, he should check that two chosen adjacent vertices v and w are colored with exactly two different colours. To achieve this, he computes $g_i^{y_v}$ and $g_i^{y_w}$, $\forall i = 1, 2, \ldots, c$ and compares these values with the corresponding components in the committed vectors V_i. If A has built an appropriate coloring, B finds that there is a unique vector V_h where the v-th component coincides with $r_h \cdot g_h^{y_v}$, whereas the v-th component in the other vectors is equal to $g_i^{y_v}$. The same occurs for some vector V_l and w-th component.

If B chooses bit $b = 1$ as challenge, both the Hamming weight of coloring vectors a_i, and the value y provided by A in the response step, allow B to check the following items:

- $\sum_{i=1}^{c} W_H(a_i) = n$,, so all the vertices are colored using only one colour.
- $\exists i \in \{1, 2, ..., c\} \, | W_H(a_i) = k$, so there is at least a size k independent vertex set to whose vertices the coloring has assigned the same colour.
- $(\prod_{j=1}^{n} r_i^{a_i^j} \cdot g_i^{y_j}) = (r_i^{a_i^1} \cdot g_i^{y_1}) \cdot (r_i^{a_i^2} \cdot g_i^{y_2}) \cdots (r_i^{a_i^n} \cdot g_i^{y_n}) = (r_i^{a_i^1} \cdot r_i^{a_i^2} \cdots r_i^{a_i^n}) \cdot (g_i^{y_1} \cdot g_i^{y_2} \cdots g_i^{y_n}) = r_i^{\sum_{j=1}^{n} a_i^j} \cdot g_i^{\sum_{j=1}^{n} y_j} = r_i^{W_H(a_i)} \cdot g_i^{y}$, so the c committed vectors have been properly computed.

In order to prove soundness, if the prover A does not know any size k independent vertex set in G and the verifier B follows correctly the protocol, then no matter how A plays, B should reject the proof with high probability. In such a case, A has basically two possible ways to try to fool B. She could use an incorrect c-coloring of G with some vertex subset of cardinality k, or she could compute correct coloring vectors a_i with no vertex subset of Hamming weight k. In the first case, there exists at least a vector $a_i, i \in \{1, 2, \cdots, c\}$ such that two adjacent vertices are colored with the same colour, and hence B could detect the fraud if $b = 0$ with probability at least $1/|E|$ in each iteration. When a dishonest prover A uses a correct coloring, if B chooses bit $b = 1$ he always detects the fraud when checking the existence of a vertex subset of size k colored with the same colour ($\exists i \in \{1, 2, ..., c\} | W_H(a_i) = k$).

Hence, under the assumption that a dishonest prover A chooses at random the way to commit fraud, after m successive and independent iterations with uniformly random chosen challenges, the probability that A successfully cheats B is upper bounded by $(2^{-2m} \cdot (3 - 1/|E|)^m)$.

Regarding computational zero-knowledge, we need to show that the prover A conveys no knowledge to any possible verifier, including ones that deviate arbitrarily from the protocol. From the received witnesses, B should be able to obtain the committed c-coloring and consequently the independent vertex set I, only if he is able to solve the DLP. So, according to the simulation paradigm, it is possible to build an expected polynomial time simulator that generates a probability distribution which is polynomially indistinguishable from the distribution induced during the interaction between A and B. In particular, the simulator first tries to guess B's challenge so chooses a random bit b'. Then, if $b' = 0$ the simulator generates at random correct coloring vectors a_i and consequent witnesses passing verification. Else, if $b' = 1$, false coloring vectors a_i are generated such that there is a vertex subset with cardinality k colored using the same colour that is not used for any other vertex.

The simulator tries one of both possible challenges at random and if it coincides with B's challenge b (which happens with probability exactly 1/2) then its output is polynomially indistinguishable from A's output. Else, it reinitiates.

So, in an expected polynomial time the described simulator may replace A, and therefore computational zero-knowledge is proven. □

The proposed $IVSP-ZKPK$ algorithm may be applied in a quite natural way as an identification protocol with advantages compared to other similar schemes. In the corresponding identification scheme the public file containing records for each user should consist of each name and the respective auxiliary identification information composed of a graph G and the size k of the secret embedded independent vertex set. All users should have free read access to this public file. When a user A wishes to convince B of her identity, she invokes the identification protocol with the public file record corresponding to her name as a parameter, and B verifies her record in the public file and proceeds executing his role in the protocol. So, soundness property of the $IVSP-ZKPK$ algorithm yields that an impersonation attack is practically infeasible.

A different application of the independent vertex set Problem in Cryptography has been recently addressed in [5]. This application has to do with the computation of an access structure taking into account the relationship graph. By using the algorithm here proposed it is also possible to convince an adversary of the validity of such an access structure without revealing who the honest members are. Another cryptographic use of this problem was addressed in the same paper where the authors proposed tackling the key scrow problem via the independent vertex set problem. In general, different cryptologic applications of this problem may be described anywhere it is important that the participants with access privileges to a determined information form an independent vertex set.

5 Complexity Analysis

This section specifies both the techniques used to build the instances that are necessary for the $IVSP-ZKPK$ algorithm, and the complexity of the different operations that are required.

First of all, the random generation of a graph G with n vertices and an embedded independent vertex set of size k takes $O((n-k)^2)$. Then, the coloring is accomplished through a well-known greedy heuristic that takes $O(n^3)$ under the worst-case analysis. In order to build the instances of the DLP, A should generate c prime numbers and use the modular exponentiation algorithm $c \cdot n$ times whose complexity is $O(c \cdot n \cdot \log^3 l)$ (where $l = \max_{i=1,2,\ldots,c} \{p_i\}$). The response associated to the challenge $b = 0$ takes constant time, whereas if the challenge chosen by B is $b = 1$ the computation of the answer takes linear time. Hence, the computational complexity associated to A may be estimated by $O(n^3)$, which confirms that the assumed restriction of the capabilities of A to polynomial time is possible.

The operations corresponding to B when he has to verify the answers provided by A depend on the challenge chosen by him. So, if the challenge is $b = 0$, then B should compute several exponential operations that take $O(c \cdot log^3 l)$. On the other hand, also the operations associated to the process of verification when the challenge is $b = 1$ are of order $O(c \cdot log^3 l)$.

Regarding the communication complexity, in the pre-processing stage the prover should publish the graph $G = (V, E)$, so $O(|E| \cdot log(n))$ determines the size of the file containing it. Furthermore, the committed vectors are formed by $n \cdot c$ integers and the response to both challenges is composed by two integers, so the corresponding communication complexity of the algorithm is $O(m \cdot c \cdot log(n))$.

So, a question regarding the use of $IVSP - ZKPK$ algorithm that deserves special attention is the choice of parameters such as n, c and p_i. In order to guarantee the security of the scheme, graph's size n and primes p_i should be large enough, whereas it is convenient that the number of colours c be small in order to reduce communication complexity. Also generation and computer representation of random graphs G are important factors having a profound effect on the complexity of the algorithm. In particular, experimental analyzes of generations of graphs guaranteeing the difficulty of the $IVSP$ recommends the use of $n > 1000$, [2].

6 Conclusions

In this work a new computational zero-knowledge proof of knowledge has been described for the independent vertex set Problem. Its validity based on the difficulty of the discrete logarithm problem has been formally established.

Among known schemes based on NP-complete problems, the one proposed in this paper is one of the most efficient when parameters are adequately chosen. Due to its efficiency and to the basic problem of the independent vertex set problem, the proposed scheme may be used for identification and access control systems, as well as for the design of secure network protocols for broadcast channels and for the computation of access structures.

A full comparison among the computational efficiency of the proposed algorithm and other previous schemes is part of a work in progress. Also a forthcoming version of this work will include some open questions such as the range of parameters that guarantees both security and efficient performance.

References

1. Brassard, G., Chaum, D., Crépeau, C.: Minimum disclosure proofs of knowledge. Journal of Computer and System Sciences **37** (1988) 156–189.
2. Brockington, M., Culberson, J.: Camouflaging independent vertex sets in quasi-random graphs. Johnson, D., Trick, M., eds.: Cliques, Coloring and Satisfability. Volume XXVI. American Mathematical Society (1994) 75–88.
3. Caballero, P., Hernández, C.: Strong Solutions to the Identification Problem. Wang, J., ed.: Computing and Combinatorics, Berlin, Springer-Verlag (2001) 257–261 Lecture Notes in Computer Science Volume 2108.
4. Chaum, D., Evertse, J.H., van de Graaf, J., Peralta, R.: Demonstrating possession of a discrete logarithm without revealing it. Odlyzko, A.M., ed.: Advances in Cryptology - Crypto '86, Berlin, Springer-Verlag (1986) 200–212 Lecture Notes in Computer Science Volume 263.

5. Desmedt, Y., Wang, Y.: Efficient Zero-Knowledge Protocols for Some Practical Graph Problems.Third Conference on Security in Communication Networks'02, Springer-Verlag (2003) 296–308 Lecture Notes in Computer Science Volume 2576.
6. Feige, U., Fiat, A., Shamir, A.: Zero-knowledge proofs of identity. Journal of Cryptology **1** (1988) 77–95.
7. Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. Odlyzko, A.M., ed.: Advances in Cryptology - Crypto '86, Berlin, Springer-Verlag (1986) 186–194 Lecture Notes in Computer Science Volume 263.
8. Fortin, S.: The Graph Isomorphism Problem. Technical Report TR 96-20, University of Alberta, Department of Computer Science (1996).
9. Fortnow, L.: The complexity of perfect zero-knowledge. Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing, STOC 87. (1987) 204–209.
10. Franklin, M., Wrigh, R.: Secure communication in minimal connectivity models. Journal of Cryptology **13** (2000) 9–30.
11. Goldreich, O., Micali, S., Wigderson, A.: How to prove all NP-statements in zero-knowledge, and a methodology of cryptographic protocol design. Odlyzko, A.M., ed.: Advances in Cryptology - Crypto '86, Berlin, Springer-Verlag (1986) 171–185 Lecture Notes in Computer Science Volume 263.
12. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems.Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing (STOC 85). (1985) 291–304.
13. Santis, A.D., Crescenzo, G.D., Goldreich, O., Persiano, G.: The graph clustering problem has a perfect zero-knowledge proof. Information Processing Letters **69** (1999) 201–206.
14. Schnorr, C.P.: Efficient identification and signatures for smart cards. Brassard, G., ed.: Advances in Cryptology - Crypto '89, Berlin, Springer-Verlag (1989) 239–252 Lecture Notes in Computer Science Volume 435.

Improvement of the Edit Distance Attack to Clock-Controlled LFSR-Based Stream Ciphers*

Pino Caballero-Gil[1] and Amparo Fúster-Sabater[2]

[1] D.E.I.O.C. University of La Laguna. 38271 La Laguna, Tenerife, Spain
`pcaballe@ull.es`
[2] Institute of Applied Physics. C.S.I.C. Serrano 144, 28006 Madrid, Spain
`amparo@iec.csic.es`

Abstract. The main idea behind this paper is to improve a known plaintext divide-and-conquer attack that consists in guessing the initial state of a Linear Feedback Shift Register component of a keystream generator, and then trying to determine the other variables of the cipher based on the intercepted keystream. While the original attack requires the exhaustive search over the set of all possible initial states of the involved register, this work presents a new and simple heuristic optimization of such an approach that avoids the evaluation of an important number of initial states when launching a constrained edit distance attack on irregularly clocked shift registers.

1 Introduction

Stream ciphers have extensive applications in secure communications, e.g. wireless systems, due to different practical advantages such as easy implementation, high speed and good reliability. When designing a stream cipher, the main goal is to expand a short key into a long pseudorandom keystream in such a way that it should not be possible to reconstruct the short key from the keystream. In this work we focus on stream ciphers based on Linear Feedback Shift Registers ($LFSRs$), such as A5 for GSM [12] or the function E0 for Bluetooth [2]. Other examples of $LFSR$-based generators are LILI-II [3], Toyocrypt [5], Shrinking [4] and Alternating Step [7] generators. All these generators produce keystream sequence with high linear complexity, long period and good statistical properties, [10]. In particular, the two last generators were thoroughly analyzed in [14] through a correlation attack based on a decoding problem.

The main idea behind this paper is to improve a known plaintext divide-and-conquer attack that consists in guessing the initial state of an $LFSR$ component of the generator, trying to determine the other variables of the cipher based on the intercepted keystream, and then checking that the initial guess was consistent with the observed keystream sequence. Such an attack was first proposed in [8] by means of a theoretical model and a distance function known as Levenshtein or

---

* Research supported by the Spanish Ministry of Education and Science and the European FEDER Fund under Projects SEG2004-04352-C04-03 and SEG2004-02418.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 355–364, 2005.
© Springer-Verlag Berlin Heidelberg 2005

edit distance. This distance was also used in [9] to attack a single $LFSR$-based generator. On the other hand, it has been proven [13] that when the length of the intercepted sequence is large enough, the number of candidate initial states is small. The attack considered here may be seen as an extension of the constrained edit distance attack to clock-controlled $LFSR$-based generators presented in [15]. Our main aim is to investigate whether the number of initial states to be analyzed can be reduced independently of the length of the intercepted sequence. In fact, this feature has already been pointed out in [6] as one of the most interesting problems in the cryptanalysis of stream ciphers today. So, while according to the original method, the attacker needs to traverse an entire search tree including all the possible $LFSR$ initial states, in this work we try to improve such an attack by simplifying the search tree in such a way that only the most efficient branches are retained. This new approach produces a significant improvement in the computing time of the original edit distance attack since it implies a dramatic reduction in the number of initial states that need to be evaluated.

This work is organized as follows. Section 2 introduces some definitions and basic concepts regarding the computation of constrained edit distances. In Section 3, some ideas for an efficient initial state selection method are introduced. Such a method allows us both to deduce a threshold value for the edit distance, and to discard beforehand an important number of initial states. Section 4 provides the full description of the improved algorithm, which takes advantage of the threshold value described in the previous Section. Finally, Section 5 contains simulation results while in Section 6 several conclusions are drawn.

2 Constrained Edit Distance Attack

The Levenshtein or edit distance may be defined as the minimum number of elementary operations (insertions, deletions and substitutions) required to transform one sequence X of length N into another sequence Y of length M. Some of the different applications of the edit distance are, for instance, file revision, spell correction, plagiarism detection, molecular biology, and speech recognition. The dynamic programming approach is a classical solution for computing the edit distance matrix, where the distances between longer and longer prefixes of the sequences are successively evaluated until the final result is achieved. When applying an edit distance attack to a stream cipher and depending on the generator design, some edit operations may be restricted. In this case, a so-called constrained edit distance may be necessary.

The specific theoretical model considered in this work for the attacked generator is described in Fig. 1. As usual, it is assumed that the $LFSR$ feedback polynomial is known. The use of this general model implies that the known plaintext attack is applicable not only to those generators that fit exactly the simplest version of such a model but also to all the sequences produced by more complex generators that also fulfills the description. In this latter case, it is understood that the attack will provide a simpler equivalent description of the original attacked generator.

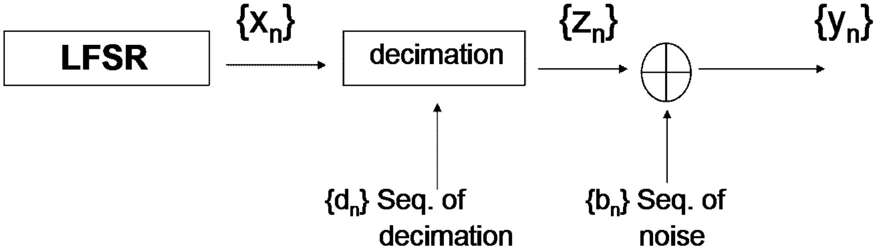


Fig. 1. Theoretical model

An essential step in edit distance attacks is the computation of the edit distance matrix $W = (w_{i,j}), i = 0, 1, \ldots, N - M, j = 0, 1, \ldots, M$ associated with each possible couple of sequences X and Y where Y represents the intercepted keystream sequence while X is each one of the $LFSR$ sequences produced by each one of the possible initial states. In the following, some of the parameters of such a matrix are described. Firstly, its dimension is $(N-M+1)(M+1)$. Secondly, the element $w_{N-M,M}$ represents the edit distance between the sequences X and Y. Lastly, each element of the matrix $w_{i,j}$ corresponds exactly to the edit distance between prefix subsequences $x_1, x_2, \ldots, x_{i+j}$ and $y_1, y_2, \ldots, y_j$.

In the constrained edit distance attack here analyzed only deletions and substitutions are allowed. Those two elementary operations may be seen as the result of an irregular decimation on the $LFSR$ sequence plus the addition of a noise sequence respectively. Furthermore, in this work it is assumed that the number of consecutive deletions is 1 (constrained edit distance). Under this hypothesis, the length of X may be estimated as $N \approx 3M/2$, which coincides with the mathematical expectation. The previous hypothesis implies that the computation of $2(N - M)(N - M + 1)$ elements of the matrix W corresponding to the two triangles: $\{w_{i,j} : i = 1, \ldots, N - M, j = 0, \ldots, i - 1\}$ and $\{w_{i,j} : i = 0, \ldots, N - M - 1, j = 2M - N + 1 + i, \ldots, M\}$ can be avoided. The remaining elements $w_{i,j}$ of the constrained edit distance matrix W may be computed recursively by columns according to the formulas in Equation (1).

$$w_{0,0} = 0$$
$$w_{i,j} = min\{w_{i,j-1} + P_s(x_{i+j}, y_j), w_{i-1,j-1} + P_d(x_{i+j}, y_j)\} \text{ where}$$
$$P_s(x_{i+j}, y_j) = \begin{cases} 0 \; if \; x_{i+j} = y_j \\ 1 \; if \; x_{i+j} \neq y_j \end{cases}$$
$$P_d(x_{i+j}, y_j) = \begin{cases} 1 \; if \; x_{i+j} = y_j \\ 2 \; if \; x_{i+j} \neq y_j \end{cases} \tag{1}$$

The elements of the matrix W may be seen as costs of optimal paths in an induced graph with as many vertices as elements in the matrix W. Moreover, the arcs have costs 0,1 or 2 depending on the coincidences between the corresponding bits of Y and X, (see equation (1)). In such an induced graph, the optimal paths between the source associated with the element $w_{0,0}$ and the sink corresponding to the element $w_{N-M,M}$ give us the solution of the cryptanalytic attack by specifying both decimation and noise sequences $D = \{d_n\}$ and $B = \{b_n\}$, respectively.

Example : For an intercepted keystream sequence Y:1101011 of length M=7 and a candidate sequence X:1110110111 of length N=10, the constrained edit distance matrix is: $W = \begin{pmatrix} 0 & 0 & 0 & 1 & 2 & - & - & - \\ - & 1 & 1 & 1 & 1 & 2 & - & - \\ - & - & 3 & 3 & 2 & 2 & 2 & - \\ - & - & - & 5 & 5 & 4 & 3 & 3 \end{pmatrix}$. The graph induced by this matrix is shown in Fig. 2 where the twelve optimal paths are remarked in bold.

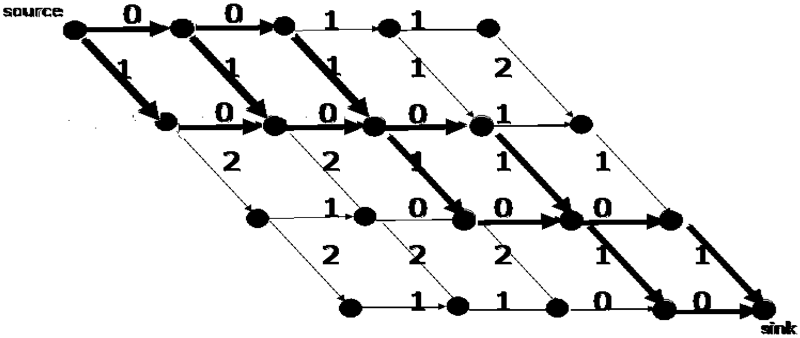


Fig. 2. Induced graph and optimal paths

From those optimal paths, the 12 possible solutions to the cryptanalysis corresponding in this case to decimation without noise are expressed in terms of decimation sequences D.

$$D = \{d_n\} : \begin{cases} 0010010010 : Solution 1; \; 0010010100 : Solution 2 \\ 0010100010 : Solution 3; \; 0010100100 : Solution 4 \\ 0100010010 : Solution 5; \; 0100010100 : Solution 6 \\ 0100100010 : Solution 7; \; 0100100100 : Solution 8 \\ 1000010010 : Solution 9; \; 1000010100 : Solution 10 \\ 1000100010 : Solution 11; \; 1000100100 : Solution 12 \end{cases}$$

3 Threshold Search

The main idea behind the method described in this section comes from the association between bits x_{i+j} of X and arcs of the graph induced by the matrix W. In particular, we consider cut sets between the source and the sink in the induced graph, which allow us to define two sets of conditions for the sequences X either to establish a threshold edit distance or to discard a set of initial states. In this way, once a subsequence of Y fulfills some of the previous conditions, the cost of the corresponding cut set can be guaranteed either to be minimum or not to be minimum, respectively. This fact has direct consequences on the costs of the optimal paths, that is to say, on the edit distances.

The cut sets that we use in this work are defined as follows. Each cut set $C_{i+j}, 1 \leq i+j \leq N$ contains both the set of all the arcs corresponding to the vertex x_{i+j}, and all those arcs corresponding to bits x_w with $w > i+j$ whose output vertex is one of the output vertices of the former set. From these cut sets, we deduce several independent conditions on the sequence Y that may be used to guarantee both a decrease and an increase on the edit distances of different sequences X. In particular, the conditions obtained from the defined cut sets may be described by the formulas in Equation (2).

$$\begin{array}{l}
\forall j : 1, 2, \ldots, \lfloor M/2 \rfloor ; y_j = y_{j-1} = y_{j-2} = \cdots = y_{\lceil j/2 \rceil} \\
\forall j : \lfloor M/2 \rfloor + 1, \lfloor M/2 \rfloor + 2, \ldots, \lceil 3M/4 \rceil - 1 ; y_j = y_{j-1} = \cdots = y_{j - \lceil (M-2)/4 \rceil} \\
\forall j : \lceil 3M/4 \rceil, \lceil 3M/4 \rceil + 1, \ldots, M ;\ y_j = y_{j-1} = y_{j-2} = \cdots = y_{2j-M}
\end{array} \qquad (2)$$

The checking procedure of these hypothesis takes polynomial time as it implies a simple verification of the lengths of the runs in Y. After having checked each hypothesis separately, the tools used to verify both sets of conditions on X are described in terms of a pattern and a counterpattern, which are made out of independent bits of X according to the formulas in Equation (3).

$$\begin{array}{l}
\forall j : 1, 2, \ldots, \lfloor M/2 \rfloor ; \text{ if } y_j = y_{j-1} = y_{j-2} = \cdots = y_{\lceil j/2 \rceil} \text{ then} \\
\quad \begin{cases} x_j = x_{j+1} = y_j & X - Pattern \\ x_j = x_{j+1} \neq y_j & X - Counterpattern \end{cases} \\
\forall j : \lfloor M/2 \rfloor + 1, \ldots, \lceil 3M/4 \rceil - 1 ; \text{ if } y_j = y_{j-1} = \cdots = y_{j - \lceil (M-2)/4 \rceil} \text{ then} \\
\quad \begin{cases} x_{2j - \lceil M/2 \rceil} = x_{2j - \lceil M/2 \rceil + 1} = x_{2j - \lceil M/2 \rceil + 2} = y_j & X - Pattern \\ x_{2j - \lceil M/2 \rceil} = x_{2j - \lceil M/2 \rceil + 1} = x_{2j - \lceil M/2 \rceil + 2} \neq y_j & X - Counterpattern \end{cases} \\
\forall j : \lceil 3M/4 \rceil, \lceil 3M/4 \rceil + 1, \ldots, M ; \text{ if } y_j = y_{j-1} = y_{j-2} = \cdots = y_{2j-M} \text{ then} \\
\quad \begin{cases} x_{2j - \lceil M/2 \rceil} = x_{2j - \lceil M/2 \rceil + 1} = x_{2j - \lceil M/2 \rceil + 2} = y_j & X - Pattern \\ x_{2j - \lceil M/2 \rceil} = x_{2j - \lceil M/2 \rceil + 1} = x_{2j - \lceil M/2 \rceil + 2} \neq y_j & X - Counterpattern \end{cases}
\end{array} \qquad (3)$$

The X-pattern allows to discover initial states producing sequences X with a low edit distance. Furthermore, the X-pattern provides a good quality threshold for the method that will be described in the following Section. On the other hand, sequences X fulfilling the X-counterpattern lead to high edit distance values and consequently may be directly discarded.

Example : Given a sequence Y of length M=7 and a candidate sequence X of length N=10, we may define the cut sets shown in Fig. 3. The 6 independent hypothesis on runs in Y that are deduced from those cut sets are $y_2 = y_1$, $y_3 = y_2$, $y_4 = y_3 = y_2$, $y_5 = y_4 = y_3$, $y_6 = y_5 = y_4$, and $y_6 = y_5$. Consequently, since the example Y:1101011 only fulfills the first hypothesis, the second and third bits are fixed in both the X-pattern and the X-counterpattern. If the length of the $LFSR$ equals 9 and its feedback polynomial is $1 + x + x^3 + x^4 + x^9$, then only 16 initial states (out of the $2^9 = 512$ possible) will satisfy the X-pattern:111.....11 and may be considered as the most promising initial states. Consequently, they need to be fully evaluated in order to deduce a threshold on the edit distance. On the other hand, the counterpattern of X is defined by: 000.....00. In a similar

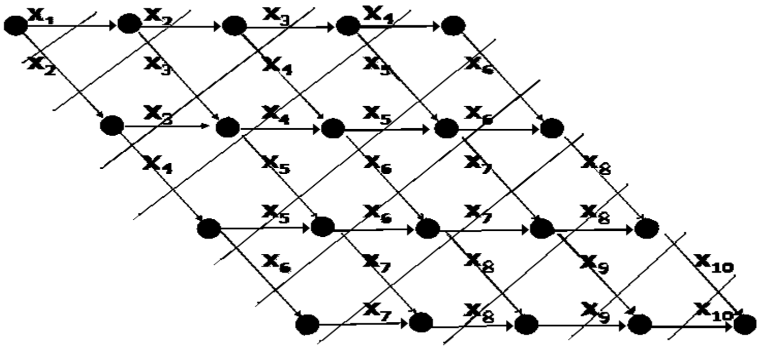


Fig. 3. Cut sets

way as before, only 16 initial states (out of the $2^9 = 512$ possible) will satisfy the X-counterpattern. They must be rejected as they are the less promising initial states. Indeed, in this example, it may be checked that there are 10 sequences X fulfilling the X-pattern that are solutions to the cryptanalysis. In fact, their edit distance is 3 that is precisely the minimum edit distance for this example. Moreover, it might be also verified that all the 16 initial states fulfilling the X-counterpattern lead to edit distances greater than 3.

4 Improved Attack

The threshold obtained through the X-pattern as well as the discarded initial states deduced from the X-counterpattern are items of the improved attack that is described in this Section. The algorithm here developed makes use of a new concept, the so-called *bad column*, which leads to a considerable saving in the computation of the edit distance matrices. A *bad column* with respect to a threshold T may be defined as a column j_0 of the edit distance matrix W such that each one of their elements fulfills the Equation (4):

$$w_{i,j_0} > T - (N - M - i), \forall i \tag{4}$$

Once an edit distance threshold has been obtained, we may use such a threshold to stop the computation of any matrix W as soon as a *bad column* has been detected. This is due to the knowledge that the edit distance corresponding to the candidate initial state will be greater than the threshold. In this simple way, two new improvements on the original attack may be achieved. On the one hand, as yet mentioned, the computation of any matrix may be stopped as soon as a *bad column* is obtained. On the other hand and thanks to the association between bits x_{i+j} and arcs of the graph, we may define a new counterpattern on the initial states of the $LFSR$, the so-called IS-counterpattern. This new concept will allow us to discard the set of initial states fulfilling such an IS-counterpattern when an early *bad column* has been detected. This is so because once a *bad column* has been obtained, it is possible to discard directly all the initial states whose first bits coincide with those used within the computation of the *bad column.* Note that in order to take full advantage of *bad columns*, it is convenient to have some efficient way of obtaining soon a good threshold. That is exactly the effect of the X-pattern described in the previous section.

We are now ready to describe the improved edit distance attack algorithm

Algorithm

Input: The intercepted keystream sequence Y of length M, and the feedback polynomial of the $LFSR$ of length L.

Output: The initial states of the $LFSR$ producing sequences X of length $3M/2$ whose constrained edit distance is minimum, and the corresponding decimation and noise sequences D and B, respectively.

1. Verification of the hypothesis on Y described in Equation (2) .
2. Definition of the X-pattern and X-counterpattern according to Equation (3).
3. Rejection of all initial states that produce sequences X fulfilling the X-counterpattern.
4. For each initial state that produces a sequence X fulfilling the X-pattern:
   (a) Computation of the edit distance matrix according to Equation (1).
   (b) Updating of the threshold T.
5. For each initial state producing a sequence X that does not fulfill the X-pattern and that has not been previously discarded:
   (a) Computation of the edit distance matrix stopping when detection of *bad columns* according to threshold T and Equation (4).
   (b) Definition of the IS-counterpattern.
   (c) Rejection of all initial states producing sequences X fulfilling the IS-counterpattern.
6. For each initial state producing a sequence X with minimum edit distance:
   (a) Recovery of the optimal paths from the graph induced by the edit distance matrix.
   (b) Translation from each optimal path into a couple of decimation and noise sequences (D,B).

Example :

Given a sequence Y:1101011 of length M=7, and the feedback polynomial: $1 + x + x^3 + x^4 + x^9$ of the $LFSR$ of length L=9. Since the unique hypothesis fulfilled by Y is: $y_2 = y_1$, the X-pattern:111.....11, and the X-counterpattern:000.....00. So, we have the consequent rejection of the 16 initial states producing sequences of the form 000.....00. Also, for each one of the 16 initial states generating sequences of the form 111.....11 the edit distance matrix is computed, and from this computation the threshold T=3 is obtained. For the remaining 480 initial states, we start computing the edit distance matrix, and stop as soon as a *bad column* for the threshold T=3 is detected. In particular, we have to start to evaluate:

- 1 initial state in order to discard $2^7 = 128$ initial states, including those ones discarded by the X-counterpattern, corresponding to matrices W containing the *bad column* j_0=1.
- 3 initial states in order to discard 96 initial states due to the fact that the column j_0=2 of W is a *bad column.*
- 5 initial states which allow us to discard 40 initial states due to the fact that the column j_0=3 of W is a *bad column.*
- 12 initial states which allows us to discard 48 initial states (including 2 that fulfill the pattern) due to the fact that the column j_0=4 of W is a *bad column.*
- 24 initial states in order to discard 48 initial states (including 3 states fulfilling the pattern) due to the fact that the column j_0=5 of W is a *bad column.*
- 42 initial states where the column j_0=6 of W is a *bad column.*
- 39 initial states where the column j_0=7 of W is a *bad column.*

Regarding solutions, in this example we find exactly 48 initial states producing sequences X with edit distance equal to 3. In fact, 10 initial states were directly detected in Step 4, while the remaining 38 solution states turned up as a result of the last step. For each one of the 48 initial states producing a sequence X with minimum edit distance, we have to recover the optimal paths from the graph in order to translate them into decimation and noise sequences.

5 Simulation Results

The next table shows some results for experimental implementations of the algorithm. At column denoted Pol., the positive exponents of the feedback polynomial of the $LFSR$ are represented. The columns denoted Seq.count.pat. display the number of sequences that fulfill the X-counterpattern (X-pattern). The column marked with Sol.pat. gives the number of initial states producing sequences X that are solutions. Thres. and Dist. are the columns where the obtained threshold and the minimum edit distance are shown. Finally, %Sav reflects a lower bound on the percentage of saving in the computing time and memory of the proposed algorithm compared with the original constrained edit distance attack.

From these randomly generated examples, we may deduce a general classification of inputs into several cases. The best ones correspond to patterns which directly identify solutions. Contrarily, bad cases are those in which the pattern is not fulfilled by any initial state. Such cases are generally associated with long runs at the beginning and at the end of the sequences Y. Finally, the medium cases are those for which, despite the non existence of solutions fulfilling the pattern, a good threshold is obtained. Such cases allow a good percentage of saving in computing thanks to the detection of many early *bad columns.*

N	M	L	Pol.	2^L	Seq.count.pat.	Sol.pat.	Thres.	Dist.	%Sav.
20	15	7	1,7	128	0	0	-	5	28.3
30	20	9	1,3,4,9	512	1	0	11	10	56.3
33	22	7	1,7	128	2	1	12	12	30.7
45	30	7	1,7	128	0	0	-	16	20.8
75	50	7	1,7	128	8	0	27	29	24
150	100	9	1,3,4,9	512	32	0	57	55	36.3
300	200	13	1,3,4,13	8192	128	0	115	114	25.9
450	300	17	3,17	131072	512	0	174	173	20.6
750	500	14	1,3,5,14	16384	1024	1	291	291	28.33

These empirical results show that in most cases the improvement in time complexity of the attack is greater than 25%. Furthermore, it is clear that the worst outcomes appear when the initial results in steps 1 to 4 are not adequate as there are not initial states fulfilling the pattern or the counterpattern. Since the hypothesis on Y are independent, the groups of bits in the X-pattern and in the X-counterpattern are also independent. Consequently, the conditions may be considered separately defining in this way a relaxed X-pattern and X-counterpattern which may lead to sequences that fulfill them. In addition, empirical results have shown that intercepted sequence Y with short runs at the beginning and at the end cause a greater improvement in the time complexity of the attack. Thus, another way to avoid a bad behavior of the original algorithm is by choosing subsequences from the intercepted sequence Y that have no too long runs at the beginning and at the end, and applying the algorithm to each one of these subsequences.

6 Conclusions

In this work a new algorithm based on two different and independent ways to improve a known constrained edit distance attack on clock-controlled *LFSR*-based generators has been proposed. The described algorithm avoids the exhaustive search over all the initial states of the involved *LFSRs*. The most remarkable aspect of this work is that the general ideas that have been proposed may be applied to attack any clock-controlled LFSR-based stream cipher.

References

1. Anderson, R.J.: A Faster Attack on Certain Ciphers, Electronics Letters, Vol. 29 No. 15, July (1993) 1322-1323.
2. Bluetooth, *Specifications of the Bluetooth system,* Version 1.1, February 2001, available at http://www.bluetooth.com/
3. Clark, A. *et al.*: The LILI-II Keystream Generator. Proc. ACISP 2002, Lecture Notes in Computer Science Vol.2384 Springer-Verlag (2002) 25-39.

4. Coppersmith, D., Krawczyk, H., Mansour, H.: The Shrinking Generator, Proc. Crypto'93, Lecture Notes in Computer Science Vol.773 Springer-Verlag (1994) 22-39.
5. CRYPTREC project- cryptographic evaluation for Japanese Electronic Government, www.ipa.go.jp/security/enc/CRYPTREC/index-e.html
6. Golic, J.D.: Recent Advances in Stream Cipher Cryptanalysis. Publication de l'Institut Mathematique Tome 64 (78) (1998) 183-204.
7. Golic, J.D., Menicocci, R.: Correlation Analysis of the Alternating Step Generator. Design Codes and Cryptography 31 (1) (2004) 51-74.
8. Golic, J.D., Mihaljevic, M.: A Generalized Correlation Attack on a Class of Stream Ciphers Based on the Levenshtein Distance, Journal of Cryptology, Vol. 3, No. 3 (1991) 201-212.
9. Golic, J.D., Petrovic, S.: A Generalized Correlation Attack with a Probabilistic Constrained Edit Distance, Proc. Eurocrypt'92, Lecture Notes in Computer Science Vol.658 Springer-Verlag (1993) 472-476.
10. Gollmann, D., Chambers, W.C.: Clock-Controlled Shift Registers: A Review, IEEE Transactions on Selected Areas in Communications SAC-7 May (1989) 525-533.
11. Golomb,S.W.: Shift Register-Sequences, Aegean Park Press, Laguna Hill, 1982.
12. GSM, *Global Systems for Mobile Communications,* available at http://cryptome.org/gsm-a512.htm
13. Jiang, S., Gong, G.: On Edit Distance Attack to Alternating Step Generator, Technical Report Corr2002-28, University of Waterloo (2002).
14. Johansson, T.: Reduced Complexity Correlation Attacks on Two Clock-Controlled Generators, Proc. Asiacrypt'98, Lecture Notes in Computer Science Vol.1514 Springer-Verlag (1998) 342-356.
15. Petrovic, S., Fúster, A.: Clock Control Sequence Reconstruction in the Ciphertext Only Attack Scenario, Proc. ICICS 2004, Lecture Notes in Computer Science Vol.3269 Springer-Verlag (2004) 427-439.

Protocol Analysis for Concrete Environments

Dieter Gollmann

TU Hamburg-Harburg,
Hamburg, Germany
diego@tu-harburg.de

Abstract. For protocol analysis, we have to capture the protocol specification, the security goals of the protocol, and the communications environment it is expected to run in. In the research literature, the emphasis is usually on verification techniques and on the modelling of security properties, while in most cases the default for the communications environment is an unstructured network totally controlled by the attacker. This paper will argue that for the analysis of the kind of protocols developed today, more specific models of the communications network are required. To support this argument, a number of recently proposed security protocols with novel features will be briefly discussed.

1 Introduction

The design of security protocols has a reputation of being 'difficult and error prone'. While this difficulty is sometimes exaggerated, there is certainly a case for a proper formal analysis of widely deployed security protocols. When analyzing a protocol we are given a description of the protocol, a set of security goals, and a model of the underlying communications system. We then check whether the protocol meets its desired goals.

The research community has discussed conventions for describing protocols, and there exist proposals for protocol description languages such as CAPSL[1]. There has been a change in conventions from defining protocols as sequences of messages exchanged, to defining protocols as collections of interacting processes. The latter approach has the advantage that all checks a protocol participant makes before moving to the next stage are made explicit.

Equally, the importance of distinguishing security properties in all their nuances is widely acknowledged and there has been much research on this topic. Current 'application level' definitions for the properties of cryptographic primitives and mechanisms can be found in [9]. A discussion of various authentication properties is given in [5]. There has also been much work in the field of cryptographic theory on teasing out subtle differences in the security properties cryptographic mechanisms might be asked to fulfil.

---

[1] For a CAPSL tutorial see `http://www.csl.sri.com/users/millen/capsl/`

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 365–372, 2005.
© Springer-Verlag Berlin Heidelberg 2005

2 The Dolev-Yao Model

In contrast, less effort has been spent on modelling the communications environment. Indeed, protocol analysis often tries to assume as little as possible about the communications system and gives all messages to the adversary for delivery. This approach is often presented as analysis in the Dolev-Yao model [4]. The model makes two independent assumptions:

- Cryptography is ‘perfect’. The adversary does not try to exploit any weakness in the underlying cryptographic algorithms but only algebraic properties of cryptographic operators and interactions between protocol messages.
- The adversary can observe and manipulate all messages exchanged in a protocol run and can itself start protocol runs.

The second assumption was already stated by Needham and Schroeder [10]:

> We assume that the intruder can interpose a computer in all communication paths, and thus can alter or copy parts of messages, replay messages, or emit false material. While this may seem an extreme view, it is the only safe one when designing authentication protocols.

Analysing protocols in a setting as general as possible is, however, not necessarily a route to higher security. Protocols may make use of features of the particular environment they were designed for so showing that a protocol does not meet is goal in a more general setting is useful side-information but should not be automatically classified as an attack.

3 Agility

We can analyze protocols that should meet well established security requirements and use established security primitives. Typical examples are the protection of message confidentiality through encryption, the protection of message integrity through message authentication codes or digital signatures, and the establishment of a security association between two peer nodes. Today, these mechanisms are found in networks at the IP layer (IPsec) at the transport layer (SSL) and now also at the web services layer. When dealing with established goals and mechanisms, security goals, assumptions about the environment, and standard cryptographic primitives can be integral parts of the methodology.

As an example, the BAN logic of authentication [3] assumes that attackers are outsiders and this is reflected even in its axioms, in particular in its message meaning rule for shared secrets. This rule says that if principal A receives a message containing a secret shared with principal B, A can conclude that the message came from B. However, a dishonest principal B might pass the secret to a third party and thus potentially deceive A about the source of messages. If assumptions like this are hard-coded into the verification methodology, changes about goals, primitives, and environment would require some redesign of the methodology.

There is a second direction in protocol analysis, viz the study of protocols that should meet novel requirements. In this case, we need *agile* methodologies where specific adversaries (rather than the general Dolev-Yao adversary) and new security requirements can be defined conveniently. Note that in most cases new protocols are designed because new requirements have emerged, so that traditional security assumptions have to be adjusted. For illustration, we briefly sketch four specific scenarios, together with observations on how established security assumptions may change.

4 Mobile IPv6 Binding Updates

In mobile IP, each host has a home address (HoA) at its home network and can always be reached via this address. Moreover, the mobile node has a secure tunnel to its home network. When a mobile node moves to a new location, it might tell its correspondent node that it has moved to a new care-of-address (CoA). The correspondent node could then update its binding cache that links the home address and care-of-address of the mobile node. If the correspondent node cannot check that the binding updates it receives are factually correct, an attacker could spread misinformation about the location of other nodes (can be prevented by authenticating the origins of update requests) or could lie about its own location (authentication is of no help) as part of denial-of-service attacks that flood the victim with data the attacker had requested for itself (bombing attacks).

The binding update protocol for mobile IPv6 [2, 7] works as follows (figure 1). The mobile node starts by sending a *Home Test Init* message (HoTI) via the home network and a *Care-of Test Init* message (CoTI) directly to the correspondent. The correspondent replies to both requests independently. A *Home Test* (HoT) message containing a 64-bit home keygen token K_0 and a home nonce index i is sent to the mobile node via the mobile's home address. A *Care-of Test* (CoT) message containing a 64-bit care-of keygen token K_1 and a care-of nonce index j is sent directly to the claimed current location[2]. The mobile node uses both keygen tokens to compute a binding key

$$K_{bm} := \text{SHA1}(\text{home keygen token}||\text{care-of keygen token}),$$

and the *Binding Update* (BU) authenticated by a 96-bit MAC

$$\text{HoA},\ i,\ j,\ \text{HMAC\_SHA1}(K_{bm},\ \text{CoA}||\text{CN}||\text{BU})\_96.$$

This protocol does not rely on the secrecy of cryptographic keys but on *return routability*. The correspondent checks that it receives a confirmation from

---

[2] Nonces are used to make the protocol stateless for the correspondent. The keygen tokens are derived from a long term node key and nonces. The mobile node returns the indices in its final message allowing the correspondent node to look up the nonces and recalculate the keys.

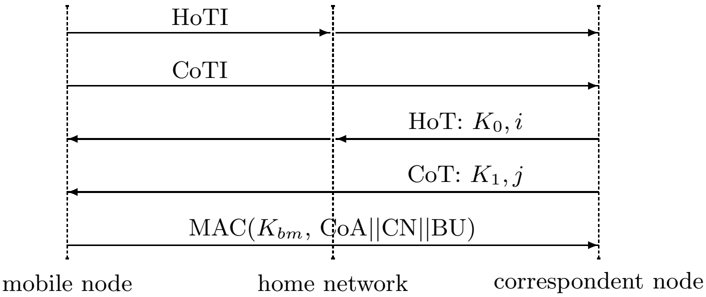


Fig. 1. Mobile IUPv6 binding update protocol

the advertised location. The threat model assumes that messages over the fixed Internet are considered secure or can be protected otherwise. Hence, keys K_0 and K_1 may be sent in the clear. These keys could also be interpreted as challenges (nonces) that bind identity to location through the binding key K_{bm}. In communications security the term authentication typically refers to the corroboration of a link between an identity of some kind and an aspect of the communications model, like a message or a session [5]. In this interpretation, binding update protocols provide *location authentication.*

5 Middleboxes

Protocols for the 'real' Internet have to consider so-called middleboxes like Network Address Translators (NATs) and firewalls. Protocols like HIP (Host Identity Protocol) provide mobile nodes with identifiers above the IP layer that do not change when nodes move and the IP address changes, and maintain a mapping between the identifier and the IP address. However, when the protocol has to traverse middleboxes several problems can arise. For example, a node may be behind a NAT so its true address is not visible to its peer so the middlebox may have to act as a proxy and provide the mapping between identifier and actual address. There may also be problems with firewalls that permit traffic only in one direction as the protocols updating the address often include messages in both directions.

As a further problem, a node may tell its firewall to let packets from its correspondent pass, but when the correspondent changes its location the firewall rules have to be updated. Then, schemes for protecting the instructions to the firewall have to be implemented. Issues of this kind are discussed, e.g. in [11]. These examples should illustrate why a simple 'Alice & Bob' model of communications is no longer appropriate when designing, and analysing security protocols at the IP layer.

6 Multi-layered Protocols

Multi-layered authentication protocols try to derive security properties at a higher protocol layer from guarantees given at a lower layer. For example, the variants of EAP [1] use identifiers at different layers. A principal thus can be known by distinct identities at each layer. Hence, security analysis also has to check the binding that is intended (or not intended, when privacy is a goal) between the different identifiers of a single principal. An example for the pitfalls one has to be aware of when dealing with this issue is reported in [8]. Again, we note that the Alice & Bob view of the world is too simplistic.

7 Sensor Networks

The Canvas protocol [12] provides data integrity in a sensor network by relying on independent witnesses but does not provide data origin authentication at the same time. We will give a slightly abbreviated version of the discussion in [6]. Let us assume that nodes can communicate with their direct neighbours and have information about nodes in their vicinity, but no means for authenticating arbitrary nodes in the network. I.e., there is nothing like a public key infrastructure. Nodes share secret keys with nodes that are one or two hops away. Message authentication codes protect the integrity of messages transmitted between nodes that have shared keys.

Nodes can inject new messages into the network and forward messages they receive. We assume an algorithm exists for routing message sin the network. We want to achieve *data integrity*. Forwarded messages cannot be manipulated or inserted[3]:

Definition 1. *Data integrity is the property whereby data has not been altered in an unauthorized manner since the time it was created, transmitted, or stored by an authorized source [9].*

Defence against the creation of messages with bad content is a separate issue that is not being addressed here.

In this network, the creator of a message cannot vouch for its integrity as nodes further away would not share a key with the originator. The Canvas protocol thus uses interwoven authentication paths for data integrity. Forwarding of messages works as follows. Let K_{xy} denote a symmetric key shared by nodes X and Y and let A, B, C, D, denote nodes in the network. A message m is forwarded from B to C as follows (figure 2):

$$B \to C : m, A, B, D, h(K_{ac}, m), h(K_{bc}, m), h(K_{bd}, m)$$

Node A had forwarded the message to B, nominated C as the next node, and included the authenticator $h(K_{ac}, m)$. In turn, B nominates D as the next node and constructs authenticators $h(K_{bc}, m)$ and $h(K_{bd}, m)$. The recipient C checks the two authenticators $h(K_{ac}, m)$ and $h(K_{bc}, m)$, and discards m if authentication fails.

---

[3] Source [12] refers to message authentication, but in [9] this term appears as a synonym for data origin authentication.

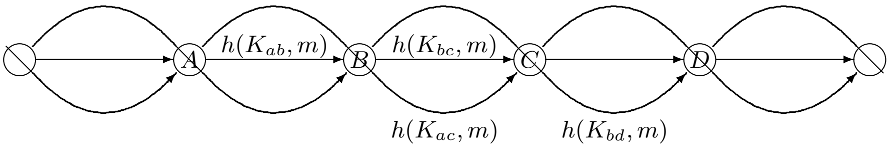


Fig. 2. The Canvas protocol

Obviously, if A and B collude they can modify m without being detected by C. However, it can be shown that the protocol achieves its goal if no two adversarial nodes are direct neighbours [12]. This observation contradicts a view widely held in communications security that *data integrity* and *data origin authentication* are equivalent properties, see e.g. [9, page 359].

Definition 2. *Data origin authentication (message authentication) is a type of authentication whereby a party is corroborated as the source of specified data created at some time in the past [9].*

By definition, data origin authentication includes data integrity. Conversely, in a communications system where the sender's identity (address) is an integral part of a message, a message with a forged sender address must not be accepted as genuine. To check the integrity of a message we would also have to verify its origin. Moreover, if messages pass through a network that is controlled by the adversary, we can only rely on evidence provided by the sender to verify that a message has not been altered in transit. For both reasons data integrity includes data origin authentication, but only under the specific assumptions we have made about the communications system.

As a final twist to this discussion, we show that there exists an attack if we adjust assumptions about the adversary. Adversarial nodes still cannot be direct neighbours but they may agree a-priori on a strategy for modifying messages and know their respective routing strategies. Two adversarial nodes A and C separated by a honest node B can collude to change a forwarded message m to $\tilde{m}$. The attack in figure 3 targets a node that can be reached in one hop from one of the adversarial nodes and in two hops from the other.

1. Adversary A forwards message m to B, naming C as the next node and including $h(K_{ae}, \tilde{m})$ in place of the authenticator $h(K_{ac}, m)$; E has to be a node that can be reached in one hop from C and in two hops from A.
2. Node B successfully checks the authenticators for m, names D as the next node, and forwards $h(K_{ae}, \tilde{m})$ unchecked.
3. Adversary C receives m from B, changes it according to the pre-arranged strategy to $\tilde{m}$, generates authenticators for the modified message, and forwards those together with $h(K_{ae}, \tilde{m})$ to E.
4. Node E receives the modified message $\tilde{m}$ with valid authenticators from A and C and accepts it as genuine.

This attack could be prevented if E knows about valid routes in the network. By assumption, A and C are not direct neighbours so messages could not arrive

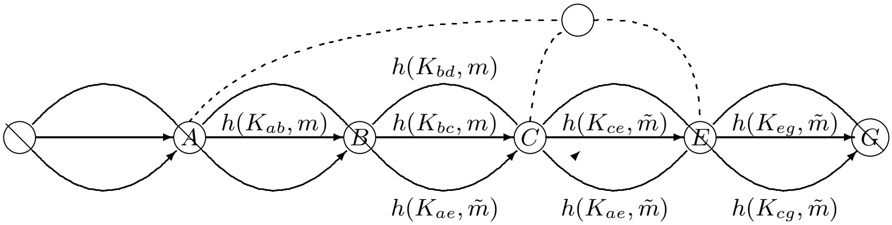


Fig. 3. An 'attack' on the Canvas protocol; dotted lines indicate unused links

along the route $A \to C \to E$. However, this would constitute yet another change in assumptions. So far, nodes were only storing keys for some neighbours but had no further information about the network topology.

8 Conclusion

We have given examples that have introduced location and return routability as new aspects that have to be captured in protocol analysis. We have pointed to issues that arise when parties have to communicate via middleboxes so that traffic identifiers change along a route. We have mentioned layered protocols and the problems of matching identifiers at different layers of the protocol stack. In the final example, security was relying on the fact that adversarial nodes are sufficiently isolated so that they cannot violate message integrity.

For the analysis of such protocols, we need methodologies that allow us to capture relevant aspects of the communications environment. We have to be able to specify which nodes are honest, and which communication links are not controlled by the adversary. We might have to accommodate new security properties, and maybe even new axioms for location-based arguments like return routability. For the analysis of such protocols, we may also have to change the way we think about security. Axioms in the logical derivation systems used may only hold under certain assumptions, and even the familiar security terminology may implicitly reflect assumptions about the communications system. The major challenge here is to check evaluation methodologies for traces of the Dolve-Yao model so that we understand which aspects of the methodology depend on its assumptions.

References

1. B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz. *Extensible Authentication Protocol (EAP)*, June 2004. RFC 3775.
2. Tuomas Aura, Michael Roe, and Jari Arkko. Security of Internet location management. In *Proceedings of the 18th Annual Computer Security Applications Conference*, pages 78–87, December 2002.
3. Michael Burrows, Martín Abadi, and Roger Needham. A logic of authentication. *DEC Systems Research Center*, Report 39, revised February 22 1990.

4. Danny Dolev and Andrew C. Yao. On the security of public key protocols. *IEEE Transactions on Information Theory*, IT-29(2):198–208, March 1983.
5. Dieter Gollmann. Authentication by correspondence. *IEEE Journal on Selected Areas in Communications*, 21(1):88–95, January 2003.
6. Dieter Gollmann. Challenges in protocol design and analysis. In J.D. Tygar D.T. Lee S.P. Shieh, editor, *Computer Security in the 21st Century*, pages 7–22. Springer, 2005.
7. D. Johnson, C. Perkins, and J. Arkko. *Mobility Support in IPv6*, June 2004. RFC 3775.
8. Catherine Meadows and Dusko Pavlovic. Deriving, attacking and defending the gdoi protocol. In *Proceedings ESORICS 2004, LNCS 3193*, pages 53–72. Springer Verlag, 2004.
9. Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography.* CRC Press, Boca Raton, FA, 1997.
10. Roger M. Needham and Michael D. Schroeder. Using encryption for authentication in large networks of computers. *Communications of the ACM*, 21:993–999, 1978.
11. Hannes Tschofenig, Andrei Gurtov, Aarthi Nagarajan, Murugaraj Shanmugam, and Jukka Ylitalo. Traversing middleboxes with the host identity protocol. In *Proceedings of ACISP 2005*, July 2005.
12. Harald Vogt. Integrity preservation for communication in sensor networks. Technical Report 434, ETH Zürich, February 2004.

Pattern Recognition in AVHRR Images by Means of Hibryd and Neuro-fuzzy Systems

Jose Antonio Piedra, Francisco Guindos, Alberto Molina, and Manuel Canton

[1] Universidad de Almeria, Departmento de Languajes y Computación,
Phone +34 950 0140 36, 04120 Almeria, Spain
`{jpiedra, fguindos, mcanton}@ual.es`
`alberto@altoalmanzora.org`

Abstract. The main goal of this work is to improve the automatic interpretation of ocean satellite images. We present a comparative study of different classifiers: Graphic Expert System (GES), ANN-based Symbolic Processing Element (SPE), Hybrid System (ANN – Radial Base Function & Fuzzy System), Neuro-Fuzzy System and Bayesian Network.. We wish to show the utility of hybrid and neuro-fuzzy system in recongnition of oceanic structures. On the other hand, other objective is the feature selection, which is considered a fundamental step for pattern recognition. This paper reports a study of learning Bayesian Network for feature selection [1] in the recognition of oceanic structures in satellite images.

1 Structure of the Automatic Interpretation System

Fig. 1 depicts the overall structure of the system that has been developed. In a first step, the raw image is processed by means of algorithms such as radiometric correction, map projection and land masking. These are well known techniques also used when the analysis is made by human experts. However, we don't make any image enhancement like histogram equalization or contrast stretching that are appropriate to make some features visible to human eye but have no positive effects when the images are going to be processed by digital systems.

The second step aims to detect clouds pixels that are opaque to radiance data measured in the AVHRR infrared and visible scenes. Cloudy images are distorted in such a way that the zone affected isn't of any value for our later processing, so we build a mask of 0s that will exclude these pixels [2].

The following task is the segmentation that will divide the whole image in regions. The idea is that each phenomenon of interest should coincide with one or a small set of the segmented regions. The nature of ocean dynamics makes very difficult this process that is nevertheless fundamental, so we've designed an iterative knowledge-driven method to perform this part of the process pipeline [3].

The next task is the features or descriptors selection, which consists of selecting an optimal or sub-optimal feature subset from a set of candidate features. The most common framework for features selection is to define some criteria for measuring the goodness of a set of features, and then use a search algorithm to find an optimal or sub-optimal set of features [4]. We have used Bayesian networks for features selection in the recognition of oceanic structures in satellite images [5].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 373 – 378, 2005.
© Springer-Verlag Berlin Heidelberg 2005

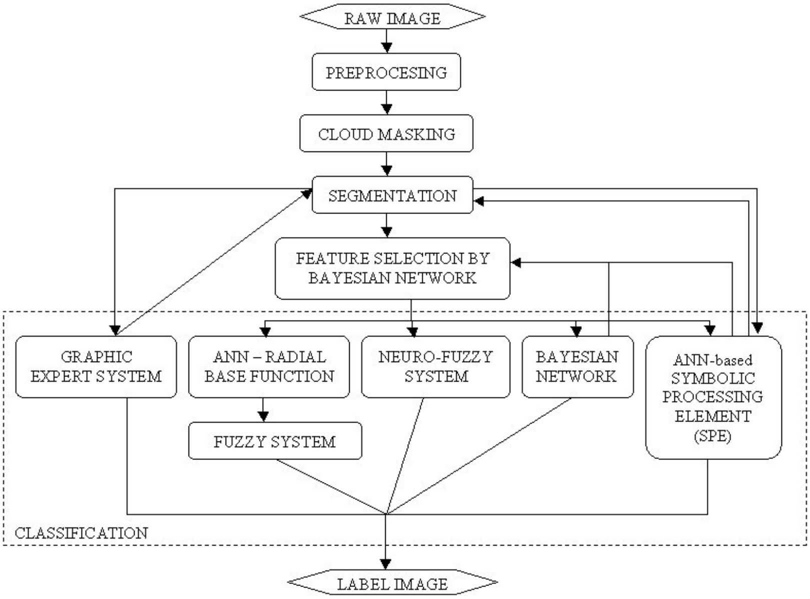


Fig. 1. Structure of the ocean feature recognition system

In the last step, each region produced in the segmentation is analyzed and, if the recognition is positive, it is labeled with the identifier of the matching structure. The structures of interest in the Canary Islands zone as defined in are: coastal upwelling, warm eddies, cold eddies and island wakes. The vast majority of the regions that appear in the segmentation are of no special interest and they are labeled with a 0.

We have implemented a redundant recognition subsystem. It has an ANN-based Symbolic Processing Element (SPE) module [6], a rule-based Graphic E.S. (GES) [3], Bayesian Network, Hybrid System (Artificial Neural Network based Radial Base Function and Fuzzy System based Sugeno) and Neuro-Fuzzy Systems (NEFPROX, ANFIS) performing the same task. The purpose is to test different methodologies and to provide a way to validate and compare these results.

2 Detailed Process

2.1 Feature Selection by Bayesian Network

The most common framework for feature selection is to define criteria for measuring the goodness of a set of features, and then use a search algorithm that finds an optimal or sub-optimal set of features. Our goal in this step is to apply the theory of learning Bayesian Network to the reduction of irrelevant features [1] in the recognition of oceanic structures in satellite images [6].

The experiment was done over the feature set obtained in the work [6]. The learning algorithms (K2, VNSST, Naive-Bayes) have been evaluated with different configurations of parameters to select the best configuration. Table 1 shows the summary of comparative results generated from SPE (Neural Network) and Learning Bayesian Method. This method allows choosing a feature subset obtaining the same accuracy rate (about 80%) that with the initial feature set.

Table 1. Comparative results of feature selection

Algorithms	Number of Features
Symbolic Processing Element	50
K2 Learning	15
VNSST Learning	15
Naive-Bayes Learning	50

2.2 Classification

All classifiers has been designed and tested with the same image data set.

2.2.1 Hybrid System (Artificial Neural Network and Fuzzy System)

Fuzzy modeling is one of the techniques currently being used for modeling nonlinear, uncertain, and complex systems. An important characteristic of fuzzy models is the partitioning of the space of system variables into fuzzy regions using fuzzy sets [7]. One of the aspects that distinguish fuzzy modeling from other black-box approaches like neural nets is that fuzzy models are transparent to interpretation and analysis (to a certain degree).

Radial basis function networks and fuzzy rule systems are functionally equivalent under some conditions. Therefore, the learning algorithms developed in the field of artificial neural networks can be used to adapt the parameters of fuzzy systems. We use the relationship between RBF neural network and fuzzy system for classification purpose [8].

The operation begins training a neuronal network under initial conditions (equivalence with fuzzy system). Neural network obtains a set of rules, that are used to build the fuzzy system. Both systems are used in classification processes.

2.2.2 Neuro-fuzzy System

Neuro-Fuzzy System refers to the combination of fuzzy set theory and neural networks with the advantages of both:

1. Manage imprecise, partial, vague or imperfect information
2. Handle any kind of information (numeric, linguistic, logical, etc.)
3. Self-learning, self-organizing and self-tuning capabilities
4. No need of prior knowledge of relationships of data
5. Reduce human decision making process
6. Fast computation using fuzzy number operations

We use ANFIS [9] (Adaptative Network based Fuzzy Inference System), which is a fuzzy inference system implemented in the framework of adaptative network. ANFIS can serve as a base for constructing a set of fuzzy if-then rules with appropiate membership functions to generate the stipulated input-output pairs.

On the other hand, we use a general approach to function approximation by a neuro-fuzzy model based on supervised learning. NEFPROX [10] (NEuroFuzzyfunction apPROXimator) has a similar structure as the NEFCON model, but it is an extension, because it does not need reinforcement learning. On the other hand it also extends the NEFCLASS model [11], that can only be used for crisp classification tasks.

2.2.3 Bayesian Networks

Bayesian Networks provide an intuitive graphical visualization of the knowledge including the interactions among the various sources of uncertainty. Bayesian Networks are models for representing uncertainty in our knowledge. We use them in feature selection and classification.

3 Results

The information provided by knowledge driven classifiers are refurbished with the original segmentation to create images (Fig. 2) that show in a visual way each the labeled ocean feature of interest. Each feature is represented by different colored region in a map where each color represents one of the ocean phenomena we are looking for.

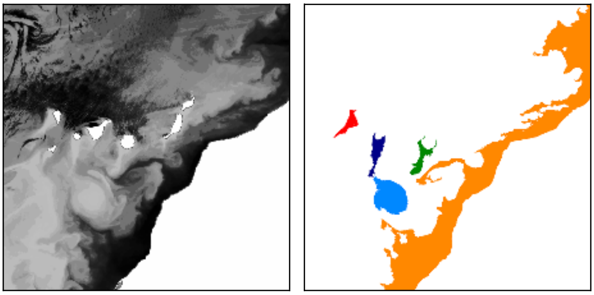


Fig. 2. AVHRR scene (equalized) and feature map (orange:upwelling, red-green-blue: warm wakes,light blue: warm gyre)

Results from systems SPEs,GES, HS, BN and NFS, depend mainly on the quality of the images. In general, a good number of training images are needed. For GES, the

Table 2. Comparative results of classification

Classifier	Accuracy Rate
Hybrid System	60 %
NEFPROX	60 %
ANFIS	60 %
Bayesian Networks (K2,VNSST Learning)	80 %
S.P.E.	80 %
G.E.S.	95 %

best results are achieved when the human expert provides the system with specific knowledge about the target area. When these requisites are met, the systems produce positive ocean structures recognition, which is shown in table 2.

4 Conclusions

We have explored the use of Bayesian networks as a mechanism for feature selection in a system for automatic recognition of ocean satellite images. The use of Bayesian networks has provided benefits with respect to SPE, not only in the reduction of relevant features, but also in discovering the structure of the knowledge, in terms of the conditional independence relations among the variables. In future works we plan to improve the accuracy rate of the system including more variables. Furthermore, we expect to use models to avoid the discretisation of the continuous features when learning Bayesian networks.

On the other hand, obtained results aren´t expected for classification process by hybrid and neuro-fuzzy system. They don´t correspond with results of other cassifiers (EPS, GES and Bayesian Network). The main problem is the number of generated rules, which is excessive. In future works we plan to improve the accuracy rate of the hybrid and neuro-fuzzy systems including more variables, tunning training parameters, optimizing the obtained structure and automating the construction of neurofuzzy system.

Finally, the structure of the automatic interpretation system that has been introduced, it is a complex and independent structure for the following tasks:

1. Iterative segmentation-recognition cycle is made by means of graphic expert system.
2. Feature selection and validation of knowledge is done by bayesian learning
3. Multiple classification allows to find different interpretations of the existing knowledge to validate the knowledge of the classifiers.

Acknowledgments

This work is being partially supported by the Spanish Commission for Science and Technology (CICYT) through the project 2004-TIN-05346.

References

1. Inza I., Larrañaga P., Etxeberria R. and Sierra B.: Feature Subset Selection by Bayesian networks based optimization. Artificial Intelligence, nº 123, (2000) 157–184.
2. Torres J.A., Guindos F., Peralta M. and Cantón M.: An Automatic Cloud-Masking System Using Backpro Neural Nets for AVHRR Scenes, IEEE Transactions On Geoscience and Remote Sensing, vol. 41, nº 4, (2003) 826–831.
3. Guindos F., Piedra J.A. and Cantón M.: Ocean features recognition in AVHRR images by means of bayesian net and expert system, 3rd International Workshop on Pattern Recognition in Remote Sensing, Kingston University, United Kingdom. August (2004).

4. Langley P. and Sage S.: Induction of selective Bayesian classifiers. In Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence, Seattle. Morgan-Kaufmann, (1994) 399–406.
5. Yamagata Y. and Oguma H.: Bayesian feature selection for classifying multi-temporal SAR and TM Data. IEEE, (1997) 978–980.
6. Torres J.A., Guindos F., López M. and Canton M.: Competitive neural-net-based system for the automatic detection of oceanic mesoscalar structures on AVHRR scenes. IEEE Trans. on Geoscience and Remote Sensing, vol.41, issue 4, April (2003) 845–852.
7. Zadeh L.A.: Fuzzy Logic. IEEE Computer, vol. 21 , issue 4 , April (1988) 83–93.
8. Yaochu Jin, Werner von Seelen, Bernhard Sendhoff: Extracting Interpretable Fuzzy Rules from RBF Neural Networks. Institut für Neuroinformatik, Ruhr-Universität Bochum, FRG. Internal Report (2000–2002).
9. Roger Jang: ANFIS Adaptative Network based Fuzzy Inference System. IEEE Transactions On Systems, Man and Cybernetics, vol 23, nº 3. May/June (1993).
10. Nauck D. and Kruse R.: A neuro-fuzzy approach to obtain interpretable fuzzy systems for function approximation. Fuzzy Systems Proceedings, IEEE World Congress on Computational Intelligence., vol 2 , May (1998) 1106 –1111.
11. Nauck D.: Knowledge discovery with NEFCLASS. Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies, vol. 1 , 30 August-1 September (2000) 158–161.

Image Processing Techniques for Braille Writing Recognition

Néstor Falcón, Carlos M. Travieso, Jesús B. Alonso, and Miguel A. Ferrer

Dpto. de Señales y Comunicaciones, Universidad de Las Palmas de Gran Canaria
Campus de Tafira s/n, E-35017, Las Palmas de Gran Canaria, Spain
`nesmofalcon@gmail.com, ctravieso@dsc.ulpgc.es`

Abstract. In this paper we present the development of *BrailLector,* a system able to speak from Braille writing. By means of dynamic thresholding, adaptive Braille grid, recovery dots techniques and TTS software (Text-To-Speech), *BrailLector* translates Braille scanned images into normal text, and not only that, it speaks the translated text. *BrailLector* is a robust application with innovative thersholding and Braille grid creation algorithms which detects and read Braille characters with 99.9% of correct symbols and an error variance below 0.012. The conversion time is only 26 secs for double-sided documents by MATLAB programming language.

1 Introduction

Nowadays, lack or problem of vision has been an important obstacle to access to printed contents and to the information society. For this reason, some people have tried to achieve that blind people are able to access to the printed culture, for example Valentin Haüy and Luis Braille who have understood the importance of a communication code. Globally, an estimated 40 to 45 million people are blind and 135 million have low vision according to the World Health Organization (WHO) [1] and this number grows every year.

A Braille Optical Character Recognizer is interesting due to the following reasons;

- It is an excellent communication tool for sighted people (who do not know Braille) with the blind writing.
- It is a cheap alternative Braille to Braille copy machine instead of the current complex devices which use a combination of heat and vacuum to form Braille impressions.
- Braille writing is read using the finger so is necessary touch the document, for this reason the book after many readings is possible has been deteriorated.
- It is interesting to store a lot of document of blind authors which were written in Braille and were never converted to digital information.
- *Braillector* offers a better integration of blind people to the "information society".

Since the most part of Braille books are written with two kinds of points, we will have to distinguish between each one. One kind is like "mountains" and the other is as little "valleys", hence the finger of the blind person who reads the text only detects

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 379 – 385, 2005.
© Springer-Verlag Berlin Heidelberg 2005

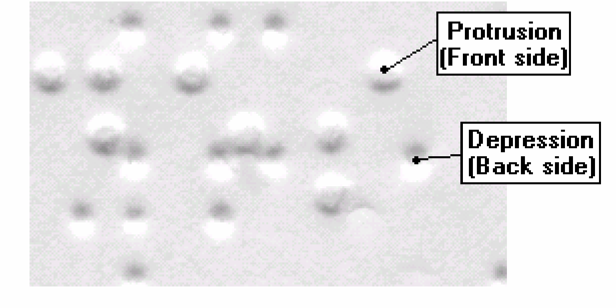


Fig. 1. Protrusions and Depressions on a Braille sheet

the protrusions, while depressions are points written to be read from the other side of the sheet. If we are able to distinguish between each point we will have a big advantage, we will be able to recognize the two sides with only one scan.

The structure of this paper is the next: In the next text we will describe the characteristics of the database created. After, image processing techniques used for dots detection and recovering will be described. Then, we will explain the conversion from Braille text to standard text. Conclusions close this paper.

2 Database

A big database has been created in order to check the global system with as many characters as we could. This database provides single and double-sided documents, which have dots in one or both sides of the sheet respectively. The number of characters in this database ensures the correct testing of the developed system and a good analysis of the error variance. The next table gives a full explanation of this database.

Table 1. Database created

Braille sheets	26
Total number of characters	30862
Mean number of character per sheet	1235
Digital format	Gray scale
Resolution	100 dpi (horizontal and vertical)
Image size	1360 Kbytes
Image format	Bitmap (‘bmp’)
Braille type	Double sided – grade 1
Document size	29.5 cm. (horizontal) x 30.5 cm. (vertical)

The mechanism used for image acquisition has been a flat-bed scanner instead of a digital camera because it is a cheap alternative which can be used for so many other applications and it is easy and quick to use. The system is able to work with images of different resolution than 100 dpi since it uses interpolation methods to resize the input image.

3 Image Processing Techniques

The next image represents the global blocks of image processing techniques;

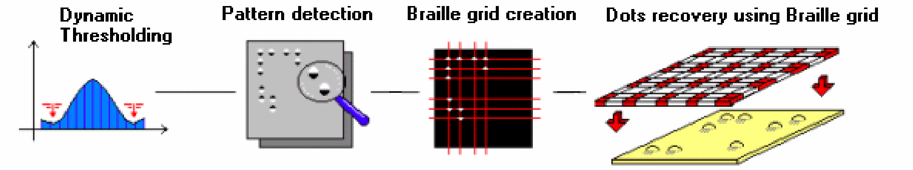


Fig. 2. Braille Image Processing

The different steps of the scanned image for its translation are;

1. An innovative thresholding method has been used to extract the useful information of Braille images instead of traditional methods [2]. Once we know the optimum area of Braille spots (it reduces the number of wrong symbols detected), an iterative algorithm looks for the best threshold according to these areas of Braille dots. This area criterion for thresholds selection offers an accuracy way to get the optimum levels to separate black, white and grey.

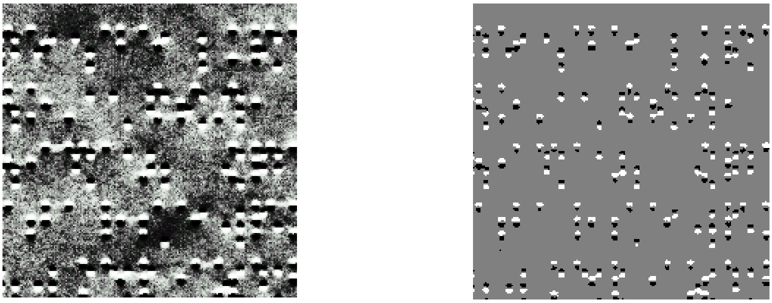


Fig. 3. Automatic threshold selection process

After this first step, no useful information has been rejected and the image is now ready to be processed in the pattern detection block.

2. Once this primary process is done, we take advantage of the shadows which make dot patterns. As we have seen, these protrusions on a Braille sheet have a brilliant zone above and a dark zone below. The depressions have exactly the opposite pattern (these shadows are created by the skew angle of the light beam

in reflection scanners) so this difference will facilitate the separation between front and back side dots;

- Moving white "islands" 4 pixels downwards and doing a logical "and" with black spots we will extract front side dots.
- Moving white "islands" 4 pixels upwards and doing a logical "and" with black spots we will extract back side dots.

The goal of this secondary process is to separate each side of the document in different images. This algorithm consists of a "shift and overlap" process since it only moves the spots downwards or upwards and carries out a logical and. It is the fastest method we have tried because it avoids the sequential reading of the image matrix and it is very simple and efficient. In this stage, skew angle of the scanned document is detected by means of horizontal histogram and mass centers calculation and corrected rotating the original image.

3. Not all the dots are detected with overlapping process, some of them are missed, either because they are very small or their shadows do not overlap with only 4 pixels. For this reason, a new stage was added to the system: The Braille grid.

   An adaptive algorithm has been developed in order to make this mesh from the detected dots. The algorithm builds columns in a first stage: since distances between points are normalized [3], the process begins searching for groups of dots in the same vertical plane that respect these distances. Then it builds the columns according to the pattern of Braille columns adapting itself to the layout of the document. Thus, we get a flexible mesh that tolerates small differences between columns. The process for the rows is quite similar but in this case the pattern to search is a Braille row. Detected columns and rows will be arranged together to create the final structure. This grid is flexible and respects the layout of the original document which makes it suitable for copying Braille sheets without losing the format.

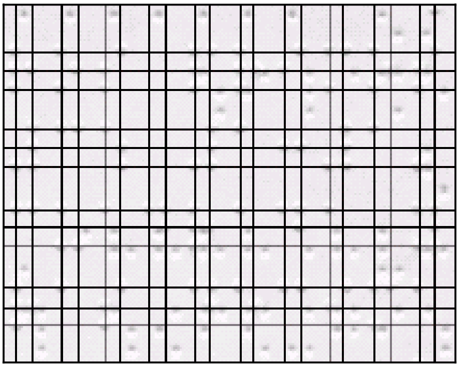


Fig. 4. Braille grid layout for frontal side

4. After mesh building, all valid Braille positions are known. Those intersections between rows and columns will define a valid position for a Braille dot. First, dilation techniques [4] are used to expand the search zone. Then, we fit this

image on the dots image after thresholding in order to check in detail those positions where dots were not found. In this point lays the intelligence of the global system, only potential positions of dots are checked; this means time saving and efficient search. Original dots will be recovered (they belong to correct Braille positions) and false dots will be discriminated as they are out of the valid places for a Braille dot. In the following figure, we show this effect on the recuperation of Braille dots.

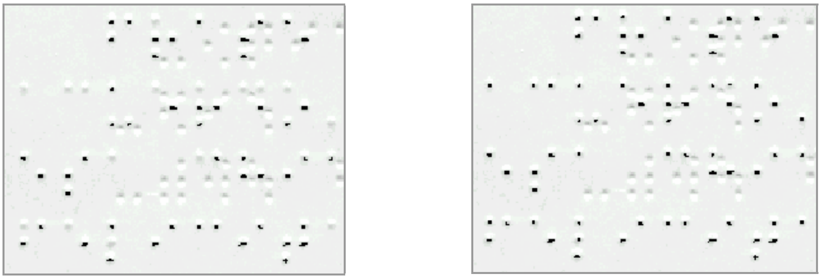


Fig. 5. Front side detected dots before and after the use of the recovered algorithm

4 From Braille to Normal Text

At this point all valid Braille dots have been detected in both sides of the document. This final image contains the Braille dots represented by spots, so now it is analyzed and text is segmented in rows and characters. For this segmentation process we will take advantage of the Braille mesh one more time since it marks all the positions of Braille dots. Every character will be converted into a binary number according to the active dots. The process consists of reading character by character and each one of the six positions that make the basic cell [5]. Hence we will have six possible values for each character (either raised or flat). This way of coding is simple and fast. It can be explained better by means of the next figure.

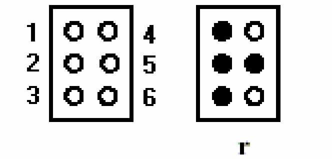


Fig. 6. Standard Braille Character (left) and ‘r’ symbol

In this way and looking at the previous example, ‘r’ symbol can be coded like ‘111010’ where each black dot is an active dot or a “mountain” on the paper and they become ‘1’ in the binary number. This way of Braille text binarization makes the global system independent of the language of the document and easily configurable for adding different alphabets. The output of this step will be a file with each character coded like a binary number; we will only have to translate each number for its equivalent letter in normal text to get the final output like a text file.

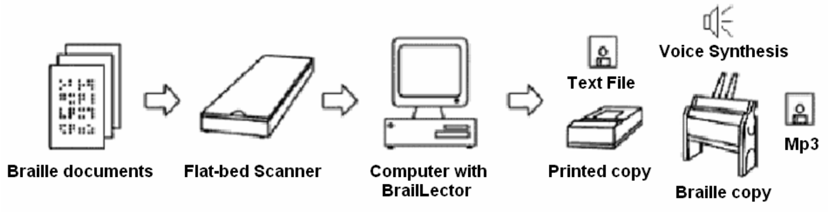


Fig. 7. Global translation process of Braille documents

This final output can be presented in different formats such as a text file, a new Braille printed copy, voice (by means of TTS software [6]) or even mp3 audio format.

5 Conclusions

In this paper we have explained the development of an automatic system for translating Braille text to normal text or voice. The global algorithm is very fast and robust. It has been divided in different modules for each part of the image processing. For achieving this system, dynamic thresholding and adaptive Braille grid has been used, adding some intelligence to the global process and making it able to detect dots in both sides of the document with only one scan. This process has an efficiency of 99.9% and it takes only 26 secs to translate a double-sided document improving all the main references found in the bibliography [7], [8], [9].

Acknowledgments

As with most works, many have provided help in the development of this research. Many thanks to Dr. Fatih Kurugollu, Lecturer of School of Computer Science at Queen´s University, Belfast, United Kingdom where a big part of this work was done. He provided invaluable help and clear ideas to solve many problems about image processing. He has my admiration for his patience and my deep appreciation for his assistance.

This work was supported by the research Project "Pi 2003/005" from Canary Autonomous Government (Spain).

References

1. http://www.who.int Active on April 1st 2005
2. N. Otsu. "A threshold selection method from gray-level histograms" In IEEE Transactions on Systems, Man, and Cybernetics, vol 9, no.1, pp 62-66. Jannuary 1979
3. Dubus J., Benjelloun M., Devlaminck V., Wauquier F., Altmayer P.: Image processing techniques to perform an autonomous system to translate relief Braille into black-ink, called: Lectobraille. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, (1988) 1584–1585.

4. Rafael C. González, Richard E. Woods "Digital Image Processing" Prentice Hall (2002)
5. http://www.fbraille.com.uy/alfabeto Active on April 1st 2005
6. http://www.textreader.net Active on April 1st 2005
7. Wong L., Abdulla W., Hussman S.: A Software Algorithm Prototype for Optical Recognition of Embossed Braille. The University of Auckland, New Zealand, (Technical Report) (2004).
8. Mennens J.,. Tichelen L. V, Francois G., Engelen J.: Optical recognition of braille writing using standard equipment. IEEE Transactions on Rehabilitation Engineering, 2(4) (1994) 207– 212.
9. Ng C., Ng V., Lau Y.: Regular feature extraction for recognition of Braille. In Proceedings of Third International Conference on Computational Intelligence and Multimedia Applications, (1999) 302-306.

Retinal Based Authentication via Distributed Web Application

C. Mariño, M.G. Penedo, and M. Penas

Grupo de Visión, Artificial y Reconocimiento de Patrones, (VARPA),
Universidade de A Coruña, Campus de Elviña s/n
castormp@fi.udc.es, {cipenedo, infmpc00}@dc.fi.udc.es

Abstract. Traditional authentication systems, employed to gain access to a private area in a building or to data stored in a computer, are based on something the user *has* (an authentication card, a magnetic key) or something the user *knows* (a password, an identification code). But emerging technologies allow for more reliable and comfortable for the user, authentication methods, most of them based on biometric parameters. Much work could be found in literature about biometric based authentication, using parameters like iris, voice, fingerprints, face characteristics, and others. We have developed a new methodology for personal authentication, where the biometric parameter employed for the authentication is the retinal vessel tree, acquired through a retinal angiography. It has already been asserted by expert clinicians that the configuration of the retinal vessels is unique for each individual and that it does not vary in his life, so it is a very well suited identification characteristic. In this work we will present the design and implementation stages of an application which allows for a reliable personal authentication in high security environments based on the retinal authentication method.

1 Introduction

Reliable authentication of people has long been an interesting goal, becoming more important as the need of security grows, so that access to a reliable personal identification infrastructure is an essential tool in many situations (airport security controls, all kinds of password-based access controls, ...). Conventional methods of identification based on possession of ID cards or exclusive knowledge are not altogether reliable. ID cards can be lost, forged or misplaced; passwords can be forgotten or compromised. A solution to that problems has been found in the biometric based authentication technologies. A biometric system is a pattern recognition system that establishes the authenticity of a user's specific physiological or behavioral characteristic. Identification can be in the form of verification, authenticating a claimed identity, or recognition, determining the identity of a person from a database of known persons (determining who a person is without knowledge of his/her name).

Many authentication technologies can be found in the literature, with some of them already implemented in commercial authentication packages [1,2,3]. But today most of the efforts in authentication systems tend to develop more secure environments, where it is harder, or ideally, impossible, to create a copy of the properties used by the system to discriminate between authorized individuals and unauthorized ones, so that an impostor

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 386–391, 2005.
© Springer-Verlag Berlin Heidelberg 2005

could be accepted by the biometric system as a true sample. In that sense, the method employed by our system [4] uses as the authentication biometric parameter the blood vessel pattern in the retina of the eye.

In this paper, a distributed client-server framework which allows for personal authentication is outlined. From the design stage, where design patterns are the fundamental tool, to the implementation (server, communication channels and clients) are described. The main goal of this work is to build a robust system which allow for the automatic personal authentication, which would allow or deny the access to critical resources of the system.

2 Authentication Methodology

In many cases it is almost impossible to acquire the biometric parameter in the same conditions than the stored template used for the authentication, so that a first step of normalization of both parameters (the acquired and the reference one) is needed in order to make the system reliable enough, avoiding the rejection of legitimate users by changes due to illumination, translations or rotations in the image. The main drawback of retinal angiographies is the different position of the vessels used in the authentication, because it is very difficult that the user places the eye in the same position in different acquisitions, so that an alignment is necessary prior to the authentication. To perform that alignment, an image registration algorithm is employed [5].

Here a feature-based authentication method is employed, as commented in [4], where features are extracted from the angiographies using a differential geometry operator (Figure 1), the MLSEC-ST [6, 7], and then a multi-resolution alignment algorithm is performed until both images (the reference image, stored in the database and the acquired image) are registered. Finally, similarity between the aligned images is measured by means of the Normalized Cross-Correlation Coefficient [8] γ, and if the value γ is higher than the acceptance threshold the individual is accepted, or rejected otherwise.

The whole authentication process is depicted in Figure 2. Firstly, in the *enrollment* stage, authorized individuals are entered in the database. After that, each time the user requests for a resource or for accessing to a protected area, the authentication process is performed (second stage).

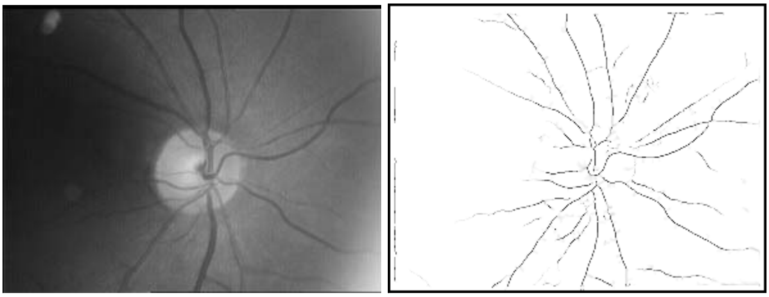


Fig. 1. Features extracted from angiographies are employed as the reference points in the registration process. (a) Original angiography. (b) Creases extracted from (a).

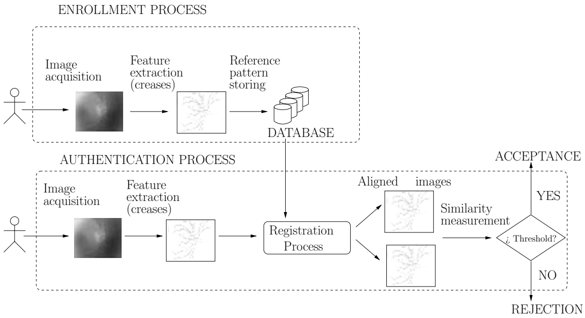


Fig. 2. Representation of the authentication process, composed by two stages: an initial enrollment stage, where the reference pattern is stored in the database of authorized users, and an authentication step, which is performed each time the individual requests for a protected resource

3 Description of the Distributed System

Our system employs a conventional client-server architecture [9]. Once the retinal pattern of the authorized person is stored in the database, the system performs an authentication procedure each time the individual tries to access to the protected resources of the system.

To fulfill the main requirements of the system, a framework which allows the communication between the server authentication system and the acquisition device located at the access points must be designed. That communication must be above all reliable and secure. That two requirements could be solved by means of a secure communication protocol, such as Secure Socket Layer (SSL) communications (Figure 3).

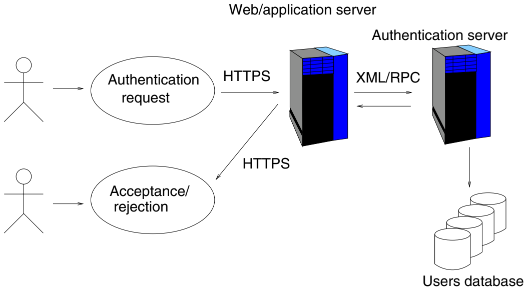


Fig. 3. The distributed authentication system

The most important tools employed for designing the system were the design patterns [10] and the UML [11, 12], which allow for a more robust and reliable software applications. For example, the strategy pattern encapsulates several methods employed to authenticate the individuals (although, by the moment, only the one described here has been reliably tested). But also distributed patterns appeared in the design: single threaded execution pattern is needed to allow only a client updates the data in the authorized people database at a time or the read-write lock pattern, which avoids unnecessary waiting to read data from the database by allowing concurrent reads when data are being updated.

A client application initiates a connection to authenticate an individual, and if authentication is successful, resources can be accessed. The graphical user interface for the interaction with data (like the depicted in Figure 4) is obtained through the web application.

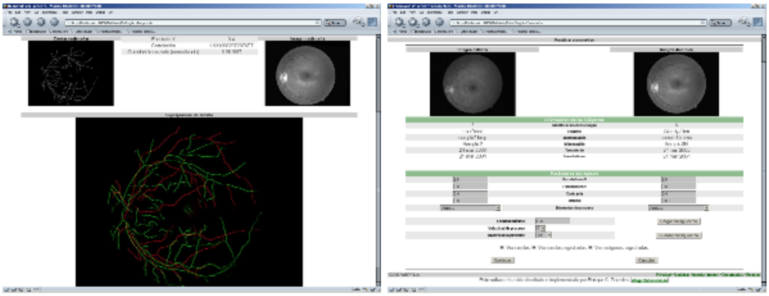


Fig. 4. Screenshots from the client program-application with retinographies, which are the biometric authentication parameter employed by the system. (a) Extracted features employed by the system. (b) Authentication parameters obtained in a typical access situation.

The overall operation flow at the client-side can be described as follow (see also Figure 3):

- An individual makes HTTP request to web server first. The web server will handle the request and will communicate with the authentication server to perform the authentication process.
- The authentication server will access the database and will allow or deny the access to the requested resource. Upon request, procedure execution will be invoked (using XML-RPC) based on the data sent by the client.
- Once the procedure ends, the related information is sent to the client and displayed in the browser window, allowing or denying access depending on the result of the authentication process.

4 Communication Technology

To assure confidentially and reliability in the communications, the most widely used secure Web encryption has been employed: the Secure Hypertext Transport Protocol (HTTPS) on the Secure Socket Layer (SSL) version 3.0 with 40 and 128 bit RC4 encryption. The SSL [13] protocol has become most widely used method for encrypting and authenticating Web communications.

Our approach to deal with the heterogeneity of the clients focuses on documents or data and transformations that visualize the data to fit the computing capabilities of different users, while preserving semantics of data. This division of the underlying data and the way it gets displayed mimics the Model-View-Controler (MVC), a well known and frequently used design pattern to develop interactive applications with flexible human-computer interfaces. This would allow for a future implementation of mobile authentication clients, for example.

XML has been chosen as the medium to represent the data sent to the server for the authentication. XML is a markup language for documents containing structured information. Once the data is written in XML, an associated XSL document can be written to define the way it gets displayed. The XML document is platform-independent and thus corresponds to the *model*, whereas the style sheet depends on the displaying device, and corresponds to the view.

MVC separation of view and model offers the following advantages over conventional application-centric environments:

- Model-view separation at the document level via XML/XSL documents.
- Simple communication protocol with a standard message format based on (encrypted) ASCII XML messages.

Finally, using XML-RPC [14], procedure calls are wrapped in XML establishing that way a simple pathway for calling authentication functions. XML-RPC is a quick-and-easy way to make procedure calls over the Internet. It converts the procedure call into XML document, sends it to a remote server using HTTP, and gets back the response as XML. Other considered (although finally discarded) protocols where SOAP and CORBA, both of them popular protocols for writing distributed, object-oriented applications, well-supported by many vendors, although they are much more complex than XML-RPC, and require fairly sophisticated clients. In addition, XML-RPC has fewer interoperability problems than them.

5 Conclusions and Future Work

In this paper an authentication web application has been presented. Using that program restricted access to system resources (buildings, data, etc.) can be granted only to authorized individuals, rejecting those requests from unauthorized persons. XML, which serves as the communications medium, is a standard that has already gained wide acceptance and provides a powerful medium for data exchange, visualization specifications and procedure execution by means of the XML-RPC method invocation procedure. Moreover all the communications take place over secure channels, through to the employment of encryption with HTTPS on the SSL version 3.0. Currently, system is under evaluation and many tasks as robustness assessment, performance verification, formal verification of whole the server-procedures, so as testing in a real environment must be performed.

Acknowledgments

This paper has been partly funded by the Xunta de Galicia through the grant contract PGIDIT03TIC10503PR

References

1. J.G. Daugman. Biometric personal identification system based on iris analysis. United States Patent No.5,291.560, 1994.
2. J.Bigüin, C.Chollet, and G.Borgefors, editors. *Proceedings of the 1st.International Conference on Audio- and Video-Based Biometric Person Authentication*, Crans-Montana,Switzerland, March 1997.
3. R. Zunkel. Hand geometry based verification. In *BIOMETRICS:Personal Identification in Networked Society*. Kluwert Academic Publishers, 1999.
4. C. Mariño, M.G. Penedo, and F. González. Retinal angiographies based authentication. *Lecture Notes in Computer Science*, 2905:306–313, 2003.
5. L.G. Brown. A survey of image registration techniques. *ACM Computer Surveys*, 24(4):325–376, 1992.
6. A. López, D. Lloret, J. Serrat, and J.J. Villanueva. Multilocal creasness based on the level set extrinsic curvature. *Computer Vision and Image Understanding*, 77:111–144, 2000.
7. C. Mariño, M. Penas, M.G. Penedo, D. Lloret, and M.J Carreira. Integration of mutual information and creaseness based methods for the automatic registration of slo sequences. In *Proceedings of the SIARP'2001, VI Simpósio Ibero-Americano de Reconhecimento de Padrões*, volume I, 2001.
8. J.P.Lewis. Fast template matching. *Vision Interface*, pages 120–123, 1995.
9. Robert Orfali, Dan Harkey, and Jeri Edwards. *Client/Server Survival Guide.* John Wiley & sons, 3rd edition, 1999.
10. E. Gamma, R. Helm, R. Johnson, and Vlissides J. *Design Patterns, Elements of Reusable Object-Oriented Software*. Professional Computing Series. Addison-Wesley, 1995.
11. Mark Grand. *Patterns in Java: a catalog of reusable design patterns illustrated with UML*, volume 1. New York,John Wiley & sons, 1998-1999.
12. Mark Grand. *Patterns in Java: a catalog of reusable design patterns illustrated with UML*, volume 1. New York,John Wiley & sons, 1998-1999.
13. K.E.B. Hickman. The SSL protocol. http://www.netscape.com/newsref/ssl.html, December 1995.
14. Simon St.Laurent, Joe Johnston, and Edd Dumbill. *Programming Web Services with XML-RPC*. O'Reilly, 2001.

Skeleton Extraction of 2D Objects Using Shock Wavefront Detection

Rubén Cárdenes[1] and Juan Ruiz-Alzola[2]

[1] Medical Technology Center, University of Las Palmas GC, Spain
`http://www.ctm.ulpgc.es`
[2] Canary Islands Institute of Technology (ITC), Spain
`http://www.itccanarias.org`
`ruben@ctm.ulpgc.es`, `jruiz@itccanarias.org`

Abstract. This paper proposes a method for computing the medial axis transform (MAT) or the skeleton of a general 2D shape using a technique with a high performance, based on a distance transform computation from the shape's boundaries. The distance transform is computed propagating a wavefront from the boundary, and the skeleton is obtained detecting the points where the wavefronts collide themselves, and applying connectivity rules during the process. This method has two main advantages: the efficiency and the preservation of the skeleton properties.

1 Introduction

The medial axis transform (MAT) or skeleton of a geometric figure is a set of curves that approximate the local symmetry axis of this figure. The concept of medial axis consists in the reduction of geometric shape information to the minimum. The first definition was given by Blum [1], who stated the analogy with a prairie fire: the MAT is the set of points where the propagation wavefront initiated from the shape boundary "intersects itself". Despite the extreme simplicity of this definition, the implementation in digital images without loss of important properties is surprisingly difficult. For this reason there exists a large number of algorithms and methods in the literature to address this problem. The relevance of MATs was introduced initially by Blum mainly for visual perception analysis and it is used especially for pattern recognition.

Formally there exist four definitions for medial axis transform, all of them equivalent, described by Montanari in [2]. In the first of them the medial axis transform is the prairie fire model proposed by Blum commented above where the fire lines represent the propagation fronts. The second definition states that the MAT is the set of ridges of the distance map constructed from the shape boundary. A distance map from a set of objects in an image, is the image where the value of each pixel is the distance to the nearest object [3],[4]. In the third definition, one of the more common, the MAT is the set of centers of the maximal discs of the figure, where a maximal disc is a circumference contained in the shape for which there exists no other circumference inside the shape that contains it. The last model define the MAT as the set of points that do not belong to any straight line segment connecting other points to their respective closest boundary points.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 392–397, 2005.
© Springer-Verlag Berlin Heidelberg 2005

The principal MAT properties are the connectivity, the preservation of the main topological features of the shape, i.e. to not lose any main branch in the skeleton and to not add irrelevant branches to it, and the Euclidean equidistance of every point of the skeleton to two or more points on the boundary. Regarding the discrete MAT, it is desirable to obtain one pixel thickness results to accomplish the narrowness property. Obviously in a discrete domain the equidistance property is not possible to obtain, so it is necessary to relax this property to get the most approximated result.

Methods to construct skeletons are well described in the literature, and can be divided into three categories. First, the topological thinning methods, Ammann [5], Zhang [6], that work eroding iteratively the shape until the skeleton is obtained. The criterion used to delete a point is local, so it is necessary to take care in some cases because the skeleton is a global property of the shape. These methods are computationally heavy, because a great number of iterations are needed. Second, the distance maps based methods, Arcelli [7], compute the MAT as the ridges of the DT computed from the boundary, as stated in the second Montanari's definition, and detecting the ridges as local maxima points in the distance maps. The main problem with these methods is the detection of saddle points, where two or more ridges intersect, so they are local maxima in some directions, but local minima in the directions tangent to the ridges that intersect, so these points are likely to be undetected, and the skeleton could be disconnected. Finally the Voronoi diagrams based methods that became very popular at the beginning of the 90's, see Ogniewwicz [8], Brandt [9], Sugihara [10] and Kimmel [11]. It is well known that the skeleton of a polygonal figure can be obtained from the Voronoi diagram of the polygon edges. In the case of an arbitrary shape, it is necessary to determine which segments of the shape should be separated to generate the Voronoi diagram. This is achieved in [8] by means of a residual function that avoids the branch generation in the skeleton from points that are close in the boundary. In [11] the Voronoi diagram is generated from boundary segments separated by curvature local maxima.

2 Method

In this paper we propose a method to extract the skeleton from a 2D shape, where the distance map from the shape boundaries is computed using a propagation scheme and then, the skeleton is constructed detecting the points where the collision from the wavefronts occurs. The distance map used in this paper is taken from a previous work [12], where a wavefront from each object is started and propagated until the domain is completely filled. This technique is based on ordered propagation [13] for the shake of efficiency, and instead of using the classical approach of bucket sorting to implement the propagation fronts, we use a double list structure to handle the wavefronts. This is more efficient than raster scan, and the distances are a good approximation to the Euclidean, and they are equivalent to those obtained by Danielsson's method [4].

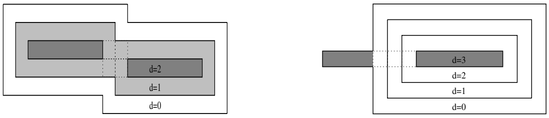
The way of detecting the points where the wavefront intersection takes place is carried out by a non propagation criterion. The discrete propagation fronts initiated from the shape boundary, are implemented with a list of pixels at Euclidean distance d from it, where d is an integer number, and these pixels are the ones which have real distance values closer to d inside the shape. The wavefront at distance d from the boundary will generate the wavefront at distance $d+1$, as long as there exist more inner non visited pixels. When a pixel can not be propagated deeper into the shape, it is automatically labeled as member of the skeleton. With this criterion the skeleton obtained is not connected and has not one pixel thickness.

The connectivity is not assured because two effects. The first one appears because points which should be labeled as belonging to the MAT, are not detected in cases where the propagation fronts coming from opposite directions do not intersect at the same pixel but adjacent ones, as in figure 1(a). The second effect happens when the shape becomes wider as illustrated in figure 1(b). The first effect is corrected a posteriori using the distance map computed, connecting the pixels of the computed MAT through the points with maxima values in the distance map. The second effect is solved in the propagation process, by connecting the extremal points of the skeleton computed in d (if any) with its nearest point in the front at distance d.

The skeletonization simplified pseudo code is as follows.

Input: Q: The set of n objects belonging to the boundary shape
Output: Sk : The set of points corresponding to the skeleton

```
for $i = 1$ to $n$ (Initialization)
    $F_i^0 = \mathbf{q_i}$
end for
d=0
while (There exist non visited elements in $\overline{Q}$) do
    propagate front $F^d$: $F^d \rightarrow F^{d+1}$
    for $i = 1$ to number of elements in $F^d$
        if ($\mathbf{p_i}$ agrees shock criterion)
            add $\mathbf{p_i}$ to $Sk$
        end if
    end for
    if ($Sk \neq \emptyset$)
            For each extremal point in $Sk$: $\mathbf{e}$, add the nearest
            point to $\mathbf{e}$ that belongs to $F^d$, to $Sk$
    end if
    d=d+1
end while
Thinning(Sk) (one pass)
Reconnection(Sk)
```



(a) (b)

Fig. 1. Disconnection effects: wavefronts coming from opposite directions do not intersect at the same pixel but adjacent ones (a), and aperture zones (b)

F^d represents the propagation front at distance d and $\overline{Q}$ represents the interior pixels of the shape.

The narrowness property is easily achieved by a regular thinning which is highly efficient because only a small percentage of the image has to be scanned, and only one thinning iteration is needed.

3 Results and Conclusions

Figure 2 shows the distance map obtained from the boundaries of a 2D object, and the skeleton extracted with our method. In figure 3, it is shown other results, where the skeleton from different 2D shapes are successfully obtained using shock wavefront detection. This algorithm presents several advantages. First, the MAT is computed very efficiently because it uses an ordered propagation scheme [13] which is a very fast operation. The post-processing steps, connection and thinning, are computed also very fast because a very small number of points are involved and only one thinning iteration is needed. Second, the topological properties of the shape are always preserved as shown in our results. Third, irrelevant branches in the MAT do not appear, i.e. it is not very sensitive to boundary noise, and no pruning of the resulting skeleton is needed. And finally this method can be also extended straightforwardly to 3D shapes, obtaining the medial surface transform.

On the other hand, the skeleton branches obtained with our method do not start in the boundary, in general, but they could be extended if necessary. Skele-

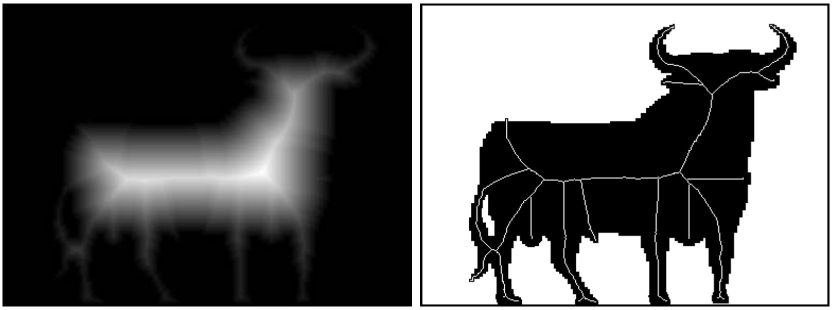


Fig. 2. Distance Transform from the boundary of a shape and skeleton

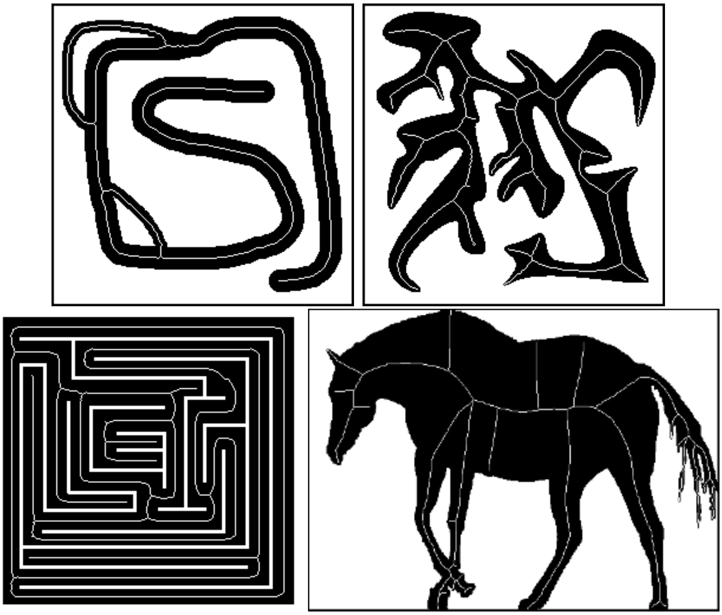


Fig. 3. Medial axis transform in several shapes

tons are used in multiple applications, from motion path planning, to pattern recognition and shape analysis, and we strongly believe that the method presented here is suitable for many of those applications, especially when the efficiency is one of the key factors.

Acknowledgments

This work has been partially funded by an European grant FP6-507609 from the European Commission, and has been partially supported by the Spanish Ministry of Science and Technology and European Commission, co-funded grant TIC-2001-38008-C02-01.

References

1. H. Blum, “A transformation for extracting new descriptors of shape,” in *Proc. Models for the Perception of Speech and Visual Form*, Weiant Wathen-Dunn, Ed., Cambridge, MA, November 1967, pp. 362–380, MIT Press.
2. U. Montanari, “A method for obtaining skeletons using a quasi-euclidean distance,” *Journal of the Association for Computing Machinery*, vol. 15, no. 4, pp. 600–624, 1968.
3. A. Rosenfeld and J.L. Pfaltz, “Sequential operations in digital picture processing,” *J. Assoc. Comp. Mach.*, vol. 13, pp. 471–494, 1966.
4. P.E. Danielsson, “Euclidean distance mapping,” *Computer Graphics and Image Processing*, vol. 14, pp. 227–248, 1980.

5. C.J. Ammann and A.G. Sartori-Angus, "Fast thinning algorithm for binary images," *Image Vision Comput.*, vol. 3, no. 2, pp. 71–79, 1985.
6. Y.Y. Zhang and P.S.P Wan, "A parallel thinning algorithm with two-subiteration that generates one-pixel-wide skeletons," in *Proceedings of International Conference on Patter Recognition.* 1996, vol. 4, pp. 457–461, IEEE Computer Society.
7. C. Arcelli and G. Santini di Baja, "Finding local maxima in a pseudo-euclidean distance transformation," *CVGIP*, vol. 43, pp. 361–367, 1988.
8. R. Ogniewicz and M. Ilg, "Voronoi skeletons: Theory and applications," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR*, Los Alamitos, California, 15–18 1992, pp. 63–69, IEEE Computer Society.
9. J.W. Brandt and V.R. Algazi, "Continuous skeleton computation by Voronoi diagram," in *CVGIP Image Understanding*, 1992, vol. 55, pp. 329–338.
10. K. Sugihara, "Approximation of generalized Voronoi diagrams by ordinary Voronoi diagrams," *CVGIP: Graphical Models and Image Processing*, vol. 55(6), pp. 522–531, 1993.
11. R. Kimmel, D. Shaked, N. Kiryati, and A. M. Bruckstein, "Skeletonization via distance maps and level sets," *Computer Vision and Image Understanding: CVIU*, vol. 62, no. 3, pp. 382–391, 1995.
12. R. Cardenes, *Esquemas eficientes de geometría computacional aplicados a la segmentacióon de imágenes médicas*, Ph.D. thesis, University of Las Palmas de Gran Canaria, 2004.
13. B.H. Verwer, P.W. Verbeek, and S.T. Dekker, "An efficient uniform cost algorithm applied to distance transforms," *IEEE Transactions on Pattern Analysis an Machine Intelligence*, vol. 11(4), pp. 425–429, 1989.

Cue Combination for Robust Real-Time Multiple Face Detection at Different Resolutions*

M. Castrillón-Santana, O. Déniz-Suárez,
C. Guerra-Artal, and J. Isern-González

IUSIANI, Edif. Ctral. del Parque Científico Tecnológico,
Universidad de Las Palmas de Gran Canaria, 35017, Spain
`mcastrillon@iusiani.ulpgc.es`

Abstract. This paper describes a face detection system conceived to process video streams in real-time. Cue combination allows the system to tackle the temporal restrictions achieving a notable detection rate. The system developed is able to detect simultaneously at different resolutions multiple individuals building a feature based model for each detected face.

1 Introduction

If Human Computer Interaction (HCI) were more similar to human to human communication, HCI would be non-intrusive, more natural and comfortable for humans [8]. Human beings are sociable and communicate not only with words but with sounds and gestures. In this context, the human face is a main information channel during the communication process.

The face detection problem is a revisited topic in the literature, and it is commonly defined as: *to determine any face -if any- in the image returning the location and extent of each* [3,13]. According to some recent works [6] [9] [11] the problem seems to be solved. However, those detectors focus the problem using approaches which are valid for restricted face dimensions, and with the exception of the first reference, to a reduced head pose range. Particularly for video stream processing, they try to solve the problem in a monolithic fashion, neglecting elements that the human system employs: temporal and contextual information, and cue combination.

Section 2 describes our approach for robust real-time multiple face detection based on the combination of different cues. The resulting approach achieves better detection rates for video stream processing and cheaper processing costs than outstanding and publicly available face detection systems, as suggested by different experiments in Section 3.

---

* Work partially funded by research projects Univ. of Las Palmas de Gran Canaria UNI2003/06, UNI2004/10 and UNI2004/25, Canary Islands Autonomous Government PI2003/160 and PI2003/165 and the Spanish Ministry of Education and Science and FEDER funds (TIN2004-07087).

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 398–403, 2005.
© Springer-Verlag Berlin Heidelberg 2005

2 The Face Detection Approach

The system developed assumes that a single technique alone can not solve the problem properly for any circumstance. Therefore, our approach combines different single techniques which are not robust and fast enough individually, but as a whole they outperform any individual approach included in terms of correct detection rate and processing cost.

The basic system working makes use of an exhaustive face detection approach, which allows the system to extract different features of the face detected. Those features are used considering temporal coherence in the next frames, obviously only after a detection, reducing the processing cost which any exhaustive approach would have if it were applied to every frame.

On the one side, the approach makes use for the first detection or after a failure, of two window shift detectors based on the general object detection framework described in [11], which provide acceptable performance and processing rates for their particular context although an exhaustive search is performed. These two brute force detectors, recently integrated in OpenCV computer vision library [4], are the frontal face detector described in [11], and the local context based face detector described in [5]. The last one achieves better recognition rates for low resolution images and non frontal faces whenever the head and shoulders are visible.

On the other side, as we mentioned above, the approach extracts different features from each detected face in a frame, therefore multiple face detection is considered. Indeed the exclusive use of a monolithic approach based on the Viola framework has the disadvantage of not using a main cue needed for video processing: temporal coherence. Any face detected in a frame provides information which can be used in the next frames to speed up the process. Therefore, for each detected face, the system stores not only its position and size, but also its average color using red, green normalized color space. Those features are certainly useful useful to speed up the process, e.g., it is used to define a Window of Attention where the previous detection will likely be, or if the face size is big enough to be worth the application of the object centered detector. In any case, this information is valuable to reduce the time consumption.

Among the different features, the skin color is a valid cue extensively used in the literature. Skin color based approaches for face detection have a lack of robustness for different conditions. A well known problem is the absence of a general skin color representation for any kind of light source and camera [10]. However, the skin color extracted from the face previously detected by the frontal face Viola detector can be used to estimate facial features position by means of the color blob, which provides valuable information to detect eye positions for frontal faces [1].

Additional features have been considered in order to develop a more robust system. Each face in a frame is characterized by different features $f = \langle pos, size, red, green, leye_{pos}, leye_{pattern}, reye_{pos}, reye_{pattern}, face_{pattern} \rangle$.These features direct different cues in the next frame which are applied opportunistically in an order based on the computational cost and the reliability.

- Eye tracking: A fast tracking algorithm [2] is applied in an area that surrounds previously detected eyes, if available.
- Face detector: The Viola-Jones detector is applied in an area that covers the previous detection [11].
- Local context face detector: If previous techniques fail, it is applied in an area that includes the previous detection [5].
- Skin color: Skin color, defined using red-green normalized color space [12], is searched in the window that contains the previous detection, and the new sizes and positions coherently checked.
- Face tracking: If everything else fails, the prerecorded face pattern is searched in an area that covers previous detection [2].

These techniques are applied until one of them finds the face, or the process will be restarted using the Viola-Jones based detectors applied to the whole image. Whenever a face is detected, the skin color is used for facial features detection [1]. Obviously, if there were no recent detection, there is no face model active, and therefore the object-centered and local context detectors are applied sequentially to the whole image.

A single or multiple faces detected in consecutive frames are related according to their specific features. During the video stream processing, the face detector gathers a set of detection threads, $IS = \{dt_1, dt_2, ..., dt_n\}$. A detection thread contains a set of continuous detections, i.e. detections which take place in different frames but are related by the system in terms of position, size and pattern matching techniques.

The Viola-Jones based detectors have some level of false detections. For that reason a new detection thread is created only if the eyes are located too. The use of the weakest cues, i.e. color and tracking, after a recent detection is reserved to detections which are already considered part of a detection thread. In this way, spurious detections do not launch cues which are not robust enough, in the sense that they are not able to recover from a false face detection. The final results is that for each detection thread, the face detector system provides a number of facial samples, $dt_p = \{x_1, ..., x_{m_p}\}$, which correspond to those detections for which also the eyes were located.

The resulting system is able to manage in real-time complex scenes in which the human face experiences large scale, pose and appearance transformations. Each specialized detector is described in more detail below.

3 Experiments

The system here described has been applied to still images and video streams. For still images the lower boundary is the combination of both Viola-Jones based detectors performances [5,11]. If both detectors return a face in the same position, it is preferred the frontal face detector data as it is more precise. For still images the added value of the approach is the likely eye detection for almost frontal views and the combination of two Viola-Jones based detectors, see Figure 1.

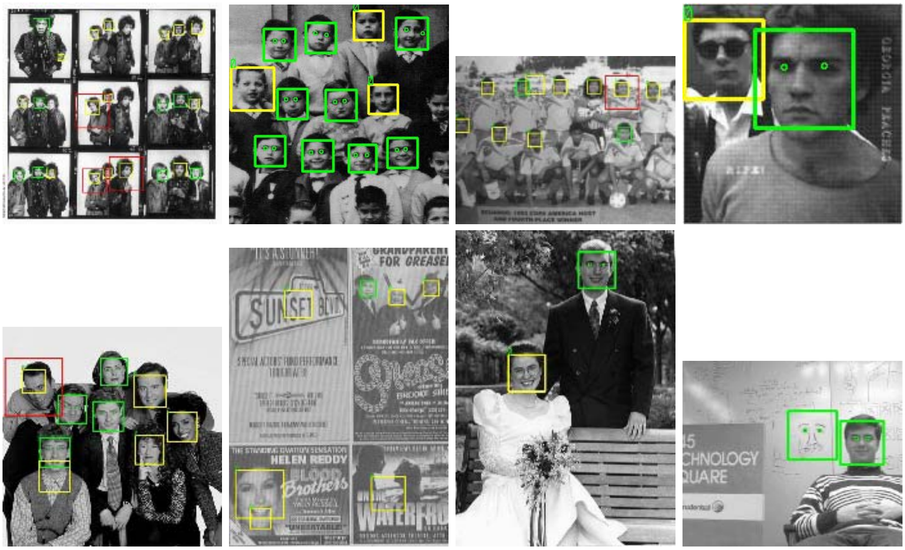


Fig. 1. Detection examples for some CMU database [9] samples. Color indicates the technique used: green means that the eyes were detected, yellow means that they were not detected, and red containing a yellow rectangle means that the local context detector was used. The images have been scaled down to fit the paper size. The size of the images are originally 814 × 820, 256 × 256, 611 × 467 and 159 × 160 respectively in the first row and in the second 500×500, 539×734, 336×484 and 258×218. Obviously for still images there are no detections based on skin color or tracking.

The benefits of our approach are evidenced in video stream processing. 70 desktop sequences containing more than 30000 images of different individuals for typical webcam resolutions 320 × 240, were processed at an average rate of 20 fps providing therefore multiple face detection in real-time. In Table 1, it is observed that the detection rates of the approach are slightly better than those achieved by the OpenCV implementation of the Viola frontal face detector [7] (3 percentage points greater). However, those results are obtained five times faster, and with the added value of correct eye detection for more than 60% of the faces detected. The approach is also suitable for sequences with resolution changes, see Figure 3.

The temporal coherence applied to video stream processing not only speeds up the process but also allows to despise most false detections which appear when a still image is processed, see some false detections in Figure 1. In at least 10 of the sequences analyzed some detections were non face patterns, and they were correctly not assigned to any detection thread as the eyes were not found and their position, or their color and size were not coherent with any active detection thread. Or in the worst case, a non face detection was associated to a detection thread, but the system observed soon an incoherence and decided to remove the detection thread and wait for a new eye pair detection.

Table 1. Results for face and eye detection. TD reflects correct detection ratio and FD means false detection ratio.

	Viola		Our approach	
	TD	FD	TD	FD
Faces	90.1%	8.2%	92.9%	8%
Eyes	0.0%	-	64.3%	3.7%
Proc. time	117.5 msecs.		21 msecs.	

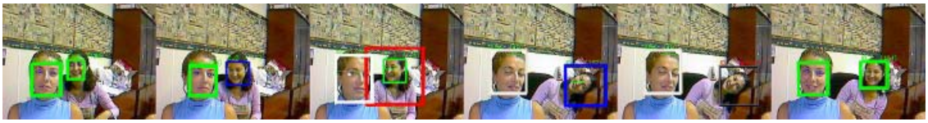


Fig. 2. From left to right: 1) Both faces are detected and their eyes, 2) the Viola based detectors failed detecting the right face, it is detected by tracking the face pattern, 3) the left face is detected using skin color and the right one by means of the local context face detector, 4) the same for the left face, the right one is found by tracking, 5) face pattern tracking is not allowed to be the only valid cue for many consecutive frames, so the right face detection thread is considered missed, and 6) the right face recover its vertical position and fused with the latent detection thread.

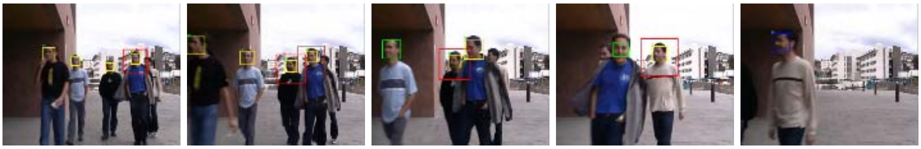


Fig. 3. Frames extracted from a video stream with 720 × 576 resolution. The color has the same meaning than in Figure 1, but observe that the last frame depicts a blue rectangle which means that tracking was used.

4 Conclusions and Future Work

We have developed a face detection system which provides robust multiple face detection at frame rate using a standard webcam. The system has been tested with 70 sequences containing around 30000 images achieving higher detection rate (aprox. five times faster) than the Viola-Jones based face detector, and providing additionally the location of the eyes for more than 60% of the images.

The results achieved processing video streams have been possible thanks to the integration of different cues and particularly the temporal coherence. The average processing time of 21 msecs. reported by the system, makes it suitable for further use in the field of perceptual user interfaces.

References

1. M. Castrillón Santana, F.M. Hernández Tejera, and J. Cabrera Gámez. Encara: real-time detection of frontal faces. In *International Conference on Image Processing*, Barcelona, Spain, September 2003.
2. Cayetano Guerra Artal. *Contribuciones al seguimiento visual precategórico.* PhD thesis, Universidad de Las Palmas de Gran Canaria, Octubre 2002.
3. Erik Hjelmas and Boon Kee Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, 2001.
4. Intel. Intel Open Source Computer Vision Library, b4.0. www.intel.com/research/mrl/research/opencv, August 2004.
5. Hannes Kruppa, Modesto Castrillón Santana, and Bernt Schiele. Fast and robust face finding via local context. In *Joint IEEE Internacional Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, pages 157–164, October 2003.
6. Stan Z. Li, Long Zhu, ZhenQiu Zhang, Andrew Blake, HongJiag Zhang, and Harry Shum. Statistical learning of multi-view face detection. In *European Conference Computer Vision*, pages 67–81, 2002.
7. Rainer Lienhart and Jochen Maydt. An extended set of haar-like features for rapid object detection. In *IEEE ICIP 2002*, volume 1, pages 900–903, September 2002.
8. Alex Pentland. Looking at people: Sensing for ubiquitous and wearable computing. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 107–119, January 2000.
9. Henry Schneiderman and Takeo Kanade. A statistical method for 3d object detection applied to faces and cars. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1746–1759, 2000.
10. Moritz Storring, Hans J. Andersen, and Erik Granum. Physics-based modelling of human skin colour under mixed illuminants. *Robotics and Autonomous Systems*, 2001.
11. Paul Viola and Michael J. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition*, pages 511–518, 2001.
12. Christopher Wren, Ali Azarrbayejani, Trevor Darrell, and Alex Pentland. Pfinder: Real-time tracking of the human body. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.
13. Ming-Hsuan Yang, David Kriegman, and Narendra Ahuja. Detecting faces in images: A survey. *Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.

Evolutionary Color Constancy Algorithm Based on the Gamut Mapping Paradigm

Cristian Munteanu[1,2], Agostinho Rosa[1], Manuel Galan[2],
and Enrique Rubio Royo[2]

[1] Instituto de Sistemas e Robotica, Instituto Superior Tecnico, Av. Rovisco Pais 1,
Torre Norte 6.21, 1049-001, Lisboa, Portugal
`{cmunteanu, acrosa}@isr.ist.utl.pt`
`http://laseeb.isr.ist.utl.pt/`
[2] CICEI, Universidad de Las Palmas de Gran Canaria, Edificio Central del Parque,
Científico y Tecnológico, Campus Universitario de Tafira, 35017 – Las Palmas, Spain
`rubio@cicei.com`

Abstract. In recent years, extensive work has been done to design algorithms that strive to mimic the robust human vision system which is able to perceive the true colors and discount the illuminant from a scene viewed under light having different spectral compositions (the feature is called "color constancy"). We propose a straightforward approach to the color constancy problem by employing an Interactive Genetic Algorithm [1] (e.g. a Genetic Algorithm [2], [3] guided by the user) that optimizes a well known and robust variant of color constancy algorithm called "gamut mapping" [4]. Results obtained on a set of test images and comparison to various color constancy algorithms, show that our method achieves a good color constancy behavior with no additional knowledge required besides the image that is to be color-corrected, and with minimal assumptions about the scene captured in the image.

1 Introduction

Color Constancy algorithms are fundamental both in Computer Vision whenever recognition of objects in a scene is performed based on the color of the respective objects, as well as for color correction of digital pictures with applications in medicine, remote sensing, arts and media. When compared to the human visual system, algorithms that perform Color Constancy fail short to achieve the same effectiveness in recuperating the colors in scenes taken under different illuminants. This is due to the fact that such algorithms make restrictive assumptions about the world from which the image was taken. More-over, several algorithms require additional information about the scene and the technical characteristics of the camera that captures the image, knowledge that is not always readily available. We tackle the color constancy problem by employing an Interactive Genetic Algorithm [1] (e.g. a Genetic Algorithm guided by the user) that optimizes a well known and robust variant of color constancy algorithm called "gamut mapping" [2]. Results are given on a set of test images and comparison is made to various color constancy algorithms that proved efficient and that pertain to the following categories: Gray World (GW) methods, Gamut Mapping methods, and Neural Networks methods [2].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 404 – 409, 2005.
© Springer-Verlag Berlin Heidelberg 2005

1.1 Gamut Mapping as Basis Transformation for IGA-GM

The set of all possible responses due to known or expected surface reflectances, as seen under a known, canonical illuminant (usually taken to be the daylight, or an approximation of it), is a convex set, referred to as the canonical gamut. Similarly, the set of responses due to an unknown illuminant is also a convex set. Illumination change is assumed to be modeled by a simple diagonal transformation according to the research of von Kries which was later confirmed to hold as a reasonable assumption in [3]. It was shown that under the von Kries adaptation assumption the two gamuts are within a diagonal transformation of each other. Due to the fact that GM proposes a simple diagonal transformation to each pixel in the image, and because it has been shown that under not very restrictive assumptions such diagonal transformations suffice for good color constancy [3], in this paper we adopt the GM method as the basis for the color constancy transformation applied to correct a given image. The main drawbacks of the classical GM methods are: a) they use extensive knowledge about the camera used to take the picture; b) they use a priori information about the world in which the image was taken in order to construct the gamuts.

2 IGA-GM. Algorithm Presentation

In the present paper we employ an Interactive Genetic Algorithm (IGA) in which a human evaluator iteratively gives a subjective score for each result of a diagonal mapping (corrected image), until the best corrected image (and implicitly the best diagonal mapping) is discovered. Thus IGA-GM adapts to the "best" gamut mapping (diagonal transformation) according to human subjective criteria. This is done without making use of any a priori knowledge about the surrounding world conditions, or the camera. The Genetic Algorithm [4] is given in pseudocode in Fig.1. Each individual x_i in the population (chromosome) codes a diagonal transformation which is next applied to each pixel in the original (input) image, to give the corrected (output) image:

$$\forall i, j : (R^{\text{corrected}}_{\text{pixel}(i,j)}, G^{\text{corrected}}_{\text{pixel}(i,j)}, B^{\text{corrected}}_{\text{pixel}(i,j)}) = (x_{i,1} \cdot R^{\text{original}}_{\text{pixel}(i,j)}, x_{i,2} \cdot G^{\text{original}}_{\text{pixel}(i,j)}, x_{i,3} \cdot B^{\text{original}}_{\text{pixel}(i,j)}) \tag{1}$$

How well a chromosome performs (that is, how good the diagonal mapping is in recuperating the original colors and how well it corrects the input image) is judged mainly by the human evaluator who looks at the corrected image and gives a numeric score, called fitness value: $f(x_i)$ ranging from 0 (worst appearing image) to 10 (best appearing image). The user shouldn't evaluate all images corrected by each chromosome in the population, in each generation of the algorithm, because such a process would become too tedious (there would be a lot of images to evaluate). Many of the chromosomes to be evaluated are actually given a fitness value automatically, using a clustering algorithm over fitness values previously allocated by the user. Clustering is performed by two distinct procedures (see Fig. 1): `evaluate_1` and `evaluate_2`. `evaluate_1` is applied in the first generation of the GA, when the population is initialized randomly. The population is clustered and for each cluster an individual is picked up at random. The respective chromosome is evaluated by the user. The rest of the members of the respective cluster get the same fitness value as the one allocated by the user. In `evaluate_2` again the population is first clustered, then for each individual in the population we identify the cluster in which this individual lies. There is a fixed probability P_E that this individual will be evaluated by the user. If not evaluated by the user we proceed as follows: if the size of the respective cluster is greater than 2, the fitness of the respective individual is computed as the average of the fitness of all individuals in the cluster. If the cluster contains just one individual (e.g. the individual for which we calculate the fitness), then the respective chromosome is evaluated by the user.

```
$t := 0$
initialise $P(t)$
evaluate_1 $P(t)$
while (terminate($P(t)$) $\neq$ true) do
{
        $P'(t)$ := select_tournament ($P(t) \mid q$)
        $P''(t)$ := crossover ($P'(t) \mid P_c$)
        $P'''(t)$ := mutate ($P''(t) \mid P_m$)
        $P(t+1)$ := elitist ($P'''(t) \cup P(t) \mid K$)
        evaluate_2 $P(t+1)$
        $t := t+1$
}
endwhile
```

Fig. 1. Pseudocode of IGA-GM. t represents the time/generation index; $P(t)$ is the population at generation t, $P'(t)$ is the population at generation t after selection, $P''(t)$ is the population at generation t after selection and crossover, and $P'''(t)$ is the population at generation t after selection, crossover and mutation. K is the number of elites.

The selection strategy has been adopted to insure a steady convergent behavior of the algorithm. The trade-off we had to make is the well-known trade-off between exploration and exploitation present in any search method including EA. The convergent exploitation assured by selection and crossover should well-balance the wide exploration effect achieved by our mutation operator [5]. The selection method (`select_tournament` in Fig. 1) was chosen as a combination between binary tournament for which groups of $q = 2$ individuals are randomly formed and the chromosome with better fitness in the group is selected into the next generation [5], and a K-elitist scheme that attempts the preservation of the K best individuals in the population [6]. Binary tournament was chosen because it has a constant and relatively high selection pressure [7], and it is simple to implement and computationally light. The elitist scheme does not assure with probability 1 the preservation of the best individuals in the population, it just attempts this preservation. This is due to the subjective nature of fitness allocation by the user: the human evaluator might score the same image (corresponding to the best chromosome) with different fitness values on different occasions, and when this score diminishes in value the respective chromosome (i.e. the best solution) might be lost from the population due to selection effects.

The crossover operator should efficiently exploit the search space around the parents (i.e. the pair of chromosomes that undergo crossover). The advantage of using SBX is that it generates children (i.e. the pair of chromosomes resulted after crossing-over the parents) that are spread symmetrically around the parents and proportionally to the spread of the parents, as discussed in [8]. SBX was implemented as described in [8], the parameter that controls the spread of the children around the parents being η, and the probability of applying crossover is P_c.

The mutation operator should counter-balance the reduction of diversity due mainly to selection (associated with exploitation of the search space) and reintroduce diversity into population and thus increase exploration of the search space. The reduction of diversity is a cause of premature convergence [5] and it becomes more apparent when small populations of short in length chromosomes are used. This is precisely the case of our application. For such cases we have designed a novel mutation operator based on Principal Component Analysis, called PCA-mutation in [9] which maintains high levels of diversity in the population and increases the probability of discovering better solutions, as shown in [9]. The parameter of PCA-mutation (see [9]) is denominated c, and the probability of applying mutation is P_m.

3 Experimental Results

The experimental part comprises two main sections: a) firstly we perform a comparison between IGA-GM and other Color Constancy algorithms for a single unknown illuminant (e.g. different than canonical or daylight illuminant). b) secondly, we check the effectiveness of IGA-GM on a set of scenes taken under two quite different than daylight illuminants. In Table 1 we give the parameters of IGA-GM used in both experimental parts and for all images in the test set.

Table 1. IGA-GM parameters: N - population size, l- number of genes in each chromosome that undergo the evolution process, Pc - crossover probability , Pm - mutation probability, P_E - user evaluation probability, q – size of the tournment selection, c – parameter of PCA-mutation (see [35]), η – parameter of the SBX crossover, T_{max} -number of generations the GA is allowed to run, vlb_j - lower bound of the genes, vub_j - upper bound of the genes, χ - inconsistency threshold for the clustering algorithm

N	l	P_c	P_m	P_E	K	q	c	η	T_{max}	vlb_j	vub_j	χ
50	3	0.9	0.25	0.04	1	2	100	2	10	0	100	0.95

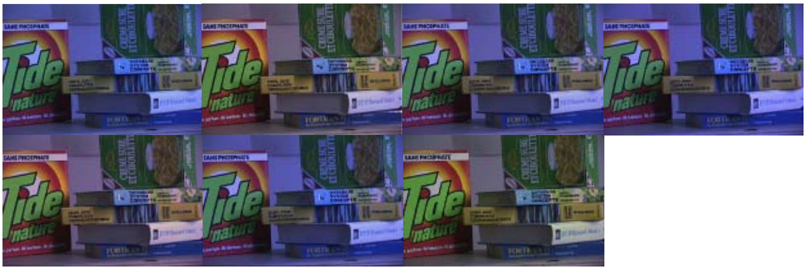


Fig. 2. Comparison to Color Constancy algorithms (Detergent image). Upper-row from left to right: original image (to be color-corrected), target image (image under canonical illuminant), GW, Retinex; Lower-row from left to right: NN, GM, IGA-GM.

Table 2. RMS error calculated for *Detergent* image on pairs between target image (T) and Color Constancy corrected image. The image that is not corrected (the original - input image) is denoted as O.

	(T, O)	(T,GW)	(T, Retinex)	(T,NN)	(T, GM)	(T, IGA-GM)
RMS$_{(R,G,B)}$	2.34e+04	18.02	49.13	13.38	22,30	16.16
RMS$_{(r,g)}$	16.42	12.88	16.27	7.78	21.74	7.99

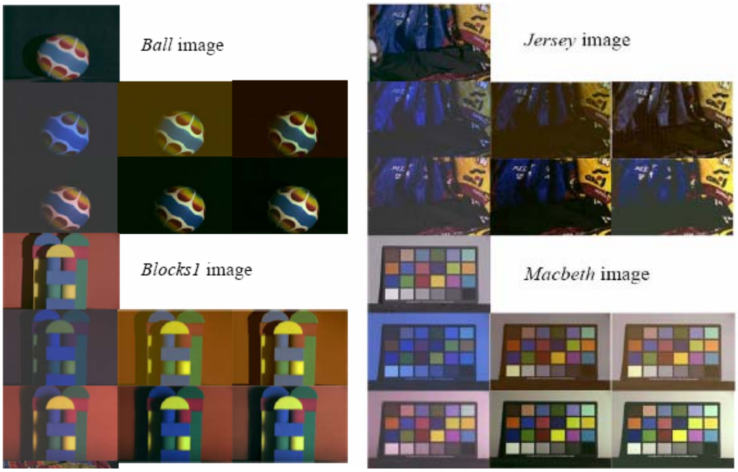


Fig. 3. IGA-GM solutions on different illuminants. Figure is divided in horizontal bands for each scene. First column first line: target image; second line first column: image under "solux_4700+3202"; second line second column: IGA-GM; second line third column: "IGA-GM + intensity correction"; third line first column: image under "syl_wwf"; third line second column: IGA-GM; third line third column: IGA-GM + intensity correction.

4 Conclusions

The main advantages of IGA-GM are the following: the method does not require previous information and knowledge about the scene being captured in the image, it does not make further restrictive assumptions about the world from which the image was taken, it is a simple algorithm with no pre-processing or training phases, it takes into account very subtle subjective criteria when judging the quality of an image, criteria that so far cannot be successfully "coded" in an automatic or machine controlled way. Moreover, results obtained on a wide set of test images show that IGA-GM achieves an effectiveness that is close to that of the best Color Constancy methods operating on the respective images. Consequently, the advantage lies in the robustness of IGA-GM, that is: we may use a single method (i.e. IGA-GM) to correct a wide range of images, instead of testing several Color Constancy methods (such as GM, GW, NN, etc.) and see which performs better on the respective images. Though results are good even when the human evaluator that analysis the outputs of IGA-GM doesn't have any knowledge about the scene captured in the input image, an increase in efficiency is expected when human evaluator experts are used to correct images which pertain to fields for which they have acquired the necessary expertise. Thus, for future work, we will first divide the images into groups pertaining to various fields and letting a human expert on the respective field correct the images using IGA-GM. Such fields of application may include medical images, remote sensing images, consumer and commercial images, or artistic photography.

References

1. Takagi, H.: Interactive Evolutionary Computation: Fusion of the Capabilities of EC Optimization and Human Evaluation. In: Proceedings of the IEEE 89 (2001) 1275–1296
2. Barnard, K., Cardei, V., Funt, B. V.: A Comparison of Computational Color Constancy Algorithms-Part I: Methodology and Experiments with Synthesized Data. IEEE Trans. on Image Processing 11(9) (2002) 972–984
3. Finlayson, G., Drew, M. S., Funt, B.V.: Diagonal transforms suffice for color constancy. In: Proceedings IEEE Int. Conf. on Computer Vision (1993) 164–171
4. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)
5. Back, T., Fogel, D., Michalewicz, Z., Bäck, T.(eds.): Handbook of Evolutionary Computation. Institute of Physics Publishing (1997)
6. Bäck, T., Hoffmeister, F.: Extended selection mechanisms in Genetic Algorithms. In: Proceedings 4th Int. Conf. Genetic Algorithms (1991) 92–99
7. Miller, B. L., Goldberg, D. E.: Genetic Algorithms, tournament selection and the effects of noise. Complex Systems 9 (1996) 193–212
8. Deb, K., Beyer, H.-G.: Self-Adaptation in Real-Parameter Genetic Algorithms with Simulated Binary Crossover. Evolutionary Computation 9(2) (2001) 197–221
9. Munteanu, C., Lazarescu, V.: Improving mutation capabilities in a real-coded GA. Proceedings of EvoIASP'99. Lecture Notes in Computer Science, Vol. 1596. Springer-Verlag, Berlin Heidelberg New York (1999) 138–149

Vision Based Automatic Occupant Classification and Pose Recognition for Smart Airbag Deployment

Min-Soo Jang[1], Yong-Guk Kim[2], Sang-Jun Kim[1], Jeong-Eom Lee[1],
Soek-Joo Lee[3], and Gwi-Tae Park[1]

[1] Dept. of Electrical Engineering, Korea University, Seoul, Korea
gtpark@korea.ac.kr
[2] School of Computer Engineering, Sejong University, Seoul, Korea
ykim@sejong.ac.kr
[3] Hyundai Autonet Co., Korea
gidung@haco.co.kr

Abstract. Airbags have been saved thousands of lives and reduced the number of serious injuries from collisions. However, the car occupant can be often hurt, or killed, by the airbag itself. For reducing the risk caused by airbag, designing a smart airbag is an important issue. This paper presents a vision based automatic system that can control triggering and intensity of airbag deployment. The system consists of an occupant classification system and an occupant pose recognition system, by which we aim to control whether the airbag should be triggered or not, and how strongly it should be deployed when it is triggered. Results suggest that the system is feasible as a vision based airbag controller.

1 Introduction

During the last decade, the safety of occupants in the cars has been increased by improving the airbag system, which can prevent life-threatening head (or chest) injuries by avoiding direct contact to the dashboard during the accident. However, the National Highway Traffic Safety Administration (NHTSA) reported that since 1990, over 200 fatalities have been recorded as a direct result of the airbag deployment [1]. The majority of these deaths have been children (or infants in rear-facing infant seat). In response, the NHTSA had issued a set of regulations mandating low risk deployment and smart airbag that adapt intelligently to the occupant.

In this paper, we present a vision based automatic system that consists of occupant classification system to control triggering of airbag deployment and occupant pose recognition system to control intensity of it. Vision sensor has some advantages, such as diverse information about occupant, high accuracy, and low restriction of installation position. Our vision system consists of a pair of CCD cameras mounted on rear-view mirror position and it is pointed towards the occupant. To classify the occupant, we use 2-stage SVM classifier, which input is down-sampled disparity map images of the occupant [2, 3]. And the occupant pose recognition system uses two algorithms in parallel, head detection algorithms based on a head contour model and head tracking algorithm using the inverse compositional image alignment method [4, 5, 6].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 410 – 415, 2005.
© Springer-Verlag Berlin Heidelberg 2005

This present paper consists of several sections as follow. Architecture of the vision-based airbag control system is discussed in section 2. Occupant classification system and occupant pose recognition system are described in section 3 and 4, respectively. Result of experiments is reported in section 5. Finally, our result is summarized, and the performance of the whole system is discussed in section 6.

2 The Airbag Control System

In this section, we describe our airbag control system and how the system is operated. Fig. 1(a) shows the architecture of the system. Our vision-based airbag control system consists of two parts: vision-based occupant classification (VOC) system and vision-based occupant pose recognition (VOPR) system. The VOC system classifies the occupant into 4 classes, such as 5th%tile female (or adult), 6 years-old child, infant seat and empty seat. According to the classification outcome, the system will deploy the airbag only if the occupant is classified as the 5th%tile female or adult, otherwise it will not be deployed. For the VOPR system, we assume that the head position of the occupant indicates his pose. So that, we divide the passenger's space into 7 areas and the system recognizes which area the occupant's head belongs to as shown in Fig. 1 (b). When the occupant's head belongs to the area 1 or 2, the system will regard the occupant is too close to dashboard and deploy the airbag weakly. Otherwise, the system will deploy it normally.

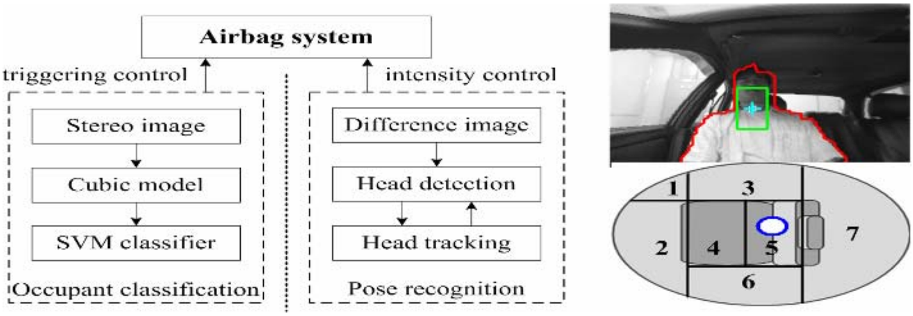


Fig. 1. The architecture of the airbag control system (a) and the head tracking for the pose recognition (b) - the solid circle indicates the head position of an occupant and the numbers show the area where the occupant can take a pose.

3 Occupant Classification

Fig. 2 shows an overall sequence of the VOC system. First, a disparity map image is obtained by using a fast SAD algorithm [7], and we apply the cubic model to the disparity map image for eliminating unnecessary information, such as dashboard, window and door. Then, the SVM classifier takes the down-sampled image of the disparity map image as an input [2, 3]. Since the disparity map of the 5th%tile female and 6 years-old child are similar compared to other cases, the SVM classifier consists of two stages: in the first stage, it classifies the occupant into three classes, such as the 5th%tile female or 6 years-old child as the first class, the infant seat as the second class, and the empty seat as the third class; in the second stage, the first class case is classified as either the 5th%tile female or 6 years-old child.

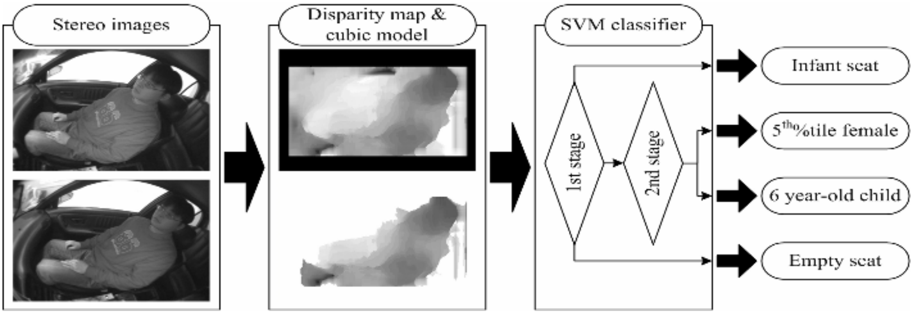


Fig. 2. Operation sequence of the VOC system

4 Occupant' Pose Recognition

In the VOPR system, to make the system robust, two algorithms are working in parallel. Fig. 3 shows the flow chart of the VOPR system.

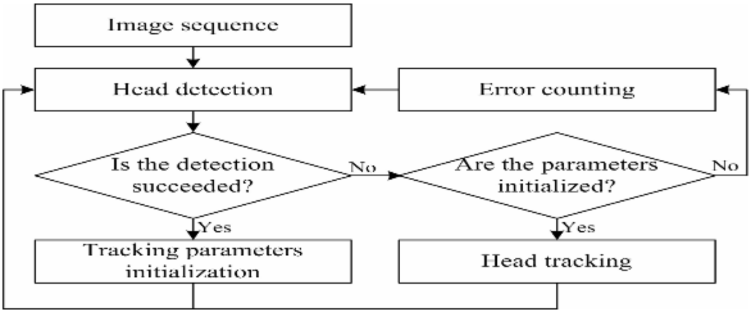


Image sequence
Head detection
Error counting
Is the detection succeeded?
No
Are the parameters initialized?
No
Yes
Yes
Tracking parameters initialization
Head tracking

Fig. 3. Flow chart of the VOPC system

Initially, the system detects the occupant's head in the given image by using a SVM classifier with head contour model. When the head detection succeeds, the system initializes tracking parameters and repeats the sequence; otherwise, it checks whether the tracking parameters are initialized or not in the previous stage. For the head tracking, it tracks the head by using an inverse compositional image alignment method based on Lucas-Kanade algorithm [6]. However, when the tracking parameters are not initialized, it is counted as an error and returns to the head detection stage. If the number of error becomes larger that the permitted number of errors, the system halts and sends a warning signal.

4.1 Head Detection

Fig. 4 illustrates several steps of how the occupant's head can be detected within the given image. First, we accumulate difference images of the image sequence until the difference value is greater than the predefined threshold as shown in Fig. 4 (b). And then, we apply some binary morphological operations such as dilation and erosion for filling holes and gaps of the image and extract feature points of the occupant from the image, the image becomes Fig. 4 (c) and Fig. 4 (d), respectively. We define the feature points as points on the occupant's contour at intervals of 2.5 degrees. Finally, we try to detect the occupant's head on the feature points by using a head contour model by combining a SVM classifier that was trained using different sizes of head contours as illustrated in Fig. 5 (a) where each head contour model is made up different number of feature points, 97, 81 and 65, respectively.

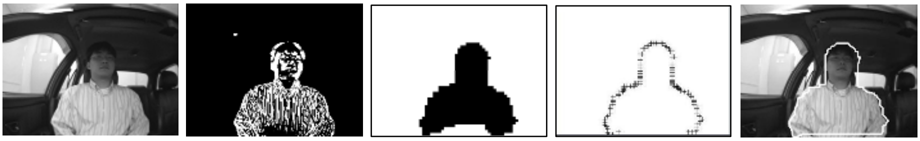

(a) (b) (c) (d) (e)

Fig. 4. Steps of the head detection algorithm

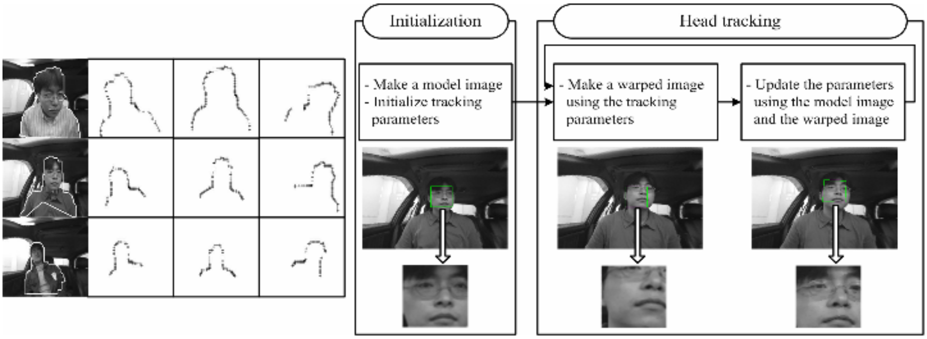

Initialization
Head tracking
- Make a model image
- Initialize tracking parameters
- Make a warped image using the tracking parameters
- Update the parameters using the model image and the warped image

Fig. 5. The head contour model with three different sizes of head (a) and the block diagram of the head tracking algorithm (b)

4.2 Head Tracking

Fig. 6 (b) illustrates the detail schematic diagram of the head tracking. In the initialization part, a model image is extracted from the given image and then it initializes the tracking parameters, such as warping and motion parameters. In the head tracking part, a warped image is made using the tracking parameters and it updates the tracking parameters to minimize the sum of squared error between the model image T and the warped image of input image I as shown in following equation [5, 6]:

$$\sum_{\mathbf{x}}[T(\mathbf{W}(\mathbf{x};\Delta\mathbf{p}))-I(\mathbf{W}(\mathbf{x};\mathbf{p}))]^2 \tag{1}$$

$$\mathbf{W}(\mathbf{x};\mathbf{p})=\begin{pmatrix}1+p_1 & p_3\\ p_2 & 1+p_4\end{pmatrix}\begin{pmatrix}x\\ y\end{pmatrix}+\begin{pmatrix}p_5\\ p_6\end{pmatrix} \tag{2}$$

where $\mathbf{x}=(x,y)^T$ containing the pixel coordinates; $\mathbf{W}(\mathbf{x};\mathbf{p})$ is the parameterized set of allowed warps; $\mathbf{p}$ is a vector of parameters, so $\mathbf{p}=(p_1,p_2,\ldots,p_6)^T$. Such step is iterated until the tracking parameter $\mathbf{p}$ converges. The update rule is given as follow:

$$\Delta\mathbf{p}=H^{-1}\sum_{\mathbf{x}}\left[\nabla T\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]^T[I(\mathbf{W}(\mathbf{x};\mathbf{p})-T(\mathbf{x})] \tag{3}$$

$$H=\sum_{\mathbf{x}}\left[\nabla T\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]^T\left[\nabla T\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right] \tag{4}$$

where H is Hessian matrix. Since we can calculate some parameters before tracking sequence, it allows the fast iteration and leads to the real time basis tracking of the object.

5 Experiments

The performance of occupant classification system was evaluated with our stereo image database acquired within the experimental car. The stereo images totaled 802 and the resolution of each image was 320x240. We used down-sampled images (resolution: 80x60) to reduce the dimension of the image. We randomly divided the data sets into a training set (50%) and a test set (50%) and used the public domain implementation of SVM, called LibSVM and two standard kernels [8]. Table 2 (a) shows average results of occupant classification. In case of RBF kernel, the correction rate was 96.57%, and in case of polynomial kernel, it was, 96.43% for each stage. Table 2 (b) shows results of occupant' pose recognition. The experiment was accomplished by using two image sequences, and the sequences consist of 608, 411 frames, respectively. We counted it as a success case if any one of the two algorithms recognize the occupant' head, as shown in Fig. 1 (b).

Table 1. Occupant classification rate for each kernels (a) and pose recognition result of two image sequences (b)

(a)

Kernel	Classification rate
Polynomial	96.57%
RBF	96.43%

(b)

	Recognition rate
Sequence set I	83.11%
Sequence set II	83.42%

6 Conclusions and Future Work

We proposed a vision based automatic system that consists of occupant classification system and occupant pose recognition system to control triggering and intensity of airbag deployment, respectively. To classify the occupant, 2-stage SVM classifier is used. To recognize the occupant pose, we used two algorithms in parallel: one is the head detection algorithm based on the head contour model and the other is the head tracking algorithm. We tested the system with our stereo image database acquired within the experimental car. Results show that the performance of the system is satisfactory, suggesting that the vision based airbag control has a potential. We plan to carry out further research for solving an occlusion problem caused by the occupant' arm, hat, and so on.

Acknowledgments. This work was supported by Hyundai Autonet Co and 21st century Frontier Program.

References

1. http://www.nhtsa.dot.gov
2. V. N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, NY, USA, 1995
3. H. –G. Lee et al., Occupant Classification for Smart Airbag Using Stereovision and Support Vector Machines, *Springer Lecture Notes in Computer Science*, Vol. 3173, pp.642-647, 2004
4. M. W. Lee, Cohen I., S. K. Jung, Particle filter with analytical inference for human body tracking, *Motion and Video Computing, Workshop on.*, pp.159-165, 2002
5. B. Lucas, T. Kanade, An Iterative Image Registration Technique with an Application to Stereovision, *Proceedings of IJCAI 81*, pp.674-679, 1981
6. S. Baker, I. Mattews, Lucas-Kanade 20 Years On: A Unifying Framework: Part 1. Technical Report, CMU-RI-TR-02-16, Carnegie Mellon University Robotics Institute, 2002
7. C. L. Zitnick, T. Kanade, A Cooperative Algorithm for Stereo Matching and Occlusion Detection, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 22, Issue 7, pp.675-684, 2000
8. http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html

A Wiener Neuronal Model with Refractoriness*

Virginia Giorno[1], Amelia G. Nobile[1], and Luigi M. Ricciardi[2]

[1] Dipartimento di Matematica e Informatica, Università di Salerno,
Via Ponte don Melillo, 84084 Fisciano (SA), Italy
{giorno, nobile}@unisa.it
[2] Dipartimento di Matematica e Applicazioni,
Università di Napoli Federico II, Via Cintia, 80126 Napoli, Italy
luigi.ricciardi@unina.it

Abstract. A mathematical characterization of the membrane potential as an instantaneous return process in the presence of random refractoriness is investigated for the Wiener neuronal model. In the case of constant refractoriness, simple closed form expressions are obtained.

1 Introductory Remarks

The beginning of the history of neuronal models based on diffusion processes can be set at the year 1964, when Gerstein and Mandelbrot [3] postulated that for a number of experimentally monitored neurons subject to spontaneous activity the firing probability density function (pdf) could be modeled by the first passage time (FPT) pdf of a Wiener process.

Let $\{X(t), t \geq 0\}$ be a Wiener process defined in $\mathbb{R}$ and characterized by drift $A_1(x) = \mu$ and infinitesimal variance $A_2(x) = \sigma^2$, with $\mu > 0$ and $\sigma > 0$. As is well-known, the FPT of $X(t)$ through S, with $X(0) = y < S$, is defined as follows:

$$T_y = \inf_{t \geq 0} \{t : X(t) \geq S\}, \qquad X(0) = y < S, \tag{1}$$

and its pdf is:

$$g(S, t \mid y) := \frac{\partial}{\partial t} P(T_y < t) = \frac{S - y}{\sigma \sqrt{2 \pi t^3}} \exp\Big\{ - \frac{(S - y - \mu t)^2}{2 \sigma^2 t} \Big\} \quad (y < S). \tag{2}$$

In the neuronal modeling context, the state S represents the neuron's firing threshold, the FPT through S the firing time and $g(S, t \mid y)$ the firing pdf.

If $\mu > 0$ and $y < S$, the Laplace transform of $g(S, t \mid y)$ is:

$$g_\lambda(S \mid y) := \int_0^{+\infty} e^{-\lambda t} g(S, t \mid y) \, dt = \exp\Big\{ \frac{(S - y) \mu}{\sigma^2} - \frac{S - y}{\sigma^2} \sqrt{\mu^2 + 2 \sigma^2 \lambda} \Big\}. \tag{3}$$

---

[*] This work has been performed under partial support by MIUR (PRIN 2003) and by GNCS-INdAM.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 416–425, 2005.
© Springer-Verlag Berlin Heidelberg 2005

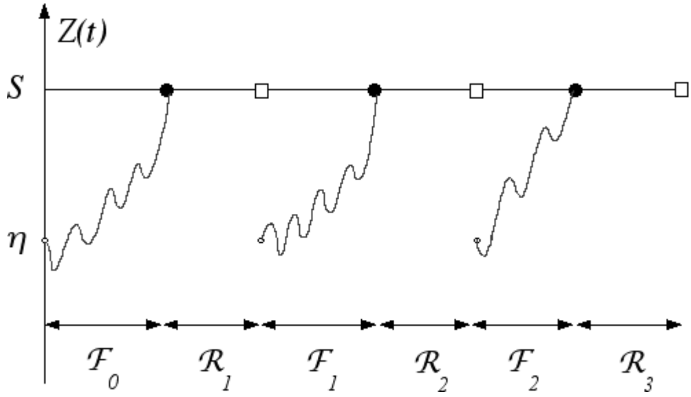


Fig. 1. An hypothetical sample path of $Z(t)$. Circles indicate firing times, and squares ends of refractoriness periods. The reset value has been denoted by η.

Hence, $P(T_y < +\infty) = 1$ and the mean and variance of T_y are given by:

$$t_1(S \mid y) = \frac{S-y}{\mu}\,, \qquad \mathrm{Var}(S \mid y) = \frac{(S-y)\,\sigma^2}{\mu^3}\,\cdot \tag{4}$$

For the Wiener neuronal model (cf., for instance, [5]) the membrane potential can be described by means of a return process $\{Z(t), t \geq 0\}$ defined in $(-\infty, S)$, constructed as follows. Starting at a fixed state $\eta \in (-\infty, S)$ at time zero, a firing takes place when $X(t)$ attains for the first time the firing threshold S, after which the neuron is unable to fire again for a period of refractoriness of random duration. At the end of the period of refractoriness, $Z(t)$ is instantaneously reset to η. The subsequent evolution of the process then goes on as described by $X(t)$, until the boundary S is again reached. A new firing then occurs, followed by the period of refractoriness, and so on.
The process $\{Z(t), t \geq 0\}$ consists of recurrent cycles $\mathcal{F}_0, \mathcal{R}_1, \mathcal{F}_1, \mathcal{R}_2, \mathcal{F}_2, \ldots$ (cf. Fig. 1), each of random duration, where the durations F_i of $\mathcal{F}_i$ $(i = 0, 1, \ldots)$ and the durations of refractory period R_i of $\mathcal{R}_i$ $(i = 1, 2, \ldots)$ are independently distributed random variables. Here, F_i $(i = 0, 1, \ldots)$ denotes the time interval elapsing between the i-th reset of the membrane potential at the value η and the $(i+1)$-th FPT from η to S. Instead, R_i $(i = 1, 2, \ldots)$ indicates the duration of i-th refractory period. The random variables $F_0, F_1, \ldots$ will be assumed to be independent and identically distributed (iid), each with pdf $g(S, t \mid \eta)$ depending only on the length of the corresponding firing interval. Furthermore, we assume that $R_1, R_2, \ldots$ are iid random variables, each with pdf $\varphi(t)$ depending only on the duration of the refractory period.

Let $\{M(t), t \geq 0\}$ be the random process representing the number of firings released by the neuron up to time t and, for $\eta \in (-\infty, S)$, let

$$q_k(t \mid \eta) = P\{M(t) = k \mid Z(0) = \eta\} \qquad (k = 0, 1, \ldots) \tag{5}$$

be the probability of occurrence of k firings up to time t. Then (cf. [6]),

$$\begin{aligned}
q_0(t \mid \eta) &= 1 - \int_0^t g(S, \tau \mid \eta)\, d\tau, \\
q_1(t \mid \eta) &= g(S, t \mid \eta) * \varphi(t) * \Big[1 - \int_0^t g(S, \tau \mid \eta)\, d\tau\Big] \\
&\quad + g(S, t \mid \eta) * \Big[1 - \int_0^t \varphi(\tau)\, d\tau\Big], \\
q_k(t \mid \eta) &= \big[g(S, t \mid \eta) * \varphi(t)\big]^{(k)} * \Big[1 - \int_0^t g(S, \tau \mid \eta)\, d\tau\Big] \\
&\quad + g(S, t \mid \eta) * \big[\varphi(t) * g(S, t \mid \eta)\big]^{(k-1)} * \Big[1 - \int_0^t \varphi(\tau)\, d\tau\Big] \\
&\qquad (k = 2, 3, \ldots),
\end{aligned} \tag{6}$$

where $(*)$ means convolution and exponent (r) indicates (r)-fold convolution.

For the Wiener process, in Section 2 the probabilities of occurrence of multiple firings up to time t are calculated and exact formulas for the first two moments of the number of firings released by the neuron up to time t are given. Furthermore, the interspike pdf is determined for any pressigned pdf of the refractoriness period. Finally, in Section 3 the case of constant refractoriness is analyzed.

2 Effect of Refractoriness

We shall now provide a description of the number of firings released by the neuron up to time t and of the interspike pdf for the Wiener neuronal model in the presence of random refractoriness periods.

Theorem 1. *The probability that zero firings occur up to time t is:*

$$q_0(t \mid \eta) = \frac{1}{2}\left[1 + \mathrm{Erf}\Big(\frac{S - \eta - \mu\, t}{\sigma\sqrt{2\,t}}\Big) - \exp\Big\{\frac{2\,\mu\,(S - \eta)}{\sigma^2}\Big\}\, \mathrm{Erfc}\Big(\frac{S - \eta + \mu\, t}{\sigma\sqrt{2\,t}}\Big)\right], \tag{7}$$

where

$$\mathrm{Erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-z^2}\, dz, \qquad \mathrm{Erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{+\infty} e^{-z^2}\, dz \qquad (x \in \mathbb{R})$$

denote the error function and the complementary error function, respectively.

Proof. We evaluate $q_0(t \mid \eta)$ from the first of (6). To this purpose, we note that

$$\int_0^t g(S, \tau \mid y)\, d\tau + \int_{-\infty}^S \alpha(x, t \mid y)\, dx = 1 \qquad (y < S), \tag{8}$$

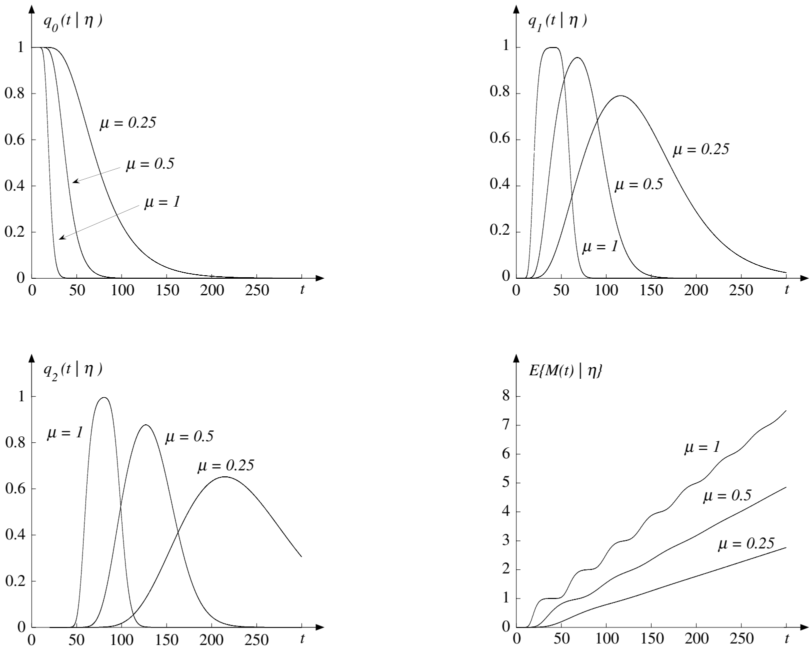


Fig. 2. Probabilities $q_0(t \,|\, \eta)$, $q_1(t \,|\, \eta)$, $q_2(t \,|\, \eta)$ and mean $E\{M(t) \,|\, \eta\}$ are plotted as function of t for $S = -50$, $\eta = -70$, $\sigma^2 = 1$, $E(R) \equiv \zeta = 20$ and $\mu = 0.25, 0.5, 1$

where (see, for instance, [1])

$$\alpha(x,t \mid y) = \frac{1}{\sigma\sqrt{2\,\pi\,t}} \left[\exp\Bigl\{-\,\frac{(x-y-\mu\,t)^2}{2\,\sigma^2\,t}\Bigr\} - \exp\Bigl\{\frac{2\,\mu\,(S-y)}{\sigma^2} - \frac{(x+y-2\,S-\mu\,t)^2}{2\,\sigma^2\,t}\Bigr\}\right] \qquad (x,\ y < S) \qquad (9)$$

denotes the transition pdf of the Wiener process in the presence of an absorbing boundary at S. Making use of (9), from (8) one has:

$$\int_0^t g(S,\tau \mid y)\,d\tau = 1 - \int_{-\infty}^S \alpha(x,t \mid y)\,dx = \frac{1}{2}\left[\mathrm{Erfc}\Bigl(\frac{S-y-\mu\,t}{\sigma\sqrt{2\,t}}\Bigr) + \exp\Bigl\{\frac{2\,\mu\,(S-y)}{\sigma^2}\Bigr\}\,\mathrm{Erfc}\Bigl(\frac{S-y+\mu\,t}{\sigma\sqrt{2\,t}}\Bigr)\right] \qquad (y < S). \qquad (10)$$

Hence, (7) follows from the first of (6) making use of (10) with $y = \eta$. □

Theorem 2. *The probabilities of occurrence of k firings up to time t are:*

$$q_1(t \mid \eta) = \frac{1}{2}\left[\mathrm{Erfc}\Bigl(\frac{S-\eta-\mu\,t}{\sigma\sqrt{2\,t}}\Bigr) + \exp\Bigl\{\frac{2\,\mu\,(S-\eta)}{\sigma^2}\Bigr\}\,\mathrm{Erfc}\Bigl(\frac{S-\eta+\mu\,t}{\sigma\sqrt{2\,t}}\Bigr)\right]$$

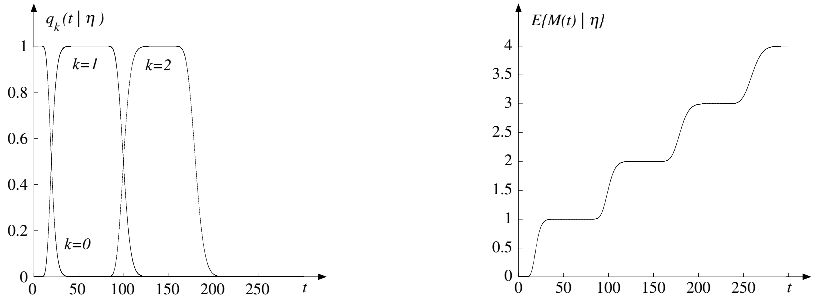


Fig. 3. Probabilities $q_k(t\,|\,\eta)$ $(k=0,1,2)$ and mean $E\{M(t)\,|\,\eta\}$ are plotted as function of t for $S=-50$, $\eta=-70$, $\mu=1$, $\sigma^2=1$ and $E(R)=60$

$$-\frac{1}{2}\,\varphi(t) * \left[\mathrm{Erfc}\Big(\frac{2\,(S-\eta)-\mu\,t}{\sigma\,\sqrt{2\,t}}\Big) + \exp\Big\{\frac{4\,\mu\,(S-\eta)}{\sigma^2}\Big\}\,\mathrm{Erfc}\Big(\frac{2\,(S-\eta)+\mu\,t}{\sigma\,\sqrt{2\,t}}\Big)\right], \tag{11}$$

$$\begin{aligned} q_k(t\mid\eta) &= \frac{1}{2}\,\big[\varphi(t)\big]^{(k-1)} * \left[\mathrm{Erfc}\Big(\frac{k\,(S-\eta)-\mu\,t}{\sigma\,\sqrt{2\,t}}\Big) + \exp\Big\{\frac{2\,\mu\,k\,(S-\eta)}{\sigma^2}\Big\}\right.\\ &\quad\left.\times\mathrm{Erfc}\Big(\frac{k\,(S-\eta)+\mu\,t}{\sigma\,\sqrt{2\,t}}\Big)\right] - \frac{1}{2}\,\big[\varphi(t)\big]^{(k)} * \left[\mathrm{Erfc}\Big(\frac{(k+1)\,(S-\eta)-\mu\,t}{\sigma\,\sqrt{2\,t}}\Big)\right.\\ &\quad\left.+\exp\Big\{\frac{2\,\mu\,(k+1)\,(S-\eta)}{\sigma^2}\Big\}\,\mathrm{Erfc}\Big(\frac{(k+1)\,(S-\eta)+\mu\,t}{\sigma\,\sqrt{2\,t}}\Big)\right] \qquad (k=2,3,\ldots). \end{aligned}$$

Proof. Let

$$\pi_k(\lambda\mid\eta) := \int_0^{+\infty} e^{-\lambda t}\, q_k(t\mid\eta)\, dt \qquad (\lambda>0) \tag{12}$$

be the Laplace transform of $q_k(t\mid\eta)$. Denoting by $\Phi(\lambda)$ the Laplace transform of the refractoriness pdf $\varphi(t)$, from the second and the third of (6) we have:

$$\begin{aligned} \sum_{j\ge k} \pi_j(\lambda\mid\eta) &= \frac{1}{\lambda}\,\Big[g_\lambda(S\mid\eta)\Big]^k\,\Big[\Phi(\lambda)\Big]^{k-1}\\ &= \frac{1}{\lambda}\,g_\lambda(k\,S\mid k\,\eta)\,\Big[\Phi(\lambda)\Big]^{k-1} \qquad (k=1,2,\ldots), \end{aligned} \tag{13}$$

where the last identity follows from (3). Taking the inverse Laplace transforms of (13), one is lead to:

$$\sum_{j\ge k} q_j(t\mid\eta) = \begin{cases} \displaystyle\int_0^t g(S,\tau\mid\eta)\,d\tau, & k=1\\[2ex] \big[\varphi(t)\big]^{(k-1)} * \displaystyle\int_0^t g(k\,S,\tau\mid k\,\eta)\,d\tau, & k=2,3,\ldots. \end{cases} \tag{14}$$

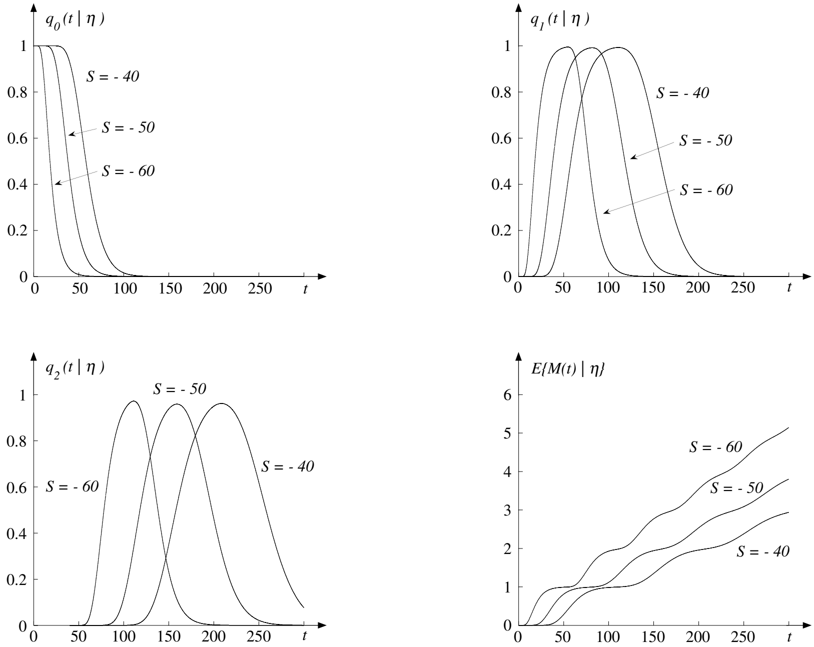


Fig. 4. Probabilities $q_0(t\,|\,\eta)$, $q_1(t\,|\,\eta)$, $q_2(t\,|\,\eta)$ and mean $E\{M(t)\,|\,\eta\}$ are plotted as function of t for $\eta = -70$, $\mu = 0.5$, $\sigma^2 = 1$, $E(R) = 40$ and $S = -60, -50, -40$

Finally, recalling that

$$q_k(t \mid \eta) = \sum_{j \geq k} q_j(t \mid \eta) - \sum_{j \geq k+1} q_j(t \mid \eta),$$

and making use of (10) and (14), relations (11) follow. □

Probabilities $q_k(t \mid \eta)$ can be used to evaluate the explicit expressions of the first two moments of the number of firings released by the neuron up to time t. Indeed, by virtue of (10) and (14), one obtains:

$$\begin{aligned} E\big\{M(t) \mid \eta\big\} &:= \sum_{k \geq 1} k\, q_k(t \mid \eta) \equiv \sum_{k \geq 1} \sum_{j \geq k} q_j(t \mid \eta) \\ &= \frac{1}{2} \Bigg\{ \mathrm{Erfc}\Big(\frac{S-\eta-\mu\, t}{\sigma\sqrt{2\,t}}\Big) + \exp\Big\{\frac{2\,\mu\,(S-\eta)}{\sigma^2}\Big\}\, \mathrm{Erfc}\Big(\frac{S-\eta+\mu\, t}{\sigma\sqrt{2\,t}}\Big) \\ &\quad + \sum_{k \geq 2} \big[\varphi(t)\big]^{(k-1)} * \Bigg[\mathrm{Erfc}\Big(\frac{k\,(S-\eta)-\mu\, t}{\sigma\sqrt{2\,t}}\Big) \\ &\quad + \exp\Big\{\frac{2\,\mu\, k\,(S-\eta)}{\sigma^2}\Big\}\, \mathrm{Erfc}\Big(\frac{k\,(S-\eta)+\mu\, t}{\sigma\sqrt{2\,t}}\Big)\Bigg]\Bigg\}, \end{aligned}$$

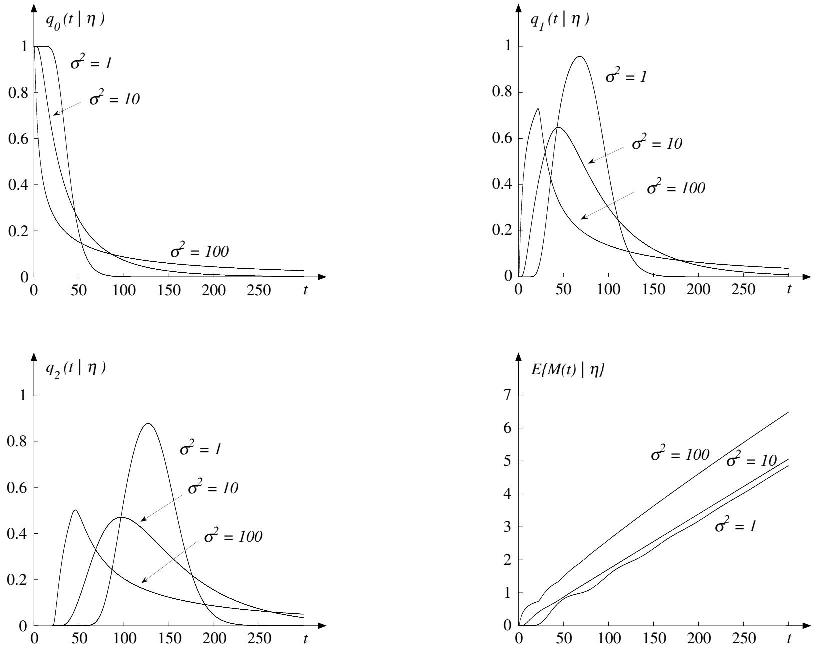


Fig. 5. Probabilities $q_0(t \mid \eta)$, $q_1(t \mid \eta)$, $q_2(t \mid \eta)$ and mean $E\{M(t) \mid \eta\}$ are plotted as function of t for $\eta = -70$, $S = -50$, $\mu = 0.5$, $E(R) = 20$ and $\sigma^2 = 1, 10, 100$

$$\begin{aligned} E\big\{[M(t)]^2 \mid \eta\big\} &:= \sum_{k \geq 1} k^2\, q_k(t \mid \eta) \equiv \sum_{k \geq 1} (2\,k-1) \sum_{j \geq k} q_j(t \mid \eta) \\ &= \frac{1}{2} \bigg\{ \mathrm{Erfc}\Big(\frac{S-\eta-\mu\,t}{\sigma\sqrt{2\,t}}\Big) + \exp\Big\{\frac{2\,\mu\,(S-\eta)}{\sigma^2}\Big\}\, \mathrm{Erfc}\Big(\frac{S-\eta+\mu\,t}{\sigma\sqrt{2\,t}}\Big) \\ &\quad + \sum_{k \geq 2} (2\,k-1)\, \big[\varphi(t)\big]^{(k-1)} * \bigg[\mathrm{Erfc}\Big(\frac{k\,(S-\eta)-\mu\,t}{\sigma\sqrt{2\,t}}\Big) \\ &\quad + \exp\Big\{\frac{2\,\mu\,k\,(S-\eta)}{\sigma^2}\Big\}\, \mathrm{Erfc}\Big(\frac{k\,(S-\eta)+\mu\,t}{\sigma\sqrt{2\,t}}\Big)\bigg]\bigg\}. \end{aligned} \tag{15}$$

As proved in [4], for large t, the mean and variance of the number of firing released by the neuron up to time t are approximatively linear function of t.

Let now $I_0, I_1, I_2, \ldots$ denote the random variables describing the interspike intervals, with I_0 representing the time of occurrence of the first firing and I_k $(k = 1, 2, \ldots)$ the duration of the time interval elapsing between k-th and $(k+1)$-th firing. Furthermore, let $\gamma_k(t)$ denote the pdf of I_k $(k = 0, 1, \ldots)$. Therefore, the pdf of I_0 is $\gamma_0(t) \equiv g(S, t \mid \eta)$ and the interspike intervals $I_1, I_2, \ldots$ are iid random variables having pdf (cf. [2]):

$$\gamma(t) \equiv \gamma_k(t) = \int_0^t \varphi(\vartheta)\, g(S, t-\vartheta \mid \eta)\, d\vartheta \qquad (k = 1, 2, \ldots), \tag{16}$$

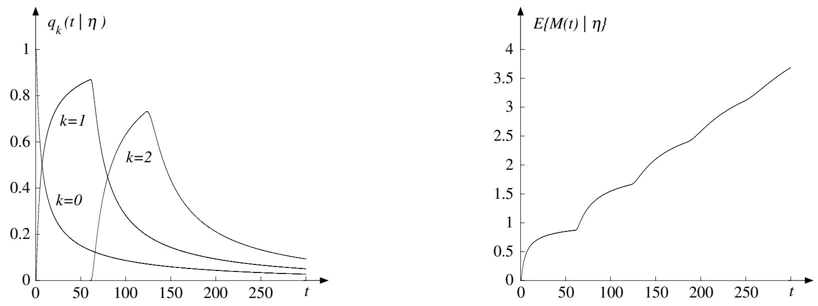


Fig. 6. Probabilities $q_k(t \mid \eta)$ $(k = 0, 1, 2)$ and mean $E\{M(t) \mid \eta\}$ are plotted as function of t for $S = -50$, $\eta = -70$, $\mu = 0.5$, $\sigma^2 = 100$ and $E(R) = 60$

where g is given in (2). Hence, by virtue of (4) and (16), the mean and the variance of the interspike intervals $I_1, I_2, \ldots$ are given by:

$$E(I) = \frac{S-\eta}{\mu} + E(R), \qquad \mathrm{Var}(I) = \frac{(S-\eta)\,\sigma^2}{\mu^3} + \mathrm{Var}(R), \tag{17}$$

where $E(R)$ and $\mathrm{Var}(R)$ denote the mean and variance of refractory periods.

3 Constant Refractoriness

We now assume that after each firing a refractoriness period of constant duration ζ occurs. Hence, the pdf of the refractoriness period is given by:

$$\varphi(t) = \delta\Big(t - \zeta\Big), \tag{18}$$

where $\delta(t)$ is the Dirac delta function. Since $[\varphi(t)]^{(k)} = \delta\big(t - k\,\zeta\big)$, from (11) then follows:

$$\begin{aligned} q_k(t \mid \eta) = {} & \frac{1}{2}\, H\Big[t - (k-1)\,\zeta\Big] \left\{ \mathrm{Erfc}\left(\frac{k\,(S-\eta) - \mu\,\big[t - (k-1)\,\zeta\big]}{\sigma\,\sqrt{2\,\big[t - (k-1)\,\zeta\big]}} \right) \right. \\ & \left. + \exp\left\{ \frac{2\,\mu\,k\,(S-\eta)}{\sigma^2} \right\} \mathrm{Erfc}\left(\frac{k\,(S-\eta) + \mu\,\big[t - (k-1)\,\zeta\big]}{\sigma\,\sqrt{2\,\big[t - (k-1)\,\zeta\big]}} \right) \right\} \\ & - \frac{1}{2}\, H\Big(t - k\,\zeta\Big) \left[\mathrm{Erfc}\left(\frac{(k+1)\,(S-\eta) - \mu\,\big(t - k\,\zeta\big)}{\sigma\,\sqrt{2\,\big(t - k\,\zeta\big)}} \right) \right. \\ & \left. + \exp\left\{ \frac{2\,\mu\,(k+1)\,(S-\eta)}{\sigma^2} \right\} \mathrm{Erfc}\left(\frac{(k+1)\,(S-\eta) + \mu\,\big(t - k\,\zeta\big)}{\sigma\,\sqrt{2\,\big(t - k\,\zeta\big)}} \right) \right] \\ & (k = 1, 2, \ldots), \end{aligned} \tag{19}$$

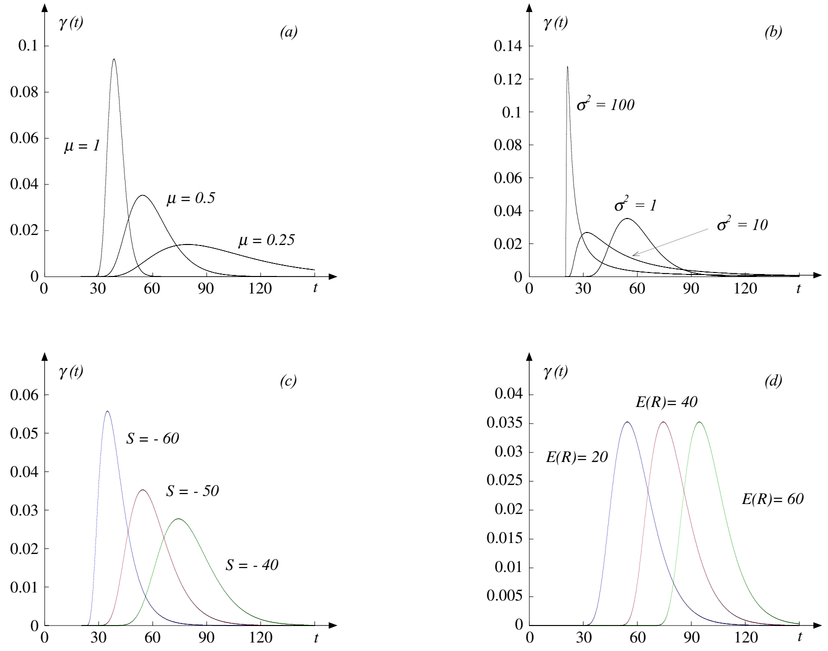


Fig. 7. The interspike pdf $\gamma(t)$ is plotted as function of t: *(a)* for $\eta = -70, S = -50$, $\sigma^2 = 1$, $E(R) = 20$ and $\mu = 0.25, 0.5, 1$; *(b)* for $\eta = -70$, $S = -50$, $\mu = 0.5$, $E(R) = 20$ and $\sigma^2 = 1, 10, 100$; *(c)* for $\eta = -70$, $\mu = 0.5$, $\sigma^2 = 1$, $E(R) = 20$ and $S = -60, -50, -40$; *(d)* for $\eta = -70$, $S = -50$, $\mu = 0.5$, $\sigma^2 = 1$ and $E(R) = 20, 40, 60$

where $H(t)$ denotes the Heaviside unit step function.

The first two moments of the number of firings released by the neuron up to time t are then obtained from (15):

$$E\{M(t) \mid \eta\} = \frac{1}{2} \sum_{k=0}^{\lfloor t/\zeta \rfloor} \Bigg[\mathrm{Erfc}\bigg(\frac{(k+1)\,(S-\eta) - \mu\,(t - k\,\zeta)}{\sigma\sqrt{2\,(t-k\,\zeta)}}\bigg) + \exp\Big\{\frac{2\,\mu\,(k+1)\,(S-\eta)}{\sigma^2}\Big\}\,\mathrm{Erfc}\bigg(\frac{(k+1)\,(S-\eta) + \mu\,(t - k\,\zeta)}{\sigma\sqrt{2\,(t-k\,\zeta)}}\bigg)\Bigg], \tag{20}$$

$$E\{[M(t)]^2 \mid \eta\} = \frac{1}{2} \sum_{k=0}^{\lfloor t/\zeta \rfloor} (2\,k+1) \Bigg[\mathrm{Erfc}\bigg(\frac{(k+1)\,(S-\eta) - \mu\,(t - k\,\zeta)}{\sigma\sqrt{2\,(t-k\,\zeta)}}\bigg) + \exp\Big\{\frac{2\,\mu\,(k+1)\,(S-\eta)}{\sigma^2}\Big\}\,\mathrm{Erfc}\bigg(\frac{(k+1)\,(S-\eta) + \mu\,(t - k\,\zeta)}{\sigma\sqrt{2\,(t-k\,\zeta)}}\bigg)\Bigg]$$

where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x.

The pdf of the interspike intervals $I_1, I_2, \ldots$, obtained from (16), is:

$$\gamma(t) = H\Big(t-\zeta\Big)\,\frac{S-\eta}{\sigma\,\sqrt{2\,\pi\,(t-\zeta)^3}}\;\exp\bigg\{-\,\frac{\big[S-\eta-\mu\,(t-\zeta)\big]^2}{2\,\sigma^2\,(t-\zeta)}\bigg\}, \tag{21}$$

so that the mean and variance are:

$$E(I) = \zeta + \frac{S-\eta}{\mu}\,, \qquad \mathrm{Var}(I) = \frac{\sigma^2\,(S-\eta)}{\mu^3}\,\cdot \tag{22}$$

Figures 2÷6 show the probabilities $q_0(t\,|\,\eta)$, $q_1(t\,|\,\eta)$, $q_2(t\,|\,\eta)$, evaluated via (7) and (19), and the mean $E\{M(t)\,|\,\eta\}$, obtained from the first of (20), as function of t for various choices of μ, σ^2, S, η and of $E(R) \equiv \zeta$.

In Fig. 2 $S=-50$, $\eta=-70$, $\sigma^2=1$ and $E(R)=20$, so that $E(R) \equiv t_1(S\,|\,\eta)$ if $\mu=1$ and $E(R) < t_1(S\,|\,\eta)$ if $\mu=0.25, 0.5$. Instead, in Fig. 3 $E(R) > t_1(S\,|\,\eta)$ for $S=-50$, $\eta=-70$, $\mu=1$, $\sigma^2=1$ and $E(R)=60$. Fig. 2 and Fig. 3 exibit a drastically different behavior as $E(R)$ increases.

Fig. 4, instead, refers to the case $\eta=-70$, $\mu=0.5$, $\sigma^2=1$, $E(R)=40$. Then, $E(R) > t_1(S\,|\,\eta)$ if $S=-60$, $E(R)=t_1(S\,|\,\eta)$ if $S=-50$ and $E(R) < t_1(S\,|\,\eta)$ if $S=-40$.

In Fig. 5 one has $E(R) < t_1(S\,|\,\eta)$ for $\eta=-70$, $S=-50$, $\mu=0.5$, $E(R)=20$ and $\sigma^2=1, 10, 100$, whereas in Fig. 6 it is $E(R) > t_1(S\,|\,\eta)$ for $S=-50, \eta=-70$, $\mu=0.5$, $\sigma^2=100$ and $E(R)=60$. In the case $\sigma^2=100$, Fig. 5 and Fig. 6 exibit a different behavior as $E(R)$ increases.

Finally, in Fig. 7 the interspike pdf $\gamma(t)$ is plotted as function of t for different choices of parameters μ, σ^2, S, η and $E(R)$.

Extensions to random refractoriness for other diffusion neuronal models will be the object of future works.

References

1. Cox, D.R. and Miller, H.D.: *The Theory of Stochastic Processes.* Methuen, London (1970)
2. Esposito, G., Giorno, V., Nobile, A.G., Ricciardi, L.M., Valerio, C.: Interspike analysis for a single neuron's activity in presence of refractoriness. In Trappl, R. (ed.): Cybernetics and Systems. Vol. 1. Austrian Society for Cybernetics Studies, Vienna (2004) 199-204
3. Gerstein, G.L., Mandelbrot, B.: Random walk models for the spike activity of a single neuron. Biophys. J. 4 (1964) 41-68
4. Giorno, V., Nobile, A.G. and Ricciardi, L.M.: On the moments of firing numbers in diffusion neuronal models with refractoriness. In: Mira, J. and Alvarez, J.R. (eds): IWINAC 2005. LNCS 3561 Springer-Verlag (2005) 188-196
5. Ricciardi, L.M., Di Crescenzo, A., Giorno, V., Nobile, A.G.: An outline of theoretical and algorithmic approaches to first passage time problems with applications to biological modeling. Math. Japonica 50, No. 2 (1999) 247-322
6. Ricciardi, L.M., Esposito, G., Giorno, V., Valerio, C.: Modeling Neuronal Firing in the Presence of Refractoriness. In Mira, J., Alvarez, J.R. (eds): Computational Methods in Neural Modeling. IWANN 2003. LNCS 2686 Springer-Verlag (2003) 1-8

On Myosin II Dynamics: From a Pulsating Ratchet to a Washboard Potential*

A. Buonocore, L. Caputo, E. Pirozzi, and L.M. Ricciardi

Dipartimento di Matematica e Applicazioni, Università di Napoli Federico II,
Via Cintia, I-80126 Napoli, Italy
{aniello.buonocore, enrica.pirozzi, luigi.ricciardi}@unina.it
luigia.caputo@dma.unina.it

Abstract. As a model of Brownian motor, we consider the motion of particles in an asymmetric, single-well, periodic potential undergoing half-period shifts driven by two Poisson processes. Probability currents and stopping force are explicitly obtained as a function of the model parameters, and use of the notion of driving effective potential is made to bridge the present model with our previous works involving washboard potentials.

1 Introduction

In the present paper we resort to an idea originally proposed in [4], and re-implemented in [11], to provide an explanation of a possible genesis of the washboard potential phenomenologically introduced by us in previous papers (see [2] and its re-interpretation [3]). Namely, we show that the presence of two equally periodic potentials, shifted by a half-period and alternating in time according to a Poisson distribution, as far as the particles' dynamics is concerned, is equivalent to a washboard potential. Under particular conditions, the period of such potential turns out to be a half of that of the two alternating potentials. We are thus led to obtain a unique periodic potential modified by means a linear term, such as the one instrumentally hypothesized in [3] without any speculation on its origin. This resulting potential identifies with the "effective driving potential" defined in [16] and therein determined by means of a time series yielded by simulations of trajectories that are solutions of a Langevin stochastic differential equation.

Borrowing the framework and some of the technical tools of [16], hereafter we shall provide a quantitative analysis of the resulting Brownian motor and of the detailed features of the generated washboard potential, with specific reference to experimental data of [9] concerning the displacement of Myosin II (henceforth abridged as "myosin").

2 The Model

As is customary, the point-size particle miming a myosin head is viewed as moving along an axis, say x, subject to a viscous drag force as well as to the random

---

* Work performed under partial support by MIUR (PRIN 2003), by GNCS-INDAM and by Campania Region.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 426–435, 2005.
© Springer-Verlag Berlin Heidelberg 2005

force $(2k_{\mathrm{B}}T/\rho)^{1/2} \times \Lambda(t)$, where $\Lambda(t)$ is a standard Gaussian white noise. Furthermore, the overall effect of the interaction of the particle with the molecules of the actin filament is synthesized into a space-periodic potential whose period is twice the distance L (henceforth the "motor step") between consecutive actin monomers. Under the reasonable assumption of overdamped motion (see, for instance, [1], [13], [3]), the equation of motion for the particle is:

$$\dot{x}(t) = -\frac{U_S'(x) - F_e}{\rho} + \sqrt{\frac{2k_{\mathrm{B}}T}{\rho}}\Lambda(t), \tag{1}$$

where F_e indicates a possibly external extra force, ρ the drag coefficient, T the absolute temperature and k_{B} the Boltzmann constant. Finally, $U_S(x)$ denotes the above $2L$-periodic potential and S is a two-valued random variable, whose values will be henceforth denoted by $s = 1, 2$. Such a variable is related to a fluctuation source and specifies the current chemical state of the actin-myosin system. For $s = 1, 2$, the transition from the chemical state s to the chemical state $3 - s$ is assumed to occur according to a Poisson process of rate a_s. Setting $a = a_1 + a_2$, we then have $P(S = 1) = a_2/a$ and $P(S = 2) = a_1/a$. Here we shall assume that $U_1(x)$ and $U_2(x)$ are asymmetric saw-tooth shaped potentials with their minima at L_A and $L_A + L$ $(0 < L_A < L)$, respectively, and possessing equal depths U_0 (see Fig. 1). Hence:

$$U_s(x + L) = U_{3-s}(x), \qquad \text{for } x \in [0, L] \text{ and } s = 1, 2. \tag{2}$$

As is well known (see, for instance, [14]) the Fokker-Plank differential formulation of (1) is:

$$\begin{cases} \dfrac{\partial p_1(x,t)}{\partial t} = \eta(x,t) + \dfrac{\partial}{\partial x}\left[\dfrac{U_1'(x) - F_e}{\rho} p_1(x,t)\right] + D\dfrac{\partial^2}{\partial x^2} p_1(x,t) \\[2ex] \dfrac{\partial p_2(x,t)}{\partial t} = -\eta(x,t) + \dfrac{\partial}{\partial x}\left[\dfrac{U_2'(x) - F_e}{\rho} p_2(x,t)\right] + D\dfrac{\partial^2}{\partial x^2} p_2(x,t) \end{cases} \tag{3}$$

where $\eta(x,t) = -a_1(x,t)p_1(x,t) + a_2 p_2(x,t)$ is the contribute of the state's transitions and $D = k_{\mathrm{B}}T/\rho$ is the diffusion constant.

We have to look for the solution of the equations (3) with the constraint that the position x of the particle is confined in $[0, 2L]$. Hence, the obtained ($2L$-periodic) solution $p_s(x,t)$ includes the contributions of all positions that are modulo $2L$ congruous to x. For all $x \in [0, 2L]$ and for all integer k, the quantity $p_s(x,t)dx$ $(s = 1, 2)$ then expresses the fraction of particles that at time t are located in the neighborhood of $x + 2kL$, in the chemical state s. Under such reduced dynamics, considerations of physical nature (see, for instance, [12]) indicate that a steady state regime is attained in which the solutions of Eqs. (3) are time-independent: $p_s(x,t) = p_s(x)$. As Fig. 1 shows, four consecutive intervals $1, 2, 3, 4$ are present, such that one of the slopes of the two potentials varies when moving from an interval to the next one. Such intervals have end points $e_0 = 0$, $e_1 = L_A$, $e_2 = L$, $e_3 = L_A + L$, $e_4 = 2L$, and in each of them the functions $U_1'(x)$ and $U_2'(x)$ are constant.

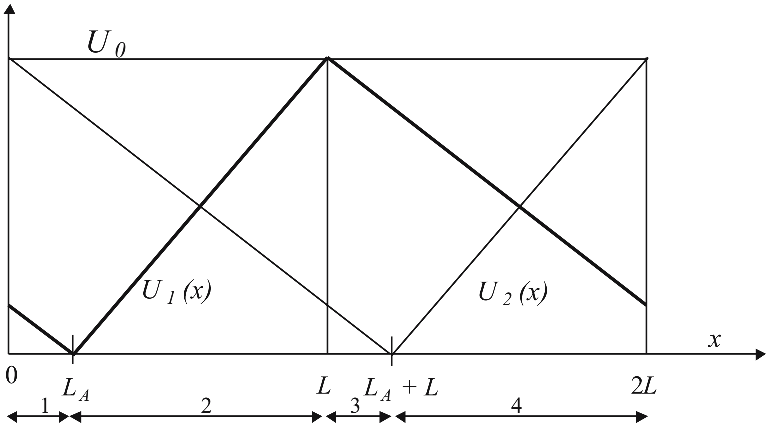


Fig. 1. The figure shows the intervals $1, 2, 3, 4$ partitioning the periodicity interval $[0, 2L]$ of the two saw-tooth potentials $U_1(x)$ and $U_2(x)$. Passing from each interval to the next one, a change occurs of one of the slopes of the potential.

From now on unless differently stated, we shall tacitly assume that indices s and r are such that $s = 1, 2$ and $r = 1, 2, 3, 4$. Denote now $U_s'(x) = m_{sr}$, meaning that x is in the interval r while chemical state is s. Then, Eqs. (3) read:

$$\begin{cases} p_{1r}''(x) = -\dfrac{m_{1r} - F_e}{\rho D} p_{1r}'(x) + \dfrac{a_1}{D} p_{1r}(x) - \dfrac{a_2}{D} p_{2r}(x) \\ \\ p_{2r}''(x) = -\dfrac{m_{2r} - F_e}{\rho D} p_{2r}'(x) - \dfrac{a_1}{D} p_{1r}(x) + \dfrac{a_2}{D} p_{2r}(x). \end{cases} \tag{4}$$

The search of the solution of Eqs. (4) will be performed along the lines indicated in [1]. To this end, with $i = 1, 2, 3, 4$ let λ_{ri} denote the eigenvalues of (4) in the r-th interval. (One of the eigenvalues, say λ_{r4}, can be seen to be equal to zero, so that one is led to a cubic equation for the remaining eigenvalues.) In each interval, we look for a solution of the form

$$p_{sr}(x) = \sum_{i=1}^{4} C_{ri}^{(s)} e^{\lambda_{ri} x}. \tag{5}$$

Note that only 16 out of the 32 coefficients $C_{ri}^{(s)}$ must be specified. Indeed, imposing that (5) satisfies (4), leads to the specification of the ratios $C_{ri}^{(2)}/C_{ri}^{(1)}$ as functions of the previously determined eigenvalues.

Let now

$$J_s(x) = -D \left[\frac{U_s'(x) - F_e}{\rho D} p_s(x) + p_s'(x) \right], \tag{6}$$

denote the probability currents in the s-th chemical state. Then, the unknown 16 coefficients $C_{ri}^{(1)}$ are determined by imposing the 15 continuity conditions ($r = 1, 2, 3$):

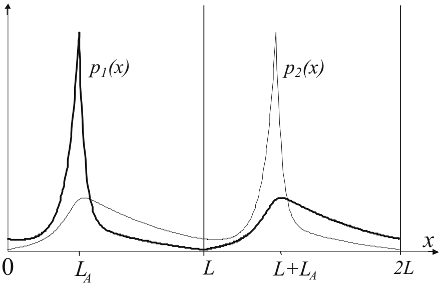


Fig. 2. Plots of probability densities $p_1(x)$ and $p_2(x)$

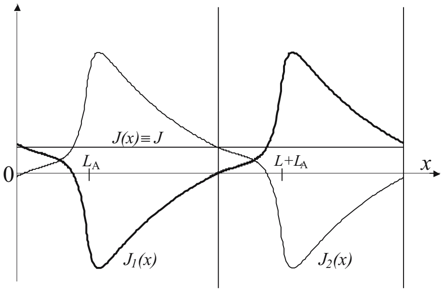


Fig. 3. Probability currents $J_1(x)$ and $J_2(x)$ sum up to yield the constant total probability current J

$$p_s(e_0^+) = p_s(e_4^-), \qquad p_s(e_r^-) = p_s(e_r^+) \tag{7}$$

$$J_1(e_0^+) = J_1(e_4^-), \qquad J_s(e_r^-) = J_s(e_r^+) \tag{8}$$

and the normalization condition

$$\sum_{s=1}^{2} \int_0^{2L} p_s(x)\, dx = 1. \tag{9}$$

We further note that functions $J_s(x)$, can be determined in terms of the $p_{sr}(x)$:

$$J_{sr}(x) = -D \sum_{i=1}^{4} C_{ri}^{(s)} \left[\frac{m_{sr} - F_e}{\rho D} + \lambda_{ri} \right] e^{\lambda_{ri} x}. \tag{10}$$

In summary, in each of the above considered four intervals, probability densities $p_s(x)$ and probability currents $J_s(x)$ are obtained via Eqs. (5) and (6). Note, in addition, that the total probability current $J_1(x) + J_2(x)$ is a constant, hereafter denoted by J.

Fig. 2, for a somewhat arbitrary choice of parameters $a_1, a_2, \rho, T, L, L_A, U_0, F_e$ shows the typical plot of probability densities $p_1(x)$ and $p_2(x)$ obtained via the above procedure in the case $a_1 = a_2$. For the same parametric choice, Fig. 3 shows the plot of probability currents $J_1(x)$ and $J_2(x)$, as well as the total probability current J.

3 Origin of the Washboard Potential

We re-write Eqs. (3) for the stationary regime $\left(\text{i.e. } \dfrac{\partial p_1(x,t)}{\partial t} = \dfrac{\partial p_2(x,t)}{\partial t} = 0 \right)$ and add them up term by term to obtain:

$$\frac{d}{dx} \left[\frac{\widetilde{V}\,'(x) - F_e}{\rho} p(x) \right] + D \frac{d^2}{dx^2} p(x) = 0, \tag{11}$$

where

$$\widetilde{V}\,'(x) = \sum_{s=1}^{2} U_s'(x)\frac{p_s(x)}{p(x)} \tag{12}$$

and $p(x) = p_1(x) + p_2(x)$. Note that $\widetilde{V}\,'(x)$ is $2L$-periodic. Hence, $\widetilde{V}(x)$, that we shall call "effective driving potential" by borrowing a definition given in [16], turns out to be the sum of a $2L$-periodic function, say $U(x)$, and of a linear term:

$$\widetilde{V}(x) = U(x) - F_i \cdot x. \tag{13}$$

Here, F_i has dimension of a force, identifies with the "effective driving force" of [16], and depends on all parameters of the model, including F_e. Such force can be calculated as follows:

$$F_i = -\frac{1}{2L}\int_0^{2L} \widetilde{V}\,'(x)\,dx = -\frac{1}{2L}E\left[\frac{U_S'(X)}{p(X)}\right]. \tag{14}$$

Indeed, since $\widetilde{V}(x)$ is $2L$-periodic, setting $\Delta\widetilde{V} = \widetilde{V}(0) - \widetilde{V}(2L)$ from (13) we have $F_i = \Delta\widetilde{V}/(2L)$. On the other hand, by integration of (12) between 0 and $2L$, we obtain:

$$\Delta\widetilde{V} = -\int_0^{2L} \widetilde{V}\,'(x)\,dx = -\sum_{s=1}^{2}\int_0^{2L} \frac{U_s'(x)}{p(x)}p_s(x)\,dx = -E\left[\frac{U_S'(X)}{p(X)}\right]$$

having made use of the well-known definition

$$E\left[g(X,S)\right] = \sum_{s=1}^{2}\int_0^{2L} g(x,s)p_s(x)\,dx$$

where g is any measurable function.

As for $\widetilde{V}(x)$, it can be obtained, apart from an arbitrary constant, by numerical integration of (12). Hence, from (13) and (14) the periodic component $U(x)$ of $\widetilde{V}(x)$ is immediately obtained. Fig. 4, obtained by taking $\widetilde{V}(0) = [U_1(0) + U_2(0)]/2$, shows plots of $U(x)$ and $\widetilde{V}(x)$ in $[0, 2L]$ for the same choice of parameters as in Fig. 2. Note that the linear behaviors of $\widetilde{V}(x)$ in $[0, L_A]$ is due to being therein $U_1'(x) = U_2'(x)$ so that, by virtue of (12), one has $\widetilde{V}(x) = m_{11} \cdot x$, with $m_{11} < 0$. Hence, due to (13), a similar conclusion follows for the function $U(x)$.

We now recall that $\widetilde{V}\,'(x)$ is $2L$-periodic. However, in the case of symmetric transitions (i.e. $a_1 = a_2$), $\widetilde{V}\,'(x)$ becomes L-periodic, and hence the period of the potential $U(x)$ equals the motor step L. The proof follows in 3 steps.

Step 1. For $x \in [0, L]$ one has:

$$p_s(x + L) = p_{3-s}(x) \quad\Longleftrightarrow\quad a_1 = a_2$$

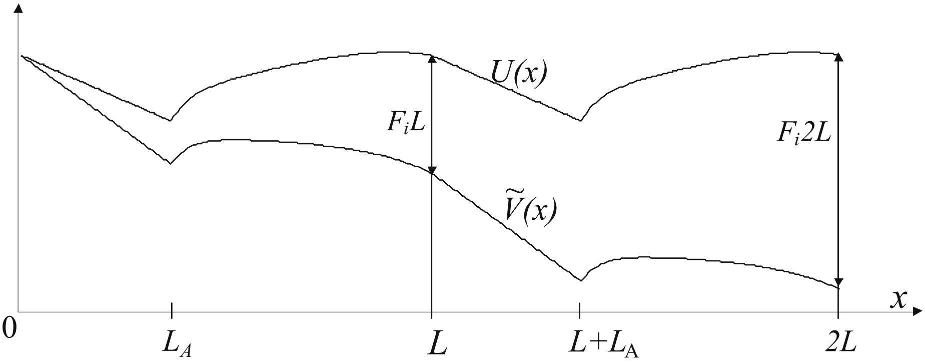


Fig. 4. Plots of the effective driving potential $\widetilde{V}(x)$ and of its periodic component $U(x)$

Indeed, if $a_1 = a_2$, Eq. (2) implies that, apart from a shift of magnitude L, the Brownian particles in the stationary regime are equally distributed in states 1 and 2:

$$p_1(x+L) = p_2(x), \quad p_2(x+L) = p_1(x) \qquad \forall x \in [0, L]\,. \tag{15}$$

Vice versa, if (15) hold, then from Eqs. (3), written for the stationary case, after some algebra one obtains $a_1 = a_2$. Hence, iff state transitions are symmetric, then $p_s(x+L) = p_{3-s}(x)$.

Step 2. We prove that if positive constants α_1 and α_2 exist such that:

$$p_s(x+L) = \alpha_s\, p_{3-s}(x), \quad \text{for } x \in [0, L], \tag{16}$$

then $\alpha_1\alpha_2 = 1$. This follows from (16) after imposing continuity conditions (7), so that

$$p_1(L^-) = p_1(L^+) = \alpha_1 p_2(0^+) = \alpha_1 p_2(2L^-) = \alpha_1\alpha_2 p_1(L^-).$$

Step 3. We show that for $x \in [0, L]$,

$$\widetilde{V}\,'(x+L) = \widetilde{V}\,'(x) \text{ iff } a_1 = a_2.$$

Indeed,

$$\begin{aligned}\widetilde{V}\,'(x+L) = \widetilde{V}\,'(x) &\Leftrightarrow \sum_{s=1}^{2} U'_{3-s}(x)\frac{p_s(x+L)}{p(x+L)} = \sum_{s=1}^{2} U'_s(x)\frac{p_s(x)}{p(x)} \\ &\Leftrightarrow \sum_{s=1}^{2} U'_s(x)\,[p_2(x)p_2(x+L) - p_1(x)p_1(x+L)] = 0 \\ &\Leftrightarrow \frac{p_1(x+L)}{p_2(x+L)} = \frac{p_2(x)}{p_1(x)}. \end{aligned} \tag{17}$$

Hence,

$$\widetilde{V}'(x+L) = \widetilde{V}'(x) \Rightarrow p_1(x+L) = \alpha p_2(x) \text{ and } p_2(x+L) = \alpha p_1(x), \tag{18}$$

with α any positive constant. Applying the result stated in Step 2 to Eqs. (18), it follows $\alpha = \alpha_1 = \alpha_2 = 1$, which by Step 1 implies $a_1 = a_2$. On the other hand, if $a_1 = a_2$, then (15) hold, so that the last equality of (17) holds, which in turn proves the announced L-periodicity of $\widetilde{V}'(x)$.

Note that in the case of symmetric transitions, F_i can be evaluated also as follows:

$$F_i = -\frac{1}{L}\int_0^L \widetilde{V}'(x)\,dx. \tag{19}$$

In conclusion, in the case of symmetric transitions, Eq. (11) describes the stationary regime of the diffusion process responsible for the particle's dynamics under the stochastic differential equation

$$\dot{x}(t) = -\frac{\widetilde{V}'(x) - F_e}{\rho} + \sqrt{\frac{2k_{\mathrm{B}}T}{\rho}}\Lambda(t). \tag{20}$$

Here, differently from Eq. (1), no stochastically alternating drift is present, the entire dynamics being determined exclusively by the washboard potential $\widetilde{V}(x)$ and by the effects of thermal fluctuation. From Eqs. (13) and (20), one sees that the Brownian particle is subject to a diffusion having drift $\left[U'(x) - (F_i + F_e)\right]/\rho$, and hence is subject to a tilted potential $U(x) - (F_i + F_e)x$. The tilt disappears whenever $F = F_i + F_e = 0$, namely when the effective driving force F_i is balanced by an opposite external force F_e. In such a case the total probability current vanishes. The stopping force F_{st}, namely the external force able to stop the net motion of the particle, from Eq. (14) is recognized to be the fixed point of $E[U_S'(X)/p(X)]/(2L)$ viewed as a function of F_e.

4 Numerical Results

The model discussed in the foregoing has been implemented with reference to experimental data provided in [9]. In this work, myosin head displacements have been monitored in the practical absence of external loads ($F_e \simeq 0$). The measurements, performed at environmental temperature $T = 293$ K, yielded a mean dwell time $\hat{\mu} \simeq 5$ ms and forward jump frequency $\hat{p} \in (0.8, 0.9)$. Than the estimated average velocity $\hat{v} = (2(\hat{p} - 1)L/\hat{\mu}$ for $L = 5.5$ nm (as reported, for instance, in [7] e [8]) falls between 6.6×10^{-7} and 9.9×10^{-7} nm/ns. Hence, since the motor step is a half of the periods of $U_1(x)$ and $U_2(x)$, the total probability current of Brownian particles in our model is given by $\hat{J} = \hat{v}/(2L)$, thus falling between 6×10^{-8} and 8×10^{-8} ns^{-1}. In our numerical evaluation, we have chosen ρ =90 pN ns/nm as reported in [5] and also supported by the estimation in [6] for a slightly smaller protein. For the symmetric transition $\overline{a} = a_1 = a_2$ and for different choice of U_0 and L_A we have calculated the total probability

Table 1. For T =293 K, L =5.5 nm, ρ =90 pN ns/nm, $F_e = 0$ and for some choices of U_0, L_A and $\overline{a}$, total probability current J, stopping force F_{st} and effective driving force F_i have been listed

U_0 ($k_\mathrm{B}T$)	L_A (nm)	$\overline{a}$ (ns^{-1})	J (ns^{-1})	F_{st} (pN)	F_i (pN)
30	2.0	1.4×10^{-7}	6.97×10^{-8}	-8.34	0.63
31	2.0	1.4×10^{-7}	6.87×10^{-8}	-8.89	0.86
32	2.0	1.4×10^{-7}	7.14×10^{-8}	-9.40	1.19
33	2.0	1.4×10^{-7}	7.00×10^{-8}	-9.94	1.45
34	2.0	1.4×10^{-7}	6.85×10^{-8}	-10.47	1.90
32	1.0	1.5×10^{-7}	7.24×10^{-8}	-10.46	1.24
32	2.0	1.5×10^{-7}	7.63×10^{-8}	-9.43	1.24
32	3.0	1.5×10^{-7}	7.35×10^{-8}	-8.46	1.16
32	4.0	1.5×10^{-7}	8.00×10^{-8}	-7.69	1.23
32	2.0	1.3×10^{-7}	6.20×10^{-8}	-9.37	1.14
32	2.0	1.4×10^{-7}	7.14×10^{-8}	-9.40	1.19
32	2.0	1.5×10^{-7}	7.63×10^{-8}	-9.43	1.24
32	2.0	1.6×10^{-7}	7.74×10^{-8}	-9.48	1.26

current. As Table 1 shows, the obtained values are compatible with the above mentioned experimental estimations. Table 1 also lists the obtained values of stopping force F_{st} and effective driving force F_i. It is interesting to remark that the obtained values of the total probability current are very close to a half of the chosen transition rates that, for our symmetric transition model, identify with the harmonic mean of the rates, in agreement with a similar result in [4].

Figure 5 shows the calculated behavior of total probability current J as function of external force F_e. The intercepts of the curves with the horizontal axis ($J = 0$) represent the stopping forces. Note that the total probability current stays constant for a wide range of external applied forces, to quite abruptly drop and rapidly change sign, which implies a sudden inversion of the direction of motion.

It is instructive to analyze the implications of model (1) where time alternating potentials $U_1(x)$ and $U_2(x)$ are present, and of the reduced model (20), under symmetric transitions, that involves the unique washboard potential $\widetilde{V}(x)$ given by (13). Denoting by p and μ the theoretical analogue of $\hat{p}$ and $\hat{\mu}$, one has [10]:

$$p = \frac{1}{1 + e^{-(F_i + F_e)L/k_\mathrm{B}T}}. \tag{21}$$

As for μ, we can evaluate it in two ways. The first way is by [10]:

$$\mu = \frac{p}{D} \int_0^L dx \exp\left\{ \frac{\widetilde{V}(x) - F_e \cdot x}{k_\mathrm{B}T} \right\} \int_{x-L}^{x} dy \exp\left\{ -\frac{\widetilde{V}(y) - F_e \cdot y}{k_\mathrm{B}T} \right\}. \tag{22}$$

Note that the arbitrary constant arising from integration in (12) cancels out in (22). The second way is via the equation

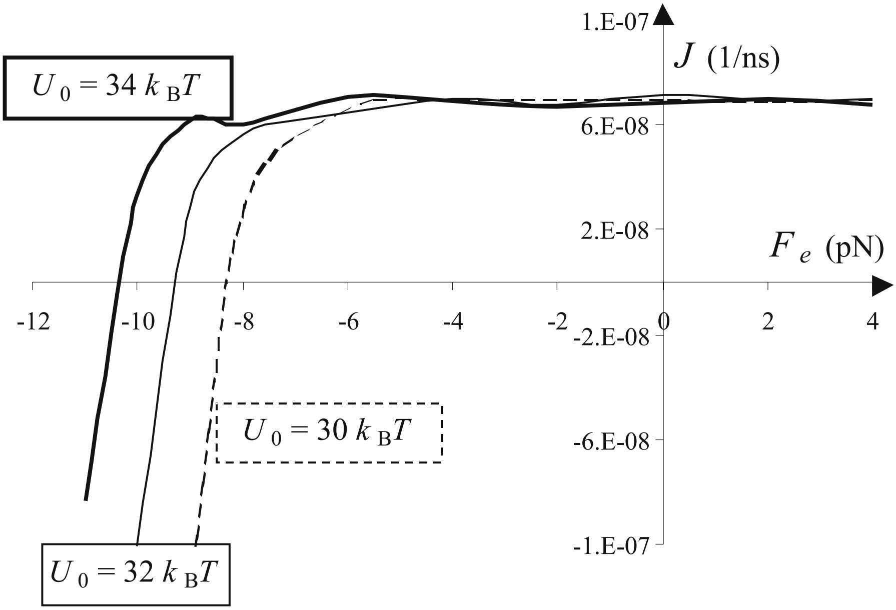


Fig. 5. Total probability current J as a function of external force F_e for the indicated three choices of U_0. Here $L_A = 2$ nm and $\overline{a}$ =1.4×10^{-7} ns^{-1}. The remaining parameters are those of Table 1.

Table 2. For some choices of U_0, L_A and $\overline{a}$, the probability p given by (21) is listed. In columns 5 and 6 the mean dwell time calculated via (22) and (23), respectively, are also listed. The last column lists the depths U_0^* of the wells in the periodic component of the washboard potential $\widetilde{V}(x)$. All parameters have been chosen as in Table 1.

U_0 (k_BT)	L_A (nm)	$\overline{a}$ (ns^{-1})	p	μ (ns) Eq. (22)	μ (ns) Eq. (23)	U_0^* (k_BT)
30	2.0	1.4×10^{-7}	0.701	0.2878×10^{7}	0.2887×10^{7}	13.94
31	2.0	1.4×10^{-7}	0.762	0.3830×10^{7}	0.3818×10^{7}	14.29
32	2.0	1.4×10^{-7}	0.835	0.4639×10^{7}	0.4696×10^{7}	14.57
33	2.0	1.4×10^{-7}	0.878	0.5859×10^{7}	0.5743×10^{7}	14.86
34	2.0	1.4×10^{-7}	0.929	0.6288×10^{7}	0.6265×10^{7}	15.14
32	1.0	1.5×10^{-7}	0.843	0.4741×10^{7}	0.4738×10^{7}	14.66
32	2.0	1.5×10^{-7}	0.844	0.4464×10^{7}	0.4514×10^{7}	14.54
32	3.0	1.5×10^{-7}	0.829	0.4507×10^{7}	0.4479×10^{7}	14.49
32	4.0	1.5×10^{-7}	0.841	0.4073×10^{7}	0.4248×10^{7}	14.34
32	2.0	1.3×10^{-7}	0.825	0.4834×10^{7}	0.4896×10^{7}	14.65
32	2.0	1.4×10^{-7}	0.835	0.4639×10^{7}	0.4696×10^{7}	14.57
32	2.0	1.5×10^{-7}	0.844	0.4464×10^{7}	0.4514×10^{7}	14.54
32	2.0	1.6×10^{-7}	0.848	0.4398×10^{7}	0.4399×10^{7}	14.55

$$\mu = \frac{2p-1}{2J}. \tag{23}$$

This should yield the same value as (22) if models (1) and (20) are truly equivalent as far as the predicted average velocity is concerned. Table 4 evidently supports such an equivalence, as shown by columns 5 and 6 referring to calculations made via (22) and (23), respectively. In the last column of Table 4, the depths U_o^* of the periodic component in $\widetilde{V}(x)$ have been listed. For the considered choices of parameters, this turns out to be very close to the estimated counterparts in [3].

References

1. Astumian D.R. and Bier M. Fluctuation Driven Ratchets: Molecular Motor. *Phys. Rev. Lett.* **72**, 1766–1769 (1994).
2. Buonocore A. and Ricciardi L.M. Exploiting thermal noise for an efficient actomyosin sliding mechanism. *Math. Biosci.* **182**, 135–149 (2003).
3. Buonocore A., Caputo L., Ishii Y., Pirozzi E., Yanagida T. and Ricciardi L.M. A Phenomenological model of Myosin II dynamics in the presence of external loads. *BioSystems* in press. Also in *arXiv:q-bio. BM/0411025 v1* (2004).
4. Chauwin J.F., Ajdari A. and Prost J. Force-Free Motion in Asymmetric Structures: a Mechanism without Diffusive Steps. *Europhys. Lett.* **27**, 421–426 (1994).
5. Esaki S., Ishii Y. and Yanagida T. Model describing the biased Brownian movement of myosin. *Proc. Japan Acad.* **79**, Ser. B, 9–14 (2003).
6. Howard J. *Mechanism of Motor Proteins and the Cytoskeleton.* Sinauer Associates, Inc. Publishers, Sunderland, Massachusetts (2001).
7. Ishii Y. and Yanagida T. Single Molecule Detection in Life Science. *Single Mol.* **1**, 5–16 (2000).
8. Kitamura K., Ishijima A., Tokunaga M. and Yanagida T. Single-Molecule Nanobiotechnology. *JSAP International* **4**, 4–9 (2001).
9. Kitamura K., Tokunaga M., Iwane A.H. and Yanagida T. A single myosin head moves along an actin filament with regular steps of 5.3 nanometers. *Nature* **397**, 129–134 (1999).
10. Lindner B., Kostur M. and Schimanky-geier L. Optimal diffusive transport in a tilted periodic potential. *Fluc. Noise Lett.* **1**, R25–R39 (2001).
11. Makhnovskii Yu. A., Rozenbaum V. M., Yang D.-Y. and Lin S. H. Flashing ratchet model with high efficiency. *Phys. Rev. E* **69**, 021102-1–021102-7 (2004).
12. Reimann P. Brownian motors: noisy transport far from equilibrium. *Phys. Rep.* **361**, 57–265 (2002).
13. Reimann P. and Hänggi P. Introduction to the physics of Brownian motors. *Appl. Phys. A* **75**, 169–178 (2002).
14. Risken H. *The Fokker-Planck Equation: Methods of Solutions and Applications.* Springer-Verlag, Berlin (1989).
15. Sekimoto K. Kinetic Characterization of Heat Bath and the Energetics of Thermal Ratchet Models. *J. Phys. Soc. Jpn.* **66**, 1234–1237 (1997).
16. Wang H. and Oster G. Ratchets, power strokes, and molecular motors. *Appl. Phys. A* **75**, 315–323 (2002).

Feedback Effects in Simulated Stein's Coupled Neurons*

A. Di Crescenzo[1], B. Martinucci[1], and E. Pirozzi[2]

[1] Dipartimento di Matematica e Informatica, Università di Salerno,
Via Ponte don Melillo, I-84084 Fisciano (SA), Italy
adicrescenzo@unisa.it, barbara.martinucci@dma.unina.it
[2] Dipartimento di Matematica e Applicazioni, Università di Napoli Federico II,
Via Cintia, I-80126 Napoli, Italy
enrica.pirozzi@unina.it

Abstract. A network consisting of two Stein-type neuronal units is analyzed under the assumption of a complete interaction between the neurons. The firing of each neuron causes a jump of constant amplitude of the membrane potential of the other neuron. The jump is positive or negative according to whether the firing neuron is excitatory or inhibitory.

Making use of a simulation procedure designed by ourselves, we study the interspike intervals of the two neurons by means of their histograms, of some descriptive statistics and of empirical distribution functions. Furthermore, via the crosscorrelation function, we investigate the synchronization between the neurons firing activity in the special case when one neuron is excitatory and the other is inhibitory.

1 Introduction

The dynamics of a pair of Stein's neuronal units serially connected has been recently analyzed under the hypothesis that they are subject to excitatory and inhibitory stimuli with constant or alternating rates ([6], [7]). Making use of an ad hoc simulation procedure designed by ourselves, various quantitative results have been already obtained. In particular, the existence has been disclosed of an "optimal" value for the amplitude γ of the jumps of the membrane potential of the second unit (the "receiving neuron") attained as effect of the firing of the first unit (the "sending neuron"). We also studied the reaction time, defined as the random time elapsing between a firing of the sending neuron and the subsequent firing of the receiving neuron. The reaction times have been analyzed in [7], under the assumption of constant inhibitory rate, for both constant and alternating excitatory rates, with emphasis on the effects of various choices of γ and of the amplitude of the refractory period.

The model described in [6] and [7] is re-considered here under the novel assumption that a complete interaction between the two neurons exists. The

---

* This work has been performed under partial support by MIUR (cofin 2003) and by G.N.C.S. (INdAM).

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 436–446, 2005.
© Springer-Verlag Berlin Heidelberg 2005

effect of the firing of each neuron is a sudden constant magnitude jump of the membrane potential of the other neuron.

By means of our simulations, we obtain the neurons interspike intervals (ISIs) histograms and their empirical distribution functions. The synchronization of the firing activity of the neurons is studied in the special case in which one neuron is excitatory and the other is inhibitory.

We point out that the analysis of firing activity of coupled neurons under various mutual interactive paradigms is particularly relevant in various respects: for instance, it is known that the cerebellar cortex includes pairs of coupled neurons, which has motivated some previous studies (see, for instance, [4] and [5]). Synchronization will be studied via the firing times crosscorrelation function (see [2]). Furthermore, by means of the autocorrelation function, we have been able to disclose particular patterns of spikes elicited by each single neuron.

2 The Method

In previous papers we analyzed the behavior of two coupled neuronal units that interact according to a sending-receiving model, assuming that an unidirectional connection from the first to the second neuron exists. We now present a statistical analysis of the firing activity of the two neurons under the hypothesis of complete interaction. For each neuronal unit, changes in the membrane potential between two consecutive spikes are described by the Stein's differential equation; however, the further hypothesis is now added that whenever a neuron fires, the membrane potential of the other neuron undergoes a jump of constant magnitude. Let $\{[X_1(t), X_2(t)];\ t \geq 0\}$ be the stochastic process describing the time-evolution of the membrane potential of the pair of neurons between consecutive firings. We consider the following stochastic differential equations:

$$\mathrm{d}X_1(t) = -\frac{1}{\tau}\, X_1(t)\,\mathrm{d}t + \alpha\,\mathrm{d}N_1^+(t) - \beta\,\mathrm{d}N_1^-(t) + \gamma_1\,\mathrm{d}M_2(t) \tag{1}$$

$$\mathrm{d}X_2(t) = -\frac{1}{\tau}\, X_2(t)\,\mathrm{d}t + \alpha\,\mathrm{d}N_2^+(t) - \beta\,\mathrm{d}N_2^-(t) + \gamma_2\,\mathrm{d}M_1(t), \tag{2}$$

where τ is the positive time constant according to which, in absence of stimuli, the membrane potential exponentially decays to the resting level. The effects of excitatory and inhibitory stimuli, that are assumed to occur according to independent Poisson processes, consist of instantaneous jumps of the membrane potential of magnitudes α and $-\beta$, respectively, where α and β are positive constants. The stochastic processes $N_i^+(t)$ and $N_i^-(t)$ $(i = 1, 2)$ in the above equations are assumed to be independent time-homogeneous Poisson processes describing the arrival of excitatory and inhibitory stimuli on the i-th neuronal unit originating from the environment. Processes $M_1(t)$ and $M_2(t)$ count the number of firings produced in [0,t] by the first and second neuron, respectively. The amplitude of the jump of the first (second) neuron membrane potential caused by the spike of the second (first) neuron is γ_1 (γ_2). Its value is positive or negative according to the excitatory or inhibitory nature of the firing neuron.

As implied by Eqs. (1) and (2), in absence of stimuli the neuronal membrane potential exponentially decays with time constant τ to the resting level that, without loss of generality, is set to be 0. A firing occurs when the membrane potential of a neuronal unit crosses the constant firing threshold S. We also assume that after each firing a refractory period of fixed duration takes place, at the end of which the membrane potential is reset in the neighborhood of the resting level according to some probability density. Here we shall assume that the reset occurs according to some probability density. Here we shall assume that the reset occurs according to the truncated Gaussian probability density $f(x) = C\, e^{-x^2/2}$, $-3 < x < 3$.

Since an analytical solution of the membrane potential dynamics for the model described by Eqs. (1) and (2) is not available, a Monte-Carlo simulation method has been devised and implemented by us. This procedure, described in [6], is very suitable for the statistical description of coupled neurons firing activity.

3 Statistical Results on ISIs

In this Section we study the dependence of interspike intervals of both neurons on parameters γ_1 and γ_2. Hereafter we shall focus our attention on three related matters: (i) to discuss the shape exhibited by ISIs histograms, (ii) to calculate the relevant statistical indices of ISIs, and (iii) to perform some comparisons between ISIs empirical distribution functions.

3.1 ISI Histograms

A measure of the variability in the timing of the sequence of spikes generated by the neurons is provided by the histograms interspike intervals. We consider the case in which the first neuron is excitatory whereas the second may possess excitatory or inhibitory nature. Computational results based on extensive simulations show that when the second neuron sends large inhibitory inputs, the probability mass of first neuron's ISI spreads over the temporal axis. The opposite occurs for large positive values of γ_1, i.e. when the second neuron is highly excitatory (see, for instance, Figure 1).

Feedback effects appear in the firing activity of the two coupled neurons. For instance, if the first neuron is excitatory and causes a jump of amplitude γ_2 of the membrane potential of the second neuron, then the interspike intervals of the latter exhibit a dependence on the behavior of γ_1. This is shown in Figure 2, where the second neuron ISI histograms are more spread when $\gamma_1 < 0$.

3.2 Some Statistical Indices of ISIs

Some relevant descriptive statistics of both neurons' ISIs are analyzed in this Section. The intrinsic symmetry property of the model, evident from Eqs. (1) and (2), is reflected in the ISIs statistical indices. The mean of the first neuron's

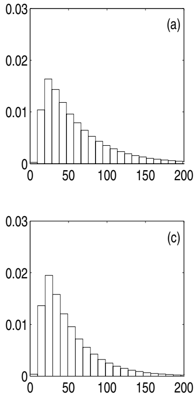


Fig. 1. First neuron's ISI histograms for $\gamma_2 = 2$ and (a) $\gamma_1 = -2$, (b) $\gamma_1 = -1$, (c) $\gamma_1 = 1$, (d) $\gamma_1 = 2$

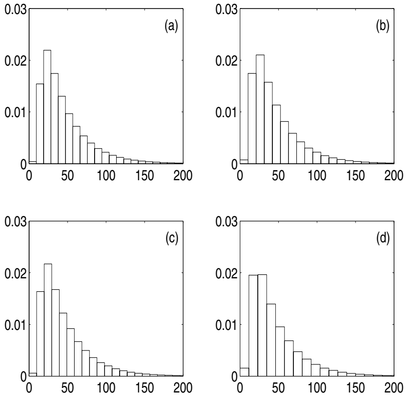


Fig. 2. Second neuron's ISI histograms for $\gamma_2 = 2$ and (a) $\gamma_1 = -2$, (b) $\gamma_1 = -1$, (c) $\gamma_1 = 1$, (d) $\gamma_1 = 2$

ISI evaluated for the couple of parameters (γ_1, γ_2) equals the mean of second neuron's ISI for (γ_2, γ_1). A similar behavior is exhibited also by the standard deviation and by the coefficient of variation of the interspike intervals.

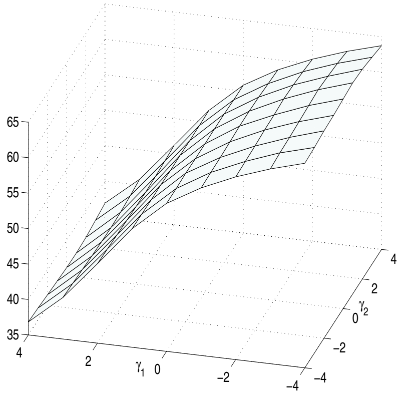


Fig. 3. Mean of first neuron's ISI

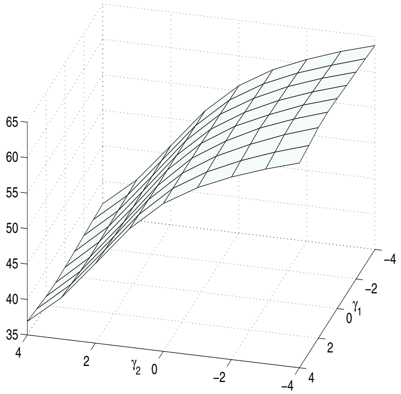


Fig. 4. Mean of second neuron's ISI

The mean of the first neuron's ISI decreases when γ_2 is fixed and γ_1 increases. Similarly, the mean of second neuron's ISI decreases for fixed γ_1 when γ_2 increases (see Figures 3 and 4). Moreover, mean of first neuron's ISI is larger (smaller) than the mean of the second neuron's ISI when γ_2 is larger (smaller) than γ_1. The means of the two neurons' ISIs are closer when γ_1 approaches γ_2, and viceversa. These remarks suggest that firing activity properties of the couple of neurons are globally dependent on the difference between γ_1 and γ_2.

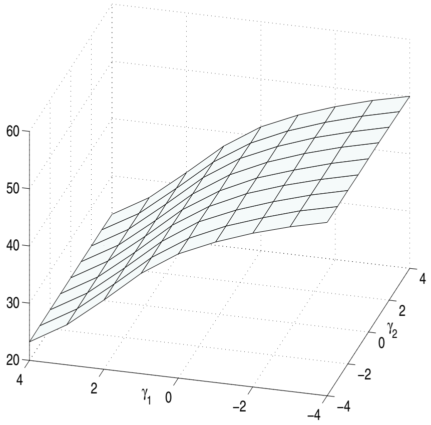


Fig. 5. Standard deviation of first neuron's ISI

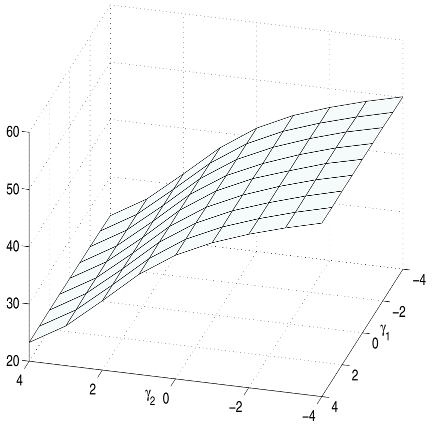


Fig. 6. Standard deviation of second neuron's ISI

Similar remarks about symmetry hold for the standard deviation (see Figures 5 and 6) and for the coefficient of variation (see Figures 7 and 8). Both statistical indices for the ISI of the first neuron decrease when γ_1 increases and γ_2 is fixed. By symmetry, the standard deviation and the coefficient of variation of the second neuron's ISI decrease when γ_1 is fixed and γ_2 increases. Moreover, if γ_1 is larger (smaller) than γ_2, the standard deviation and the coefficient of variation of second neuron's ISI are larger (smaller) than those of first neuron's ISI.

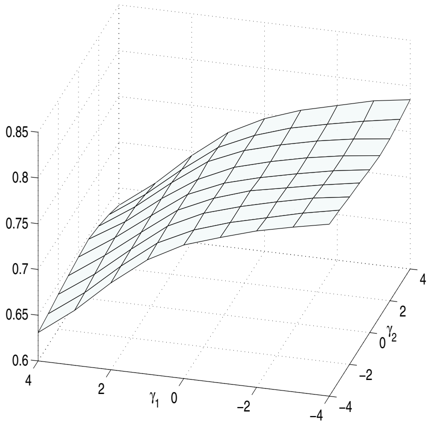


Fig. 7. Coefficient of variation of first neuron's ISI

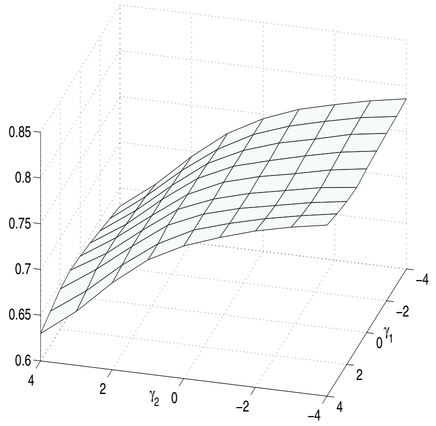


Fig. 8. Coefficient of variation of second neuron's ISI

3.3 ISIs Distribution Functions

Due to the symmetry of the model, first neuron's ISI distribution function is identical to that of the second neuron when the values of parameters γ_1 and γ_2 are exchanged. An example of this property is shown in Figures 9 and 10, where the ISIs cumulative distribution functions of first and second neuron are plotted for some choices of γ_1 and γ_2, respectively.

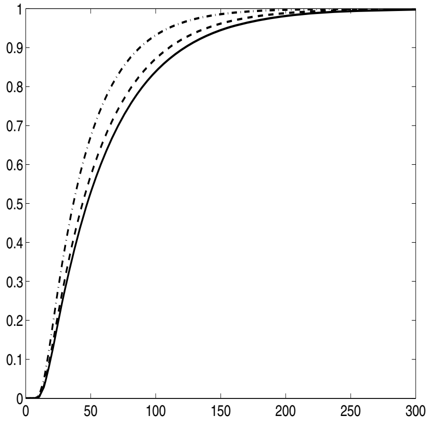


Fig. 9. First neuron's ISIs distribution functions for $\gamma_2 = 2$, and $\gamma_1 = -2$ (solid line), $\gamma_1 = 0$ (dashed line), $\gamma_1 = 2$ (dashed-dotted line)

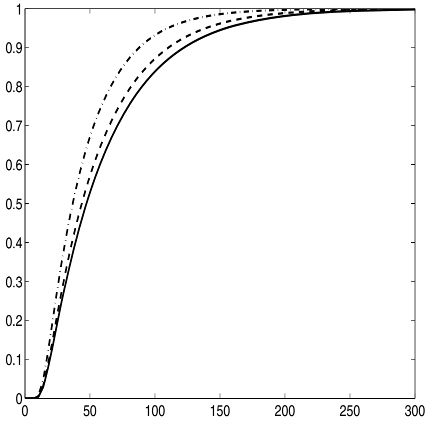


Fig. 10. Second neuron's ISIs distribution functions for $\gamma_1 = 2$, and $\gamma_2 = -2$ (solid line), $\gamma_2 = 0$ (dashed line), $\gamma_2 = 2$ (dashed-dotted line)

Comparing first neuron's ISIs distribution functions for different values of γ_1 we notice that the cumulative distribution function becomes larger when γ_1 increases (see Figure 9). The same property holds for the second neuron's ISI distribution function when γ_2 increases (see Figure 10). This suggests the existence of some kind of stochastic ordering. Indeed, denoting by $Y_i^{(\gamma_1,\gamma_2)}$ the random variable describing the i-th neuron interspike intervals and by $H_i^{(\gamma_1,\gamma_2)}(t)$ its distribution function ($i = 1, 2$), for $\delta > 0$ we have:

$$H_1^{(\gamma_1,\gamma_2)}(t) \leq H_1^{(\gamma_1+\delta,\gamma_2)}(t) \quad \text{and} \quad H_2^{(\gamma_1,\gamma_2)}(t) \leq H_2^{(\gamma_1,\gamma_2+\delta)}(t), \quad \text{for all } t \geq 0. \tag{3}$$

Eq. (3) shows that

$$Y_1^{(\gamma_1,\gamma_2)} \geq_{st} Y_1^{(\gamma_1+\delta,\gamma_2)} \quad \text{and} \quad Y_2^{(\gamma_1,\gamma_2)} \geq_{st} Y_2^{(\gamma_1,\gamma_2+\delta)},$$

where $\geq_{st}$ denotes the usual stochastic order. (For the definition of usual stochastic order see, for instance, [9]).

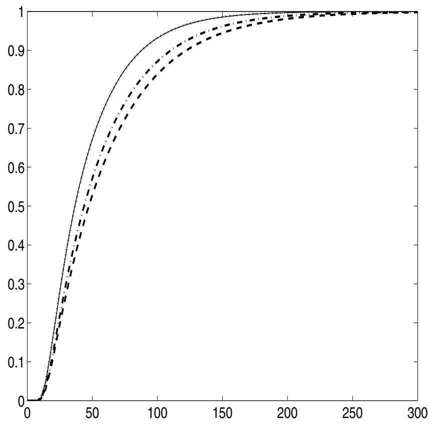


Fig. 11. ISIs distribution functions for the first neuron, with $\gamma_1 = 2$ (solid line), and for the second neuron, with $\gamma_2 = -2$ (dashed line), $\gamma_2 = 0$ (dashed-dotted line), $\gamma_2 = 2$ (dotted line)

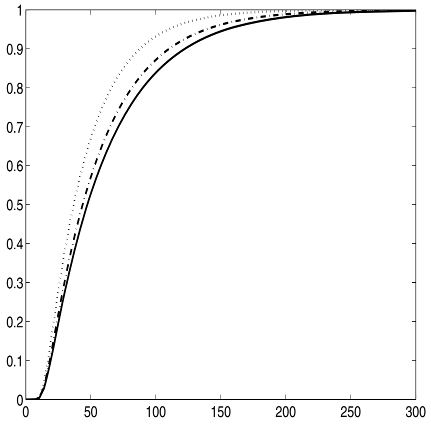


Fig. 12. ISIs distribution functions for the first neuron, with $\gamma_1 = -2$ (solid line), and for the second neuron, with $\gamma_2 = -2$ (dashed line), $\gamma_2 = 0$ (dashed-dotted line), $\gamma_2 = 2$ (dotted line)

Let us now compare the ISIs distribution functions of the two neurons. We note that when γ_1 is larger (smaller) than γ_2, the first neuron's ISI distribution function is larger (smaller) than that of second neuron. The distribution functions are equal when $\gamma_1 = \gamma_2$. Hence, since

$$\begin{aligned} H_1^{(\gamma_1,\gamma_2)}(t) \geq H_2^{(\gamma_1,\gamma_2)}(t), &\quad \text{for all } t \geq 0, \text{ with } \gamma_1 \geq \gamma_2, \\ H_1^{(\gamma_1,\gamma_2)}(t) \leq H_2^{(\gamma_1,\gamma_2)}(t), &\quad \text{for all } t \geq 0, \text{ with } \gamma_1 \leq \gamma_2, \end{aligned}$$

we conclude that

$$Y_1^{(\gamma_1,\gamma_2)} \leq_{st} Y_2^{(\gamma_1,\gamma_2)} \quad \text{for} \quad \gamma_1 \geq \gamma_2 \quad \text{and} \quad Y_1^{(\gamma_1,\gamma_2)} \geq_{st} Y_2^{(\gamma_1,\gamma_2)} \quad \text{for} \quad \gamma_1 \leq \gamma_2.$$

Figures 11 and 12 show the distribution functions of the two neurons' ISIs for different values of (γ_1, γ_2).

4 Entropies and Correlation Functions

Information theory is widely used in neuronal coding to quantify the information on the received stimuli conveyed by the neural response [3]. Aiming to obtain a measure of information on the coupled neuronal activity, hereafter we consider the following discrepancy measure between the distributions of $Y_1^{(\gamma_1,\gamma_2)}$ and $Y_2^{(\gamma_1,\gamma_2)}$:

$$I_{(Y_1,Y_2)}^{(\gamma_1,\gamma_2)} = \int_0^{+\infty} h_1^{(\gamma_1,\gamma_2)}(u) \log \frac{h_1^{(\gamma_1,\gamma_2)}(u)}{h_2^{(\gamma_1,\gamma_2)}(u)} \, \mathrm{d}u, \tag{4}$$

$$I_{(Y_2,Y_1)}^{(\gamma_1,\gamma_2)} = \int_0^{+\infty} h_2^{(\gamma_1,\gamma_2)}(u) \log \frac{h_2^{(\gamma_1,\gamma_2)}(u)}{h_1^{(\gamma_1,\gamma_2)}(u)} \, du. \tag{5}$$

Here, $h_i^{(\gamma_1,\gamma_2)}(t)$ denotes the probability density function (pdf) of $Y_i^{(\gamma_1,\gamma_2)}$ $(i = 1, 2)$. Functions $I_{(Y_i,Y_j)}^{(\gamma_1,\gamma_2)}$ $(i, j \in \{1, 2\}, i \neq j)$ are called relative entropies, or discrimination measures. They provide a measure of the inefficiency of assuming that the probability density function of interspike intervals is $h_j^{(\gamma_1,\gamma_2)}(u)$ when the true p.d.f. is $h_i^{(\gamma_1,\gamma_2)}(u)$.

According to the nature of our simulation scheme, $I_{(Y_1,Y_2)}^{(\gamma_1,\gamma_2)}$ and $I_{(Y_2,Y_1)}^{(\gamma_1,\gamma_2)}$ are evaluated by means of a discretization of the right-hand sides of Eqs. (4) and (5). Figures 13 and 14 show the ISIs relative entropies as γ_1 and γ_2 vary. The symmetry is again evident by comparing these entropies. We point out that, as expected, the minimum of such functions is attained for $\gamma_1 = \gamma_2$.

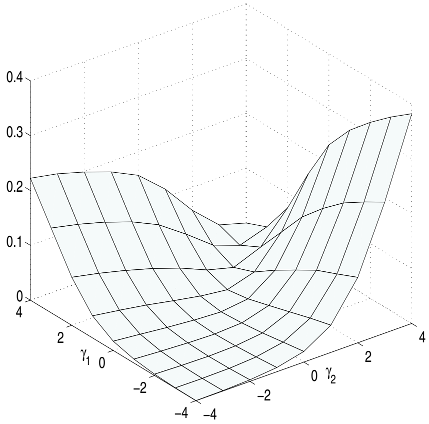


Fig. 13. Relative entropy of first neuron's ISI for some choices of γ_1 and γ_2

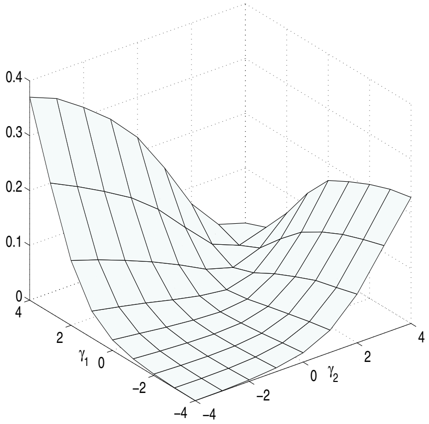


Fig. 14. Relative entropy of second neuron's ISI for some choices of γ_1 and γ_2

Let us denote by $\{T_n^i, n \in \mathbb{N}\}$ the stochastic process describing i-th neuron firing time $(i = 1, 2)$, where T_n^i denotes the random time in which the i-th neuron $(i = 1, 2)$ fires for the n-th time $(n \in \mathbb{N})$. We adopt the following definition of autocorrelation function (see [1] for a more general definition):

$$AC^i(\tau) = \frac{\sum_k (T_{k+j}^i - \overline{T}^i)(T_k^i - \overline{T}^i)}{\sum_k (T_k^i - \overline{T}^i)^2} \quad (i = 1, 2), \tag{6}$$

where $\overline{T}^i = \dfrac{1}{n_{tot}} \sum\limits_n T_n^i$ is the mean firing time of the i-th neuron, with n_{tot} denoting the total number of spikes released by the i-th neuron. The index j appearing in the right-hand-side of (6) is larger than k, and it is such that $T_{k+j}^i - T_k^i = \tau$.

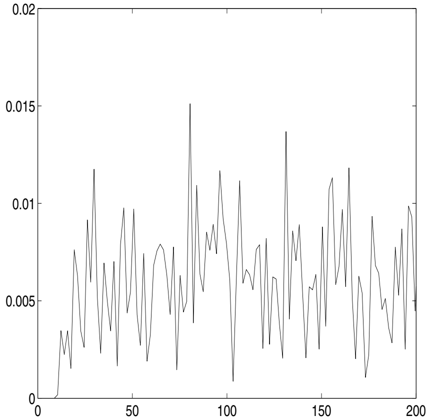


Fig. 15. First neuron's autocorrelation function for $\gamma_1 = -1$ and $\gamma_2 = 2$

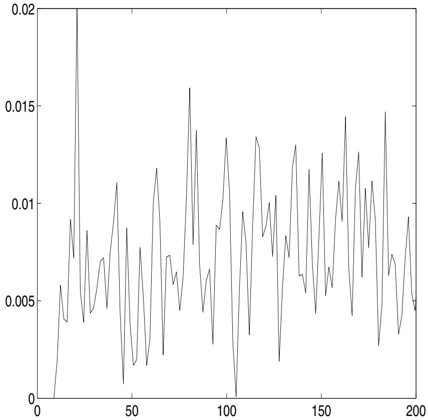


Fig. 16. Second neuron's autocorrelation function for $\gamma_1 = -1$ and $\gamma_2 = 2$

As Eq. (6) shows, the autocorrelation function is an even function of τ that may take both positive and negative values. When τ is small, the firing times T^i_{k+j} and T^i_k involved in $T^i_{k+j} - T^i_k = \tau$ are very close, and the autocorrelation function takes positive values. On the contrary, for large values of τ we have observed negative values for the function $AC^i(\tau)$. A plot of the autocorrelation function when the first neuron is excitatory and the second inhibitory is shown in Figures 15 and 16. Second neuron's autocorrelation function is larger than that of the first neuron and shows numerous peaks. Hence, for the second neuron's firing times, some lags are more likely than others: in particular, for the case described in Figure 16, the firing times show a very frequent lag of about 30 ms.

Aiming to analyze the synchronization between the two neurons firing activity we consider the crosscorrelation function defined as:

$$CC^1(\tau) = \frac{\sum_k (T^1_{k+j} - \overline{T}^1)\,(T^2_k - \overline{T}^2)}{\sqrt{\sum_k (T^1_k - \overline{T}^1)^2}\,\sqrt{\sum_k (T^2_k - \overline{T}^2)^2}}, \tag{7}$$

$$CC^2(\tau) = \frac{\sum_k (T^2_{k+m} - \overline{T}^2)\,(T^1_k - \overline{T}^1)}{\sqrt{\sum_k (T^1_k - \overline{T}^1)^2}\,\sqrt{\sum_k (T^2_k - \overline{T}^2)^2}}, \tag{8}$$

where $\overline{T}^i$ is the mean firing time of the i-th neuron ($i = 1, 2$), and $j \geq 1$ ($m \geq 1$) is such that $T^1_{k+j} - T^2_k = \tau$ ($T^2_{k+m} - T^1_k = \tau$). Due to Eqs. (7) and (8) it is

$$CC^1(\tau) = CC^2(-\tau), \qquad \tau > 0.$$

Figure 17 shows $CC^1(\tau)$ when $\gamma_1 = -1$ and $\gamma_2 = 2$. The peak exhibited by this function at the lag -1.75 ms suggests that the second neuron is very likely to fire about 1.75 ms after a spike of the first neuron. The crosscorrelation function $CC^1(\tau)$ for $\gamma_1 = 1$ and $\gamma_2 = -1$ is plotted in Figure 18. In this case the crosscorrelation function does not show significant peaks.

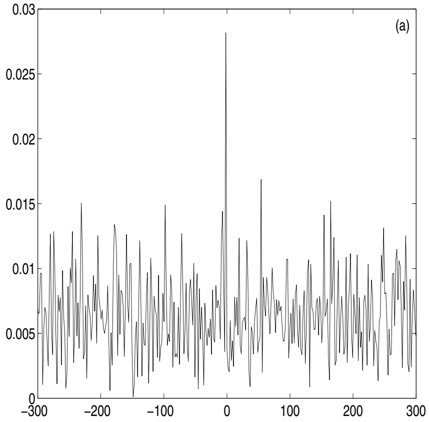


Fig. 17. Crosscorrelation function in the case $\gamma_1 = -1$ and $\gamma_2 = 2$

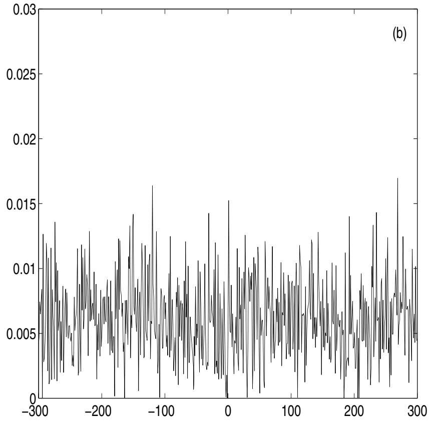


Fig. 18. Crosscorrelation function in the case $\gamma_1 = 1$ and $\gamma_2 = -1$

5 Concluding Remarks

Under the assumption of complete interaction between the neurons, the firing activity of the coupled Stein-type neuronal units is characterized by the presence of evident feedback effects. These effects can be observed looking at the shape of histograms of neurons interspike intervals when both γ_1 and γ_2 are different from 0 (see Section 3.1).

The comparison between the means of the ISIs of the two neurons suggests that the firing activity properties of the pair of neurons are globally dependent on the difference between γ_1 and γ_2. Indeed, in Section 3.2 it is emphasized that the mean of the first neuron's ISI is larger (smaller) than that of the second neuron's when γ_2 is larger (smaller) than γ_1, and is equal to the mean of the second neuron's ISI when $\gamma_1 = \gamma_2$. The standard deviation and the coefficient of variation show a similar behavior. Moreover, in Section 3.3 the analysis of ISIs distribution functions proves the existence of a stochastic ordering between the random variables describing the two neurons interspike intervals.

Finally, the crosscorrelation function studied in Section 4 when the coupled neurons have different nature, reveals the phase-locked connection between the neurons firing activity existing when γ_1 and γ_2 take very distant opposite values.

References

1. Dayan, P. Abbott L.F.: Theoretical Neuroscience. Computational and Mathematical Modeling of Neural Systems. The Mit Press (2001).
2. Borisyuk, R.: Oscillatory activity in the neural network of spiking elements. BioSystems, 67 (2002) 3–16.
3. Borst, A., Theunissen F.E.: Information Theory and neural coding. Nature Neuroscience, 2, n. 11 (1999) 947–957.
4. Burkitt, A.N.: Balanced neurons: analysis of leaky integrate-and-fire neurons with reversal potential. Biological Cybernetics, 85 (2001) 247–255.
5. Cartling, B.: Control of computational dynamics of coupled integrate-and-fire neurons. Biological Cybernetics, 76 (1997) 383–395.
6. Di Crescenzo, A., Martinucci, B., Pirozzi, E., Ricciardi L.M.: On the interaction between two Stein's neuronal units. In: Trappl, R. (ed.): Cybernetics and Systems 2004, Vol. 1. Austrian Society for Cybernetic Studies, Vienna (2004) 205–210.
7. Di Crescenzo, A., Martinucci, B., Pirozzi E.: On the dynamics of a pair of coupled neurons subject to alternating input rates. BioSystems, 79 (2005) 109–116.
8. Elson, R.C., Selverston, A.I., Huerta, R., Rulkov, N.F., Rabinovich, M.I., Abarbanel, H.D.I.: Synchronous Behavior of Two Coupled Biological Neurons. Physical Review Letters, 81, n. 25 (1998) 5692–5695.
9. Shaked M., Shantikumar J.C.: Stochastic Orders and Their Applications. San Diego: Academic Press (1994).

Upcrossing First Passage Times for Correlated Gaussian Processes*

Virginia Giorno[1], Amelia G. Nobile[1], and Enrica Pirozzi[2]

[1] Dipartimento di Matematica e Informatica, Università di Salerno,
Via Ponte don Melillo, 84084 Fisciano (SA), Italy
{giorno, nobile}@unisa.it
[2] Dipartimento di Matematica e Applicazioni, Università di Napoli Federico II,
Via Cintia, 80126 Napoli, Italy
enrica.pirozzi@unina.it

Abstract. For a class of stationary Gaussian processes and for large correlation times, the asymptotic behavior of the upcrossing first passage time probability densities is investigated. Parallel simulations of sample paths of special stationary Gaussian processes for large correlations times provide a statistical validation of the theoretical results.

1 Introductory Remarks

Upcrossing first passage time problems play a relevant role in various applied contexts including neuronal modeling [6]. Indeed, the neuronal firing can sometimes be viewed as the event that takes place when the potential difference across the membrane exceeds a certain threshold value (cf., for instance, [10], [12]).

In the context of single neuron's activity modeling, Kostyukov et al. (cf. [7], [8]) make use of the notion of correlation time to evaluate an approximation for the upcrossing FPT probability density function (pdf) of a Gaussian non-Markov process. Namely, for a one-dimensional, non-singular stationary Gaussian process with zero mean, unit variance and correlation function $\varrho(t)$,

$$\tau_c := \int_0^{+\infty} \left|\varrho(\vartheta)\right| \, d\vartheta \tag{1}$$

is defined as the correlation time of the process.

The available analytical results on upcrossing first-passage-time (FPT) problems are scarce, fragmentary and mainly centered on diffusion processes. Furthermore, if one deals with models involving processes characterized by memory effects the Markov property breaks down, and one is forced to face FPT problems for correlated processes (cf., for instance, [1], [3], [4] and [5]). Hence, in order to construct neuronal models that are based on such processes, some preliminary theoretical contributions appear to be necessary. Within such context, the

---

* This work has been performed under partial support by MIUR (PRIN 2003) and by GNCS-INdAM.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 447–456, 2005.
© Springer-Verlag Berlin Heidelberg 2005

present contribution focuses on the asymptotic behavior of the upcrossing FPT pdf for stationary mean-square differentiable Gaussian processes for large correlation times. Specifically, let $\{X(t), t \geq 0\}$ be a one-dimensional, non-singular stationary Gaussian process with zero mean, unit variance and correlation function $\varrho(t)$ such that $\varrho(0) = 1$, $\dot{\varrho}(0) = 0$ and $\ddot{\varrho}(0) < 0$. Then $\dot{X}(t)$, the derivative of $X(t)$ with respect to t, exists in the mean-square sense. Let $S(t) \in C^1[0, +\infty)$ be an arbitrary function such that $X(0) = x_0 < S(0)$. Then,

$$T = \inf_{t \geq 0}\{t : X(t) > S(t)\}, \quad X(0) = x_0 \tag{2}$$

is the FPT random variable and

$$g(t \mid x_0) = \frac{\partial}{\partial t} P(T < t) \tag{3}$$

is the FPT pdf of $X(t)$ through $S(t)$ conditional upon $X(0) = x_0$. Furthermore, $\forall n \in \mathbb{N}$ and $0 < t_1 < t_2 < \ldots < t_n$ we denote by $W_n(t_1, t_2, \ldots, t_n \mid x_0)\, dt_1\, dt_2 \cdots dt_n$ the joint probability that $X(t)$ crosses $S(t)$ from below in the intervals $(t_1, t_1 + dt_1)$, $(t_2, t_2 + dt_2)$, …, $(t_n, t_n + dt_n)$ given that $X(0) = x_0$. The function W_n can be written as:

$$\begin{aligned} W_n(t_1, t_2, \ldots, t_n \mid x_0) = & \int_{\dot{S}(t_1)}^{+\infty} d\xi_1 \int_{\dot{S}(t_2)}^{+\infty} d\xi_2 \cdots \int_{\dot{S}(t_n)}^{+\infty} d\xi_n \prod_{i=1}^{n} [\xi_i - \dot{S}(t_i)] \\ & \times p_{2n}[S(t_1), t_1; S(t_2), t_2; \ldots; S(t_n), t_n; \xi_1, t_1; \xi_2, t_2; \ldots; \xi_n, t_n \mid x_0], \end{aligned} \tag{4}$$

where $p_{2n}(x_1, t_1; x_2, t_2; \ldots, x_n, t_n; \xi_1, t_1; \xi_2, t_2; \ldots; \xi_n, t_n \mid x_0)$ is the joint pdf of $x_1 = X(t_1)$, $x_2 = X(t_2)$, …, $x_n = X(t_n)$, $\xi_1 = \dot{X}(t_1)$, $\xi_2 = \dot{X}(t_2)$, …, $\xi_n = \dot{X}(t_n)$ conditional upon $X(0) = x_0$. Furthermore, we consider the following functions:

$$\begin{aligned} & Q_1(t \mid x_0) = W_1(t \mid x_0) \\ & Q_n(t \mid x_0) = \int_0^t dt_1 \int_{t_1}^t dt_2 \cdots \int_{t_{n-2}}^t dt_{n-1}\, W_n(t_1, t_2 \ldots, t_{n-1}, t \mid x_0) \\ & \qquad\qquad\qquad\qquad\qquad\qquad (n = 2, 3, \ldots), \end{aligned} \tag{5}$$

with $t_0 > 0$. We note that $Q_n(t \mid x_0)\, dt$ gives the probability that $X(t)$ crosses $S(t)$ from below at least n times *and* the last crossing occurs in the interval $(t, t + dt)$ conditional upon $X(0) = x_0$. Denoting by $q_k(t \mid x_0)\, dt$ the probability that $X(t)$ crosses $S(t)$ for the k-th time in $(t, t + dt)$, one has (cf. [13]):

$$Q_n(t \mid x_0) = \sum_{k=n}^{+\infty} \binom{k-1}{n-1} q_k(t \mid x_0) \qquad (n = 1, 2, \ldots). \tag{6}$$

Since $g(t \mid x_0) \equiv q_1(t \mid x_0)$, setting $n = 1$ in (6) one obtains:

$$g(t \mid x_0) = W_1(t \mid x_0) - \sum_{k=2}^{+\infty} q_k(t \mid x_0), \qquad x_0 < S(0). \tag{7}$$

Making use of (6) and (7), an alternative expression for $g(t \mid x_0)$ can be obtained in terms of the functions $Q_n(t \mid x_0)$:

$$g(t \mid x_0) = W_1(t \mid x_0) - \sum_{n=2}^{+\infty} (-1)^n Q_n(t \mid x_0), \qquad x_0 < S(0). \tag{8}$$

We stress that although (8) gives a formal analytical expression for the FPT pdf through arbitrary time-dependent boundaries, no reliable numerical evaluations appear to be feasible due to the complexity of (5). However, the explicit expression of $W_1(t \mid x_0)$ can be evaluated [11]:

$$\begin{aligned} W_1(t \mid x_0) = {} & \frac{|\Lambda_3(t)|^{1/2}}{2\,\pi\,[1-\varrho^2(t)]} \exp\biggl\{-\frac{[S(t)-x_0\,\varrho(t)]^2}{2\,[1-\varrho^2(t)]}\biggr\} \\ & \times \biggl[\exp\biggl\{-\frac{\sigma^2(t \mid x_0)}{2}\biggr\} - \sqrt{\frac{\pi}{2}}\,\sigma(t \mid x_0)\,\mathrm{Erfc}\biggl(\frac{\sigma(t \mid x_0)}{\sqrt{2}}\biggr)\biggr], \end{aligned} \tag{9}$$

where

$$\begin{aligned} |\Lambda_3(t)| &= -\ddot{\varrho}(0)\,\bigl[1-\varrho^2(t)\bigr] - \bigl[\dot{\varrho}(t)\bigr]^2, \\ \sigma(t \mid x_0) &= \biggl(\frac{1-\varrho^2(t)}{|\Lambda_3(t)|}\biggr)^{1/2} \biggl\{\dot{S}(t) + \frac{\dot{\varrho}(t)\,\bigl[\varrho(t)\,S(t)-x_0\bigr]}{1-\varrho^2(t)}\biggr\}, \end{aligned} \tag{10}$$

and

$$\mathrm{Erfc}(z) = \frac{2}{\sqrt{\pi}} \int_z^{+\infty} e^{-y^2}\,dy, \qquad z \in \mathbb{R}. \tag{11}$$

We note that $W_1(t \mid x_0)$, providing an upper bound to the FPT pdf in (8), is a good approximation of $g(t \mid x_0)$ only for small values of t.

We shall now focus on the upcrossing FPT problem. We assume that a subset of sample paths of $X(t)$ originates at a state X_0 that is a random variable with preassigned pdf

$$\gamma_\varepsilon(x_0) = \begin{cases} f(x_0)\,\biggl[\displaystyle\int_{-\infty}^{S(0)-\varepsilon} f(z)\,dz\biggr]^{-1}, & x_0 < S(0)-\varepsilon \\ 0, & x_0 \geq S(0)-\varepsilon, \end{cases} \tag{12}$$

where $\varepsilon > 0$ is a fixed real number and $f(x_0)$ denotes the pdf of $X(0)$:

$$f(x_0) = \frac{1}{\sqrt{2\,\pi}}\,\exp\biggl\{-\frac{x_0^2}{2}\biggr\}, \qquad x_0 \in \mathbb{R}. \tag{13}$$

Then,

$$T_{X_0}^{(\varepsilon)} := \inf_{t \geq 0}\{t : X(t) > S(t)\} \tag{14}$$

is the ε-upcrossing FPT of $X(t)$ through $S(t)$. Its pdf is related to the conditional FPT pdf $g(t \mid x_0)$ as follows [2]:

$$g_u^{(\varepsilon)}(t) = \int_{-\infty}^{S(0)-\varepsilon} g(t \mid x_0)\, \gamma_\varepsilon(x_0)\, dx_0 \qquad (t \geq 0). \tag{15}$$

Making use of (7) in (15), one has:

$$g_u^{(\varepsilon)}(t) = W_1^{(\varepsilon)}(t) - \sum_{k=2}^{+\infty} q_k^{(\varepsilon)}(t) \tag{16}$$

where

$$W_1^{(\varepsilon)}(t) = \int_{-\infty}^{S(0)-\varepsilon} W_1(t \mid x_0)\, \gamma_\varepsilon(x_0)\, dx_0, \tag{17}$$

$$q_k^{(\varepsilon)}(t) = \int_{-\infty}^{S(0)-\varepsilon} q_k(t \mid x_0)\, \gamma_\varepsilon(x_0)\, dx_0 \qquad (k = 2, 3, \ldots). \tag{18}$$

In Section 2, under suitable assumptions on the correlation function $\varrho(t)$ and on the threshold $S(t)$, the behavior of $g_u^{(\varepsilon)}(t)$ as $\tau_c \to +\infty$ is analyzed. Furthermore, in Section 3 two different stationary Gaussian processes are considered and the results of some simulations are finally presented.

2 Asymptotic Results

Proposition 1. *Let $\{X(t), t \geq 0\}$ be a non-singular stationary Gaussian process with zero mean, unit variance and correlation function $\varrho(t)$ such that $\varrho(0) = 1$, $\dot\varrho(0) = 0$ and $\ddot\varrho(0) < 0$. Furthermore, let $S(t) \in C^1[0, +\infty)$ be an arbitrary monotonically decreasing function such that $\dot S(t) - [S(t) - S(0)]/t \leq 0$. If the correlation function of $X(t)$ satisfies*

$$\lim_{\tau_c \to +\infty} \varrho(t) = 1, \quad \lim_{\tau_c \to +\infty} \dot\varrho(t) = \lim_{\tau_c \to +\infty} \ddot\varrho(t) = 0, \quad \lim_{\tau_c \to +\infty} \frac{\dot\varrho(t)}{1 - \varrho^2(t)} = -\frac{1}{t}, \tag{19}$$

then

$$\varphi^{(\varepsilon)}(t) := \lim_{\tau_c \to +\infty} W_1^{(\varepsilon)}(t) = \begin{cases} -\dot S(t)\, \gamma_\varepsilon[S(t)], & S(t) < S(0) - \varepsilon, \\ 0, & \text{otherwise.} \end{cases} \tag{20}$$

Proof. We first note that (9) can be written as:

$$\begin{aligned} W_1(t \mid x_0) = {} & \frac{1}{\sqrt{2\pi\,[1 - \varrho^2(t)]}} \exp\Big\{ -\frac{[S(t) - x_0\, \varrho(t)]^2}{2\,[1 - \varrho^2(t)]} \Big\} \\ & \times \Big\{ -\frac{1}{2} \Big[\dot S(t) + \frac{\dot\varrho(t)\, \big[\varrho(t)\, S(t) - x_0\big]}{1 - \varrho^2(t)} \Big] \operatorname{Erfc}\Big(\frac{\sigma(t \mid x_0)}{\sqrt{2}} \Big) \\ & + \frac{1}{2\pi} \sqrt{\frac{|\Lambda_3(t)|}{1 - \varrho^2(t)}}\, \exp\Big[-\frac{\sigma^2(t \mid x_0)}{2} \Big] \Big\}, \end{aligned} \tag{21}$$

with $x_0 < S(0)$, and where $|\Lambda_3(t)|$ and $\sigma(t \mid x_0)$ are given in (10). By virtue of assumptions (19) one has:

$$\begin{aligned} &\lim_{\tau_c \to +\infty} \frac{|\Lambda_3(t)|}{1-\varrho^2(t)} = 0, \\ &\lim_{\tau_c \to +\infty} \left\{ \dot{S}(t) + \frac{\dot{\varrho}(t)\,\big[\varrho(t)\,S(t) - x_0\big]}{1-\varrho^2(t)} \right\} = \dot{S}(t) - \frac{S(t)-x_0}{t}\,. \end{aligned} \tag{22}$$

Furthermore, by noting that

$$\dot{S}(t) - \frac{S(t)-x_0}{t} < \dot{S}(t) - \frac{S(t)-S(0)}{t} \le 0$$

and recalling (19), one is led to:

$$\lim_{\tau_c \to +\infty} \sigma(t \mid x_0) = -\infty. \tag{23}$$

Hence, due to (22) and (23), from (21) one obtains:

$$\lim_{\tau_c \to +\infty} W_1(t \mid x_0) = -\Big[\dot{S}(t) - \frac{S(t)-x_0}{t}\Big]\,\delta\big[S(t)-x_0\big], \tag{24}$$

where δ denotes the Dirac delta-function. Taking the limit as τ_c diverges in (17) and making use of (24), Eq. (20) immediately follows. □

Remark 1. Under the assumptions of Proposition 1, if $\lim_{t\to+\infty} S(t) = -\infty$ one has:

$$\int_0^{+\infty} \varphi^{(\varepsilon)}(t)\, dt = 1. \tag{25}$$

Proof. Integrating both sides of (20) with respect to t in $(0, +\infty)$, we obtain:

$$\int_0^{+\infty} \varphi^{(\varepsilon)}(t)\, dt = -\int_{\mathcal{D}} \dot{S}(t)\, \gamma_\varepsilon[S(t)]\, dt,$$

where $\mathcal{D} = \{t : S(t) < S(0) - \varepsilon\}$. Hence, recalling (12), Eq. (25) immediately follows. □

Remark 1 shows that as $\tau_c \to +\infty$ the ε-upcrossing probability that, eventually, $X(t)$ crosses $S(t)$ from below at least once is unit. Hence, as $\tau_c \to +\infty$ the ε-upcrossing probability that $X(t)$ ultimately crosses $S(t)$ for the first time is unit.

Proposition 2. *Under the assumptions of Proposition 1, if* $\lim_{t\to+\infty} S(t) = -\infty$ *one has:*

$$\lim_{\tau_c \to +\infty} g_u^{(\varepsilon)}(t) = \varphi^{(\varepsilon)}(t), \tag{26}$$

with $\varphi^{(\varepsilon)}(t)$ *defined in (20).*

Proof. Taking the limit as $\tau_c \to +\infty$ in (16), for all $\varepsilon > 0$ one has:

$$\varphi^{(\varepsilon)}(t) = h^{(\varepsilon)}(t) + \psi^{(\varepsilon)}(t), \tag{27}$$

where we have set:

$$h^{(\varepsilon)}(t) = \lim_{\tau_c \to +\infty} g_u^{(\varepsilon)}(t), \qquad \psi^{(\varepsilon)}(t) = \lim_{\tau_c \to +\infty} \sum_{k=2}^{+\infty} q_k^{(\varepsilon)}(t). \tag{28}$$

Integrating both sided of (27) with respect to t between 0 and $+\infty$, and making use of Remark 1, one obtains:

$$\int_0^{+\infty} \psi^{(\varepsilon)}(t)\, dt = 0.$$

Hence, $\psi^{(\varepsilon)}(t) = 0$, so that (26) follows from (27). □

3 Examples and Simulation Results

In this Section, a simulation is used in order to disclose the essential features of the ε-upcrossing FPT pdf for a stationary Gaussian process $X(t)$ and for specified boundaries. Our approach relies on a simulation procedure by which sample paths of the stochastic process are constructed and their upcrossing first passage instants through the boundary are recorded in order to construct reliable histograms estimating the FPT pdf $\tilde{g}_u^{(\varepsilon)}(t)$. Specifically, for the construction of sample paths of the process $X(t)$ we have used the "conditional expectations method" and, to avoid numerical stability problems, we have implemented a regularization technique based on the so-called "doubled algorithm" (see, for instance, [9]). Since the sample paths of the simulated process are generated independently of one another, the simulation procedure is particularly suited for implementation on supercomputers. Hence, the related vector and parallel code has been implemented on an IBM SP-Power4 machine. To evaluate the ε-upcrossing FPT densities, we have chosen X_0 randomly according to the initial pdf $\gamma_\varepsilon(x_0)$. To this purpose, we have made use of the so-called acceptance-rejection method (cf. for instance [14]).

We now consider two stationary Gaussian processes such that the assumptions on the correlation function of Proposition 1 are satisfied.

(i) Let $\{X_1(t), t \ge 0\}$ be a stationary Gaussian process with zero mean, unit variance and correlation function:

$$\varrho(t) = e^{-\alpha|t|} \Big\{ \cos(\alpha\,\omega\, t) + \frac{1}{\omega} \sin(\alpha\,\omega\,|t|) \Big\} \qquad (t \in \mathbb{R}) \tag{29}$$

where $\alpha > 0$ and $\omega \in \mathbb{R}$. Since

$$\begin{aligned} \dot{\varrho}(t) &= -\frac{1+\omega^2}{\omega}\, \alpha\, e^{-\alpha|t|} \sin(\alpha\,\omega\, t), \\ \ddot{\varrho}(t) &= \frac{1+\omega^2}{\omega}\, \alpha^2\, e^{-\alpha|t|} \Big\{ \sin(\alpha\,\omega\,|t|) - \omega \cos(\alpha\,\omega\, t) \Big\}, \end{aligned}$$

one has $\varrho(0) = 1$, $\dot{\varrho}(0) = 0$ and $\ddot{\varrho}(0) = -\alpha^2\,(1+\omega^2) < 0$, so that the process $X_1(t)$ is mean-square differentiable. Furthermore, the correlation time of $X_1(t)$ is:

$$\tau_c = \frac{2}{\alpha\,(1+\omega^2)}\;\cdot \tag{30}$$

Hence, $\tau_c \to +\infty$ if and only if $\alpha \to 0$. It is easily proved that (19) hold as $\alpha \to 0$.

(ii) Let $\{X_2(t), t \geq 0\}$ be a stationary Gaussian process with zero mean, unit variance and correlation function:

$$\varrho(t) = \frac{1}{1+\beta\,t^2} \qquad (t \in \mathbb{R}) \tag{31}$$

with $\beta > 0$. Since

$$\dot{\varrho}(t) = -\frac{2\,\beta\,t}{(1+\beta\,t^2)^2}\,,$$
$$\ddot{\varrho}(t) = -\frac{2\,\beta\,(1-3\,\beta\,t^2)}{(1+\beta\,t^2)^3}\,,$$

one has $\varrho(0) = 1$, $\dot{\varrho}(0) = 0$ and $\ddot{\varrho}(0) = -2\,\beta < 0$, so that $X_2(t)$ is mean-square differentiable. Furthermore, the correlation time of $X_2(t)$ is:

$$\tau_c = \frac{\pi}{2\,\sqrt{\beta}}\;\cdot \tag{32}$$

Hence, $\tau_c \to +\infty$ if and only if $\beta \to 0$. One can easily prove that (19) hold as $\beta \to 0$.

In Fig. 1*(a)* the correlation function (29) is plotted for $\omega = 1$ and for $\alpha = 0.1, 0.5, 1, 2$, whereas in Fig. 1*(b)* the correlation function (31) is plotted for $\beta = 0.1, 0.5, 1, 2$,

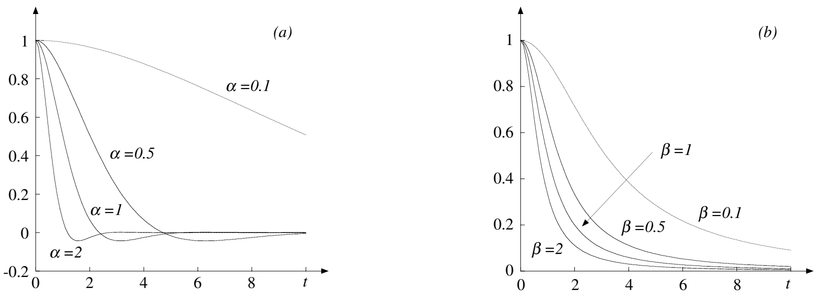


Fig. 1. Plot of correlation function (29) in *(a)* and of correlation function (31) in *(b)* as function of t. Figure *(a)* refers to the case $\omega = 1$ and $\alpha = 0.1, 0.5, 1, 2$; Figure *(b)* refers to the case $\beta = 0.1, 0.5, 1, 2$.

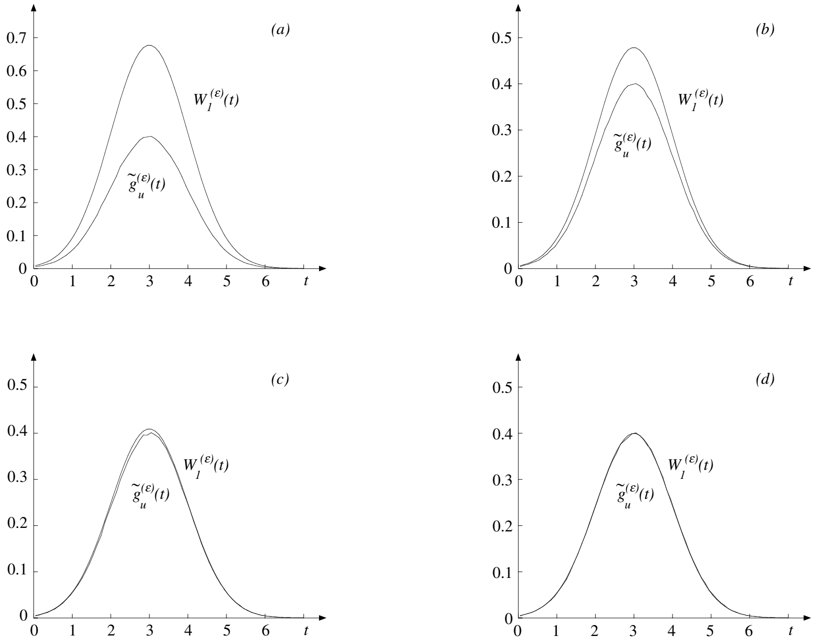


Fig. 2. Plot of $\tilde{g}_u^{(\varepsilon)}(t)$ and of $W_1^{(\varepsilon)}(t)$ for the Gaussian process with zero mean, unit variance and correlation function (29) for $S(t) = 3 - t$, $\varepsilon = 0.01$ and $\omega = 1$ in the following cases: *(a)* $\alpha = 2$, *(b)* $\alpha = 1$, *(c)* $\alpha = 0.5$ and *(d)* $\alpha = 0.1$

For both processes $X_1(t)$ and $X_2(t)$, if $S(t)$ is an arbitrary monotonically decreasing function such that *(1)* $\dot{S}(t) - [S(t) - S(0)]/t \le 0$ and *(2)* $\lim_{t\to+\infty} S(t) = -\infty$, then (26) holds. For instance, if $S(t) = a\,t + b$ $(a < 0; b \in \mathbb{R})$ conditions *(1)* and *(2)* are satisfied, whereas if $S(t) = a\,t^2 + b\,t + c$ $(a \neq 0; b, c \in \mathbb{R})$ conditions *(1)* and *(2)* are satisfied if and only if $a < 0$ and $b < 0$. Furthermore, if $S(t) = a\,e^{b\,t}$ $(a, b \in \mathbb{R})$ conditions *(1)* and *(2)* hold if and only if $a < 0$ and $b > 0$.

By making use of simulation procedure, we have performed extensive computations on processes $X_1(t)$ and $X_2(t)$ to disclose the behavior of the ε-upcrossing FPT pdf through time-dependent boundaries for large correlation times. The results of the simulations have indicated that $\tilde{g}_u^{(\varepsilon)}(t)$ is susceptible of an excellent approximation for large τ_c. Indeed, under the assumption of Proposition 2, for large τ_c the following asymptotic relation holds:

$$g_u^{(\varepsilon)}(t) \simeq W_1^{(\varepsilon)}(t) \qquad (t > 0), \tag{33}$$

where $W_1^{(\varepsilon)}(t)$, that provides an upper bound for the ε-upcrossing FPT pdf, is given in (17). This is clearly indicated in Fig. 2 and in Fig. 3 in which $S(t) = 3 - t$. Indeed, for the Gaussian process $X_1(t)$, in Fig. 2 the simulated function $\tilde{g}_u^{(\varepsilon)}(t)$ is compared with $W_1^{(\varepsilon)}(t)$ for $\alpha = 2$ in *(a)*, $\alpha = 1$ in *(b)*, $\alpha = 0.5$ in *(c)* and $\alpha = 0.1$ in *(d)*, by setting $\varepsilon = 0.01$ and $\omega = 1$. We note that already for $\alpha = 0.1$ (cf.

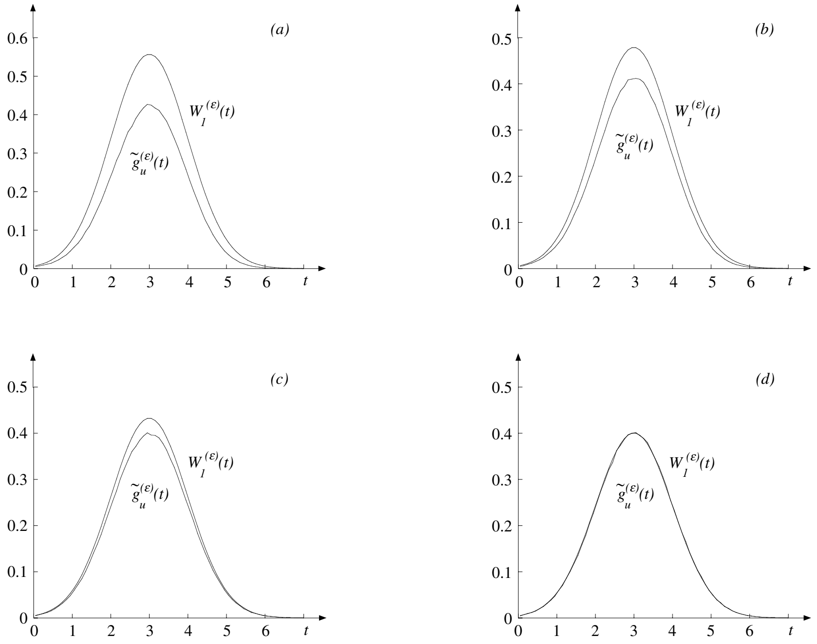


Fig. 3. Plot of $\tilde{g}_u^{(\varepsilon)}(t)$ and of $W_1^{(\varepsilon)}(t)$ for the Gaussian process with zero mean, unit variance and correlation function (31) for $S(t) = 3 - t$, and $\varepsilon = 0.01$ in the following cases: *(a)* $\beta = 2$, *(b)* $\beta = 1$, *(c)* $\beta = 0.5$ and *(d)* $\beta = 0.1$

Fig. 2*(d)*) $W_1^{(\varepsilon)}(t)$ provides a good approximation of the simulated ε-upcrossing FPT pdf. Furthermore, Proposition 1 indicates that $W_1^{(\varepsilon)}(t) \simeq \varphi^{(\varepsilon)}(t)$ for large τ_c, so that $g_u^{(\varepsilon)}(t) \simeq \varphi^{(\varepsilon)}(t)$ for all α such that $0 < \alpha < 0.1$. Instead, for the Gaussian process $X_2(t)$, in Fig. 3 the simulated function $\tilde{g}_u^{(\varepsilon)}(t)$ is compared with $W_1^{(\varepsilon)}(t)$ for $\beta = 2$ in *(a)*, $\beta = 1$ in *(b)*, $\beta = 0.5$ in *(c)* and $\beta = 0.1$ in *(d)*, by setting $\varepsilon = 0.01$. As Fig. 3*(d)* shows, already for $\beta = 0.1$, $W_1^{(\varepsilon)}(t)$ provides a good approximation of $\tilde{g}_u^{(\varepsilon)}(t)$. Hence, $g_u^{(\varepsilon)}(t) \simeq \varphi^{(\varepsilon)}(t)$ for all β such that $0 < \beta < 0.1$.

A detailed description of other simulation results for the ε-upcrossing FPT pdf of Gaussian processes will be provided in future works for different types of thresholds and for different correlation functions.

References

1. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: On a non-Markov neuronal model and its approximation. BioSystems 48 (1998) 29-35
2. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: A computational approach to first-passage-time problem for Gauss-Markov processes. Adv. Appl. Prob. 33 (2001) 453-482

3. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: On the asymptotic behavior of first passage time densities for stationary Gaussian processes and varying boundaries. Methodology and Computing in Applied Probability 5 (2003) 211-233
4. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: Computational Methods for the Evaluation of Neuron's Firing Densities. In: Moreno-Diaz, R., Pichler, F. (eds): Computer Aided Systems Theory - EUROCAST 2003. Lecture Notes in Computer Science 2809, Springer-Verlag (2003) 394-403
5. Giorno, V., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: Non-stationary Gauss-Markov Processes and Neuronal Modeling. In Trappl, R. (ed.): Cybernetics and Systems 2004 - EMCSR 2004. Austrian Society for Cybernetics Studies, Vienna (2004) 211-215
6. Lansky, P. and Smith, C.E.: The effect of a random initial value in neural first-passage-time models. Math. Biosci. 93 (1989) 191-215
7. Kostyukov, A.I.: Curve-Crossing Problem for Gaussian Stochastic Processes and its Application to Neural Modelling. Biol. Cybernetics 29 (1978) 187-191
8. Kostyukov, A.I., Ivanov, Yu. N., Kryzhanovsky, M.V.: Probability of Neuronal Spike Initiation as a Curve-Crossing Problem for Gaussian Stochastic Processes. Biol. Cybernetics 39 (1981) 157-163
9. Ogorodnikov, V.A. and Prigarin, S.M.: *Numerical Modelling od Random Processes and Fields. Algorithms and Applications.* VSP, Uthrecht, The Netherlands (1996)
10. Ricciardi, L.M.: *Diffusion Processes and Related Topics in Biology.* Springer-Verlag, New York (1977)
11. Ricciardi, L.M. and Sato, S.: On the evaluation of first–passage–time densities for Gaussian processes. Signal Processing 11 (1986) 339–357
12. Ricciardi, L.M. and Lánský, P.: Diffusion models of neuron activity. In Arbib, M.A. (ed.): The Handbook of Brain Theory and Neural Networks. The MIT Press, Cambridge (2002) 343-348
13. Roberts, J.B.: An approach to the first-passage problem in random vibration. J. Sound Vib. 8 No. 2 (1968) 301-328
14. Ross, S.M.: *Simulation.* Academic Press, San Diego (2002)

Convergence of Iterations

Paul Cull

Computer Science Dept., Oregon State University, Corvallis, OR 97331 USA
pc@cs.orst.edu

Abstract. Convergence is a central problem in both computer science and in population biology.

Will a program terminate? Will a population go to an equilibrium?

In general these questions are quite difficult – even unsolvable.

In this paper we will concentrate on very simple iterations of the form

$$x_{t+1} = f(x_t)$$

where each x_t is simply a real number and $f(x)$ is a reasonable real function with a single fixed point. For such a system, we say that an iteration is "globally stable" if it approaches the fixed point for all starting points. We will show that there is a simple method which assures global stability. Our method uses bounding of $f(x)$ by a self-inverse function. We call this bounding "enveloping" and we show that **enveloping implies global stability.** For a number of standard population models, we show that local stability implies enveloping by a self-inverse linear fractional function and hence global stability. We close with some remarks on extensions and limitations of our method.

1 Introduction

Simple population growth models have a pleasant property, they display global convergence if they have local convergence. This fact was established for a number of models by Fisher *et al* [1,2] who constructed an explicit Lyapunov function for each model they studied. Since then a number of workers have created a variety of sufficient conditions to demonstrate global stability. [3,4,5,6,7,8] Each of these methods suffer from the difficulty that either the method does not apply to one of the commonly used models or the method is computationally difficult to apply.

In this paper, we describe a simple condition which is satisfied by all the commonly used simple population models, and we show that for these models the computation for the method is not difficult. Our simple condition is that the population models are *enveloped* by *linear fractional functions*. No single linear fractional serves for all models. Instead the linear fractionals depend on a single parameter which must be adjusted for the particular model. In some cases, this parameter will also change depending on the parameters of the model. This parameter dependence may be why this simple condition has not been discovered before.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 457–466, 2005.
© Springer-Verlag Berlin Heidelberg 2005

Our pleasure with this result is not solely mathematical. There is also a psychological component. We suspect that the original creators of these population models were good biologists and not sophisticated mathematicians. If the similarity among these models required deep and complicated mathematics, we would feel that we had not captured the simple vision of the original modelers. We will argue that the usual way of writing these models suggests an implicit constraint that will force enveloping by a linear fractional.

2 Background and Definitions

In the most general sense, we want to study difference equations of the form

$$x_{t+1} = f(x_t)$$

but with this degree of generality, little can be said. If we require that f is a function which is defined for all values of x, then given an initial condition x_0, we can show that there is a unique solution to the difference equation, that is, x_t traces out a well-defined trajectory. To obtain stronger results, we will assume that f is continuous and has as many continuous derivatives as necessary. As we will see in the examples, we will assume even more structure for a population model. Intuitively, if there is no population now, there will be no population later. If the population is small, we expect it to be growing. If the population is large, we expect it to be decreasing. These ideas suggest that there should be an *equilibrium point* where the population size will remain constant. We expect the function f to be *single-humped*, that is, f should rise to a maximum and then decrease. For some models, f will go to 0 for some finite x, but for other models f will continually decrease toward 0.

We want to know what will happen to x_t for large values of t. Clearly we expect that if x_0 is near $\overline{x}$ then x_t will overshoot and undershoot $\overline{x}$. Possibly this oscillation will be sustained, or possibly x_t will settle down at $\overline{x}$. The next definitions codify these ideas. A population model is ***globally stable*** if and only if for all x_0 such that $f(x_0) > 0$ we have

$$\lim_{t\to\infty} x_t = \overline{x}$$

where $\overline{x}$ is the unique equilibrium point of $x_{t+1} = f(x_t)$. A population model is ***locally stable*** if and only if for every small enough neighborhood of $\overline{x}$ if x_0 is in this neighborhood, then x_t is in this neighborhood for all t, and

$$\lim_{t\to\infty} x_t = \overline{x}.$$

How can we decide if a model has one of these properties? The following well-known theorem gives one answer.

Theorem 1. *If $f(x)$ is differentiable then, a population model is locally stable if $|f'(\overline{x})| < 1$, and if the model is locally stable then $|f'(\overline{x})| \leq 1$.*

For global stability, a slight modification of a very general theorem of Sarkovskii [9] gives:

Theorem 2. *A continuous population model is globally stable iff it has no cycle of period 2. (That is, there is no point except $\overline{x}$ such that $f(f(x)) = x$.)*

This theorem has been noted by Cull[7] and Rosenkranz[4].

Unfortunately, this global stability condition may be difficult to test. Further, there is no obvious connection between the local and global stability conditions.

Various authors have demonstrated global stability for some population models. Fisher *et al* [1] and Goh [2] used Lyapunov functions [10] to show global stability. This technique suffers from the drawbacks that a different Lyapunov function is needed for each model and that there is no systematic method to find these functions. Singer [3] used the negativity of the Schwarzian to show global stability. This technique does not cover all the models we will consider, and it even requires modification to cover all the models it was claimed to cover. Rosenkranz [4] noted that no period 2 was implied by $|f'(x)f'(f(x))| < 1$ and showed that this condition held for a population genetics model. This condition seems to be difficult to test for the models we will consider. Cull [7,6,5,8] developed two conditions **A** and **B** and showed that each of the models we will consider satisfied at least one of these conditions. These conditions used the first through third derivatives and so were difficult to apply. Also, as Hwang [11] pointed out these conditions required continuous differentiability. All of these methods are relatively mathematically sophisticated, and so it is not clear how biological modelers could intuitively see that these conditions were satisfied.

If we return to the condition for local stability, we see that it says if for x slightly less than 1, $f(x)$ is below a straight line with slope -1, and if for x slightly greater than 1, $f(x)$ is above the same straight line, then the model is locally stable. If we consider the model

$$x_{t+1} = x_t e^{2(1-x_t)},$$

we can see that the local stability bounding line is $2 - x$. Somewhat suprisingly, this line is an upper bound on $f(x)$ for all x in $[0, 1)$ and a lower bound for all $x > 1$. (See Figure 1a). Since $2 - (2 - x) = x$, the bounding by this line can be used to argue that for this model there are no points of period 2, and hence the model is globally stable. From this example, we abstract the following definition. A function $\phi(x)$ ***envelops*** a function $f(x)$ if and only if

$$\begin{array}{lll} \phi(x) > f(x) & \text{for} \quad x \in (0,1) & \\ \phi(x) < f(x) & \text{for} \quad x > 1 & \text{such that} \quad \phi(x) > 0 \quad \text{and} \quad f(x) > 0 \end{array}$$

We will use the notation $\phi(x) \bowtie f(x)$ to symbolize this enveloping.

As we will see, our example population models have one or more parameters, and a model with one choice of parameters will envelop the same model with a different choice of parameters. For example, the function $xe^{2(1-x)}$ envelops all the functions of the form $xe^{r(1-x)}$ for $r \in (0, 2)$.

While a straight line was sufficient to envelop $xe^{2(1-x)}$, a straight line fails to envelop the closely related function $x[1 + 2(1 - x)]$. To get a more general enveloping function, we consider the ratio of two linear functions and assume

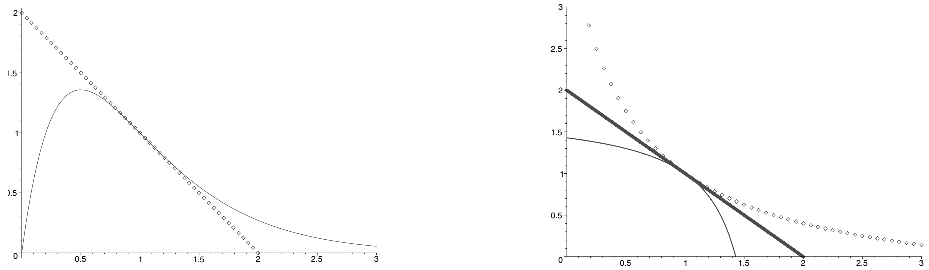


Fig. 1. (a) The function $xe^{2(1-x)}$ is enveloped by the straight line $2 - x$ which is the linear fractional with $\alpha = 1/2$. (See Model I in Section 4.). (b) Three types of linear fractionals. Dotted line $\alpha = 1/4$. Heavy line $\alpha = 1/2$. Light line $\alpha = .7$.

that the ratio is 1 when $x = 1$ and the derivative of this function is -1 when $x = 1$, which gives the following definition.

A ***linear fractional function*** is a function of the form

$$\phi(x) = \frac{1 - \alpha x}{\alpha - (2\alpha - 1)x} \qquad \text{where } \alpha \in [0, 1) \ .$$

These functions have the properties

$$\begin{aligned} \phi(1) &= 1 \\ \phi'(1) &= -1 \\ \phi(\phi(x)) &= x \\ \phi'(x) &< 0. \end{aligned}$$

The shape of our linear fractional functions changes markedly as α varies. For $\alpha = 0$, $\phi(x) = 1/x$, which has a pole at $x = 0$, and decreases with an always positive second derivative. For $\alpha \in (0, 1/2)$, $\phi(x)$ starts (for $x = 0$) at $1/\alpha$ and decreases with a positive second derivative. For $\alpha = 1/2$, $\phi(x) = 2 - x$, which starts at 2 and decreases to 0 with a zero second derivative. For $\alpha \in (1/2, 1)$, $\phi(x)$ starts at $1/\alpha$, decreases with a negative second derivative, and hits 0 at $1/\alpha$ which is greater than 1. We are only interested in these functions when $x > 0$ and $\phi(x) > 0$, so we do not care about the pole in these linear fractionals because the pole occurs outside the area of interest. Figure 1b shows the three different shapes of linear fractional functions.

3 Theorems

We are now in a position to state the necessary theorems. In what follows, we will assume that our model is $x_{t+1} = f(x_t)$, and that the model has been normalized so that the equilibrium point is 1, that is $f(1) = 1$. We will use the notation $f^{(k)}(x)$ to mean that the function f has been applied k times to x. This notation can be recursively defined by $f^{(0)}(x) = x$ and $f^{(i)}(x) = f(f^{(i-1)}(x))$ for $i \geq 1$.

Theorem 3. *Let $\phi(x)$ be a monotone decreasing function which is positive on $(0, x_-)$ and so that $\phi(\phi(x)) = x$. Assume that $f(x)$ is a continuous function such that:*

$$\begin{aligned} \phi(x) &> f(x) \quad \textit{on} \quad (0,1) \\ \phi(x) &< f(x) \quad \textit{on} \quad (1,x_-) \\ f(x) &> x \quad \textit{on} \quad (0,1) \\ f(x) &< x \quad \textit{on} \quad (1,\infty) \\ f(x) &> 0 \quad \textit{on} \quad (1,x_\infty) \end{aligned}$$

then for all $x \in (0, x_\infty)$, $\lim_{k\to\infty} f^{(k)}(x) = 1$.

A slight recasting of the above gives:

Corollary 1. *If $f_1(x)$ is enveloped by $f_2(x)$, and $f_2(x)$ is globally stable, then $f_1(x)$ is globally stable.*

Corollary 2. *If $f(x)$ is enveloped by a linear fractional function then $f(x)$ is globally stable.*

A function $h(z)$ is ***doubly positive*** iff

1. $h(z)$ has a power series $\sum_{i=0}^{\infty} h_i z^i$
2. $h_0 = 1$, $h_1 = 2$
3. For all $n \geq 1 \quad h_n \geq h_{n+1}$
4. For all $n \geq 2 \quad h_n - 2h_{n+1} + h_{n+2} \geq 0$

Theorem 4. *Let $x_{t+1} = f(x_t)$ where $f(x) = xh(1-x)$ and $h(z)$ is doubly positive, then $f(x)$ is enveloped by the linear fractional function*

$$\phi(x) = \frac{1-\alpha x}{\alpha + (1-2\alpha)x}$$

where $\alpha = \frac{3-h_2}{4-h_2} \geq \frac{1}{2}$ and the model $x_{t+1} = f(x_t)$ is globally stable.

While this doubly positive condition will be sufficient for a number of models, it is not sufficient for all the examples. The following observation will be useful in many cases.

Observation 1. *Let $\phi(x) = A(x)/B(x)$, $f(x) = C(x)/D(x)$ and $G(x) = A(x)D(x) - B(x)C(x)$. If $G(1) = 0$, $G'(1) = 0$, and $G''(x) > 0$ on $(0,1)$ and $G''(x) < 0$ for $x > 1$, then $\phi(x)$ envelops $f(x)$. (We are implicitly assuming that A, B, C, D are all positive, and all functions are twice continuously differentiable.)*

4 Simple Models of Population Growth

In this section we will apply the techniques of the previous section to 7 models from the literature. Models I, II, III, IV all turn out to be doubly positive and so we just give the model and the enveloping fractional.

Model I: The model $x_{t+1} = x_t e^{r(1-x_t)}$ is widely used (see, for example [12–14]). This model with $r = 2$ is enveloped by $\phi(x) = 2 - x$ and hence local and global stability coincide.

Model II: The model $x_{t+1} = x_t[1+r(1-x_t)]$ is widely used [12] and is sometimes considered to be a truncation of Model I. The enveloping function has $\alpha = \frac{3}{4}$ and is

$$\phi(x) = \frac{4-3x}{3-2x}.$$

Model III: The model $x_{t+1} = x_t[1 - r\ln x_t]$ is attributed to Gompertz and studied by Nobile *et al*[13]. As with the preceding two models $0 < r \leq 2$ is the necessary condition for local stability. The enveloping function has $\alpha = 2/3$ and is $\phi(x) = \frac{3-2x}{2-x}$.

Model IV: This model from [14] is

$$x_{t+1} = x_t(\frac{1}{b + cx_t} - d).$$

It differs from the previous three in that there are two parameters, b and d, remaining after the carrying capacity has been normalized to 1. The enveloping function is

$$\phi(x) = \frac{4d - (3d-1)x}{3d - 1 + 2(1-d)x}.$$

We note that $\phi(x)$ has a pole, but $\phi(x)$ goes to zero before the pole, so we can simply ignore the pole. Of course, we only need $\phi(x)$ to bound $f(x)$ on the interval $(0, \frac{4d}{3d-1})$ where $\phi(x)$ is positive.

Model V: This model from Pennycuick *et al* [15] has

$$f(x) = \frac{(1 + ae^b)x}{1 + ae^{bx}}.$$

This and the following two model are more complicated than the previous models because we have to consider different enveloping functions for different parameter ranges. For $b \leq 2$, $xe^{b(1-x)}$ envelops $f(x)$. But $xe^{b(1-x)}$ is just Model I, and as we showed it is enveloped by $2 - x$.

For larger values of b, we use $a(b-2)e^b = 2$ from local stablity, and show that the enveloping linear fractional is

$$\phi(x) = \frac{b - (b-1)x}{(b-1) - (b-2)x}$$

by using the Observation.

Model VI: Model VI is from Hassel [16] and has

$$f(x) = \frac{(1+a)^b x}{(1+ax)^b} \qquad \text{with } a > 0,\, b > 0.$$

There are two cases to consider $0 < b \leq 2$ and $b > 2$. The enveloping function for $b \leq 2$ is $\phi(x) = 1/x$. Cross multiplication shows that we want $(1 + ax)^b \bowtie$

$(1+a)^b x^2$. Taking b^{th} roots and rearranging shows that we want $1 - x + ax(1 - x^{\frac{2\ \ b}{b}}) \bowtie 0$. Clearly, each of the two terms is positive (nonnegative) below 1 and negative (nonpositive) above 1, and so enveloping is established. For $b > 2$, we need to use the Obsevation to establish enveloping.

Model VII: Model VII is due to Maynard Smith [17] and has

$$f(x) = \frac{rx}{1 + (r-1)x^c}.$$

This seems to be the hardest to analyze model in our set of examples. For example, this model does not satisfy the Schwarzian derivative condition or Cull's condition **A**. Even for our enveloping analysis, we will need to consider this model as three subcases.

Similar to previous models, local stability implies $r(c-2) \le c$, and it is easy to show that this model with smaller values of r is enveloped by this model with larger values of r. For $c > 2$, we use $r = \frac{c}{c-2}$, and

$$\phi(x) = \frac{c - 1 - (c-2)x}{c - 2 - (c-3)x}.$$

For $c > 3$, the Observation shows enveloping, but for $c \in (2, 3)$ consideration of the third derivative is needed to show enveloping.

5 Enveloping by a Linear Fractional Is Only Sufficient

Here we want to give a simple model which has global stability, but cannot be enveloped by any linear fractional. Define $f(x)$ by

$$f(x) = \begin{cases} 6x & 0 \le x < 1/2 \\ 7 - 8x & 1/2 \le x < 3/4 \\ 1 & 3/4 \le x. \end{cases}$$

then $x_{t+1} = f(x_t)$ has $x = 1$ as its globally stable equilibrium point because if $x_t \ge 1$ then $x_{t+1} = 1$, for $x_t \in [1/2, 1)$, $x_{t+1} > 1$ and $x_{t+2} = 1$, and for $x_t \in (0, 1/2)$, the subsequent iterates grow by multiples of 6 and eventually surpass $1/2$. This $f(x)$ cannot be enveloped by a linear fractional because $f(1/2) = 3$ which implies that the linear fractional would have $\alpha \le -1$ and hence have a pole in $(0, 1)$ and thus it could not envelop a positive function. On the other hand, the self-inverse function

$$\phi(x) \;=\; \begin{cases} 5 - 4x & x \le 1 \\ (5 - x)/4 & x > 1 \end{cases}$$

does envelop $f(x)$ and so demonstrates global stability.

6 Extensions

The previous sections have worked with the usual applied math assumption that real phenomena are as smooth and as differentiable as necessary to get a good theorem or estimate. Of course, everyone who has ever applied mathematics knows that this assumption is false, but they also know that it serves as a useful "rule of thumb." That is, in some cases the smoothness assumption may lead to bad estimates, but in many, many cases the smooth estimate is very close to observed (experimental) values. In a few cases, a result which was initially proved assuming smoothness has been shown to hold when some of the smoothness assumptions are dropped. Here we want to mention that **enveloping implies global stability** does *not* require continuity, even though we originally assumed continuity. Further, the assumption that x_{t+1} is a *function* of x_t is also superfluous. The enveloping result will also hold for *multi-functions*, which are mappings in which $f(x)$ may return any one of several values or any value within some range. Discontinuous functions can have points y so that for some x_0's, $\lim_{k\to\infty} f^{(k)}(x_0) = y$, but $f(y) \neq y$. We call such y's *limiting points.* To apply our theorem, one must show that no such limiting points exist within the range of interest.

6.1 General Theorem

Although our enveloping method was devised for the population models discussed above, the method can also be applied to other iterations. Not all iterations are normalized so that the fixed point is at $x = 1$. In many cases, the iteration is designed to compute the fixed point.

Theorem 5. *If the iteration* $x_{t+1} = f(x_t)$ *obeys*

$$\begin{array}{lcl} f(x) > x & \text{on} & (a\,,\,p) \\ f(x) < x & \text{on} & (p\,,\,b) \end{array}$$

where $f(x)$ *may be a discontinuous multifunction but has* p *as its only fixed point or limiting point in* $(a\,,\,b)$*, and if there is a self-inverse function* $\phi(x)$ *so that*

$$\begin{array}{lcl} \phi(x) > f(x) & \text{on} & (a\,,\,p) \\ f(x) > \phi(x) & \text{on} & (p\,,\,b) \end{array}$$

then $\lim_{k\to\infty} f^{(k)}(x_0) = p$ *for every* $x_0 \in (a\,,\,b)$*.*

6.2 Some Newton Iterations

For our examples, we'll consider the Newton iterations for square root and for reciprocal. As is well known [18], $\sqrt{A}$ can be computed by the iteration

$$x_{t+1} = \frac{x_t^2 + A}{2x_t}.$$

Clearly this iteration has $\sqrt{A}$ as its sole fixed point on $(0\,,\,\infty)$ and the continuous function obeys

$$\begin{array}{lll} f(x) \,>\, x & \text{on} & (0\,,\,\sqrt{A}) \\ f(x) \,<\, x & \text{on} & (\sqrt{A}\,,\,\infty). \end{array}$$

We take $\phi(x) \;=\; A/x$ and its easy to check that $\phi(x)$ does envelop $f(x)$ on $(0\,,\,\infty)$. So we conclude that for any $x_0 \;\in\; (0\,,\,\infty)$, this Newton iteration will converge to $\sqrt{A}$.

For a slightly more complicated example, we use the well known [18] iteration

$$x_{t+1} \;=\; x_t(2 - Ax_t).$$

to compute $1/A$. Here $f(x)$ has $1/A$ as its sole fixed point in $(0\,,\,2/A)$. Notice that $f(0) = f(2/A) = 0$ so this iteration will not converge to $1/A$ when it is started at either of these fixed points. We can take the straight line $\phi(x) \;=\; 2/A \;-\; x$ and show that this $\phi(x)$ does envelop $f(x)$ on $(0\,,\,2/A)$ and hence that this iteration converges to $1/A$ when started at any point within $(0\,,\,2/A)$.

7 Conclusion

Enveloping is a simple technique to demonstrate global stability for some one-dimensional difference equations. Enveloping was introduced by Cull and Chaffee [19,20,21]. We demonstrated that the usual population models can be enveloped by linear fractional functions. Such enveloping seems to capture the idea of *simple* function in that a “free-hand” drawing of a population model can usually be enveloped by a linear fractional. (Cull [22] gives a discussion of dynamical systems defined by linear fractionals.) As we showed by example, enveloping by a linear fractional is only a *sufficient* condition for global stability. The simplest population models which have local stability without global stability are discussed by Singer [3] and by Cull [5]. While most of the examples in this paper are all one-humped population models, **enveloping implies global stability** also holds for functions with multiple peaks, for discontinuous functions, and even for multi-functions.

References

1. Fisher, M., Goh, B., Vincent, T.: Some Stability Conditions for Discrete-time Single Species Models. Bulletin of Mathematical Biology **41** (1979) 861–875
2. Goh, B.S.: Management and Analysis of Biological Populations. Elsevier, New York (1979)
3. Singer, D.: Stable Orbits and Bifurcation of Maps of the Interval. SIAM Journal on Applied Mathematics **35** (1978) 260–267
4. Rosenkranz, G.: On Global Stability of Discrete Population Models. Mathematical Biosciences **64** (1983) 227–231

5. Cull, P.: Stability of Discrete One-dimensional Population Models. Bulletin of Mathematical Biology **50** (1988) 67–75
6. Cull, P.: Local andGlobal Stability for Population Models. Biological Cybernetics **54** (1986) 141–149
7. Cull, P.: Global Stability of Population Models. Bulletin of Mathematical Biology **43** (1981) 47–58
8. Cull, P.: Local and Global Stability of Discrete One-dimensional Population Models. In Ricciardi, L.M., ed.: Biomathematics and Related Computational Problems. Kluwer, Dordrecht (1988) 271–278
9. Sarkovskii, A.: Coexistence of Cycles of a Continuous Map of a Line to Itself. Ukr. Mat. Z. **16** (1964) 61–71
10. LaSalle, J.: The Stability of Dynamical Systems. SIAM, Philadelphia (1976)
11. Huang, Y.: A Counterexample for P. Cull's Theorem. Kexue Tongbao **31** (1986) 1002–1003
12. Smith, J.: Mathematical Ideas in Biology. Cambridge University Press, Cambridge (1968)
13. Nobile, A., Ricciardi, L., Sacerdote, L.: On Gompertz Growth Model and Related Difference Equations. Biological Cybernetics **42** (1982) 221–229
14. Utida, S.: Population Fluctuation, an Experimental and Theoretical Approach. Cold Spring Harbor Symposium on Quantitative Biology **22** (1957) 139–151
15. Pennycuick, C., Compton, R., Beckingham, L.: A Computer Model for Simulating theGrowth of a Population, or of Two Interacting Populations. Journal of Theoretical Biology **18** (1968) 316–329
16. Hassel, M.: Density Dependence in Single Species Populations. Journal of Animal Ecology **44** (1974) 283–296
17. Smith, J.: Models in Ecology. Cambridge University Press, Cambridge (1974)
18. Cull, P., Flahive, M., Robson, R.: Difference Equations: from Rabbits to Chaos. Springer, New York (2005)
19. Cull, P., Chaffee, J.: Stability in simple population models. In Trappl, R., ed.: Cybernetics and Systems 2000. Austrian Society for Cybernetics Studies (2000) 289–294
20. Cull, P., Chaffee, J.: Stability in discrete population models. In Dubois, D.M., ed.: Computing Anticipatory Systems: CASYS'99. Conference Proceedings 517, American Institute of Physics, Woodbury, NY (2000) 263–275
21. Cull, P.: Stability in one-dimensional models. Scientiae Mathematicae Japonicae **58** (2003) 349–357
22. Cull, P.: Linear Fractionals - Simple Models with Chaotic-like Behavior. In Dubois, D.M., ed.: Computing Anticipatory Systems:CASYS 2001 - Fifth International Conference, Conference Proceedings 627. American Institue of Physics, Woodbury, N.Y. (2002) 170–181

Semiautomatic Snake-Based Segmentation of Solid Breast Nodules on Ultrasonography

Miguel Alemán-Flores[1], Patricia Alemán-Flores[2], Luis Álvarez-León[1],
M. Belén Esteban-Sánchez[1], Rafael Fuentes-Pavón[2],
and José M. Santana-Montesdeoca[2]

[1] Departamento de Informática y Sistemas,
Universidad de Las Palmas de Gran Canaria, 35017, Las Palmas, Spain
[2] Sección de Ecografía, Servicio de Radiodiagnóstico Hospital,
Universitario Insular de Gran Canaria, 35016, Las Palmas, Spain
`maleman@dis.ulpgc.es`

Abstract. Ultrasonography plays a crucial role in the diagnosis of breast cancer. However, it is one of the most difficult types of images to segment and analyze. The presence of speckle noise and low contrast areas limits the success of most noise reduction filters and segmentation algorithms. In this paper, we propose a combination of different techniques which provide quite satisfactory results in the segmentation of breast tumors on ultrasonography. It is performed in a semiautomatic way, which eliminates the need for a manual delineation of the contour of the nodules. These techniques include the truncated median filter, a region-growing algorithm and active contours. Furthermore, this can be the initial phase for an exhaustive analysis of the diagnostic criteria in breast ultrasound.

1 Introduction

Medical imaging can be very useful for the early detection of breast cancer. The most commonly used types of images are mammography and ultrasonography. For the latter case, radiologists have described a series of criteria which help deciding whether a solid breast nodule is malignant or benignant by analyzing ultrasound images [1]. These factors involve a precise visual examination of the shape of the nodule and its contour. Thus, a measurement of how ellipsoid the nodule is, the extraction of the ramifications, or the location of microlobulations, angular margins and spiculations require the analysis of the whole contour or certain parts of it. For this reason, it is very important to obtain an accurate delimitation of the nodule boundary. Computer Vision techniques provide some methods for the extraction of the contours, but, due to the special characteristics of ultrasound images, they must be adapted to obtain satisfactory results. This paper presents a new approach for the segmentation of breast tumors using active contours, combined with other techniques, such as the truncated median filter or the structure tensor.

Several semiautomatic segmentation methods in ultrasound images have previously been proposed. These methods include pixel and region-based segmentation, edge-based segmentation and hybrid techniques [2][3]. The success of these

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 467–472, 2005.
© Springer-Verlag Berlin Heidelberg 2005

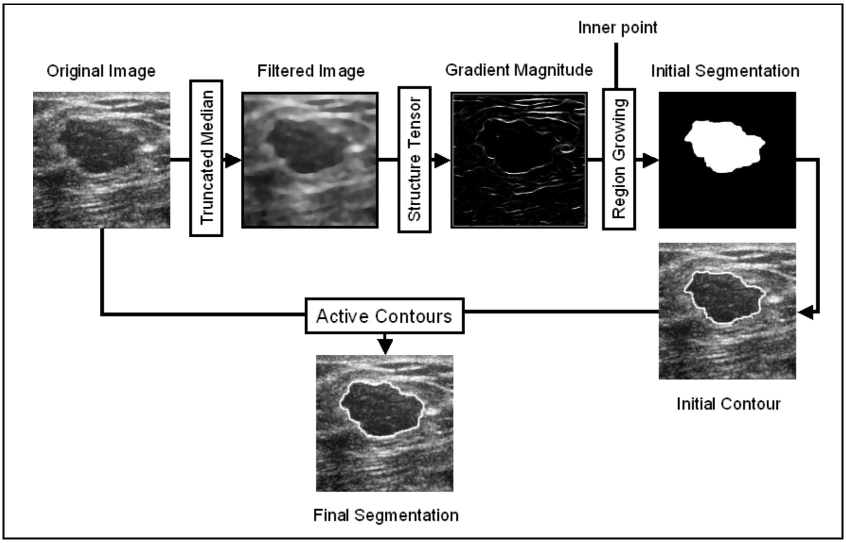


Fig. 1. General scheme of the segmentation method: truncated median, structure tensor and a region-growing algorithm are used to extract an initial segmentation of the nodule, whereas geodesic active contours are applied to obtain a more precise contour

techniques is moderate because they cannot obtain a precise shape of the region of interest due to the special characteristics of ultrasound images or the requirement of a priori knowledge of the shape.

Region-growing algorithms do not require an initial approximation, but produce quite inaccurate segmentations. On the other hand, active contours techniques generate very satisfactory results provided the initialization is close enough. This is the reason why we propose a combination of both kinds of methods. Due to the presence of speckle noise, a previous filtering of the image is required. Figure 1 shows a general scheme of our method. The rest of the paper is structured as follows: Section 2 explains how the initial segmentation is obtained. Section 3 presents the use of the active contours technique to improve the segmentation. Finally, in Sect. 4, we give an account of our main conclusions.

2 Initial Segmentation

In order to reduce the speckle noise which characterizes ultrasound images, we have used the truncated median filter [4]. This consists in an iterative filter which, for every pixel, approximates the mode of the region by means of the median of the most representative values. In a few iterations, the image is much more suitable for the application of a region-growing algorithm. On the filtered image, we apply two 3x3 masks to estimate the gradient in every pixel and, by means of the structure tensor, we obtain a better estimation of the magnitude of the gradient. From a seed point introduced by the specialist, the selection grows

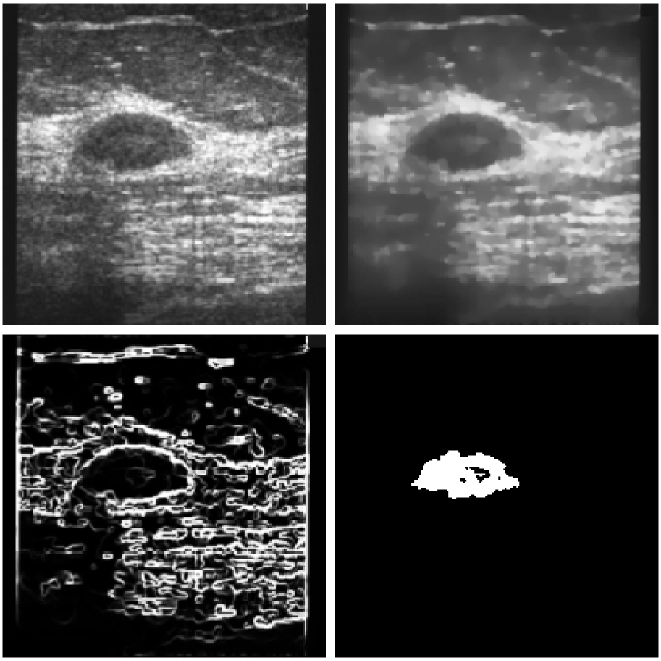


Fig. 2. From top to bottom and from left to right: original image of a nodule in ultrasonography, filtered image, magnitude of the gradient using the structure tensor, and initial segmentation by means of the region-growing algorithm

until a certain threshold for the magnitude of the gradient is reached. Figure 2 shows an example of an ultrasound image of a nodule, the filtered image, the magnitude of the gradient, and the initial segmentation of the nodule obtained through the region-growing algorithm.

3 Active Contours

The active contours technique [5][6], also called snakes, is recognized as one of the most efficient tools for image segmentation. This method consists in deforming an initial contour of the object under a set of internal and external forces. Several difficulties appear when applying this model to ultrasound images because of ultrasound characteristic speckle noise, the requirement of an initial outline of the nodule boundary to start the algorithm, and the need for a good adequacy of the external forces which guide the snake.

In our case, once the region-growing algorithm has been applied, the contour of the selected area is considered as the initial snake for the evolution of the active contours, based on the following equation:

$$\frac{\partial u(x,y)}{\partial t} = g_\sigma(I(x,y))\,|\bigtriangledown u(x,y)|\,div\left(\frac{\bigtriangledown u(x,y)}{|\bigtriangledown u(x,y)|}\right) + \lambda \nabla g_\sigma(I(x,y)) \bigtriangledown u(x,y) \quad (1)$$

where $u(x, y; 0) = u_0(x, y)$ is the initial snake contour, $\lambda \geq 0$, and $g_\sigma(I(x, y))$ represents the stopping function that attracts the snake $u(x, y)$ to the real contour of the object. It is an edge detector that must be selected according to the characteristics of the image. A typical choice of $g_\sigma(I(x, y))$ is:

$$g_\sigma(I(x,y)) = \frac{1}{\sqrt{1 + \alpha \left| \nabla I_\sigma(x,y) \right|^2}} \tag{2}$$

where $\alpha \geq 0$ is a constant and $I_\sigma(x, y)$ denotes the smoothed version of the original image, i.e. $I(x, y)$ has been convolved with a Gaussian kernel with standard deviation σ. We have used a multiscale implementation based on the geodesic active contours proposed in [6] and the level-set method proposed in [7]. Given the final standard deviation σ_0 and the number of scales to apply Ns, the corresponding standard deviations for the different scales are calculated according to the following expression:

$$\sigma_n = (n+1)\sigma_0 \quad n = Ns - 1, .., 0$$

For a given scale k, the initial contour is given by the final contour for scale $(k + 1)$, which has been calculated previously, except for the first case, in which

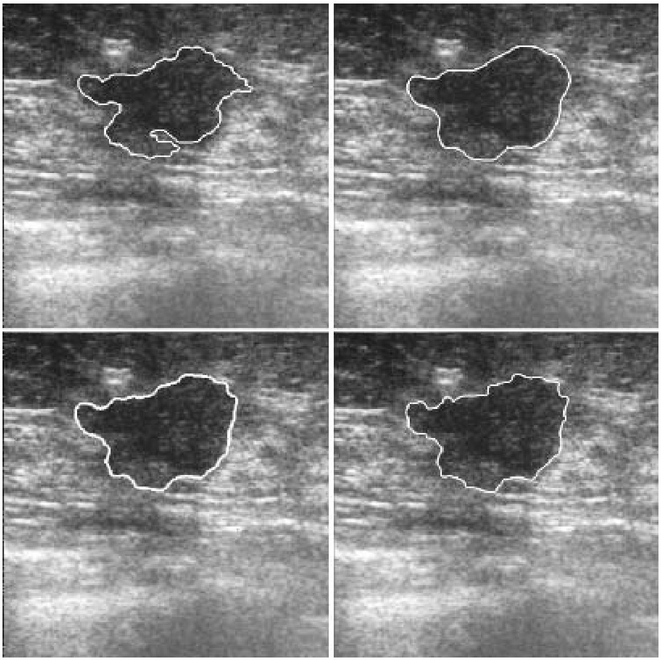


Fig. 3. Evolution of the snake: initial approximation obtained with the region-growing algorithm and final snakes using three different scales ($3\sigma_0$, $2\sigma_0$ and σ_0)

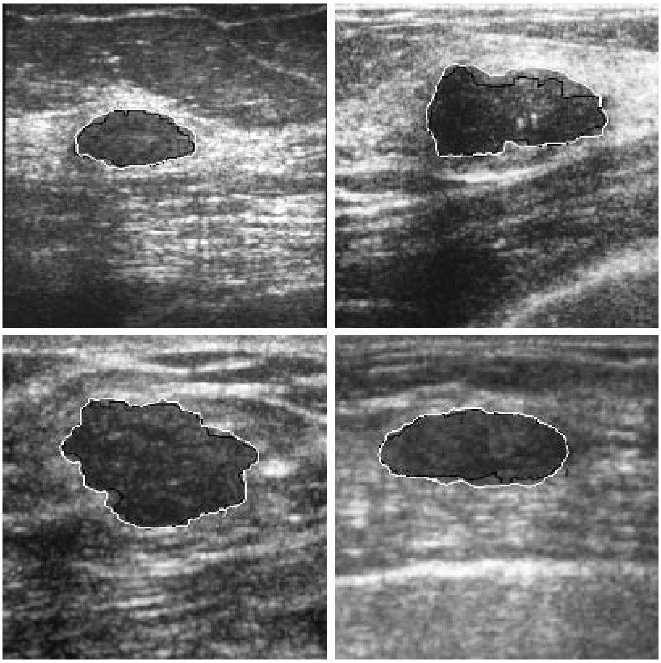


Fig. 4. Comparison of initial contours (black) and final contours (white) in four different nodules

the segmentation provided by the region-growing algorithm is used as initial approximation.

We have adapted snake adjustment from the classical model to the special characteristics of ultrasonography, and we have compared the most representative methods for speckle reduction in ultrasound images. Initially, the best results have been obtained using the truncated median filter and the geometric filters [8], but the first one was faster.

Figure 3 shows the evolution of the snakes as the different scales and iterations are applied. The multiscale implementation allows adapting the values of the parameters α and λ at each scale. This results in a faster evolution without decreasing the accuracy. Figure 4 shows a comparison of the initial and final segmentations. As observed, the active contours technique extracts the real contours of the nodules in a more accurate way.

4 Conclusion

We have proposed a new method for the segmentation of breast tumors in ultrasonography. The combination of a region-growing algorithm, guided by the structure tensor, and the active contours have provided quite promising results. The use of a region-growing algorithm does not provide very accurate segmentations, but it generates an initial contour for the active contours technique, in such a way that only a single inner point is needed, instead of a large set of contour points or a whole delineation. We have tested the proposed scheme on a set of breast ultrasound images. The comparison of the results provided by the system and those supplied by the specialists through manual delineation shows the accuracy and reliability of the proposed technique, and emphasizes its usefulness for the further analysis of the diagnostic criteria used in the classification of the nodules.

References

1. Stavros, A.T., Thickman, D., Rapp, C.L., Dennis, M.A., Parker, S.H., Sisney, G.A.: Solid breast nodules: use of sonography to distinguish between benign and malignant lesions, Radiology 196 (1995) 123-134
2. Drukker, K., Giger, M.L., Horsch, K., Kupinski, M.A. and Vyborny, C.J.: Computerized lesion detection on breast ultrasound. Medical Physics 29:7 (2002) 1438-1446
3. Revell, J., Mirmehdi, M. and McNally, D.: Applied review of ultrasound image feature extraction methods. In A Houston and R Zwiggelaar, editors, The 6th Medical Image Understanding and Analysis Conference, BMVA Press (2002) 173–176
4. Davis, E.R.: On the noise suppression and image enhancement characteristics of the median, truncated median and mode filters. Pattern Recognition Letters 7 (1988) 87-97
5. Kass, M., Witkin, A. and Terzopoulos, D.: Snakes: Active Contour Models. In 1st International Conference on Computer Vision (1987) 259-268
6. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic Active Contours, International Journal of Computer Vision, 22:1 (1997) 61-79
7. Osher, S., Sethian, J.: Fronts propagating with curvature dependent speed: algorithms based on the Hamilton-Jacobi formulation. Journal of Computational Physics, 79 (1988) 12-49
8. Busse, L.J., Dietz, D.R., Glenn, W.M. jr: A non-linear algorithm for speckle reduction, IEEE Symposium on Ultrason. Ferroelec. and Freq. Contr., IEEE 91CH3079-1 (1991) 1105-1108

Parallel Progressive Multiple Sequence Alignment

Erik Pitzer

University of Applied Sciences Hagenberg, Austria
`Erik.Pitzer@fh-hagenberg.at`

Abstract. Multiple Sequence Alignment is an essential tool in the analysis and comparison of biological sequences. Unfortunately, the complexity of this problem is exponential. Currently feasible methods are, therefore, only approximations. The *progressive* multiple sequence alignment algorithms are the most widespread among these approximations. Still, the computation speed of typical problems is often not satisfactory. Hence, the well known progressive alignment scheme of *ClustalW* has been subject to parallelization to further accelerate the computation. In the course of this action a unique scheme to parallelize sequence alignment in particular and dynamic programming in general was discovered, which yields an average of $n/2$ parallel calculations for problem size n. The scalability of $O(n)$ tasks for problem size n can be even maintained for slower networks.

1 Problem Description

Progressive multiple sequence alignment—although just an approximation of multiple sequence alignment—is often still too slow for typical applications (e.g. 80 sequences of length 3,000). Therefore, we want to further accelerate this algorithm system by parallelization.

2 Introduction and Notion

2.1 Sequence Alignment

Sequence alignment is a method to determine the similarity between two sequences $s_1, s_2 \in \mathrm{S}$ where the set of all sequences S is defined as $\{s \mid s : \mathbb{N} \rightarrow \mathcal{A}\}$ over the alphabet $\mathcal{A}$. Sequence alignment is a strong optimization problem which is usually solved by dynamic programming. A simple recurrence relation like in [1] for this sequence alignment problem can be formulated as $\mathrm{f}(i, j) = \min(\mathrm{f}(i-1, j) + g, \mathrm{f}(i, j-1) + g, \mathrm{f}(i-1, j-1) + \mathrm{cmp}(s_1(i), s_2(j))$ where cmp() is the comparison of two characters and g is the gap cost. Often the treatment of gaps is more complex, with gap opening and gap extension costs or even position-specific gap penalties. For the proposal of the distributed calculation

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 473–482, 2005.
© Springer-Verlag Berlin Heidelberg 2005

scheme we can omit these extensions without loss of generality. The comparison function cmp() can be defined implicitly as matrix of conversion costs

$$\mathrm{W_C} = \left\{ \begin{pmatrix} w_{1,1} & \cdots & w_{1,n} \\ \vdots & \ddots & \vdots \\ w_{n,1} & \cdots & w_{n,n} \end{pmatrix} \middle| \; w_{i,j} = \mathrm{cmp}(\alpha(i), \alpha(j)), \alpha : \mathbb{N} \to \mathcal{A} \right\}$$

over the ordered alphabet α. Together with the domain of gap weightings, $\mathrm{W_G}$, which can be either just a simple value as well as a set of functions, depending on the application, the whole weighting scheme will be just referenced as an element of $\mathrm{W} = \{(w_c, w_g) \mid w_c \in \mathrm{W_C}, w_g \in \mathrm{W_G}\}$.

A more demonstrative way of viewing the dynamic programming scheme of sequence alignment is a matrix of values, where every cell in that matrix depends upon three precedent values as shown in Figure 1. This matrix can be seen as a table of sub-solutions. Every cell (i, j) represents the minimal alignment cost of the two substrings $s_1[1, i]$ and $s_2[1, j]$. The bottom right element contains then the overall solution. As can be seen, the solution of one sub-problem is computed with the help of three slightly simpler problems as described in [2] and [3].

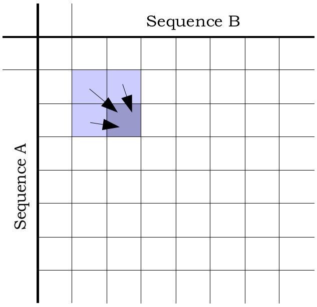


Fig. 1. Sequence Alignment

2.2 Multiple Sequence Alignment

Often the comparison of two sequences does not provide for the inclusion of enough context to obtain meaningful results. It is due to random features of the sequences that might not even come into play for the corresponding individual. Therefore, the simultaneous alignment of multiple sequences is necessary to obtain significant results. If the dynamic programming scheme is simply extended for this purpose, the matrix as shown in Figure 1 becomes multidimensional, with one dimension per sequence. This incurs an exponential runtime with respect to the number of sequences which prevents the direct application of dynamic programming for multiple sequence alignment problems. Figure 2 on the facing page shows this extension of sequence alignment. As you can see, the number of precedent values that contribute also increases quite enormously considering the fact that only one sequence was added.

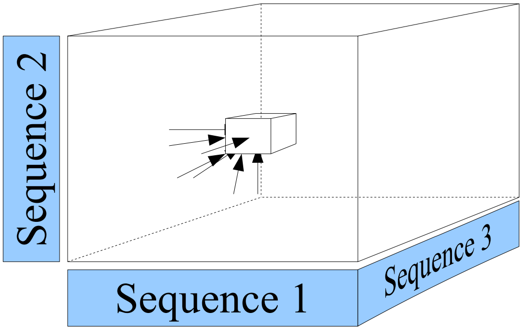
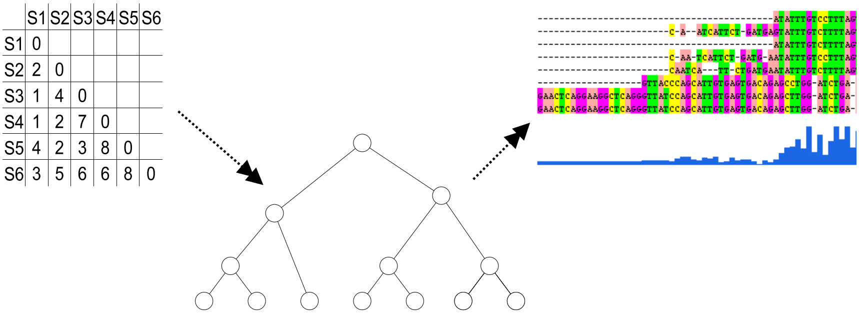


Fig. 2. Multiple Sequence Alignment



	S1	S2	S3	S4	S5	S6
S1	0					
S2	2	0				
S3	1	4	0			
S4	1	2	7	0		
S5	4	2	3	8	0	
S6	3	5	6	6	8	0

Fig. 3. Progressive Multiple Alignment

2.3 Progressive Multiple Sequence Alignment

The problem of multiple sequence alignment is not computable in reasonable time. Therefore, an approximation is calculated following a progressive alignment scheme. The idea is to replace one multidimensional calculation by a series of two-dimensional calculations. To accomplish this, the algorithm for pairwise alignment is *re-used* for multiple sequences, treating several sequences as a single sequence. In this way, smaller alignments (i.e. not containing all the sequences) can be combined to bigger alignments within quadratic computation time. To obtain reasonable results, however, the order in which these *sub-alignments* are combined is significant. Therefore, two precedent stages are necessary: First, every sequence is aligned with every other sequence to produce a pairwise distance matrix. With the help of this matrix, the phylogenetic relationship, i.e. the evolution between the sequences, can be estimated in a second step. The result of these two precedent stages is a phylogenetic tree—a tree of evolution, also called *guide tree*. Along the branches towards the root of this guide tree, the sequences are combined to sub-alignments and finally to the overall alignment. An overview of this process is depicted in Figure 3. Even if this scheme is now more complicated, the overall computation is much less complex: Instead of $O(l^n)$ we now have $O(l^2 \cdot n^2 \cdot \log n)$, where n is the number of sequences and l is their average length. Through extension of this scheme by consideration of physical and chemical processes, the approximation for biological sequences is often even more meaningful and consistent with the real development of the sequences, than the 'mathematically optimal' sequence alignment could have been. A very popular implementation of such an enhanced multiple sequence alignment scheme is called *ClustalW*, described in [4] and [5].

For the inference of evolutionary relationships, the Neighbor-Joining method is used often (see [6]). Including the enhancements described in [7], the runtime complexity of this algorithm is $O(n^3)$. Roughly speaking, the reconstruction of the phylogenetic tree is done by a repeated search for the next pair of neighbors in the tree. With this scheme the evolutionary tree can always be reconstructed if the sampled data is correct, otherwise ambiguities may arise.

3 Parallelization

3.1 Stages

As mentioned before, the progressive alignment consists of three stages: A pairwise alignment stage, that yields a comparison matrix of every pair of sequences; a phylogenetic inference stage, that estimates the evolutionary relationship between the participating sequences; and the actual progressive alignment stage that consecutively builds the alignment by gradual merging of sub-alignments. These three stages depend upon the complete result of their predecessor. Therefore, the stages alone cannot be rendered in parallel.

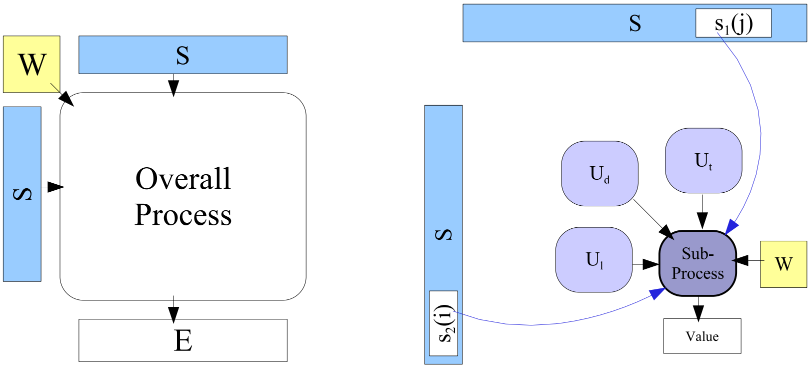
The first stage is almost obvious how to parallelize. It consists of calculating an alignment of every sequence with every other sequence. Therefore, there are $n^2/2$ sub-tasks to be computed. These sub-tasks are completely independent, so, the decomposition is evident. The easiest way to exploit this inherent parallelism is to employ the Master-Worker Pattern for distributing the work to multiple computers (see [8]). For this purpose a parallelization framework for such 'embarrassingly parallel' problems was developed (see [9] and [10]).

The second stage, consisting of the Neighbor-Joining algorithm, is currently sufficiently fast on a single computer because it does not depend on the length of the sequences but only on their number. With current problem sizes from ten to hundred sequences, the calculation is done on a single machine within seconds. Nevertheless, for larger problems also this stage will become important to accelerate. The basic idea for this stage is completely different from the first one. This time, the tasks cannot be distributed beforehand and calculated independently. However, the overall problem description can be distributed beforehand. Afterwards, the search for the next step can be parallelized again. Additionally, between the steps, the machines have to be synchronized. The data transfer is only one value for a quadratic computation, so this stage is still perfectly parallelizable as well.

When we look at the last stage, we see that it is well parallelizable in the beginning with a lot of pairwise alignment problems at the same time. As the sequences are combined to bigger and bigger sub-alignments, however, the number of such pairwise alignment problems decreases drastically. Thus, additional measures have to be taken to accelerate this stage through parallelization. The crucial element is the parallelization of the pairwise sequence alignment itself, which is described in the next section.

3.2 Distributed Dynamic Programming

While the parallelization of the pairwise alignment algorithm is not necessary for the first stage, the last stage definitely benefits from it. To further decompose the dynamic programming employed for solving the sequence alignment problem, we state the problem formally: The comparison function and the weighting scheme determine the behavior of the system. For simplicity this weighting scheme is assumed to be described by the set W. Finally, the output is often not the overall alignment cost, which is the last cell of the matrix, but instead the actual alignment of the sequences. Usually this alignment is described by an *edit string* $\mathrm{E} = \langle e_1, \ldots, e_n \rangle, e_i \in \{\text{`}\texttt{insert}\text{'}, \text{`}\texttt{delete}\text{'}, \text{`}\texttt{convert}\text{'}\}$ that describes the necessary operations to transform one sequence into the other one and can also be used to build the alignment. The overall process is, therefore, described by the mapping $\mathrm{S}^2 \times \mathrm{W} \rightarrow \mathrm{E}$. This systemic view of sequence alignment is depicted in Figure 4(a).



(a) Overall Process Model (b) Sub-Process Model

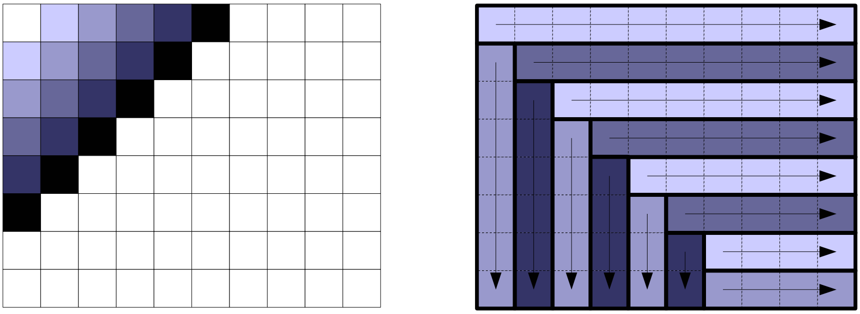
Fig. 4. Process Models

Now we can also formulate the cells of the matrix as individual processes. Every cell needs three preceding values described by $\mathrm{U} = \{(t, l, d) \mid t, l, d \in \mathbb{R}\}$ which are the values from the top, from the left and diagonally above the current cell. Additionally, the sequences' elements that have to be compared, together with the complete weighting scheme are required. Hence, the sub-processes can be sketched by the mapping $\mathrm{S}^2 \times \mathcal{A}^2 \times \mathrm{U} \times \mathrm{W} \rightarrow \mathbb{R}$, also shown in Figure 4(b).

Although every calculation of such a sub-process seems small and dependent upon a mass of inputs, challenging the usefulness of the decomposition, one important factor has to be kept in mind: The size of this seemingly ‘tiny’ process will grow quadratically during the course of the progressive alignment because, instead of comparing single elements of two sequences, every element becomes a group of sequences as soon as the sequences are whole sub-alignments. Therefore, the decomposition of the dynamic programming itself is a perfect complement to the distributed calculation of pairwise alignments.

3.3 Basic Idea

The basic idea is now to execute as many of the sub-processes in parallel. Looking at the data dependencies we see that there can only be one process in the beginning (left upper corner). After the calculation of its value, however, the preconditions for two other processes are met. These are the second element in the first row, and the second element in the first column. Next, the third elements in the first row and first column are ready to be calculated. Additionally, the second element in the second row is also ready. As we can see, the number of concurrently executable sub-processes increases until we reach the anti-diagonal. At the apogee of parallelization we can execute as many sub-processes as we have elements in the sequences; and on average half of it. Figure 5(a) shows this wave of calculation that can be conducted when as many sub-processes as possible are executed in parallel. The problem that arises now is the relatively tight data coupling: Still, every cell needs three values from precedent cells. With a dexterous workload distribution between the participating machines, however, this data coupling can be further reduced.

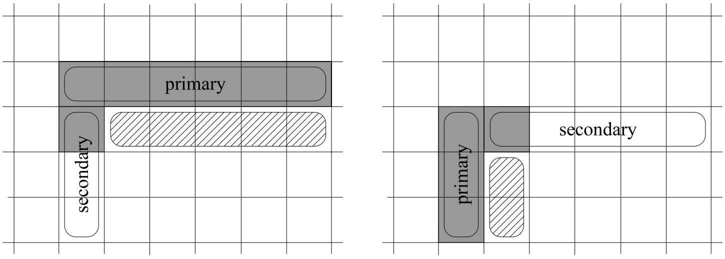


(a) Wave Pattern (b) Echelon Pattern

Fig. 5. Distributed Dynamic Programming

3.4 Further Data Decoupling

While the decomposition into individual data cells provides for great parallelism, the data coupling can still be improved. Figure 5(b) shows how the individual cells have to be grouped together to obtain a reduced need for data transfers.



(a) Horizontal Layout (b) Vertical Layout

Fig. 6. Echelon Dependencies

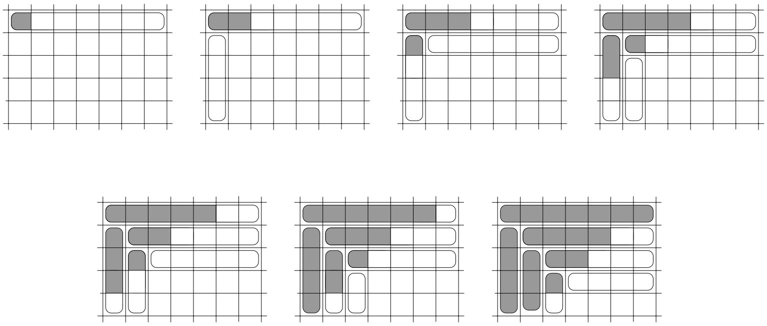


Fig. 7. Distributed Procession with Data Delay

Following this scheme, the data coupling can be reduced to almost a single value per cell on average. To accomplish this, the calculation has to be rearranged in a way that a computer will generate most of its precedent values itself. Therefore, the calculation is divided into stripes. As these stripes are filled bit by bit we call them 'echelons'.

With this scheme, echelons have data dependencies to only two other echelons. Figure 6 shows how the echelons depend on each other. Every echelon needs all values which surround them and have smaller coordinates. The preceding echelon which has the same direction is called the *primary* predecessor that will deliver all of its values to its primary client. The other preceding echelon which has an other direction is called the *secondary* predecessor. This echelon will usually just deliver an initial value to its secondary client.

To start out, an echelon needs two values from its primary predecessor, and one value from its secondary predecessor. After these values have arrived, the calculation can start. In the meantime, the primary predecessor can calculate its next value. Hopefully, this value will then be available just at the right time, when the current echelon is done with its own calculation. This means that the distributed calculation has some delay in the beginning until the first three

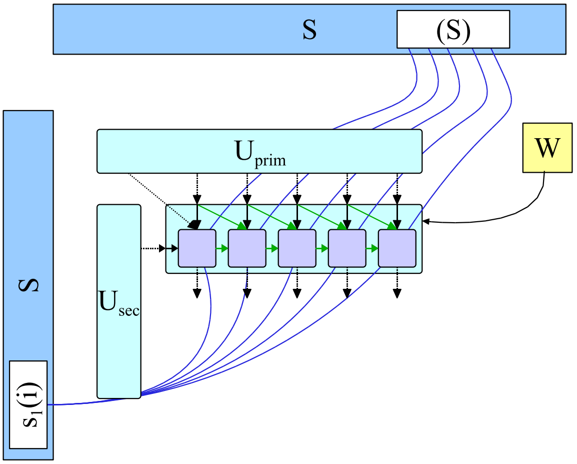


Fig. 8. New Process Model

datums have arrived at a given echelon. Later on, the values needed for the calculation will be available right on time.

An example of such a distributed progression using echelons is shown in Figure 7 on the previous page. The procession slightly hangs towards the origin, depending on the latency of the data transfer. After the echelons have been set up, however, there are no further delays, conceptually.

The new process model is shown in Figure 8. Now, every sub-process has only two predecessors U_{sec} and U_{prim} from which it will receive several values. Additionally, it will need multiple values from one of the input sequences and one value from the other. Formulated as a mapping, an echelon now calculates the function $(S) \times \mathcal{A} \times \mathbb{R}^{k-j+1} \rightarrow \mathbb{R}^{k-j}$ where j and k are the start and end index of the subsequence.

One important aspect that still remains to be solved is how to reduce the number of concurrent tasks if less computers are available or if the network is slow. In this case a lot of data transfers are made which can probably be avoided. The basic idea to solve this problem is to make the echelons bigger, and hence reduce the amount of data transfer to an arbitrary fraction adjusted to the number of available computers.

4 Results

Currently the implemented system is not ready to be fully deployed. Therefore, the shown results are based on a performance simulation. In Figure 9 on the next page the estimated runtime for a problem of size 30 × 15 is shown. The diagram shows how the total computation time is influenced by intermediate network latency. For a problem this small, the network latency can be the 28-fold computation time. For a problem of size 60 × 30, the network latency can be as high as 56-fold computation time before linear calculation would be faster.

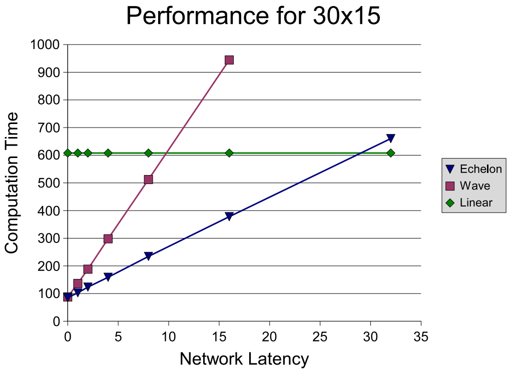


Fig. 9. Performance Simulation

Usual problem sizes, however, are much bigger. Therefore, the tolerable network latency will surely lie within reasonable boundaries. Additionally, during the progressive alignment, when the number of pairwise alignment problems decreases, the complexity of a single cell computation increases quadratically. Therefore, scalability in terms of decomposition will hardly become a problem.

5 Conclusion

Although typical sequence alignment, solved by dynamic programming, has a considerably tight data coupling between the cells in the matrix, still a lot of parallelism can be exploited. Especially within the progressive alignment scheme, the increasing complexity of comparison operations, further facilitates a distributed calculation even for slow networks. Using the idea of echelons arranged in a fish-bone pattern the network transfer can be reduced to an amount athat makes it feasible under many different situations, yielding a remarkable scalability in terms of problem size. Further research will concentrate on refining the distribution and size of these echelons, to maximize parallelism for a smaller number of computers and also for slower networks.

References

1. Hastings, N.A.J.: Dynamic Programming: With Management Applications. Butterworths (1973)
2. Needleman, S.B., Wunsch, C.D.: A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins. Journal of Molecular Biology **48** (1970) 443–453
3. Myers, E.W., Miller, W.: Optimal alignments in linear space. Computer Applications in the Biosciences **4** (1988) 11–17
4. Higgins, D.G., Sharp, P.M.: CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. Gene **73** (1988) 237–244

5. Thompson, J.D., Higgins, D.G., Gibson, T.J.: CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Research **22** (1994) 4673–4680
6. Saitou, N., Nei, M.: The Neighbor-joining Method: A New Method for Reconstructing Phylogenetic Trees. Journal of Molecular Biology and Evolution **4** (1987) 406–425
7. Studier, J.A., Keppler, K.J.: A Note on the Neighbor-Joining Algorithm of Saitou and Nei. Journal of Molecular Biology and Evolution **5** (1988) 729–731
8. Mattson, T.G., Sanders, B.A., Massingil, B.L.: A Pattern Language for Parallel Programming. Addison Wesley (2004)
9. Kern, T.: Biomedical Information Systems. `http://biomis.fh-hagenberg.at` (last visited March 2005) (2003)
10. Pitzer, E.: Acceleration of Progressive Multiple Sequence Alignment by Parallelization and Complexity Reduction of Existing Algorithms. Master's thesis, University of Applied Sciences Hagenberg, Austria (2004)

Concepts and Systems Tools for Modelling Signal Processing in Vertebrate Retina

Roberto Moreno-Díaz[1], Gabriel de Blasio[1], and Arminda Moreno-Díaz[2]

[1] Instituto Universitario de Ciencias y Tecnologías Cibernéticas,
Universidad de Las Palmas de Gran Canaria,
rmoreno@ciber.ulpgc.es
gdeblasio@dis.ulpgc.es
[2] School of Computer Science. Madrid Technical University
amoreno@fi.upm.es

Abstract. The concept of fast and slow signals interacting by non-linear lateral processing at the inner plexiform layer and its suitable mathematical formulation, allows for a coherent interpretation of simple, complex and colour coding ganglion retinal cells of various representative vertebrates. In its simplest formulation, the processing can be expressed through non-linear spatio-temporal transformations. The formulations result in a coherent unified conceptual frame to interprete signal processing in vertebrates' retinae.

1 Introduction

In a way towards a unified view of vertebrate retinal processing, several models have been proposed for various representative cases ([11], [12], [13], [14], [17]). We present here a unified frame for interpreting cat's, frog's and avian's retinae. To obtain it, we must abandon some detailed assumptions for particular retinae for the sake of consistency. Also, the fitting of quantitative data has to be postponed to further model refinement.

We first consider briefly the anatomical, physiological and formal bases. One of the keys ideas in the interpretation of retinal processing is the link of specialized ganglia computation to the interaction of amacrines and ganglion cell dendrites in the inner plexiform layer, as it is suggested by anatomy ([2]). Logical interpretations of neurophysiological recordings led also to a similar conclusion. In their account of pigeon's ganglia responses, [9] already asserted that, since different ganglia look at the world through about the same bipolars (and receptors), ganglia differences should be a consequence of their different manipulation of almost the same inputs (paraphrase).

From earlier results in cat's and monkey's retinae, a line was drawn initially between higher and lower vertebrates by which vertebrates with well developed visual cortex were on the side of simplicity whereas specialization was thought to be exclusive of poorly cortically endowed vertebrates. In principle, the whole of retinal cells were then available to engage in specialization, so that, for example, the apparently asymmetric disposition of horizontal axons made them a

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 483–491, 2005.
© Springer-Verlag Berlin Heidelberg 2005

unique candidate for directional selectivity ([1]). But this required too much of the retina, since though asymmetric units were encountered in cat, so breaking the line ([4]), they were too few to explain the number of horizontals. The conclusion is that specialization must occur, in general, after the bipolar outputs, in agreement with Maturana's observations.

In addition, experiments indicated that directional selectivity is a local property, which implies that the corresponding mechanism must be situated prior to ganglion spatial integration and it should be the result of the interaction of signals coming from relatively small areas which are close to each other. This leaves the inner plexiform layer, and not the ganglia cellular bodies, as the most probable site responsible for peculiar extraction of spatio-temporal properties of images.

Next, there is the appropriate formal representation. Different wiring diagrams of the retina were then and thereafter proposed to explain its computational properties (see, for example, [6]). But they lead to the need of adding ad hoc connections to the diagram every time a property is to be embodied, which is contrary to the spirit of any consistent approach.

It is in the formulation where the difficulties to obtain a consistent picture generate. Because of the intrinsic spatio-temporal nature of the signals being handled, wiring diagrams serve only channeled interpretations, unless they are made intricately umcomprehensible. Retinal processing is, by structure and function, a layered computation, as it is probably cortical processing too. Therefore, it should be trated as such.

Summing up, the two main points of our approach are: first, locate prominent specialization at the inner plexiform layer, prior to ganglia integration, where probable operation is a nonlinear lateral interaction of signals from bipolar, amacrines and ganglia dendrites. Second, formulate the processing there as it corresponds to a layered processing system.

A definition of layered computation for the general case has been developed somewhere ([10], [14]). We treat it here under the simplest reasonable assumptions as it applies to the retina.

2 Formulation

Simple non-specialized retinae, such as cat's, suggest that signals from outer retinal layers are two spatio-temporal versions of quasi-linearly transformed input data: a fast signal and a slow signal. Though the role of horizontals is not yet clear, it has been suggested that they might be involved in generating slow versions, which are laterally translated. In this case, those signals must be returned to ganglia (probably via amacrines), since property extraction is local.

For a continuous retina, under the simplest spatio-temporal assumptions and by considering only first order terms, the expression of the above in conventional systems analysis notation follows.

We assume linear spatio-temporal transformations, with only local nonlinearities. In this case, the signals corresponding to the action of photoreceptors and bipolars without additional delays, called f_F (fast) signals, may be expressed as:

$$f_F(\mathbf{r},t) = \int_0^t \Big[W_H(t-\tau)\int_{\mathbf{r}} W_F(\mathbf{r}-\mathbf{r}')I(\mathbf{r}',\tau)d\mathbf{r}'\Big]d\tau \tag{1}$$

under the restriction of zero initial conditions. Coordinate vector $\mathbf{r}$ is the position where the signal f_F appears and $\mathbf{r}'$ runs over a small area (the span of bipolar denditric tree or photoreceptors feet). $I(\mathbf{r}',\tau)$ is some local no linear function of the light incident in photoreceptors at $\mathbf{r}'$ and time τ (e.g. a logarithmic compression). W_F and W_H correspond to outer layers colour-space-time transformation, essentially a colour filter plus a low pass space-time filter.

Under the simplest assumption of a single speed difference for fast and slow signals, v, the linear relation between fast f_F and slow f_S signals is given by:

$$\frac{df_S}{dt} = -v[f_S - f_F] \tag{2}$$

Under zero initial conditions, the solution is:

$$f_S(\mathbf{r},t) = v\int_0^t e^{-v(t-\tau)} f_F(\mathbf{r},\tau)d\tau \tag{3}$$

Signals f_F and f_S undergo nonlinear lateral interaction at the inner plexiform layer. This interaction may be decomposed in linear lateral inhibition plus a local rectifying nonlinearity ([11]). The resulting two possibilities are linear lateral inhibition of f_F by f_S, and viceversa, that is:

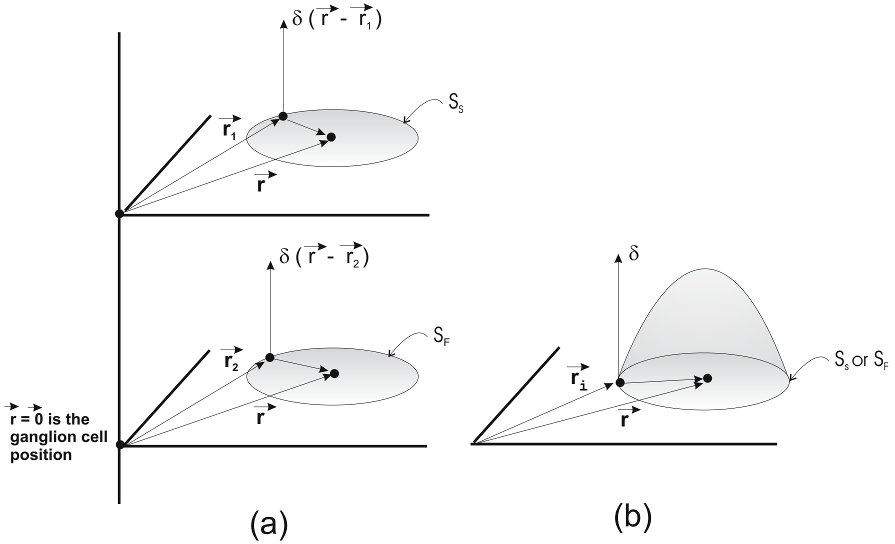


Fig. 1. Illustration of S_S and S_F. (a) Illustration of coordinates $\overrightarrow{r}$, $\overrightarrow{r}_1$, $\overrightarrow{r}_2$, and positions of S_S, S_F,and δ. (b) Illustration of the weights S_S, S_F and the relative position of $\delta(\overrightarrow{r} - \overrightarrow{r}_1)$ or $\delta(\overrightarrow{r} - \overrightarrow{r}_2)$.

$$X_1(\mathbf{r},t) = f_F(\mathbf{r}-\mathbf{r_1},t) - \int_{S_S} S_S(\mathbf{r},\mathbf{r}')f_S(\mathbf{r}',t)d\mathbf{r}' \tag{4}$$

$$X_2(\mathbf{r},t) = f_S(\mathbf{r}-\mathbf{r_2},t) - \int_{S_F} S_F(\mathbf{r},\mathbf{r}')f_F(\mathbf{r}',t)d\mathbf{r}' \tag{5}$$

Small surfaces S_S and S_F at the inner plexiform layer are supposed to be of an almost circular boundary and of inverted U shape (when plotting function $S(r')$ versus r', maximum at the center, zero at the pheriphery), from $r' = 0$ to $r' = r_0$. $\mathbf{r}$ is the position vector of the center of S_F and S_S (see 1).

According to equation (3), equation (4) can be written:

$$X_1(\mathbf{r},t) = \int_0^t \int_{S_S} \Big[\delta(\mathbf{r}-\mathbf{r_1}-\mathbf{r}')\delta(t-\tau) - S_S(\mathbf{r},\mathbf{r}')ve^{-v(t-\tau)}\Big] f_F(\mathbf{r}',\tau)d\mathbf{r}'d\tau \tag{6}$$

Similarly for equation (5):

$$X_2(\mathbf{r},t) = \int_0^t \int_{S_F} \Big[ve^{-v(t-\tau)}\delta(\mathbf{r}-\mathbf{r_2}-\mathbf{r}') - S_F(\mathbf{r},\mathbf{r}')\delta(t-\tau)\Big] f_F(\mathbf{r}',\tau)d\mathbf{r}'d\tau \tag{7}$$

The local rectifying non-linearity generates signals X_1' and X_2', given by

$$X_1'(\mathbf{r},t) = Pos(X_1(\mathbf{r},t)) \tag{8}$$

$$X_2'(\mathbf{r},t) = Pos(X_2(\mathbf{r},t)) \tag{9}$$

where $Pos(x)$ is defined as:

$$Pos(x) = \begin{cases} x \; for \; x \geq 0 \\ 0 \; for \; x < 0 \end{cases}$$

Signals X_1' and X_2' are then weighted and summated to yield, for a ganglion cell at the origin, its instantaneous frequency of firing:

$$G(t) = Pos\Big\{ \int_R \int_0^t k_1(t-\tau,\mathbf{r})X_1'(\mathbf{r},t)d\tau d\mathbf{r} + \int_R \int_0^t k_2(t-\tau,\mathbf{r})X_2'(\mathbf{r},t)d\tau d\mathbf{r} + G_0 \Big\} \tag{10}$$

where G_0 is the spontaneous response, when it exists, and R represents the Excitatory Receptive Field (ERF) of the ganglion cell.

Kernels k_1 and k_2 correspond to the actions of the dendritic tree of the ganglion cell. A Newton Polynomial type of kernels (or Hermitian) ([15]), having center and one or more inhibitory-excitatory rings is then at case. The whole layered processing is illustrated in 2.

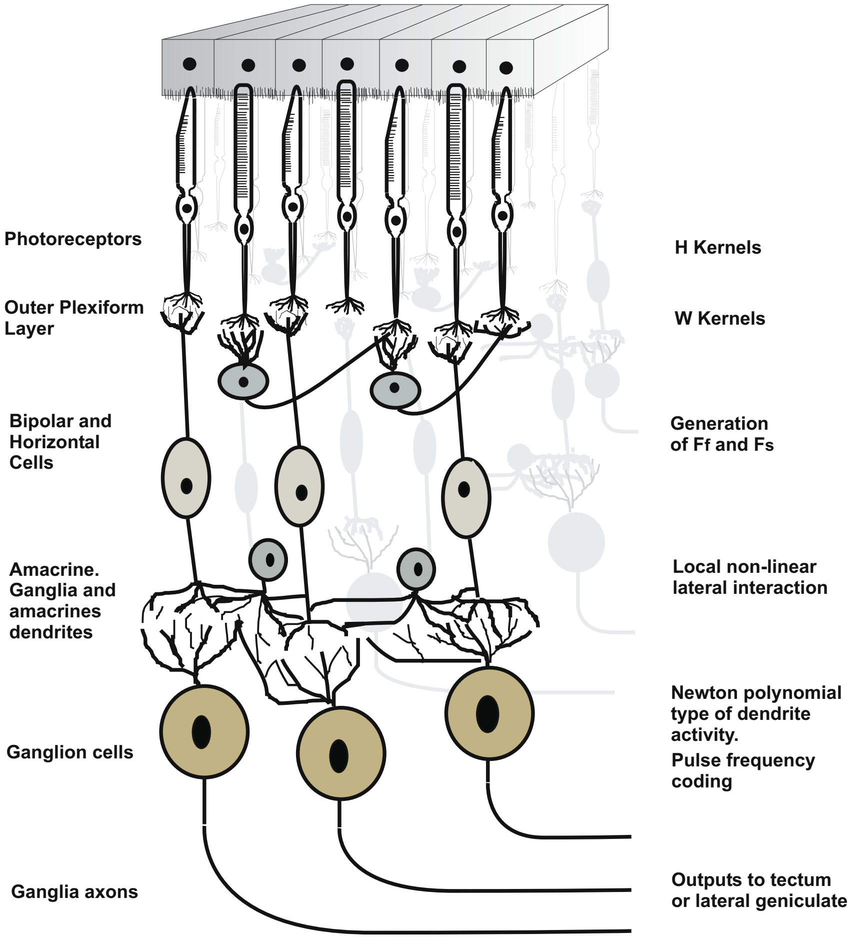

Photoreceptors
Outer Plexiform Layer
Bipolar and Horizontal Cells
Amacrine. Ganglia and amacrines dendrites
Ganglion cells
Ganglia axons
H Kernels
W Kernels
Generation of Ff and Fs
Local non-linear lateral interaction
Newton polynomial type of dendrite activity.
Pulse frequency coding
Outputs to tectum or lateral geniculate

Fig. 2. Schematic view of the structure and proposed functions of retinal cells

From (6) and (7), $X_1(\mathbf{r},t)$ and $X_2(\mathbf{r},t)$ are the results of linear space-time transformations on $f_F(\mathbf{r},t)$ given by the integral transformation kernels:

$$W_1(\mathbf{r},t) = \delta(\mathbf{r}-\mathbf{r_1})\delta(t) - S_S(\mathbf{r})ve^{-vt} \tag{11}$$

$$W_2(\mathbf{r},t) = ve^{-vt}\delta(\mathbf{r}-\mathbf{r_2}) - S_F(\mathbf{r})\delta(t) \tag{12}$$

The spatial part of (11) and (12),

$$\delta(\mathbf{r}-\mathbf{r_1}) - S_S(\mathbf{r}) \tag{13}$$

$$\delta(\mathbf{r}-\mathbf{r_2}) - S_F(\mathbf{r}) \tag{14}$$

are spatial contrast detectors, which are symmetric detectors when $\mathbf{r_1} = \mathbf{r_2} = \mathbf{r}$. If they are not equal to $\mathbf{r}$, they are asymmetric detectors. If $\mathbf{r_1}$ and $\mathbf{r_2}$ have opposite directions, one detects optimally a contrast in one direction and the other in the opposite direction. If they are distribuited randomly in directions, the whole behaves as a symmetric detector.

The temporal part of kernels (11) and (12) is

$$\delta(t) - ve^{-vt} \tag{15}$$

$$ve^{-vt} - \delta(t) \tag{16}$$

They are ON and OFF temporal kernels respectively.

In (11) and (12), if constants $\int_{S_S} S_S(r)dr = k_S$ and $\int_{S_F} S_F(r)dr = k_F$ are made $k_S = k_F = 1$, the responses to stationary spatial contrasts are zero, while there is still local ON-OFF responses and, consequently, sensitivity to moving contrast, which provoke local ON and OFF effects. In this case, if there is a random distribution of the directions of $\mathbf{r} - \mathbf{r_1}$ and $\mathbf{r} - \mathbf{r_2}$, there is sensitivity to motion without preferred direction. If $\mathbf{r} - \mathbf{r_1}$ is not random, but has a preferred overall direction $\mathbf{d}$, the overall sensitivity is higher to bright contrasts moving in the direction $-\mathbf{d}$. The inverse happens when there is a preferred direction for $\mathbf{r} - \mathbf{r_2}$ (high response to a dark contrast moving in direction $-\mathbf{d}$). Obviously, at the inner plexiform layer and prior to ganglion integration, only one of the two mechanisms given by (11) and (12) may exist.

What and how are anatomical units involved in lateral interaction processing? Amacrines suggest by themselves as responsible for the spatial spreading appearing in (11) and (12), so that k_S, k_F, S_S and S_F must depend upon ganglion-amacrines cell interaction. Therefore, the extent and geometrical shape of S_S and S_F and the values of k_S and k_F for a ganglion of a type, should be controlled by the number and geometrical pattern of its synapses with amacrines, whereas direct contact with bipolar axons provide for f_S and f_F to generate composite signals X_1' and X_2'. When the interaction of direct and amacrine signals is approximately symmetric, the resulting ERF for a ganglion is isotropic. When there is a systematic, though small, deviation in the sites of the interaction of direct and amacrines signals, anisotropic but uniform ERFs result.

3 Applications

3.1 Cat

In cat's retina, simple cells are in a large proportion ([3]), whereas specialized ganglia are rarely encountered ([4]). We consider the application of (10) to the limiting types corresponding to quasi-linear and to lightly specialized ganglia. Intermediate types correspond to intermediate values of the parameters.

Simple quasi-linear cells. Linear operation must correspond to negligible lateral interaction, that is, $k_S = k_F = 0$. Also, the spontaneous response is null, $G_0 = 0$. Equation (10) reduces to

$$G(t) = Pos\left\{\int_R \int_0^t k_1 Pos(f_F) d\mathbf{r} d\tau + \int_R \int_0^t k_2 Pos(f_S) d\mathbf{r} d\tau\right\} \tag{17}$$

If we further assume positive functions f_F and f_S, (10) leads to

$$G(t) = Pos\Big[\int_R \int_0^t \{k_1(t-\tau,\mathbf{r})f_F(\mathbf{r},\tau) + k_2(t-\tau,\mathbf{r})f_S(\mathbf{r},\tau)\}\, d\tau d\mathbf{r}\Big] \quad (18)$$

Equation (18) corresponds to a general expression for time invariant linear spatio-temporal computation ([10]).

Local contrast detectors. For these cells, k_1 and k_2 are positive in the central area and negative in the periphery. This is provided by the usual difference of gaussians. Lateral interaction kernels are symmetric, with $k_S \simeq k_F \simeq 1$. Also, $G_0 = 0$. Their higher sensitivity to moving stimuli is a consequence of the ON and OFF local contributions by the lateral interaction kernels, which add to the contrast detection contribution.

Direction selective units. In this case $G_0 = 0$, k_1 and k_2 may be expressed as a difference of gaussians and lateral interaction kernels are asymmetric. Note again that their asymmetry does not destroy the uniform ON-OFF character of the ERF.

Uniformity detectors. For these ganglia, $G_0 \neq 0$, lateral interaction kernels are symmetric, with $k_S = k_F \geq 2$, for contrast detection with no sustained effect and k_1 and k_2 are negative since any stimulus is inhibitory.

Edge inhibitory OFF-Centre units. For a model, k_1 and k_2 are negative in the central zone and lateral interaction kernels are symmetric with $k_S \simeq k_F \geq 2$ for no sustained response. In the second zone, $k_1 \simeq 0$ and k_2 is positive. In the pheriphery k_1 is positive and $k_2 \simeq 0$. Also, $G_0 = 0$.

3.2 Avian

For specificity we consider the classification of pigeon ganglion cells by [16]. None of the cells show spontaneous response, therefore $G_0 = 0$.

ON or OFF centre units. These units show ON or OFF receptive fields with an inhibitory surround and they are not more sensitive to stationary than to moving stimuli. Thus, the only contributions are from X_1' or X_2'. Lateral interaction kernels have a relatively large area of S_S or S_F and also $k_S = k_F \geq 2$.

Motion sensitive units. They show ON-OFF receptive fields and inhibitory surround. Their isotropic responses indicate symmetric lateral interaction kernels, with large areas of S_S and S_F for low sensitive units and small areas for high sensitive units. Different relative strenghts of the ON and OFF components are the result of different positive values for k_1 and k_2. The sharpness of the transitory is modulated by k_S and k_F.

Directional sensitive units. They differ from the above by the presence of asymmetric lateral interaction kernels. This selectivity is independent of the polarity of a moving contrast, as it results from (4) and (5).

3.3 Frog

Frog's ganglion retinal cells were classified in four groups by [7]. This classification has been follow thereafter, at least in what respects to limiting groups. Since there is not spontaneous response, $G_0 = 0$ in all cases.

Group 1. They are termed *fixed contrast detectors*. They have ON receptive field with no inhibitory ring. Therefore, the only term in the reponses comes from X'_1. Lateral interaction kernels are symmetric with $k_S = 1$ for a response to a stopped contrast. Ganglion weights, k_1, are representable by a gaussian.

Group 2. They are termed *bug detectors* and they detect small dark objects moving centripetally in the ERF. They are not or very little sensitive to bright stimuli. In the simplest interpretation, the only contribution is from X'_2. Lateral interaction kernels are asymmetric, with a radial distribution because their peculiar directionality, and k_F must be ≥ 2 since they do not respond to stationary contrast. Insensitivity to large objects is due to an inhibitory ring so that k_2 may be expressed as a difference of gaussians.

Group 3. They have ON-OFF receptive field and inhibitory surround. The simplest interpretation leads to lateral interaction kernels with $k_F = k_S = 1$. Ganglion weights, k_1 and k_2, are a difference of gaussians.

Group 4. They are the *dimming detectors*. They have OFF receptive fields. Therefore, the only contribution is from X'_2. Lateral interaction kernels are again local and $k_F \geq 1$. Ganglion weights, k_2, are representable by a gaussian.

3.4 Colour Coding Units

Colour coding is also embodied as a consequence of (10). For a retina containing receptors of different spectral sensitivity, local direct signals transmitted through the singularity $\delta(r - r')$ in (11) and (12) are colour filtered, while the inhibitory terms are, in general, colour wide band, since they are integrated over surfaces S_S and S_F. Therefore, (11) and (12) are narrowband ON and OFF, as well as contrast detectors in the wavelenght domain: direct colour filtered signals are locally inhibited by signals carrying the complementary colour information. The various colour coding units ([5]) appear then as a consequence.

References

1. Barlow, H. B., Levick, W.R. (1965) The Mechanism of Directionally Selective Units in Rabbit's Retina. J. Physiol. 178 (London). pp 477–504.
2. Boycott, B.B., Levick, W.R. (1974) Aspects of Comparative Anatomy and Physiology of Vertebrate Retina. Essay on the Nervous System. Oxford Clarendon Press.
3. Cleland, B.G., Levick W.R. (1974 a) Brisk and Sluggish Concentrically Organized Ganglion Cells in the Cat's Retina. J. Physiol. 240 (London) pp 421–456.
4. Cleland, B.G., Levick W.R. (1974 b) Properties of Rarely Encountered Cells in the Cat's Retina and an Overall Clasification. J. Physiol. 240 (London) pp 457–492.

5. Daw, N.W. (1973) Neurophysiology of Color Vision. Physiological Review, 53, 3, pp 571–611.
6. Grüsser, O.J., Grusser-Cornehls, U. (1973) Neuronal Mechanism of Visual Movement Perception and Some Psychophysical Behavioral Correlations. In Jung, R. (Ed) Handbook of Sensory Physiology, Vol. 7, Part 3: Central Processing of Visual Information. Berlin-Heidelberg-New York, Springer. pp 333–429.
7. Lettvin, J.Y., Maturana, H.R., McCulloch, W.S., Pitts, W.H. (1959) What the Frog's Eye Tells the Frog's Brain. Proc. of IRE 47. pp 1940–1951.
8. Maturana, H.R., Lettvin, J.Y., McCulloch, W.S., Pitts, W.H. (1960) Anatomy and Physiology of Vision in the Frog (Rana Pipiens). J. Gen. Physiol. 43 pp 129–175.
9. Maturana, H.R., Frenk, S. (1963) Directional Movement and Horizontal Edge Detector in the Pigeon Retina. Science, 42. pp 977–979.
10. Moreno-Díaz, R., Rubio, E. (1979) A Theoretical Model for Layered Visual Processing. Int. J. Bio-Med. Comp. 10, pp 231–243.
11. Moreno-Díaz, R., Rubio, E. (1980) A Model for Non Linear Processing in Cat's Retina. Biol. Cybernetics. 37 pp 25–31.
12. Moreno-Díaz, R., Rubio Royo, F., Rubio, E. (1980) A Theoretical Proposal to Account for Visual Computation in a Frog'a Retina. Int. J. Bio-Med. Comp. 11, pp 415–426.
13. Moreno-Díaz, R., Rubio, E., Núñez, A. (1980) A Layered Model for Visual Processing in Avian Retina. Biol. Cybernetics, 38, pp 85–89.
14. Moreno-Díaz, R., de Blasio, G. (2003) Systems Methods in Visual Modelling. Systems Analysis, Modelling and Simulation, 43 (9), pp 1159–1171.
15. Moreno-Díaz, R. (Jr.) (1993) Computación Modular Distribuida: Relaciones Estructura-Función en Retinas. PhD. Thesis. Universidad de Las Palmas de Gran Canaria.
16. Pearlman, A.L., Hughes, C.P. (1976) Functional Role of the Efferents to the Avian Retina. Analysis of Ganglion Cells Receptive Fields. J. Comp. Neurol. 166, pp 111–121.
17. Tvoy, J.B., Shou, T. (2002) The Receptive Fields of Cat Retinal Ganglion Cells. Progress in Retinal and eye Research, 21, pp 263–302.

Application of Multichannel Vision Concepts and Mechanisms in an Artificial Industrial Vision System

A. Quesada-Arencibia, J.C. Rodríguez-Rodríguez, and R. Moreno-Díaz Jr

Institute for Cybernetics (IUCTC), Universidad de Las Palmas de Gran Canaria
E35017 Las Palmas, Spain
aquesada@dis.ulpgc.es, jcarlos@ciber.ulpgc.es
rmorenoj@dis.ulpgc.es

Abstract. The design of visual processing systems in very demanding industrial environments is a technical field in which bioinspiration has not been explored as a developing tool. The need of extremely quick, accurate and real time responses needed in industrial applications is not usually seen as compatible with the "messy", "slow" or "inaccurate" methods and algorithms inspired in the information processing mechanisms underlying neural activity in the visual pathway. We are trying, thus, to explore the practical possibilities of interaction among concepts from both worlds: the "real" vision system designed for a real time quality control of a production line, and the "inspiration" taken from multi-channel biological vision. In previous papers [1,2] a biologically plausible parallel system for visual detection of form, movement, shape and size has been developed. The system, working off-line and skipping real time restrictions, was tested for a variety of situations, yielding very good results in estimating the mentioned visual characteristics of moving objects. Furthermore, a second parallel-computing version was designed introducing the concept of parallel channel processing, e.g., the discrimination of different visual characteristics by mean of multiprocessors and multithread computing. The architecture we present here, which includes certain concepts developed in the previously explained results [3,4], is intended to work in the production line of a beverage canning industry where cans with faulty imprinted use date and lot number have to be immediately discharged from the line.

1 Multichannel: Concept Borrowing from Natural Visual Processing

Multichannel processing is ubiquitous in natural visual systems. By that name we refer to the existence of several subpathways within a sensory modality extracting (or computing) different features in parallel, whose outputs can be combined at a subsequent level to yield a description with higher semantic content. Probably, the first detailed description of such a mechanism, both in the neural structure level and in the functional one, is included in the paper by Lettin et al "What the frog's eye tells the frog's brain". In previous papers we have developed and implemented some vision tools that make use of these characteristics.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 492 – 500, 2005.
© Springer-Verlag Berlin Heidelberg 2005

This system, being designed with two channels one for purely spatial properties of visual objects and the other for motion analysis, provides reliable and fast results in the spatial channel: the size and center of gravity values that we wanted to measure are within less than 2% error margin in average, with a low computational cost that can be adapted according to the needs. A complete description of speed and direction of movement parameters by the velocity channel, however, though being a more complicated model, take longer time but yields good results provided we control several parameters afecting the output, namely receptive field size, memory and lateral interaction. The main conlusions yielded by the system are the following:

- The operations carried out and their local amplitude make it possible to carry them out in parallel, in a two-channel fashion, thus operating the receptive fields in an independent way to combine their results in the next layer and obtain global results.
- The similarities of the different operations needed make it possible to combine them into one network from which different subnetworks that also interact are 'fed'.
- The random procedure that has been taken as a base excludes the necessity of having a deterministically organized system to obtain approximate results, proposing that the natural system on which the model is inspired does not use a perfectly established network.
- Seen as a whole, and comparing the location of the GC of the object calculated separately by both channels (e.g. the white and coloured lines in the pictures), the model gives us some perspective on channel processing in natural visual systems. Though an estimate of the position of the object can be calculated from the velocity channel, it is not as good nor as fast in its delivery as it is when presented by the purely spatial GC-size channel. The system could work properly for all descriptors, in certain cases, with only the output from the velocity channel, but if precision is needed, then a second stationary-working channel must be included. In either case, speed and direction of motion can be finely estimated, but estimating accurate position from velocity must involve extrapolation (thus introducing an error) or feedback (which introduces time delay). In cases where determining speed or other characteristics of a moving object are crucial and time-critical, having separate, dedicated channels would be more efficient. This leads to the idea of actually having *several* channels for movement description: perhaps one coarsely tuned, quick response channel and a second, finely tuned one for detailed analysis.

A diagram of the overall procedure can be seen in Fig. 1.

The practical use of this scheme has a number of restrictions. In order to be useful it has to be implemented in a multiprocessor computer and all processes have to be syncronized to yield a coherent description of the outside world. Besides, the general performance is quite domain-dependent, in the sense that the final goal of the vision system has to be perfectly defined, which is usually the case in industrial environments as it will be explained in the next section.

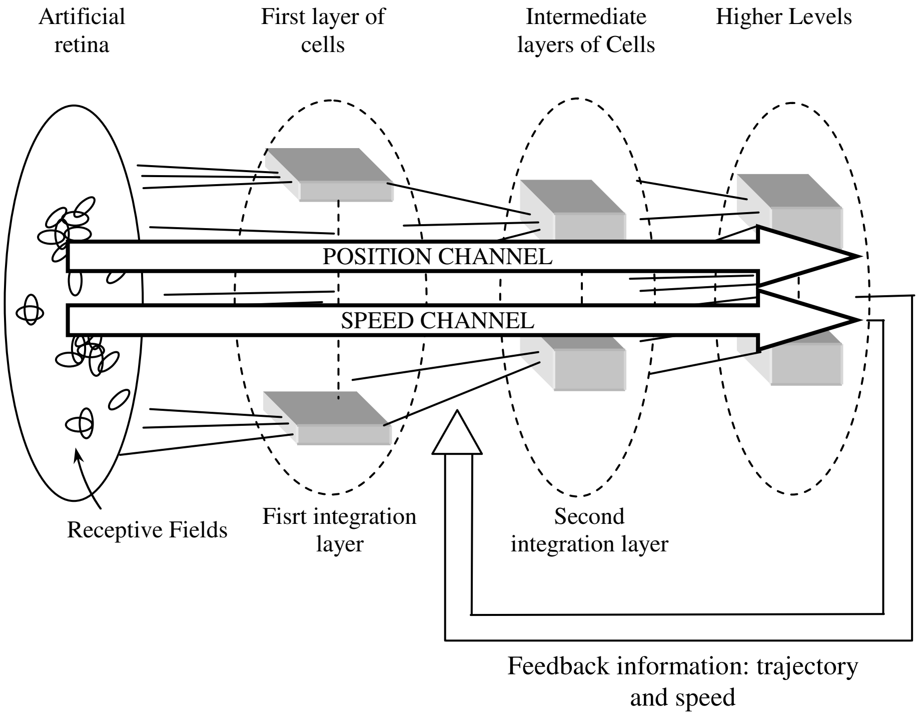


Fig. 1. Multilayer and multichannel scheme with feedback information

2 The Industrial Process. Visual Detection of Required Features

2.1 General Description

The problem to deal with is typical in the industry. The purpose is to validate imprinted codes in beverage and beer cans that go in front of the camera at very high speed. To validate means 'to verify that the code is correct" and hence, it does not imply an effective identification.

2.2 Time and Process Considerations

The critical requirements being accurate invariant from position text analysis in real time of nearly 30.000 cans per minute, the system has to be reliable and rely on specific hardware. Thus, a high speed acquisition camera, an efficient acquisition board, a strong multiprocessing system and a considerable bandwidth for main memory load are the basic requirements. Since the response of the system, e.g. whether a can is validated or not, has to be immediate, it makes the actual visual processing time very small, sensibly shorter than the 400 millisecond that biological systems need for a "good look", that is, for the effective processing of a visual frame in order to be able to recognize specific objects or events [5].

Computation simplicity can be easily achieved by opting for a point by point process (that is, one that only takes into account a single pixel in the image in each iteration) and complex operations that cannot be based on a non-exponential number of sums, such as a division or a multiplication, are not applied. It is important to work, whenever possible, with integer arithmetic.

The size of the area on which the computation is applied is also critical. And so, the effort is outstanding in the definition of windows or areas of interest, that is, rectangular sections in the acquired image that, because of their location, have a special meaning to reach a certain knowledge without requiring a complete process of the image.

A simple operation may only imply the identification of the color of a pixel. Color is a very primary information, although placed on a single pixel does not tell a lot. However, a processing by regions about the color of the pixels that make them up may give us such useful information as that of the predominance of a color in an area. An easy operation that may give us a good objective idea about the predominance of the color is color density.

The combination of these restrictions guarantees a very cheap processing in time that is able to provide real time answers.

2.3 Calibration

By calibration we mean the process in which the best physical parameters for the loaded algorithms are established. Particularly this calibration implies the physical arrangement of the camera-optics-illumination-can system and the obtained acquired image. The procedure is as follows:

An identifiable characteristic of the external object is chosen, namely the circular base of the can. A relationship between the acquired digitized circumference and the actual circumference of the can may be established. By using Hough algorithm for circumference segmentation, the circular base of a cans is extracted and from this we obtain the features that are used in the calibration mechanism:

1. A border detector is applied on the digitized image.
2. A three-dimensional space for circumference parameters is built in order to generate the equation for the circumference. In this three dimensional matrix space, the X axis contains the possible x-coordinate for circumference center, so contains the Y-axis for the y-coordinate and the Z axis, the possible values for the radius. The equation thus being:

$$(x - x')^2 + (y - y')^2 = r^2$$

3. For each one of the candidate center points we calculate a radius using the first (x, y) point of the border segmented in step 1. Thus we will have a vote for the (x’,y’,r’) position. This operation is repeated for all (x,y) points of the segmented border.
4. Then, the most voted position (x’,y’,r’) is the best probable circumference extracted using the border points of step 1.

2.4 Centring Determination

A likely first step in this problem consists in selecting as soon as possible which images deserve the extra effort of determining whether there is a code or not. This is so because during the acquisition, unless a trigger such as a photoelectric cell is used, we will have an important battery of images with cans in too dubious situations so as to tackle a search and validation of code, even for a human being.

It has been confirmed that the best group of images for validation are those that include cans whose centres are near the centre of the acquired image.

The selected scheme is particularly intuitive. It deals with an analysis by area. It is easy to demonstrate that a centred can fulfils certain requirements necessarily in a combination of areas. The most important requisite has to do with the predominance of a certain color or range of colors in a given area. Thus, for a can to be centred in the image it is essential that "metallic" colors of the can material prevail in the central part of the image.

So, we should choose the most strategic regions to inspect for the location of centred cans and the thresholds of color that have to be searched in every region.

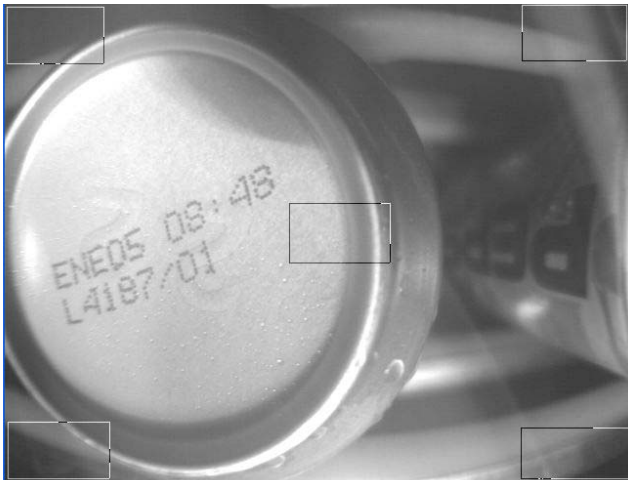


Fig. 2. A full image showing a partial extraction of four corners

2.5 Determination of Presence/Absence

The presence of ink is a necessary condition for the existence of a valid code. It seems to be natural to rest on the characteristic color of the ink, which is known. By resting on the conditions forced by the previous centring treatment we have a margin to guarantee the region in which the code should be placed (Fig. 3). Thus, a limited region in the centre of the can is traced in the code search. This is equivalent to looking for enough pixels that fulfil the ink color requisites (fixed between two thresholds).

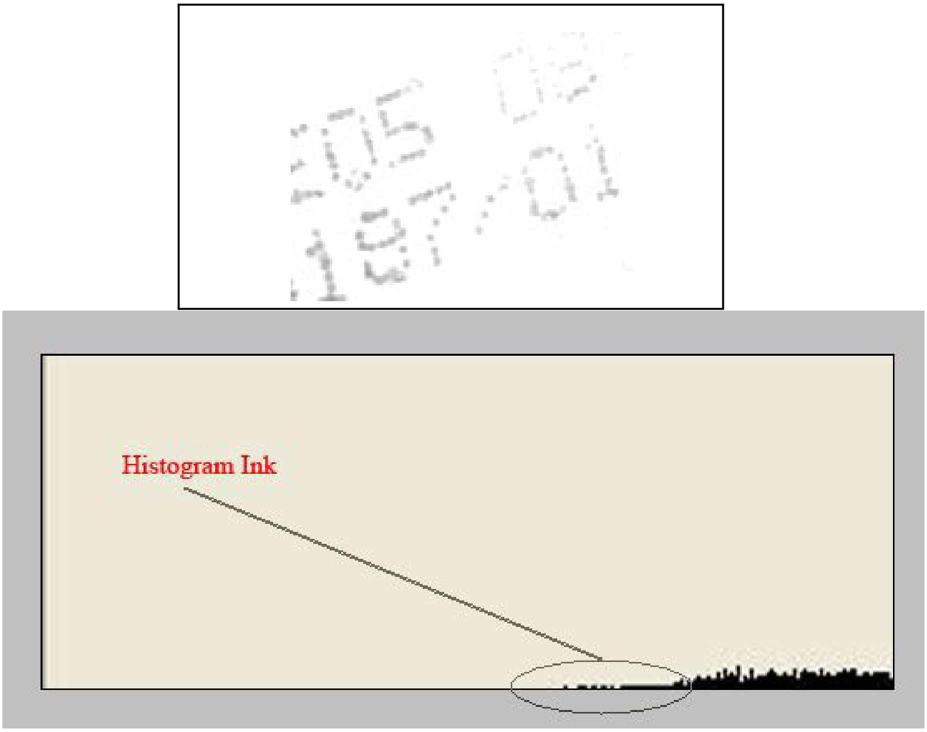


Fig. 3. "Presence/Absence of ink" window and associated histogram

2.6 Validation of Text

In Fig. 4, a biologically inspired architecture for this application is presented. The basic ideas of multichanneling in the visual tract are present. On the input image, a multiprocess task is first triggered to extract the area of interest where the first text is to be located. Thus, a second multichannel analyses the possible singularities in the text. The final validation consists of determining the coherence and plausibility of text syntactically and semantically. All these processes are independent and operate separately. Thus, the labels {1}, {2}, {3} and {4} denote different stages within the same visual tract. The illumination conditions are also relevant and have been considered to have a high contrast between the code and the surface of the can, which allows the processor concentrate in the validation process

2.7 Platform of Development

In order to support all the described machinery, a general image processing software containing a set of general purpose algorithms has been built from scratch (Fig. 6). By means of that, we intend to control almost completely the development conditions for the exam and treatment of images in real time.

Its most outstanding characteristics are the following:

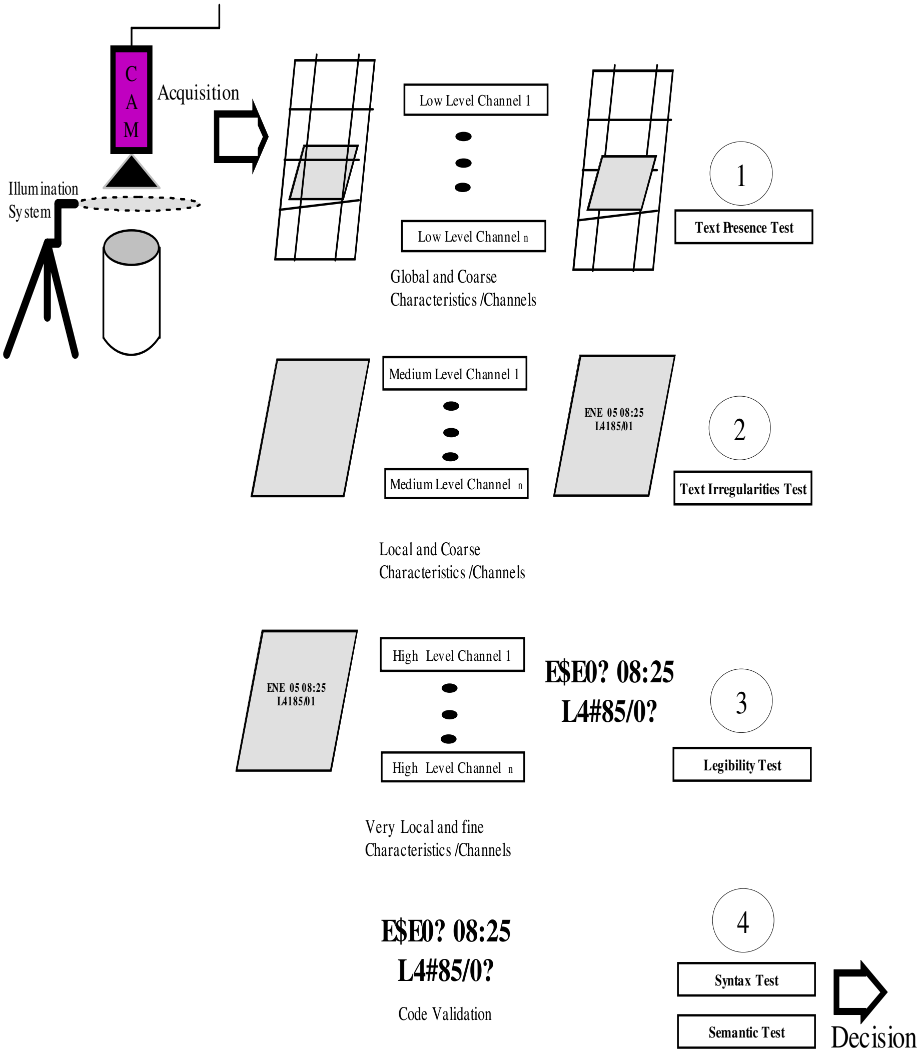


Fig. 4. Multi-level and multi-channel architecture for validation

- *Multi-platform*: Support different operating systems and machines.
- *Free Code*: It does not depend on commercial licences as much in compilation time as in execution time.
- *Modular Code*: The application is divided in independent modules and selfcontents, which are supported by an architecture of classes.
- *Architecture of Plugins*: The platform is supported by a plugins mechanism, that is, fragments of independent code that provide new functions to the general application. It is possible to create as much plugins as desired in order to incorporate capacities without knowing the internal working details of the application.

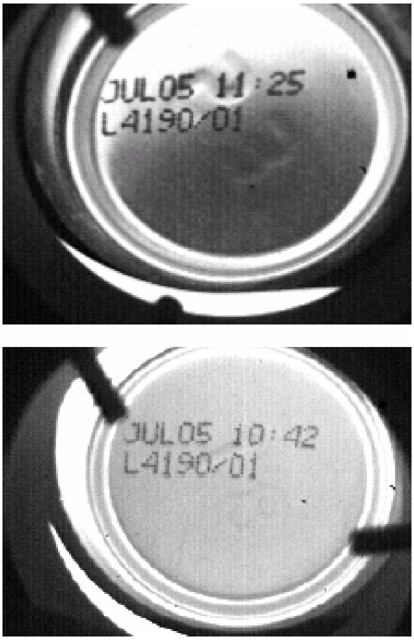


Fig. 5. Illumination conditions

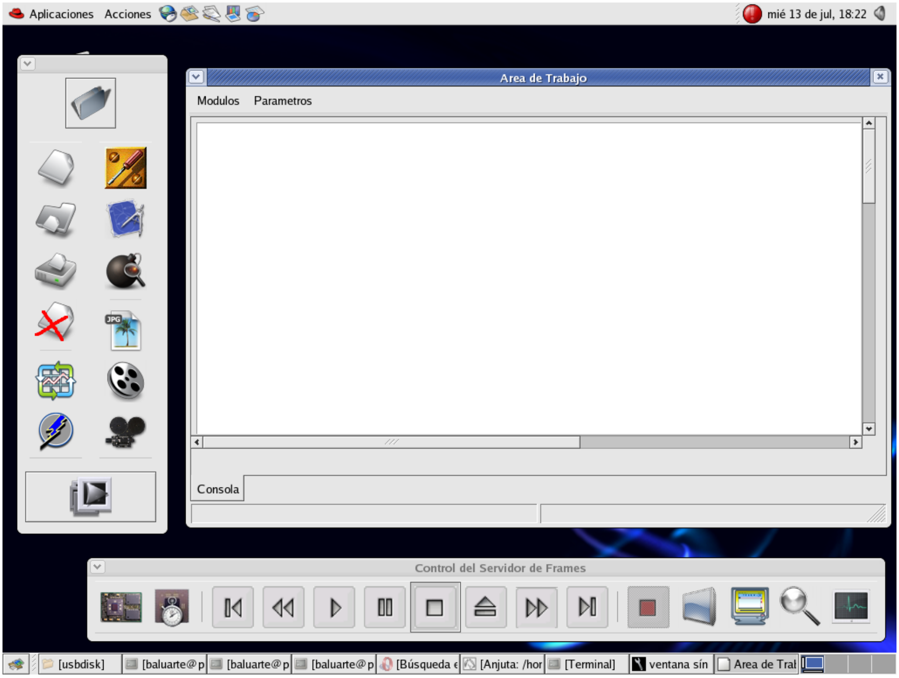


Fig. 6. A general view of the developed software. Three main windows. General options (left). Acquisition control (below). Plugins handling and configuring (center).

Efficiency: The generation of machine code is achieved by a compiler that admits four levels of optimization (gcc). This is possible because the programming language C/C++ (medium-high level language), with a long history and experience, has been chosen. The following level is the planning of multi-thread algorithms, as is the case of the platform. A thread is a sequence of independent instructions within the processor. The advantage of launching multiple threads is relevant in systems with more than a processor, as it invites the resources manager to run the threads in a balaced way. The resources manager can only distribute the sequences of instructions if it knows how independent are the sequences between themselves and a multi-thread architecture is precisely about that: To distribute the assigned tasks in independent flows of instructions.

3 Conclusions and Future Work

A multi-parallel multi-channel scheme has been presented for the solving of a daily problem in the industry. The simplification of the problem in simple tasks and these, in their turn, in simple operations has been the main priority in the interests of speed.

A special image treatment software has been implemented in order to assist the development of the global project. This software can be used in different machines and under different operating systems, encomprising all needed algorithms.

Our immediate purpose is to conclude the development of the strongly multi-level/multi-thread validation algorithm and verify whether it is an effective solution or not for a high speed treatment as the one demanded by the industry.

Along this line of work we expect to improve our lab prototype, at the same time beginning a library of specialized modules which make exhaustive use of the proposed multichannel architecture.

References

1. Alemán-Flores, M., Leibovic, K.N., Moreno Díaz jr, R.: A computacional Model for Visual Size, Location and Movement, Springer Lectura Notes in Computer Science, Vol 1333. Springer-Verlag. Berlin Heidelberg New York (1997) 406-419
2. Quesada-Arencibia, A., Moreno-Díaz jr, R., Alemán-Flores, M., Leibovic, K.N: Two Parallel Channel CAST Vision System for Motion Analysis. Springer Lecture Notes in Computer Science, Vol. 2178. Springer-Verlag. Heidelberg New York (2001) 316-327
3. Quesada-Arencibia, A.: Un Sistema Bioinspirado de Análisis y Seguimiento Visual de Movimiento. Doctoral Dissertation. PhD Thesys. Universidad de Las Palmas de Gran Canaria (2001)
4. J.C. Rodríguez Rodríguez, A.Quesada-Arencibia, R.Moreno-Díaz jr, and K.N. Leibovic: On Parallel Channel Modelling of Retinal Processes Vol 2809 Springer-Verlag. Berlin Heidelberg New York (2003) 471-481
5. Leibovic, K.N., Science of Vision, Springer Verlag, New York, 1990.

Soft Computing and Geometrical Control for Computer Aided Driving

Javier Alonso Ruiz, Teresa de Pedro, Carlos González, and Ricardo García

Industrial Automation Institute, CSIC, La Poveda, Arganda del Rey, 28500 Madrid, Spain
{jalonso, tere, gonzalez, ricardo}@iai.csic.es

Abstract. After having designed control systems for real autonomous cars in an urban environment using straight lines as reference [2], we are now trying to build a fuzzy control system based on clothoids [3], the curve used in roads and train tracks. This paper proposes a method based on soft computing and upgraded using genetic algorithms. Both sets of simulations are shown and compared.

1 Introduction

This paper is based on our previous research into real autonomous car controllers in urban environments. These control systems uses straight lines as reference lines [2]. Now, we propose a fuzzy control system based on clothoid curves [3] to navigate the bend sections of the road. Our goal is to improve the car's behaviour on bends by reducing steering wheel efforts. This improvement will reduce electrical consumption. And a decrease in excessive steering wheel movements will increase the comfort feeling.

The system designs a course map as a succession of straight lines and bends. Each bend is planned as a soft curve with a continuous and smooth curvature progression. The fuzzy controller uses this soft curvature progression to navigate on bends. To be more precise this differential equation system is shown in [4] relating the state and control variables and its integration equations for admissible paths:

$$\begin{pmatrix} x' & y' & \theta' & \kappa' \end{pmatrix} = v \cdot \begin{pmatrix} \cos\theta & \sin\theta & \kappa & 0 \end{pmatrix} + \sigma \cdot \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix} \tag{1}$$

where $(x\ y\ \theta\ \kappa)$ stand for attitude and curvature, and σ is $\sigma = \kappa' = \Phi/\cos^2\Phi$ (Φ is the wheel angle). Our proposal is to express the position of the car depending on curvature, $f(x,y,\kappa,v,t)$, related to the steering angle, and to control the car using its usual inputs, acceleration and curvature (v', $\kappa=1/r$). This curvature-based fuzzy logic control method has been optimized using genetic algorithms.

2 Trajectory Design

The car's course is considered as a succession of straight-line and bend sections. For each bend section, there are three reference points, the starting point, the ending point and one point from the middle of the curve. Therefore, the straight-line sections start at the end of a bend section and end at the beginning of the next one. As the fuzzy

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 501 – 506, 2005.
© Springer-Verlag Berlin Heidelberg 2005

control system has been tested extensively on straight-line sections [2], there will be no further references.

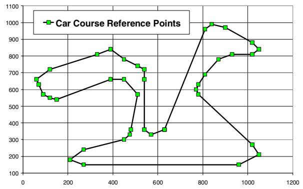


Fig. 1. Jerez circuit car course reference points

The trajectory on the bend sections is designed in the same way as bends in roads are and should be as smooth as possible [6]. Therefore, the trajectory changes its curvature as it advances in the bend. At the beginning of the bend the curvature is zero. It then starts to increase until it reaches a certain value. The curvature then remains constant until, in the last part, it falls to zero. The clothoid spiral is used to ensure that this transition to the circular curve is smooth. The clothoid, also known as Cornu's spiral or Euler's spiral, makes a perfect transition spiral, as its curvature increases linearly with distance along the spiral. The curvature of a clothoid spiral is linearly related to its arc length. And, when the path of the bend is followed at a uniform velocity, the speed of rotation is linear. The curve with the reverse relation is called the anti-clothoid and is used to return from the circular curve to the next straight-line section.

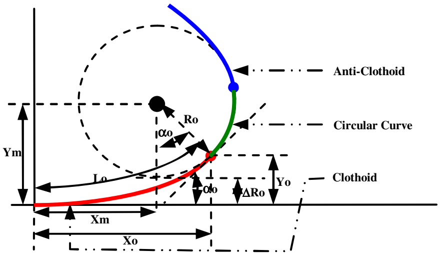


Fig. 2. Transition from the approaching clothoid to the circular curve, and the clothoid parameters

The parameters shown are the circumference radius (R_o), the approaching transition clothoid total length (L_o), the circular curve minimum distance to the initial straight line (ΔR_o), the coordinates of the tangency point between the clothoid and the circumference (X_o,Y_o), the coordinates of the circumference center (X_m,Y_m), and the angle between the initial straight-line segment and the tangent line at the tie point of the two curves (the clothoid and the circular curve) (α_o).

Each bend has its own curvature radius, but if this parameter is not available, the circular curve radius is obtained from the distance between the starting and the ending point of the bend. A clothoid curve is used to reach the circular curve. Its intrinsic equation is:

$$R \cdot L = A^2 \tag{2}$$

where R is the curvature radius at one point, L is the curve length between its inflexion point (where R is infinite) and the point of radius R, and A is the clothoid characteristic parameter. As soon as the maximum speed on the curve is fixed, the value of the transition clothoid characteristic parameter A (in meters), and the clothoid minimum length L can be inferred [6].

Once the clothoid parameters have been obtained, the center of the circumference is calculated. A distance d is defined as the sum of the circumference radius and the circular curve minimum distance to the initial straight line ($d = R_o + \Delta R_o$). The crossing point (c0) of the parallel lines at distance d to the entrance and exit straight lines is obtained. This point will be the center of the circumference that contains the circumference arc. Once the center has been found, the perpendicular lines to the entering and exiting straight lines that go through this circumference center determine the points (n1, n2), and the starting points of the clothoids (m1, m2).

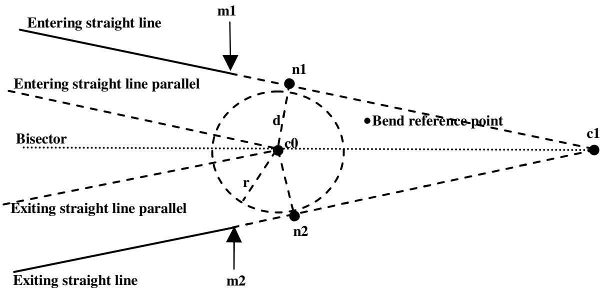


Fig. 3. Initial circumference layout

To guarantee that the designed curve goes through a given bend reference point, the center of the circumference will be moved throughout the bisector of the entering and exiting straight lines just before the calculation of m, and n points.

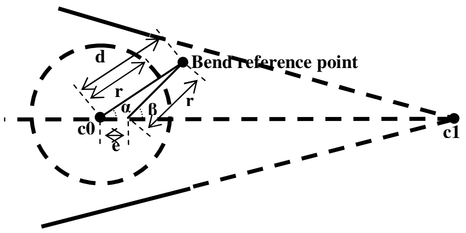


Fig. 4. Center of the circumference adjustment

The distance moved (e) depends on the distance between the circumference center(c0) and the bend reference point (d), the circumference radius (r) and the angles between the bisector of the entering and exiting straight lines, and the lines that go to the bend reference point (α and β).

$$d \cdot \sin\alpha = r \cdot \sin\beta \qquad d \cdot \cos\alpha = e + r \cdot \cos\beta \tag{3}$$

This is solved:

$$\beta = \arcsin\left(\frac{d}{r}\sin\alpha\right) \qquad e = d \cdot \cos\alpha - r \cdot \cos\beta \tag{4}$$

Moving a distance e to the center of the circumference, the designed curve goes through the intermediate point of the curves that has been provided.

Once all the necessary points have been found, the length (in degrees) of the circumference arc is calculated. This measure is calculated by reducing the angle formed by the two straight-line sections from 180º (U-turn) and, twice, the angle turned in the approach curves [6] (clothoids). Thus the vehicle will follow the entering straight line (with null curvature) as far as the point m1 at which the clothoid begins. Its curvature will increase until the circumference (to 1/radius curvature) is reached. Next, it will turn the previously calculated number of degrees, its radius of curvature remaining constant. And then it will reduce that curvature progressively until it reaches point m2.

3 Fuzzy Controller

We have used a fuzzy logic control system to get a solution that is as smooth as possible [1]. The state of the system is described with different sets of variables depending on the car's desired behavior. This article focuses on bend sections control only, because the straight-line sections control has been already presented in other papers [1], [2] and [5]. So, if the car is tracking a bend, the variables are based on the curvature. This control system must maintain the curvature goal that depends on the point of the curve where the car is. The control variables used are the curvature error and the error accumulated on that bend (a fuzzy PI fuzzy control). The car's curvature is calculated from the steering wheel position and the distance between the front and rear axes of car. These variables are "*Curvature error",* the difference between the car's desired and real curvature, and "*Curvature error summation",* the accumulated curvature error.

The input fuzzy linguistic variables will have only one fuzzy partition, but we will use the fuzzy modifiers LESS THAN and MORE THAN for operations, as we have three fuzzy partitions. Its fuzzy membership functions will be very simple to reduce the computational cost in a real-time car controller.

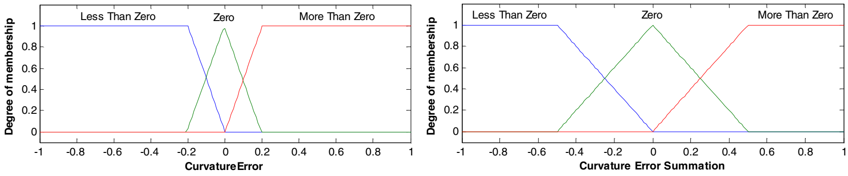


Fig. 5. The membership functions of "Curvature error" and "Curvature error summation" linguistic variables

The fuzzy rules for the bend sections are:

Table 1. Bend section fuzzy rule base

Rule	IF	Precedent	THEN	Consequent
7	**IF**	Curvature_error MORE_THAN (zero)	**THEN**	Steering negative
8	**IF**	Curvature_error zero	**THEN**	Steering unchanged
9	**IF**	Curvature_error LESS_THAN (zero)	**THEN**	Steering positive
10	**IF**	Sum_Curvature_error MORE_THAN (zero)	**THEN**	Steering negative
11	**IF**	Sum_Curvature_error zero	**THEN**	Steering unchanged
12	**IF**	Sum_Curvature_error LESS_THAN (zero)	**THEN**	Steering positive

The system actuation is defined by a goal position for the steering wheel and another goal position for the accelerator-brake set [5]. The output variable that controls the steering wheel has three normalized singleton values: negative (-1), unchanged (0), and positive (+1). And its surface control obtained is:

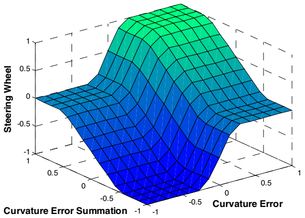


Fig. 6. Steering wheel control surface

4 GA Optimization

The control system performance is improved by modifying the membership function parameters. The objectives of the genetic algorithm fitness function are to get a control as smooth as possible (minimize steering wheel movements). Thus, the effort is defined as the difference between consecutive steering wheel goals. To ensure that the car trajectory still matches the car's designed path, the maximum error on bend sections is added to the fitness function. Both objectives are combined as follows:

$$\boldsymbol{Fitness = k1 * effort + k2 * Bends\_maximum\_error} \tag{5}$$

We have obtained different optimized set values depending on the nature of the circuit curves (motorways with small curvature or race circuits).

5 Experimental Results and Conclusions

The control system performance improvement can be seen from the simulator results. The simulator uses detailed maps of competition circuits, primarily Jerez, keeping its real proportions. And it takes into account the mechanical characteristics of our CITROËN cars. Fig. 7 left shows the simulated steering wheel effort results without optimization (top line) and after the application of the genetic algorithm (bottom line). In Fig. 7 right, the car course in the simulations goes through all the reference points and is adjusted to the circuit dimensions. Therefore, the car remains on the road.

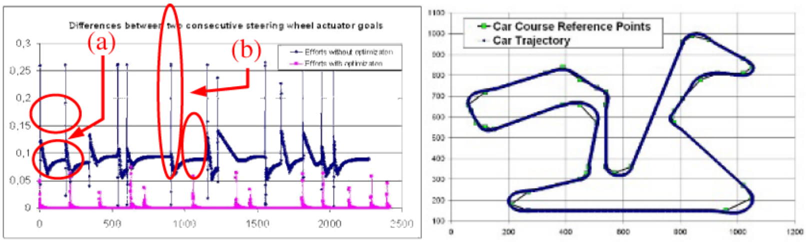


Fig. 7. On left: Differences between two consecutive steering wheel actuator goals. **On right:** Plotting a course on the Jerez circuit map

The effort reduction along each bend (Fig. 7a) ensures that electrical consumption will be reduced and the comfort feeling will increase. And the effort reduction at the starting and ending point of each bend (Fig. 7b) ensures that the expected lifetime for the steering wheel motor will be lengthened considerably. This also contributes to a reduction in electrical consumption, and an increase in the comfort feeling.

Acknowledgments

This work was supported by the Spanish Ministry of Education and Science ISAAC project, DPI2002-04064-C05-02, and the Spanish Ministry of Development COPOS project, BOE 280, November 22, Res. 22778.

References

1. García R and de Pedro T, “First Application of the ORBEXoprocessor : Control of Unmanned Vehicles” *Mathware and Soft Computing*, no. 7, vol 2-3, pp 265-273, 2000
2. García R, de Pedro T, Earanjo J E, Reviejo J and González C, “Frontal and Lateral Control for Unmanned Vehicles in Urban Tracks” *IEEE Intelligent Vehicle Symposium (IV2002)*
3. Jiménez Shaw J, “Resolución numérica completa de la Ecuación de la Clotoide”, web page and Clothos constants calculation program. http://javier.jimenezshaw.com/
4. Laugier C and Fraichar T, “Decisional Architectures for Motion Autonomy” chapter 11 in *Intelligent Vehicle Technologies*, editors: Vlacit, ParenT, Harashima, SAE International, ISBN 0 7680 0780 I
5. Naranjo J E, Reviejo J, Gonzalez C, García R, de Pedro T, “A Throttle & Brake Fuzzy Controler: Towards the Automatic Car” *Lecture Notes in Computer Science: Computer Aided Systems Theory – Eurocast 2003*, LNCS 2809 pp 291-301
6. Web page www.carreteros.org, Norm 3.1-IC “Trazado”

A Monocular Solution to Vision-Based ACC in Road Vehicles

Miguel Ángel Sotelo, Jesús Nuevo, Manuel Ocaña, and Luis Miguel Bergasa

Department of Electronics, University of Alcalá, Alcalá, Madrid, Spain
{jnuevo, michael, mocana, bergasa}@depeca.uah.es

Abstract. This paper describes a monocular vision-based Adaptive Cruise Control (ACC) System in the framework of Intelligent Transportation Systems (ITS) technologies. The challenge is to use a single camera as input, in order to achieve a low cost final system that meets the requirements needed to undertake serial production.

1 Introduction

A monocular imaging device (a single FireWire digital camera) is deployed to provide "indirect range" measurements using the laws of perspective. Some previous developments use available sensing methods such as radar [1], stereo vision [2], or a combination of both [3]. Only a few works deal with the problem of monocular vehicle detection using symmetry and color features [4], or pattern recognition techniques [5]. In the current work, the searching space is reduced in an intelligent manner in order to increase the performance of the detection module. Accordingly, road lane markings are detected and used as the guidelines that drive the vehicle searching process. The area contained by the limits of the lanes is scanned in order to find vehicle candidates that are passed on to the vehicle recognition module. This helps reduce the rate of false positive detections. In case that no lane markings are detected, a basic *area of interest* is used instead covering the front part ahead of the ego-vehicle. The description of the lane marking and vehicle detection systems is provided below, together with some graphical results.

2 System Description

2.1 Lane Tracking

The system is divided in three modular subsystems with specific functions. The first subsystem is responsible for lane detection and tracking, as well as lane crossing monitoring. Images obtained from the camera are processed and clothoid curves are fitted to the detected markings. The algorithm scans up to 25 lines in the *area of interest*, from 2 meters in front of the camera position to below the horizon. The developed algorithm implements a non-uniform spacing search that reduces certain instabilities in the fitted curve. The final state vector is composed of 6 variables [7] for each line on the road: c_{oh}, c_{1h}, c_{ov}, c_{1v}, x_o, ψ_o, where c_{oh} and c_{1h} represent the clothoid

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 507 – 512, 2005.
© Springer-Verlag Berlin Heidelberg 2005

horizontal curvature parameters, c_{ov} and c_{1v} stand for the clothoid vertical curvature parameters, while x_o and ψ_o are the lateral error and orientation error, respectively, with regard to the centre of the lane. The clothoid curves are then estimated based on lane marking measurements using a Kalman filter for each line. These lines conform the *area of interest*. Figure 1 depicts a sequence of images in which the result of the lane tracking algorithm is overprinted on the road images.

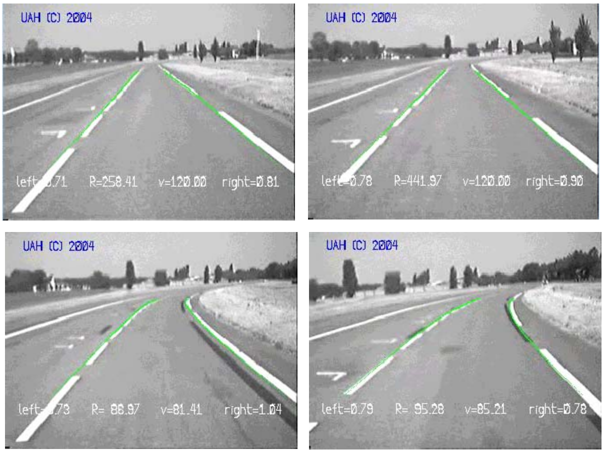


Fig. 1. Lane tracking example in a sequence of images. The green lines represent the estimated lines of the road. The example also depicts the error between the left wheel of the car the the left lane (left), the error between the right wheel of the car and the right lane (right), the radious of curvature of the road estimated at a lookahead distance of 50m (R), and the maximum recommended velocity to bend the curve (V) according to the radious of curvature.

2.2 Car Detection and Recognition

An attention mechanism has been devised with the intention of filtering out inappropriate candidate windows based on the lack of distinctive features, such as horizontal edges and symmetrical structures, which are essential characteristics of road vehicles. This has the positive effect of decreasing both the total computation time and the rate of false positive detections. Each road lane is sequentially scanned, from the bottom to the horizon line of the image, as depicted in figure 2, looking for collections of horizontal edges that might represent a potential vehicle. The scanned lines are associated in groups of three. For each group, a horizontality coefficient is computed as the ratio of connected horizontal edge points normalized by the size of the area being analysed. The resulting coefficient is used together with a symmetry analysis in order to trigger the attention mechanism. Apart from the detected road lanes, additional virtual lanes have been considered so as to cope with situations in which a vehicle is located between two lanes (for example, if it is performing a change lane manoeuvre). Virtual lanes provide the necessary overlap between lanes, avoiding both misdetections and double detections caused by the two halves of a vehicle being separately detected as two potential vehicles. A virtual lane is located to provide overlap between two adjoining lanes. On average, the system generates 5 candidate windows per frame that are passed on to the classifier. Nonetheless, this figure is bound to change depending on traffic conditions.

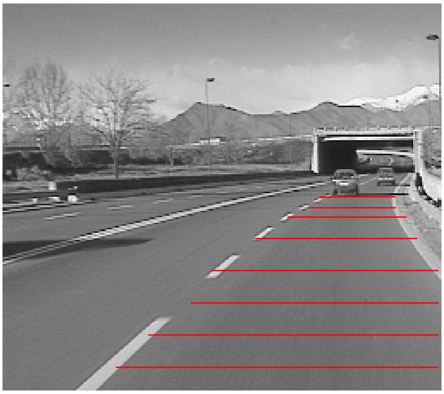


Fig. 2. Sequential vehicle candidates searching along the detected lane

The road vehicle class contains quite a large amount of different cars that makes it a non-homogeneous cluster. In consequence, it makes sense to use a distributed learning approach in which each individual part of the vehicle is independently learnt by a specialized classifier in a first learning stage. The local parts are then integrated by another classifier in a second learning stage. According to the previous statements, the proposed approach can be regarded as a hierarchical one. By using independent classifiers in a distributed manner the learning process is simplified, as long as a single classifier has to learn individual features of local regions in certain conditions. Otherwise, it would be difficult to attain an acceptable result using a holistic approach. We have considered a total of 3 different sub-regions for each candidate region. The 3 sub-regions cover the most characteristic parts of the vehicle. Two small sub-regions have been located in the area of the region where the wheels are supposed to be. A third sub-region is located in the central part of the region, covering the area where car plates and rear windshield are usually placed. The locations of the three sub-regions have been chosen in an attempt to detect coherent and structural car features.

A set of features must be extracted from each sub-region and fed to the classifier. Before doing that, the entire candidate region of interest is pre-processed using a Canny operator in order to enhance the differential information contained in it (edges). The Canny image provides a good representation of the discriminating features of the car class. On the one hand, edges, both horizontal and vertical, are clearly visible and distinguishable. On the other hand, the vertical symmetry of a car remains unchanged. In addition, edges are not affected by colours or intensity. This property makes the use of edges robust enough to different car models of the same type. In a first attempt, a set of features was extracted from each sub-region using the normalized histogram based on the co-occurrence matrix of the pre-processed sub-region (four co-occurrence matrixes were computed using four different searching vectors). This option was discarded in practice after observing the results derived from it. The use of co-occurrence matrixes proved to be non-discriminating enough as long as other parts of the image (that do not contain a car) can trigger the attention mechanism since they exhibit similar co-occurrence based values. The fact is that the information provided by co-occurrence matrixes does not uniquely reflect the 2D structure of a car. Instead, the pre-processed sub-region is directly applied to the input of the classifier, as the set of features that is finally used for learning. The dimensions of the entire region of interest are normalized before being fed to the classifier. A size of 70x80 pixels has been chosen. This size is adequate for detecting vehicles at long distances (up to 80 meters).

Several training sets were created for each sub-region in order to store representative samples in different weather and illumination conditions, as suggested in [8]. This technique allows to learn every separate training set using a specialized Support Vector Machine (SVM) [6] that yields excellent results in practice. Otherwise, the use of a global classifier would demand for excessive generalization of the classifier. General classifiers are doom to failure in practice when dealing with images acquired in outdoor scenarios, as they contain a huge variability. The global training strategy is carried out in two stages. In a first stage, separate SVM-based classifiers are trained using individual training sets that represent a subset of a sub-region. Each SVM classifier produces an output between -1 (non-vehicle) and +1 (vehicle). Accordingly, it can be stated that this stage provides classification of individual parts of the candidate sub-regions. In a second step, the outputs of all classifiers are merged in a single SVM classifier in order to provide the final classification result.

3 Results and Conclusions

The system was implemented on a Power Mac at 2.0 GHz running the Knoppix Linux Operating System. The complete algorithm runs at 25 frames/s. We created a database containing 2000 samples of road vehicles. The samples were extracted from recorded images acquired in real experiments onboard a road vehicle in real traffic conditions in Madrid. All training sets were created at day time conditions using the TsetBuilder tool [9], specifically developed in this work for this purpose. By using the TsetBuilder tool different candidate regions are manually selected in the image on a frame-by-frame basis. This allows to select candidate regions containing vehicles of different size, from different manufacturers, and so on. The number of non-vehicle samples in the training sets was chosen to be similar to the number of vehicle samples. Special attention was given to the selection of non-vehicle samples. The training of all SVM classifiers was performed using the free-licence LibTorch libraries for Linux. We obtained a detection rate of 85% in a test set containing 1000 images, and a false detection rate of 5%. The performance of the single-frame recognition process is largely increased by using multi-frame validation based on a Kalman filter. As an example, figure 3 shows a sequence of images in which a vehicle is detected and tracked along the lane of the host vehicle. A blue box is overprinted over the detected vehicle indicating the estimated distance measured from the host vehicle. Other vehicles appearing along the adjoining lane are marked with a horizontal red line. The distance between the ego-vehicle and the preceding vehicle along the lane becomes the input to the Adaptive Cruise Control (ACC) System.

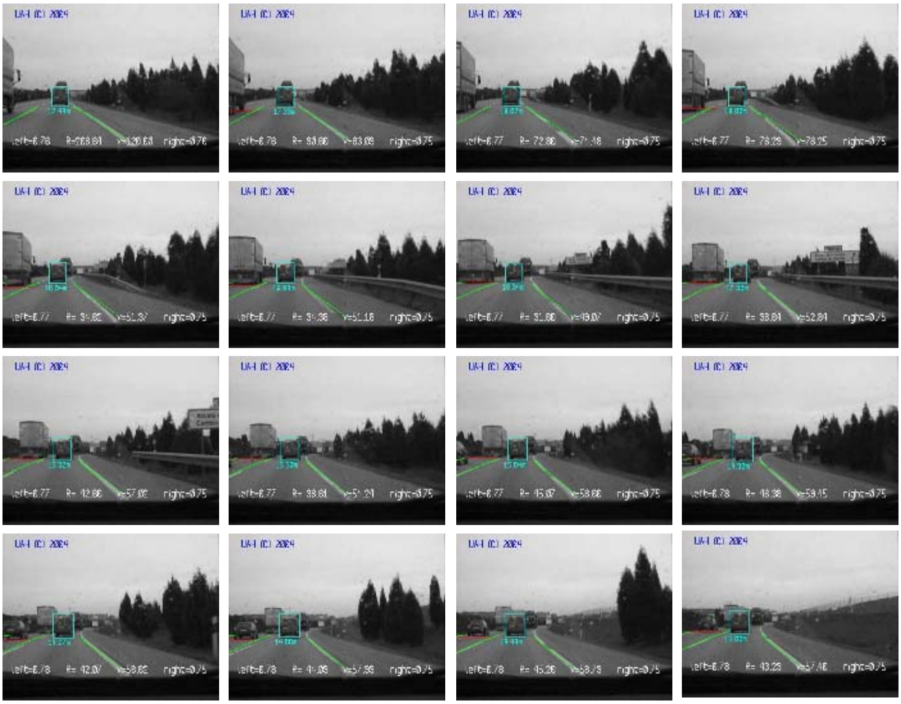


Fig. 3. Vehicle tracking example in a sequence of images

The results achieved up to date with a set of 2000 samples are encouraging. Nevertheless they still need to be improved before being safely used as an assistance driving system onboard road vehicles in real conditions. For this purpose, the content of the training sets will be largely increased by including new and more complex samples that will boost the classifier performance, in particular when dealing with difficult cases. We aim at enhancing the classifier ability to discriminate those cases by incorporating thousands of them in the database. In addition, the attention mechanism will be refined in order to provide more candidates around the original candidate region. This will reduce the number of candidate regions that only contain a part of the vehicle, i.e., those cases in which the entire vehicle is not completely visible in the candidate region due to a misdetection of the attention mechanism.

Acknowledgments

This work has been funded by Research Projects CICYT DPI2002-04064-05-04 (*Ministerio de Educación y Ciencia*) and FOM2002-002 (*Ministerio de Fomento*, Spain).

References

1. G. R. Widman, W. A. Bauson, and S. W. Alland, "Development of collision avoidance systems at Delphi Automotive Systems". In Proc. Int. Conf. Intelligent Vehicles, pp. 353-358, 1998.
2. T. Williamson and C. Thorpe, "Detection of small obstacles at long range using multibaseline stereo". In Proc. Int. Conf. Intelligent Vehicles, pp. 311-316, 1998.
3. R. Labayrade, C. Royere, D. Gruyer, and D. Aubert, "Cooperative fusion for multi-obstacles detection with use of stereovision and laser scanner". In Proc. Int. Conf. On Advanced Robotics, pp. 1538-1543, 2003.
4. A. Broggi, M. Bertozzi, A. Fascioli, C. Guarino Lo Bianco, and A. Piazzi, "The Argo autonomous vehicle's vision and control systems". International Journal of Intelligent Control and Systems. Vol. 3, No. 4, 409-441, 2000.
5. G. P. Stein, O. Mano, and A. Shashua, "Vision-based ACC with a single camera: bounds on range and range rate accuracy". In Proc. Int. Conf. Intelligent Vehicles, 2002.
6. J. C. Christopher Burges, "A Tutorial on Support Vector Machines for Pattern Recognition". Data Mining and Knowledge Discovery, 2,121-167 (1998). Kluwer Academic Publishers.1.
7. E. D. Dickmanns and B. D. Mysliwetz. "Recursive 3-D Road and Relative Ego-State Recognition". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 2, February 1992.
8. A. Shashua, Y. Gdalyahu, and G. Hayun. "Pedestrian detection for driving assistance systems: single frame classification and system level performance". In Proc. IEEE Intelligent Vehicles Symposium, pp:1-6, Parma, Italy, June 14-17 2004.
9. J. Nuevo. "TestBuilder Tutorial". Technical Report, Department of Electronics, University of Alcalá, 2005. ftp://www.depeca.uah.es/pub/vision/SVM/manual

Multi-objective Dynamic Optimization for Automatic Parallel Parking

Javier de Lope and Darío Maravall

Department of Artificial Intelligence, Faculty of Computer Science,
Universidad Politécnica de Madrid, Campus de Montegancedo, 28660 Madrid, Spain
{jdlope, dmaravall}@dia.fi.upm.es

Abstract. This paper addresses the problem of automatic parallel parking by a back-wheel drive vehicle, using a biomimetic model based on direct coupling between vehicle perceptions and actions. This problem is solved by means of a bio-inspired approach in which the vehicle controller does not need to know the car kinematics and dynamics, neither does it call for *a priori* knowledge of the environment map. The key point in the proposed approach is the definition of performance indices that for automatic parking happen to be functions of the strategic orientations to be injected, in real time, to the car-like robot controller. This solution leads to a dynamic multi-objective optimization problem, which is extremely hard to be dealt with analytically. A genetic algorithm is therefore applied, thanks to which we obtain a very simple and efficient solution. The paper ends with the results of computer simulations.

1 Introduction

Optimization is one of the most common and pervasive issues in real-world engineering and economic systems and it is at the heart of any decision-making task in which a choice must be made between several alternatives to achieve multiple, sometimes conflicting, objectives. The objectives are generally formalized as analytical functions or performance indices.

The technical literature on optimization methods is really extensive, as this fundamental subject has received a tremendous amount of attention since it came of age about 50 years ago. Single-objective optimization is by far the most researched problem in this field, although many real-life situations are multi-objective optimization problems *per se*. A multi-objective optimization problem is very often converted into a single-objective optimization case by integrating the multiple performance indices into a single one [1].

The standard solution for truly multi-objective optimization problems is to find the so-called Pareto-optimal front. The Pareto front is formed by the solutions in which any change in any of the decision variables aimed at improving a particular performance index will produce a deterioration in some of the other performance indices. Due to the inherent difficulties in calculating the analytical Pareto optimal surfaces for many real-world systems, evolutionary methods have lately been applied to solve multi-objective optimization problems [2,3].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 513–518, 2005.
© Springer-Verlag Berlin Heidelberg 2005

In most of the multi-objective optimization problems that have been tackled by evolutionary techniques so far, the performance indices to be optimized:

$$\begin{array}{l} J_i(\boldsymbol{x}) = J_i(x_1, x_2, \ldots, x_n) \\ \boldsymbol{x} \in \mathbb{R}^n \quad ; \quad i = 1, 2, \ldots N \end{array} \tag{1}$$

are not time-varying and the decision varibles x_1, x_2, $\ldots x_n$ have static constraints:

$$\begin{array}{l} g_i(\boldsymbol{x}) \geq 0 \quad ; \quad i = 1, 2, \ldots p \\ h_i(\boldsymbol{x}) = 0 \quad ; \quad i = 1, 2, \ldots q \end{array} \tag{2}$$

Only very recently have a number of papers dealing with the application of evolutionary computation to solve multi-objective optimization cases been published, in which either the performance indices $J_i(\boldsymbol{x})$, the decision variables $\boldsymbol{x}$ or both are time dependent and dynamic [4,5]. Most of this research work refers to variations on static optimization problems, in which some well-known static benchmark optimization functions are exposed to time-varying dynamics. Noise in the measurement of the performance indices or in the decision variables is sometimes also added.

In this paper, we present an engineering problem (automatic car parking) that ultimately leads to the formalization of an active multi-objective dynamic optimization problem as a cooperative game. However, instead of using evolutionary controllers to search for an optimal, or at least efficient, set of *if-then* reasoning rules, we propose a much simpler method based on what we call a biomimetic approach to sensory-motor coordination. Sensory-motor coordination is a central issue in the design of most real-world engineering systems, as the designer somehow has to deal with the coordination of perceptual information and physical actions.

2 Biomimetic Approach for Sensory and Motor Coordination in Autonomous Robots

The tailoring of this biomimetic approach to the parking problem is illustrated in Fig. 1. The robot vehicle considered in this paper is a conventional back-wheel drive car, whose dynamic equations can be modeled, for the low-speed range typical of parking maneuvers, as:

$$\begin{array}{l} \dot{x}(t) = v(t) \cos\, \theta(t) \\ \dot{y}(t) = v(t)\, \sin\, \theta(t) \\ \dot{\theta}(t) = v(t)/L\, \tan\phi(t) \end{array} \tag{3}$$

where (x, y) are the coordinates for the point of application of the force of traction on the vehicle; θ is the heading of the vehicle on the plane on which it is moving; v is its speed; L is the distance between the front and back axes, and the variable ϕ is the direction of the driving wheels with respect to the vehicle heading θ. Obviously, (v, ϕ) are the robot control variables and (x, y, θ) are its

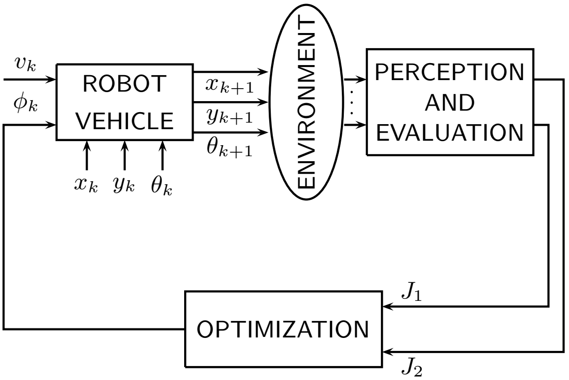


Fig. 1. Conceptual diagram of the biomimetic model

state variables. The discrete version of (3) is:

$$\begin{aligned} x_{k+1} &= x_k + v_k \cos\theta_k \\ y_{k+1} &= y_k + v_k \sin\theta_k \\ \theta_{k+1} &= \theta_k + v_k/L \tan\phi_k \\ |\phi_k| &< \phi_{\max} \end{aligned} \tag{4}$$

where $\phi_{\max}$ is the maximum angle that can be applied to the direction of the driving wheels.

In the case of automatic parking, there are two behavior indexes of interest: J_1 and J_2. These two indexes quantify the goal that the robot should park in the final position (x_d, y_d) and the goal that the robot should park in line with the parking space direction, θ_d, respectively. Hence:

$$\begin{aligned} J_1 &= \tfrac{1}{2}\left[(x-x_d)^2 + (y-y_d)^2\right] \\ J_2 &= \tfrac{1}{2}\left(\theta-\theta_d\right)^2 \end{aligned} \tag{5}$$

Supposing that the vehicle maneuvers at constant speed, the other available control variable, ϕ, should minimize both indexes:

$$\dot{\phi}(t) = -\mu_1 \frac{\partial J_1}{\partial \phi} - \mu_2 \frac{\partial J_2}{\partial \phi} \tag{6}$$

where μ_1 and μ_2 weight the importance of each goal. The discrete version of (6) is:

$$\phi_{k+1} = \phi_k - \mu_1 \left.\frac{\partial J_1}{\partial \phi}\right|_{\phi(k)} - \mu_2 \left.\frac{\partial J_2}{\partial \phi}\right|_{\phi(k)} \tag{7}$$

Equations (6) and (7) raise an important practical problem, which is what we might refer to as the relationship between the *distal* sensory information scale (given by the gradients or changes of the behavior indexes ΔJ_1 and ΔJ_2) and the *proximal* actions scale (given by the gradients of the control actions $\Delta\phi_k = \phi_k - \phi_{k-1}$ or $\Delta\phi_{k+1} = \phi_{k+1} - \phi_k$).

A practical way of solving this scale problem is to establish a tabular relationship between the distal levels and the proximal levels. In particular, the range of robot actions (speed v and steering wheel turn ϕ) can be distributed on a discrete scale of values. In view of the practical importance of the distal/proximal relationship (which in living beings takes a lot of learning), one open line of research is to develop flexible diagrams to quantify this relationship, where fuzzy linguistic variables or even genetic algorithms could play a role in adequate tuning. In this paper, however, we have not addressed this problem and have used a range of just three steering wheel action values. The action values are expressed as angular speeds of the steering wheel: $+10^o/s$, $0^o/s$, $-10^o/s$, depending on whether the ratios of the distal and proximal gradients are positive, zero or negative, respectively. As we have used a control time cycle of 100 ms, the actual steering wheel turns applied to the vehicle have been $+1^o$, 0^o, -1^o, respectively. We have achieved good results with this small scale of actions.

3 Goal Coordination in the Parking Maneuver

Looking at how humans park, we proposed [6] a method for a generic parking maneuver. We find that one very efficient maneuver, provided there are no obstacles, is to approach, in almost any direction, an area close to the position and direction of the space, as shown in Fig. 2a. As of then priority, albeit not absolute, is given to heading and, when the vehicle is aligned with the space, the approach goal takes maximum, but again not exclusive, priority.

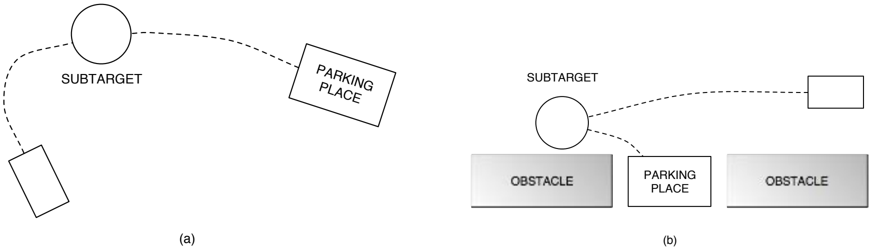


Fig. 2. Illustration of the methods described in the text

Let us take a qualitative look at the execution of this maneuver.

Phase 1. Transfer region approach. This phase can be performed, in principle, without concern for the heading goal J_2. However, the position of the vehicle in this transfer region should be as aligned as possible with respect to the direction of the space.

Phase 2. Alignment with the direction of the space. Once it is positioned in the transfer or subtarget region, the vehicle gives maximum, but not exclusive, priority to the heading goal. Obviously, the more aligned it is, the smoother the maneuver will be.

Phase 3. Parking space approach according to the desired heading. After alignment with the direction of the space, the vehicle's only concern will be to reduce its distance to the space (maximum priority of index J_1). To prevent possible losses of alignment, goal J_2 should retain some, albeit a very weak, influence.

We follow a similar approach for the parallel parking case. Now, when the vehicle is positioned in the subtarget region, it put into reverse to reach the final position and orientation in the parking zone (Fig. 2b).

4 Experimental Results

The experiments were conducted using the University of Sheffield's Genetic Algorithm Toolbox for Matlab [7]. For all cases, a 20-bit resolution binary coding was used for the parameters processed; the parameter ranges depend on the variables to be optimized.

The stochastic universal sampling method was used to select individuals. The crossover probability used is 0.7; the mutation probability is set proportionally to the size of the population, and is never over 0.02. Additionally, an elitism of a 10% from generation to generation is used.

Quality is determined by rewarding the individuals that simultaneously minimize the two indexes J_1 and J_2, that is, the closer an individual is to the position and direction defined as the target, at the end of the path, the better this individual is. Additionally, individuals who manage to reach the target along a shorter path are also considered better, although the weighting of this factor is lower.

The experiments were actually designed by defining a set of initial and final vehicle position and heading pairs that would cover the different relative situations between the source and target. Each individual generated in the evolutionary process was simulated with these initial and final conditions to thus determine its problem-solving quality.

Fig. 3 shows two different parking paths achieved using the proposed procedure. The rectangles in both sides of the parking place represent parked cars. For the maneuver, the vehicle controller also avoids the collision with these obstacles (cars). We can observe how the car have to modify the trajectory to elude the collision.

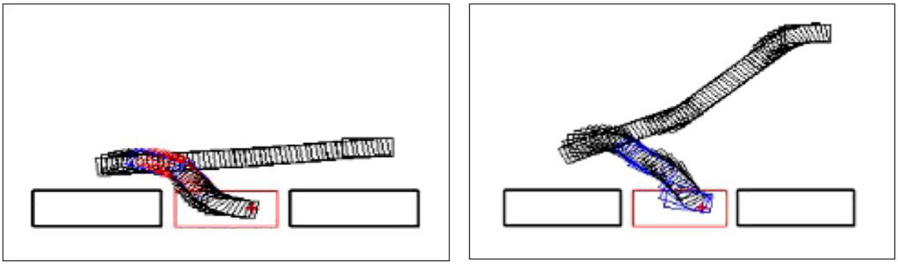


Fig. 3. Parallel parking maneuvers

5 Conclusions

We have presented a solution for conventional nonholonomic vehicle automatic parking. The proposed solution is based on a biomimetic approach that can be used to design extremely simple and robust autonomous robot control systems, as the designer has to inject only the robotic system goals. This approach means, therefore, that the use of dynamic and kinematic robot models and even the aprioristic formal descriptions of their working environments can be ignored. An automatic parking strategy has been designed, and the results obtained using genetic algorithms have been presented.

Acknowledgements

This work has been partially funded by the Spanish Ministry of Science and Technology, project DPI2002-04064-C05-05.

References

1. Deb, K. (2001) Multi-Objective Optimization Using Evolutionary Algorithms. Wiley, New York
2. Fonseca, C.M., Fleming, P.J. (1993) Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. Proc. of the 5th Int. Conf. on Genetic Algorithms, 416–423
3. Coello, C.A. (2003) Special Issue on Evolutionary Multiobjective Optimization. IEEE Trans. on Evolutionary Computation, **7**:2, 97–99
4. Branke, J. (2002) Evolutionary Optimization in Dynamic Environments, Kluwer, Boston
5. Jin, Y., Sendhoff, B. (2003) Connectedness, regularity and the success of local search in evolutionary multi-objective optimization. Proc. IEEE Congress on Evolutionary Computation (CEC-2003), 1910–1917
6. Maravall, D., De Lope, J., Patricio, M.A. (2004) Competitive Goal Coordination in Automatic Parking. Proc. 1st of the European Workshop on Evolutionary Algorithms in Stochastic and Dynamic Environments (EvoSTOC-2004)
7. Chipperfield, A., Fleming, P., Pohlheim, H., Fonseca, C. (1994) Genetic Algorithm Toolbox for Matlab, Department of Automatic Control and Systems Engineering, University of Sheffield

Electric Power Steering Automation for Autonomous Driving

J.E. Naranjo, C. González, R. García, and T. de Pedro

Instituto de Automática Industrial (CSIC),
Ctra. Campo Real Km. 0,200, La Poveda, Arganda del Rey, Madrid 28500, Spain
{jnaranjo, gonzalez, ricardo, tere}@iai.csic.es

Abstract. The automatic control of a vehicle's steering wheel is now one of the most important challenges in the Intelligent Transportation Systems field. In this paper, we present a fuzzy logic-based automatic steering control system for mass-produced electric power steering (EPS) wheel-equipped vehicles that assures human-like behavior. In the literature, we find a lot of theoretical proposals and some simulations, but only a few work teams offer real solutions for this task. One such solution is the work developed by the Autopia Program in which some vehicles have been automated and can perform some maneuvers mimicking human reactions. In this paper, we use the EPS of a Citroën C3 Pluriel to effect the car's behavior. The actuator is controlled from an onboard computer housing a fuzzy logic-based autonomous steering system. The vehicle's internal computers generate the input information, which is read by a CAN bus and a high precision GPS. Some experiments using this equipment on a private test circuit are presented, obtaining an human-like behavior in all the maneuvers.

1 Introduction

The development of Intelligent Transportation Systems (ITS) provides an opportunity to apply advanced technology to systems and methods of transport for efficient, comfortable and safer means of transport. Our work focuses on the area of road transport, and more specifically on the field of intelligent vehicles, which includes the topic of autonomous vehicles. This topic refers to vehicles that are equipped with the instrumentation and intelligence needed to provide the actual vehicle with the required service, that is, an autonomous car must control some or all of its functions without external intervention. The Autopia Program is working on this field, focusing mainly on autonomous driving using fuzzy logic controllers. The steering wheel [1], throttle [2] and brake pedal [3] have been automated, working in Citroën Berlingo vans and experiments were run on private circuits.

There are some examples of automatic steering wheel control, as a step towards achieving automatic driving. In the "Millemiglia in Automatica Tour" [4], a car was equipped with a DC motor attached to the steering wheel through a pulley, which could be moved depending on the commands of an onboard computer housing an analytical control system that received the sensor input through artificial vision. In the Autopia Program [5], two Citroën Berlingo vans have been equipped for automatic driving. Here, the steering wheel has also been modified to be moved by a DC motor

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 519 – 524, 2005.
© Springer-Verlag Berlin Heidelberg 2005

using gears attached to the steering bar because the assist mechanism of conventional cars is powered by hydraulic systems. In this case, the sensor input is provided by a Global Navigation Satellite System. Newer vehicles are equipped with electric power steering (EPS), where steering is assisted by an electric motor that acts directly on the rack bar through a pinion. The motor torque depends on the effort required by the driver. The advantage of this kind of power steering for our purposes is that no external actuator has to be added, and we can easily manage the assist motor from our onboard computer. Some developments, for example, Toyota's automatic parking system have taken this approach [6]. The aim of this paper is to present the fuzzy logic-based EPS control system developed for automatic driving that has been installed in a Citroën C3 Pluriel vehicle (Fig. 1). This vehicle has been tested on a private circuit, and human-like behaviors have been achieved.

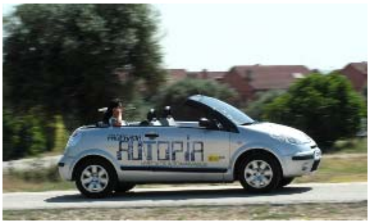


Fig. 1. Citroën C3 Pluriel testbed vehicle

2 Onboard Equipment

Electric power steering basically consists of a torque sensor and motor actuator couple. The sensor is attached to the steering column and measures the torque applied by the driver when he moves the steering wheel. This torque signal is transmitted to a control/power card that sends an amplified proportional power signal to the DC motor, which is engaged to the steering rack bar.

The first step for achieving automatic steering control is to manage the wheels from a computer that we have installed in the car. The method for this automation is to bypass the sensor and control/power card equipment and send a power signal directly to the motor. Our onboard computer runs a fuzzy logic-based control system that generates a control analog signal. An external power drive has been added for supplanting the original C3's power card. It uses as input the analog signal produced by the computer and the output is a power signal that supplies the assist motor.

The vehicle instrumentation is completed with the sensor equipment: a carrier phase differential GPS receiver, which generates to-the-centimeter accurate positioning, and a CAN bus interface.

Now we can move on to describe the control system developed to manage the assist DC motor and, consequently, the steering of the vehicle.

3 Steering Control System

A two-layer fuzzy controller has been defined for steering control. The high-level layer calculates the target position of the steering wheel to fit the vehicle to the desired route. The low level layer generates the optimum torque that must be exerted by the EPS assist motor to move the steering wheel in a human-like way.

There is also a computational reference trajectory representation represented by the set of the most representative GPS waypoints of the route to be tracked [1].

3.1 Steering Position Controller

Two variables are used as input for the fuzzy steering position control system, namely lateral and angular errors. We define lateral error as the distance between the front of the vehicle and the reference trajectory segment, measured along a line perpendicular to that reference segment. Similarly, angular error is described as the angle formed by the reference segment of the trajectory and the car’s director vector. We define two fuzzy variables, also named *angular error* and *lateral error*, each of which has two linguistic labels, *left* and *right*, that indicate where the vehicle is located with respect to the reference segment. Both variables have one associated membership function for each label, defined by their vertex, as shown in Fig. 2a and Fig. 2b.

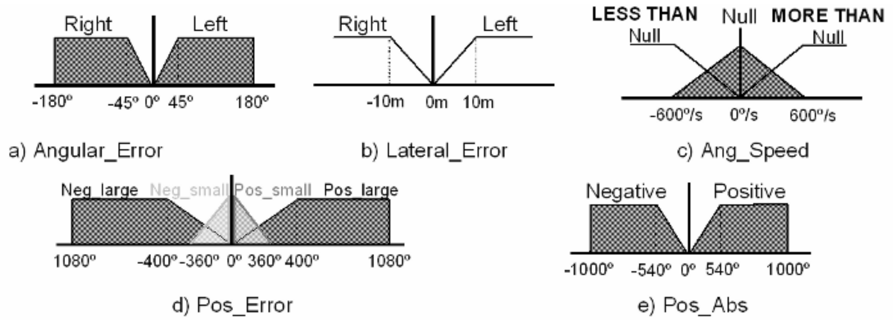


Fig. 2. Input Variables Membership Functions: a) and b) steering position controller; c), d) and e) steering torque controller

The output of the system is the target turning angle that the steering wheel must be moved to correct the trajectory deviation indicated by the input variables. There is only one fuzzy output variable, named *Steering*, with two linguistic labels, called *left* and *right*, whose membership functions are defined by singletons

The rule set for generating the steering turning angle from the input data below is the same for both bend and straight-road driving, as well as for fuzzy control and human driving. The qualitative actions for the human driver (rules) are the same in both cases, and only the quantitative part varies, which is defined in the fuzzy control by the fuzzification of the variables:

R1.1: **IF** *Lateral_Error* Left **THEN** *Steering* Right
R1.2: **IF** *Lateral_Error* Right **THEN** *Steering* Left
R1.3: **IF** *Angular_Error* Left **THEN** *Steering* Right
R1.4: **IF** *Angular_Error* Right **THEN** *Steering* Left

Where the words in italics are the fuzzy variables, the ones to the left of the term **THEN** being input variables and the variables to the right being output variables. The words in normal type are the linguistic labels associated with each one of the fuzzy variables.

3.2 Steering Torque Controller

In this case, three input variables are needed to control the torque applied to the steering wheel. The first is the angular position error of the steering wheel, that is, the difference between the target position generated by the high level fuzzy controller and the real position. The second input variable is the real position of the steering wheel, and the last one is the angular speed at which the steering wheel is turning.

When these variables are fuzzified for use in the fuzzy controller, they are transformed into fuzzy variables called *Ang_Speed* for the angular speed, *Pos_Error* for the angular position error, *Pos_Abs* for the real steering wheel position and, respectively, associated with the membership functions shown in Fig. 2c, d, and e.

The output of the fuzzy controller indicates the voltage that must be sent to the motor power card that applies a proportional amperage to the motor to move the steering wheel with the optimum torque to correctly achieve its target position. Two linguistic labels have been defined, *Positive* (right) and *Negative* (left), whose membership functions have been defined as singletons.

The definition of the rules accounts for the interaction between the input and output variables that will generate the optimum controller behavior. In this case, we have defined six rules for controlling the applied torque.

R2.1: **IF** *Pos_Error* Pos_Large **THEN** *Torque* Positive
R2.2: **IF** *Pos_Vol* Neg_Large **THEN** *Torque* Negative
R2.3: **IF** *Pos_Abs* Negative **AND** *Pos_Error* Neg_Small **THEN** *Torqu*e Negative
R2.4: **IF** *Pos_Abs* Positive **AND** *Pos_Error* Pos_Small **THEN** *Torque* Positive
R2.5: **IF** *Ang_Speed* **MORE THAN** Null **THEN** *Torque* Positive
R2.6: **IF** *Ang_Speed* **LESS THAN** Null **THEN** *Torque* Negative

4 Experiments

Having installed the described controller in the instrumented testbed car, we ran some automatic steering control experiments, one of which is shown in Fig. 3. In this figure, the black dotted line represents the reference trajectory and the gray line is the automatic vehicle route. This starts at the coordinates 459028.75m North 4462552.09m East, behind the starting point label, and is composed of eight turns, four to the left and four to the right, separated by straight segments

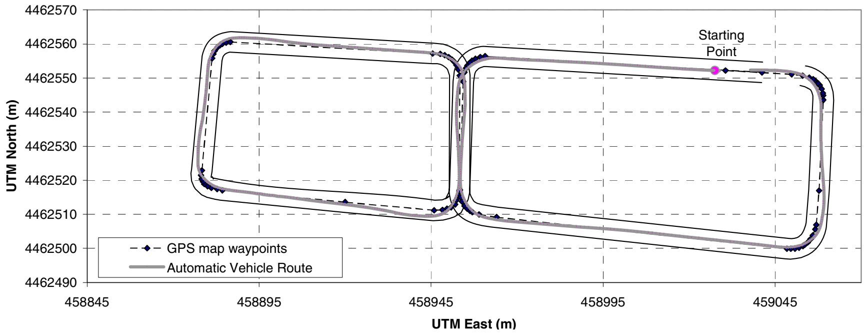


Fig. 3. Automatic route trace

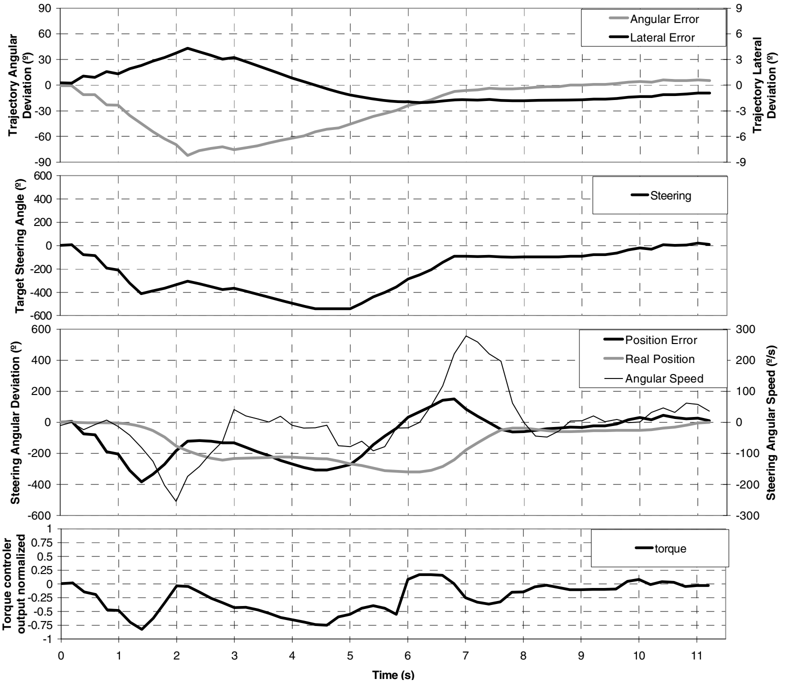


Fig. 4. Detail of the control input and output variables for the first turning to the left of the automatic tracking experiment

The controllers use the input vehicle trajectory variables and, from this information, calculate the necessary torque to move the steering wheel. Fig. 4 includes a trace of controller behavior while taking the first bend to the left to show how the system works. The top graph shows the input variable values for taking the bend. The next graph plots the output of the steering position fuzzy controller. The third graph contains the values of the input variables of the torque fuzzy controller. Finally, the bottom graph shows the output torque, normalized from -1 to 1, to be applied to the EPS motor controlling the steering wheel.

At the beginning, the car is driving centered along the first reference segment of the route. Then, a series of new points are loaded, and the lateral error increases to the left and the angular error augments to the right. This is normal behavior since the new reference segments tend to be perpendicular to the first segment, because the angle between the consecutive streets is about 90º. These input values are fuzzified, and the rule inference of the high-level controller is executed, generating a left turning command that is illustrated by the steering output variable.

The output of the low-level controller (torque) shows that the maximum effort is applied at the beginning of the turning, when a peak is needed to initiate the steering movement (1-2 sec). Once the movement is under way, the torque decreases rapidly. Finally, the controller maintains the steering position and moves the steering wheel back to the center when the turning has finished.

5 Conclusions

In this paper, we have presented a two-layer fuzzy controller for automatic electric power steering control, which we have used to run a number of automatic driving experiments discussed in the last section. These results showed that electromechanical systems, like an EPS, can be managed in a human-like way using artificial intelligence techniques, in this case, fuzzy logic. This method allows the user to mimic human behavior by extracting knowledge from experts, in this case, drivers. An additional advantage of fuzzy logic is that complex nonlinear vehicle models do not need to be developed.

References

1. R. García et al., "Frontal and Lateral Control for Unmanned Vehicles in Urban Tracks", IEEE Intelligent Vehicles Symposium, Versailles, France, 2002.
2. JE. Naranjo et. al., "Adaptive Fuzzy Control for Inter-Vehicle Gap Keeping", IEEE Trans. ITS, Special Issue on ACC, Volume 4, No. 3, September 2003, pp. 132-142.
3. J.E. Naranjo et al., "A Throttle & Brake Fuzzy Controller: Towards the Automatic Car", LNCS 2809, Springer-Verlag, July 2003, pp 291-301.
4. A: Broggi et al., "The ARGO Autonomous Vehicles Vision and Control Systems", International Journal of Int. Cont. and Syst., Vol. 3, No. 4, pp. 409-441, 1999.
5. J.E. Naranjo et al., "Overtaking Maneuver Experiments with Autonomous Vehicles", Proc. of the ICAR 2003, Coimbra, Portugal, pp. 1699-1703, June 2003.
6. T. Endo et al., "Development of reverse parking assist with automatic steering", ITS 2003 Congress Proceedings, Madrid, November 2003.

Computer Vision Application: Real Time Smart Traffic Light

Ángel Serrano, Cristina Conde, Licesio J. Rodríguez-Aragón,
Raquel Montes, and Enrique Cabello

Face Recognition and Artificial Vision Group, Universidad Rey Juan Carlos,
C/ Tulipán, s/n Móstoles E-28933 (Madrid) Spain
angel.serrano@urjc.es
http://frav.escet.urjc.es/

Abstract. The design, development, construction and testing of an Artificial-Vision controlled Traffic-Light prototype has been carried out to rule and regulate intersections. Methods, algorithms and automatons have been built up with that purpose to provide the analysis of images and decisions making at real time. The aim has been the development of an intelligent traffic-light capable of capturing the presence or absence of vehicles, pedestrians and their particular situations defined by their trajectories. Besides the above mentioned properties we have to point out the adaptation to the precise characteristics of each crossing, as its geometry, the required equipment, etc. The project has been supervised by RACE, world wide known as experts in road safety awareness, endowing the prototype with reliability and trust.

1 Introduction

A vast number of reports and statistics state the vulnerable role played by pedestrians in traffic accidents, especially in those who take place in surroundings considered safe by them. Walking is a healthy exercise with almost non-existing negative consequences except for those caused by road traffic. Walking under those circumstances is approximately ten times more dangerous than travelling as a passenger by car [1].

The availability of a wide database of accident causes is considered as one of the most important building blocks in the strategy for the development of intelligent integrated road safety systems [2]. For example, 15% of the total number of people killed on European roads are pedestrians, and 28% are vulnerable road users [3]. It is stated that most accidents take place in urban areas where serious or fatal injuries can be produced at relatively low speeds, particularly in the case of children [4].

Intersections are considered as a especially challenging problem for collision mitigation. UK statistics [5] indicate that 61% of personal injury accidents happen within 20 meters of a junction. In the USA, NHTSA [6] claims that 30% of crashes occur at intersections. The German Federal Statistical Office [7] identifies 86,497 accidents involving failure to observe priority or on entering the road. In Spain [8] the percentage of fatalities involving pedestrians represents the 15%. This suggests that feasible technical solutions reducing this type of accidents by 50% would save 6,000 – 7,000 lives per year only in Europe [2].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 525 – 530, 2005.
© Springer-Verlag Berlin Heidelberg 2005

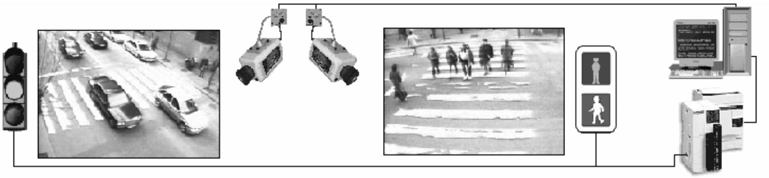


Fig. 1. In our prototype one camera focuses on the pedestrians and the other one on the vehicles. Images are then sent to the computer and as result a final working mode is sent to the automaton responsible for the traffic light switch.

A new trend in expansion is the application of computer vision techniques to traffic, in particular, for the intelligent control of traffic lights. Several factors such as number of pedestrians, situation of the crossing area, number of vehicles, etc., compete for the determination of the adequate colour of the light. This can have a substantial positive effect on the reduction of vehicle-pedestrian conflicts, especially when the system is optimized to meet the users' expectations: waiting time should be reduced to minimum, duration of green light should be adapted to the users' needs. The pre-programmed signal time allowed for pedestrians to cross a street is usually too short for some people, particularly the elderly and disabled ones, and exposes them to the oncoming traffic when the signal changes.

Despite many static cameras are being used in our cities (supermarkets, banks, underground stations, railway stations, etc.), their main commitment keeps being only to help operators make the best decision concerning security or to keep users informed of traffic fluency. Some computer vision applications have already reached the great public, as the on-board systems by Daimler Chrysler [9, 10].

From both perspectives, static and moving cameras, different approaches have been taken to detect the presence of pedestrians, using patterns of motion and appearance [11], using texture analysis and geometric features of pedestrians [12], using speed and path characteristics through a Kalman filter [13], processing background image [14], measuring motion similarity [15], using a Support Vector Machine (SVM) in night vision [16], and using an SVM to make this detection in real-time [17].

The rest of this paper is organized as follows: Section 2 describes the technical setup for our experiments, while Section 3 explains the algorithm used. Section 4 summarizes our main results and conclusions.

2 Technical Setup

The system consists of two cameras situated at only one signal post placed at an intersection. One of them focuses on the pedestrian crossing while the other one focuses on the vehicles arriving grid, as seen in Figure 1. The prototype has been developed as to be carried out by a conventional PC.

Each camera is connected to a Matrox Meteor II capture card with a resolution of 320×240 pixels. The images are alternatively taken as both capture cards share the same data bus. The image acquisition and processing is fast enough to make decisions in real time about the traffic light.

Table 1. Analysis of the different situations that may occur during the system performance. The decision to be taken is also expressed, as the resulting final working mode that will be sent to the automaton.

Vehicles Camera: Vehicles Presence	Pedestrians Camera: Waiting Pedestrians	Pedestrians Camera: Crossing Pedestrians	Automaton Mode
Yes	No	No	3
Yes	Yes	No	1
No	Yes	No	2
Yes	Yes	Yes	1
No	Yes	Yes	2
Yes	No	Yes	2
No	No	Yes	2
No	No	No	1

A TWIDO programmable automaton, by Schneider Electric, is in charge of the traffic light control. The automaton and the PC are synchronized about the working modes for an adequate functioning. These modes vary according to the presence of pedestrians or approaching vehicles, as can be seen in Tables 1 and 2. Mode 1 is a three cyclic states transition, whereas Modes 2 and 3 consist of only one state, so that the weakest part can reach a safe region.

Table 2. Specifications of the three possible working modes of the automaton

Mode 1	Vehicles:	→ Green → Yellow → Red →
	Pedestrians:	→ Red → Red → Green →
Mode 2	Vehicles:	Red
	Pedestrians:	Green
Mode 3	Vehicles:	Green
	Pedestrians:	Red

3 Algorithm Description

As Figure 2 shows, two different types of approaches have been developed. Low level procedures are in charge of the initial treatment of images to obtain the moving objects. High level procedures are in charge of analysing the movement and therefore providing the system with crucial information like position and trajectory, to make the required decision. The system can be easily adapted to different conditions in the intersection, so that it can be used for almost any type of crossing.

The images of vehicles and pedestrians are processed independently, but in a similar way. First of all, in order to track the moving objects, a background subtraction is performed [14, 18], by means of a consecutive set of 10 images organized in a FIFO queue. The mode of every pixel grey level is used for the computation of a continuously updated background image. After the subtraction, only moving objects, which will be hereafter referred to as components, can be identified (see Figure 3, a – c).

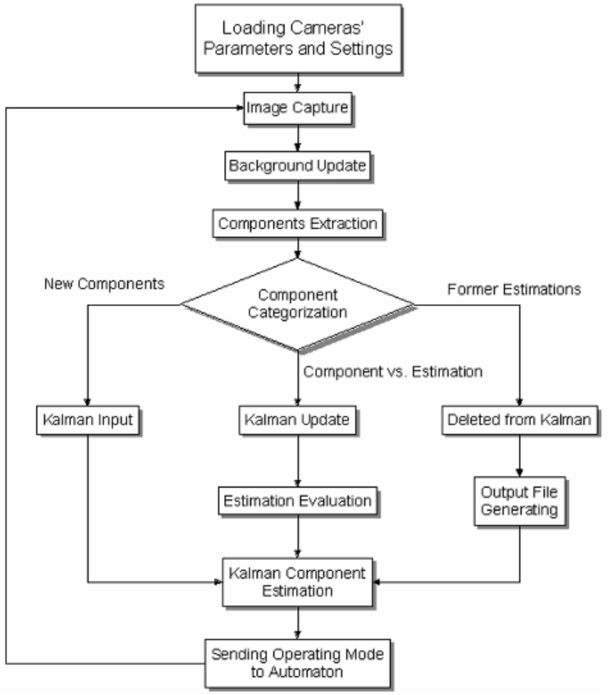


Fig. 2. The diagram shows the different steps that are taken during the algorithm

The calculation of the convex hull of the components has been established through a sequential labelling by checking each pixel in correspondence with its neighbours. A matricial labelling algorithm was put aside due to the high computational cost and the little improvement reached [19, 20]. Those components which do not reach a minimum amount of pixels are ignored, as they are considered as noise. Figure 3 (e – f) shows the subtracted image followed by the components labelled in a grey code image and the filtered result. Then each component is extracted as a separated image.

A Kalman filter is used for tracking moving objects, both vehicles and pedestrians. Initially developed for the prediction of random signals [21], it is able to estimate quantities as a function of time. An extended Kalman filter has been implemented [18, 22]. The algorithm adapts its model at each step to improve the movement estimation of the component. Position, velocity and values of the acceleration at different steps are used to predict the future object location. This allows us to track both vehicles and pedestrians in the scene. The Kalman filter can be adapted by means of a set of parameters that characterize the crossing area.

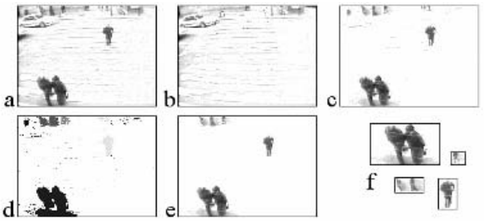


Fig 3. a. Source image taken at the pedestrian crossing. **b.** Background image computed with the mode of consecutive frames. **c.** Subtracted image with no background. **d.** Convex hull of every component labelled in a grey code. **e.** Final filtered image, without spurious components. **f.** Individual components to be tracked through the Kalman filter.

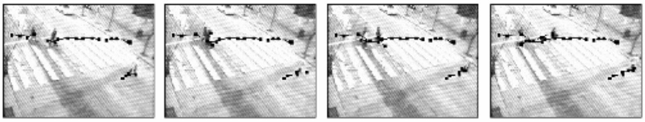


Fig. 4. Trajectories through time of three components. Consciousness and analysis of this trajectories in combination with the establishment of critical zones, let the system make the satisfactory decision in the traffic light control.

4 Results and Conclusions

Initial tests were developed in laboratory controlled conditions where light changes were not abrupt, movement was smooth and the number of objects was not very large.

Our algorithm was tested at the facilities of the Royal Automobile Club of Spain (RACE), which include an Educational Safety School at the Jarama Racecourse in Madrid. We have found that our system works well under real conditions, without influencing the presence of cameras in the behaviour of pedestrians and vehicles. After multiple tests done under RACE supervision, we can fairly state that the computer vision techniques used in this prototype are reliable and dependable to be introduced in our current daily life in roads and pedestrians crossings. Our system is always prepared, even in case of doubt, to make the choice for the weakest participant.

The prototype can also be modified to collect data on pedestrians' conducts, preferred crossing points or usual crossing paths. Behaviour of vehicles in the grid can be analysed, and an option to control density of traffic has been considered.

Our present aims focus on both, preparing the system to perform in an optimal way under extreme weather conditions and integrating several traffic lights and crossings under the control of the prototype.

We would like to thank the Royal Automobile Club of Spain (RACE) its support and sponsorship in the development of this project, showing that new technologies as computer vision have their share in the improvement of road safety.

References

1. Safety Department of Real Automóvil Club de España: RACE's Survey on Mobility in Crowded Towns: A better place to live in. Press Report, Madrid (2000). In Spanish.
2. eSafety: Final Report of the eSafety Working Group on Road Saefty. European Commision, Final Report (November 2002).
3. CARE, Community Road Accident Database. European Commission, Transport Policies. http://europa.eu.int/comm/transport/care/
4. Directive of the European Parliament and of the Council relating to the protection of Pedestrians and other vulnerable road users in the event of a collision with a motor vehicle and amending Directive 70/156/EEC, Brussels (February 2003).
5. Department for Transport, Road Accidents Statistics Branch: Personal Injury Road Accidents: Great Britain 1998. London. http://www.dft.gov.uk/
6. National Center for Statistics and Analysis of the National Highway Traffic Safety Administration.United States Department of Transportation. http://www-nrd.nhtsa.dot.gov/departments/nrd-30/ncsa/index.html
7. Statistisches Bundesamt Deutschland. Statistischer Informationsservice, Wiesbaden, Deutschland. http://www.destatis.de
8. Ministerio del Interior of Spain: Statistical Annual Report 2002. DGT. In Spanish. http://www.dgt.es/boletin/boletin02.html
9. Franke, U., Gavrilla, D. M., Görzig, S. et al.: Autonomus Driving Approaches Downtown. IEEE Intelligent Systems, vol. 13, no. 6, 40-48 (1999).
10. Gavrila, D. M.: Sensor-Based Pedestrian Protection. IEEE Intelligent Systems, vol. 16, no. 6, 77-81 (2001).
11. Viola, P., Jones, M. J., Snow, D.: Detecting Pedestrians Using Patterns of Motion and Appearance. Proc. of the 9th IEEE International Conference on Computer Vision (ICCV 2003), vol. 2, 734-741 (2003).
12. Curio, C., Edelbrunner, J., Kalinke, T. et al.: Walking Pedestrian Recognition. IEEE Trans. Intelligent Transportation Systems, vol. 1, no. 3, 155-163 (2000).
13. Bodor, R., Jackson, B., Papanikolopoulos, N.: Vision-Based Human Tracking and Activity Recognition. Proc. of the 11th Mediterranean Conf. on Control and Automation (2003).
14. Papanikolopoulos, N.: Pedestrian Control at Intersections. Intelligent transportation Systems Institute, University of Minnesota (2000).
15. Efros, A.A., Berg, A.C., Mori, G. et al.: Recognizing Action at a Distance. Proc. of the 9th IEEE International Conference on Computer Vision (ICCV 2003), vol. 2, 726-733, (2003).
16. Xu, F., Fujimura, K.: Pedestrian Detection and Tracking with Night Vision. Proc. IEEE Intelligent Vehicle Symposium, Versailles, France (2002).
17. Kang, S., Byun, H., Lee, S.: Real-Time Pedestrian Detection using Support Vector Machines. Int. J. of Pattern Recognition and Artificial Intelligence, vol 17, no. 3, 405-416, (2003).
18. Obolensky, N., Erdogmus, D., Principe, J. C.: An Time-Varing Kalman Filter to Moving Target Tracking. Proc. of CONTROLO'02, 418-422 (2002).
19. Jain, R., Kasturi, R., Schunck, B. G.: Machine Vision, McGraw-Hill (1995).
20. Gonzalez, R. C., Woods, R. E.: Digital Image Processing, Addison-Wesley (1993).
21. Kalman, R. E.: A New Approach to Linear Filtering and Prediction Problems. Trans. of the ASME-Journal of Basic Engineering, vol. 82, Series D, 35-45 (1960).
22. Hargrave, P. J.: A tutorial Introduction to Kalman Filtering. IEEE Colloquium on Kalman Filters: Introduction, Applications and Future Developments, Digest no. 27 (1989).

Permanency Memories in Scene Depth Analysis

Miguel A. Fernández[1], José M. López-Valles[2], Antonio Fernández-Caballero[1],
María T. López[1], José Mira[3], and Ana E. Delgado[3]

[1] Universidad de Castilla-La Mancha,
Escuela Politécnica Superior de Albacete, 02071 - Albacete, Spain
{miki, caballer, mlopez}@info-ab.uclm.es
[2] Universidad de Castilla-La Mancha,
Escuela Universitaria Politécnica de Cuenca, 13071 - Cuenca, Spain
JoseMaria.Lopez@uclm.es
[3] Universidad Nacional de Educación a Distancia,
E.T.S.I. Informática, 28040 - Madrid, Spain
{jmira, adelgado}@dia.uned.es

Abstract. There are several strategies of how to retrieve depth information from a sequence of images, like depth from motion, depth from shading and depth from stereopsis. In this paper, we introduce a new method to retrieve depth based on motion and stereopsis. A motion detection representation helps establishing further correspondences between different motion information. This representation bases in the permanency memories mechanism, where jumps of pixels between grey level bands are computed in a matrix of charge accumulators. For each frame of a video stereovision sequence, the method fixes the right permanency stereo memory, and displaces the left permanency stereo memory by pixel on the epipolar restriction basis over the right one, in order to analyze the disparities of the motion trails calculated. By means of this functionality, for all possible displacements of one permanency memory over the other, the correspondences between motion trails are checked, and the disparities are assigned, providing a way to analyze the depths of the objects present in the scene.

1 Stereovision-Based Depth Analysis

In general, there are several strategies of how to retrieve depth information from a sequence of images, like depth from motion, depth from shading and depth from stereopsis. In this paper, we introduce a new method to retrieve depth based on motion and stereopsis. In a conventional stereoscopic approach, usually two cameras are mounted with a horizontal distance between them. Consequently, objects displaced in depth from the fixation point are projected onto image regions, which are shifted with respect to the image center. The horizontal component of this displacement can be used to determine the depth of the object. Due to the geometry of the optic system, and considering the epipolar constraint, it is thereby sufficient to restrict disparity analysis to the projection of corresponding linear segments in the left and right camera. In some approaches, the disparity

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 531–536, 2005.
© Springer-Verlag Berlin Heidelberg 2005

is computed by searching the maximum of the cross correlation between image windows along the epipolar lines of the left and right image [1]. Similarly, this can be done by trying to match discernible image features.

So far, many algorithms have been developed to analyze the depth in a scene. Brown et al. [2] describe a good approximation to all of them in their survey article. In many previous works, a series of restrictions are used to approach the correspondence problem. The most usual restriction is the disparity restriction, which considers that is not probable that there exist objects very close to the camera. The scene uses to be limited to a medium distance. This way, too high disparities are eliminated [3]. Koenderink and van Doorn [4] expressed the necessary theory in best initial works related to disparity restriction, and Wildes [5] implemented some of their ideas [6]. More recently, disparity in stereoscopy continues showing its great interest (e.g., [7], [8]).

All these developments approach the depth analysis by different methods; but most of them have as a common denominator that they work with static images and not with motion information. In this paper, we have chosen as an alternative not to use direct information from the image, but rather the one derived from motion analysis. The system proposed uses as input the motion information of the objects present in the stereo scene, and uses this information to perform a depth analysis of the scene.

2 Motion Detection from Permanency Memories

The input to our system is a pair of stereo image sequences. The sequences have been acquired by means of two cameras arranged in a parallel configuration. The central idea behind this approach is to transpose the spatially defined problem of disparity estimation into the temporal domain and to compute the disparity simultaneously with the incoming data flow. This can be achieved realizing that in a well-calibrated fronto-parallel camera arrangement the epipolar lines are horizontal and thereby identical to the camera scan-lines. Our team has already tested the motion analysis algorithm used in this work in monocular video sequences ([9],[10],[11]).

In this case, motion analysis performs separately on both stereovision sequences in two phases. The first analysis phase is based in grouping neighboring pixels that have similar grey levels in closed and connected regions in an image frame. The method used is segmentation in grey level bands. This method consists in reducing the resolution of illumination levels of the image, obtaining this way a lower number of image regions, which potentially belong to a single object in motion. The second phase has to detect possible motions of the segmented regions through the variation of the grey level band of the pixels.

After motion detection, we now introduce a representation that may help to establish further correspondences between different motion information. This representation finds its basis in the permanency effect. This effect considers the jumps of pixels between bands, and it consists of a matrix of charge accumulators. The matrix is composed of as many units in horizontal and vertical direction as pixels there are in an image frame. Initially all accumulators are empty; that is to say, their charge is zero. When a jump between grey level bands occurs at a pixel, the charge unit (accumulator) of the permanency memory at the pixel's position is completely charged. After the complete charge, each unit of the permanency memory goes decrementing with time (in a frame-by-frame basis) down to reaching the minimum charge value, while no motion is detected, or it is completely recharged, if motion is detected again. Fig. 1 shows all these issues. Fig. 1a and Fig. 1b show two images of a monocular sequence. The advance of a car may be appreciated, as well as a more slight movement of a pedestrian. In Fig. 1c you may observe the effect of these moving objects on the permanence memory.

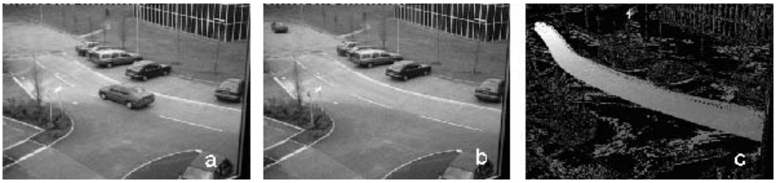


Fig. 1. Permanency effect: (a) one image of a sequence, (b) same perspective after some seconds, (c) motion trails as represented on the permanence memory

The difference between a quick object as the car, which is leaving a very long motion trail (from dark grey to white), and a pedestrian whose velocity is clearly slower and whose motion trail is nearly unappreciable with respect to the cars one, is presented. Thus, permanency memories enable representing the motion history of the frames that form the image sequence, that is to say, there is segmentation from the motion of the objects present in the scene.

3 Disparity Analysis from Permanency Memories

Now, motion-based segmentation facilitates the correspondence analysis. Indeed, motion trails obtained through the permanency memories charge units are used to analyze the disparity between the objects in the stereo pair. The set of all disparities between two images of a stereo pair is denominated the disparity map. The key idea is that a moving object causes two identical trails to appear in epipolar lines of the permanency stereo memories. The only difference relies in their relative positions, affected by the disparity of the object at each moment.

Looking at Fig. 2 it is possible to analyze the motion of each one of the three objects present in the permanency memories from their motion trails. This initial analysis is independent of the epipolar constraint studied. You may observe that object "a", which has a long trail and has his maximum charge towards the left, is advancing to the left at a high speed. Object "b", with a shorter trail, is also advancing towards the same direction but at a slower velocity. Finally, object "c", whose trail is inverted in horizontal, is moving to the right at a medium velocity, as shown by its trail.

Also from Fig. 2, but now comparing between the motion trails in both epipolar lines, disparity is analyzed. Motion trail of object "b" presents a null disparity. Therefore, we can conclude that this trail corresponds to an object that is far away from the cameras. Remember that due to our parallel cameras configuration, pixels with a null disparity are located in the infinite. Object "a" has a little greater disparity. Finally, object "c" offers the greatest disparity.

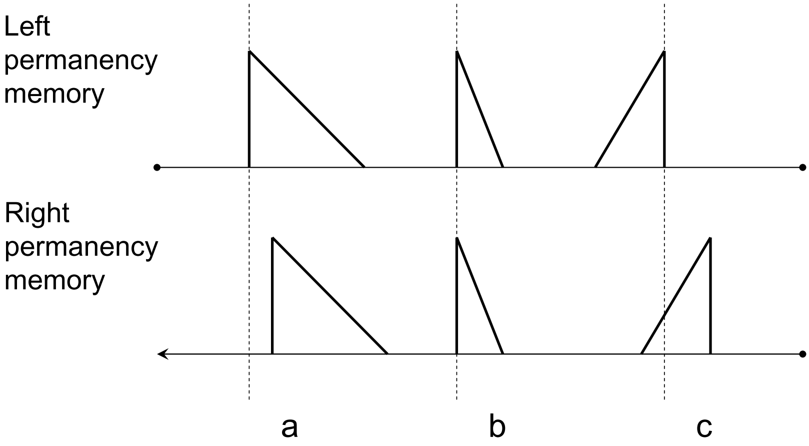


Fig. 2. Disparity of permanency memories

The generalization to global analysis on complete stereo images consists in totally superimposing the two permanency stereo memories under study, and not only their epipolar lines. One of the memories will be displaced over the other looking for motion trails that coincide in both x and y directions. Once the displacement where the coincidence of pixels of motion trails is maximum in size has been calculated, this value is assigned the disparity value. By means of this functionality, for all possible displacements of one permanency memory over the other, the correspondences between motion trails are checked and the disparities are assigned.

4 Data and Results

In order to test our algorithms, the scenario called "IndoorZoom" downloadad at labvisione.deis.unibo.it/ smattoccia/stereo.htm has been used. Fig. 3 shows the result some of the more representative results of applying our algorithms to the "IndoorZoom" scenario. In column (a) some input images of the right camera are shown, in column (b) the segmentation in grey level bands may be appreciated, in column (c) motion information as represented in the right permanency memory is offered, and in column (d) the final output, that is to say, the scene depth as detected by the cameras, is presented. You may observe on Fig. 3 that clearer

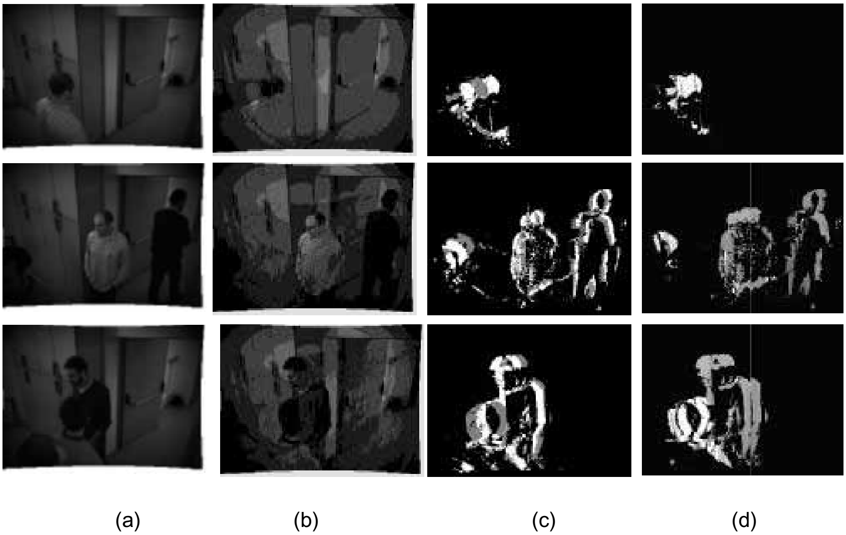


Fig. 3. Results for "IndoorZoom" scenario

colors means that persons are closer to the cameras. Black means there is no motion detected. Consider the case of occlusions, where, nevertheless, the motion trails, hence, the depths, are different, enabling this way to distinguish among different persons.

5 Conclusions

In this paper, we have introduced a new method to retrieve depth based on motion and stereopsis. A motion detection representation helps establishing further correspondences between different motion information. This representation bases in the permanency memories mechanism, where jumps of pixels between grey level bands are computed in a matrix of charge accumulators. Thus, for the purpose to analyze scene depth from stereo images, we have chosen the alternative not to use direct information from the image, but rather the one derived from motion analysis. This alternative provides an important advantage.

Trough motion information it is easier to use correspondences than by grey level information of the frames. The results are also more accurate and robust. This is due to the instantaneous motion features, such as position, velocity, acceleration and direction of the diverse moving objects. Motion information of an object is different from any other moving object's one. Nonetheless, when observing motion features of a concrete object in both stereo sequences at the same time instant, we appreciate that these features are extremely similar. This is the reason why it is easy and robust to establish correspondences between the motion information of an object at the right image respect to the object at the left image. There exist very few ambiguity possibilities.

Acknowledgements

This work is supported in part by the Spanish CICYT TIN2004-07661-C02-01 and TIN2004-07661-C02-02 grants.

References

1. Haralick, R.M., Shapiro, L.G.: Computer and Robot Vision. Addison-Wesley (1992)
2. Brown, M. Z., Burschka, D., Hager, G. D.: Advances in computational stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**:8 (2003)
3. Sumi, Y., Kawai, Y., Yoshimi, T., Tomita, F.: 3D object recognition in cluttered environments by segment-based stereo vision. International Journal of Computer Vision **46**:1 (2002) 5–23
4. Koenderink, J.A., van Doorn, A.J.: Geometry of binocular vision and a model for stereopsis. Biological Cibernetics **21** (1976) 29–35
5. Wildes, R.P.: Direct recovery of three-dimensional scene geometry from binocular stereo disparity. IEEE Transactions on Pattern Analisis and Machine Intelligence **13**:8 (1991) 761–774
6. Wilson, R., Knutsson, H.: A multiresolution stereopsis algorithm based on Gabor representation. Proceedings of the IEE International Conference on Image Processing and Applications (1989) 19–22
7. Mhlmann, K., Maier, D., Hesser, J., Mnner, R.: Calculating dense disparity maps from color stereo images, an efficient implementation. International Journal of Computer Vision **47**:1-3 (2002) 79–88
8. Gutirrez, S., Marroqun, J.L.: Robust approach for disparity estimation in stereo vision. Image and Vision Computing **22**:3 (2004) 183–195
9. Fernández-Caballero, A., Fernández, M.A., Mira, J., Delgado, A.E.: Spatio-temporal shape building from image sequences using lateral interaction in accumulative computation. Pattern Recognition **36**:5 (2003) 1131–1142
10. Fernández, M.A., Fernández-Caballero, A., López, M.T., Mira, J.: Lenght-speed ratio (LSR) as a characteristic for moving elements real-time classification. Real-Time Imaging **9** (2003) 49–59
11. Fernández-Caballero, A., Mira, J., Delgado, A.E., Fernández, M.A.: Lateral interaction in accumulative computation: A model for motion detection. Neurocomputing **50** (2003) 341–364

Pedestrian Detection for Intelligent Vehicles Based on Active Contour Models and Stereo Vision

C. Hilario, J. M. Collado, J. Ma Armingol, and A. de la Escalera

Intelligent Systems Lab. Universidad Carlos III de Madrid. Leganes, Madrid, 28911. Spain
`{chilario, jcollado, armingol, escalera}@ing.uc3m.es`

Abstract. Recently, the focus of safety systems for intelligent vehicles has been on researching and developing Advanced Driver Assistance Systems (ADAS). Most efforts have been concentrated at the driver, not taking into account the protection of the most vulnerable road users. This paper describes a pedestrian detection algorithm based on stereo vision. The use of visual information is a promising approach to cope with the different appearances of pedestrians and changes of illumination in cluttered environments. Active contour models are used to detect and track people from the images taken by an on-board vision system, performing contour extraction in sequential frames.

1 Introduction

1.1 Motivation

Over the past 20 years, the high rate of road-accidents all over the world has motivated the development of intelligent vehicles. The researchers community, the automotive industry and several organizations, have been actively involved in improving road safety through the development of ADAS[1]. However, work has been focussed on the driver, whilst the protection of pedestrians has been relegated [2].

Projects that have dealt with this case are quite recent, as it has been pointed out at the Fifth Framework Programme [3]. A possible reason for it could be the fact that detecting pedestrians with an artificial system is a difficult task. The main challenges are the high degree of variability of the human appearance, the cluttered backgrounds and the changing lighting conditions. Moreover, the applications to protect pedestrians define hard real time requirements. An open issue is which sensors are best to address this complexity. Distance sensors, like radar or laser, have the advantage of giving a direct distance measurement. Among the disadvantages stand out their lower resolution and their tendency to interfere each other if they are in closeness. On the other hand, computer vision gives a richer description of the environment, although the information is more difficult to process. Even if other sensors as lasers or radars can detect pedestrians, vision is the unique that can comprehend their motion and predict their movements. For the reason that diverse sensors could be complementary, some approaches have decided to integrate them.

The methods to detect pedestrians based on computer vision can be classified in three main groups. Those that try to find simple features that define a person are at the

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 537 – 542, 2005.
© Springer-Verlag Berlin Heidelberg 2005

lowest level. Their main drawback is that if one of those features is not enough present in the image, the pedestrian is lost. Besides, they are prone to false tracks. On the other hand, there are methods that include some kind of learning. Generally, they are based on neural networks. That type of methods require a lot of time to be trained. Model-based approaches, take advantage of the two previous. Usually, a model of the person is built, so they are more robust than feature based algorithms, but slightly slower.

1.2 Previous Work

Papageorgious and Poggio [4] presented a pedestrian detection system based on wavelet analysis and Support Vector Machines. However, the system was computationally expensive as it had to scan the whole image at multiscales. Gavrila and Philomin [5] developed a real time pedestrian detection algorithm based on distance transforms. This method performs a coarse-to-fine template matching. But the template hierarchy cannot capture the variety of human shapes. Zhao and Thorpe [6] developed a robust algorithm for detecting pedestrians in cluttered scenes through stereo-based segmentation and neural network-based recognition. Broggi *et al.*[7] also used stereo vision, combining it with a verification technique based on symmetry properties. Both systems got deceived by objects similar to humans. Recently there has been an increasing interest in using infra-red sensors [8]. Although they can detect pedestrians by the heat their bodies emit, pedestrians are not the only sources of heat in a traffic environment.

2 The Pedestrian Detection Module

Active contour models or *"snakes"* were proposed by Kass *et al* [9] in 1988 as a segmentation scheme. Its ability to extract contours, even in presence of gaps or occlusions, together with its dynamic behavior, makes this approach adequate for the detection and tracking of non-rigid objects. The main drawback is their high sensibility to the initial position. In order to overcome this limitation, a stereo module is integrated to guide the location of active contours.

2.1 Active Contour Models Initialization

The motivation for using stereo vision is manifold. When dealing with images taken by a non-static camera, most of the segmentation techniques used for non-moving camera fail due to the movement of the camera. Among the advantages of using stereo vision, it allows occlusion analysis, is robust to illumination changes and can detect both moving and motionless objects.

In the system developed, stereo vision is used to generate a disparity map of the scene (Fig. 1-d). As the pedestrians can appear in the scene at very diverse distances, the use of range-information allows filtering the images based on distance measures. Therefore, regions that are not at the desired distance are eliminated (Fig. 1-c), performing subsequent calculations only on the filtered areas. Hence, two advantages are obtained; On one hand, the algorithm is less time consuming. On the other hand, the task of initializing the snakes is eased because only the filtered area is considered. Since regions with high vertical symmetry are potential candidates for an active contour initialization, vertical symmetries are looked for. With that aim, the vertical gradient component of the filtered image is found and only pixels with high response are taken. Then, pairs of pixels on the same line vote for central pixels as their symmetry axis. An active contour is initialized in a symmetry axis, if the number of pixels that vote for that axis is over a given threshold (Fig. 1-f).

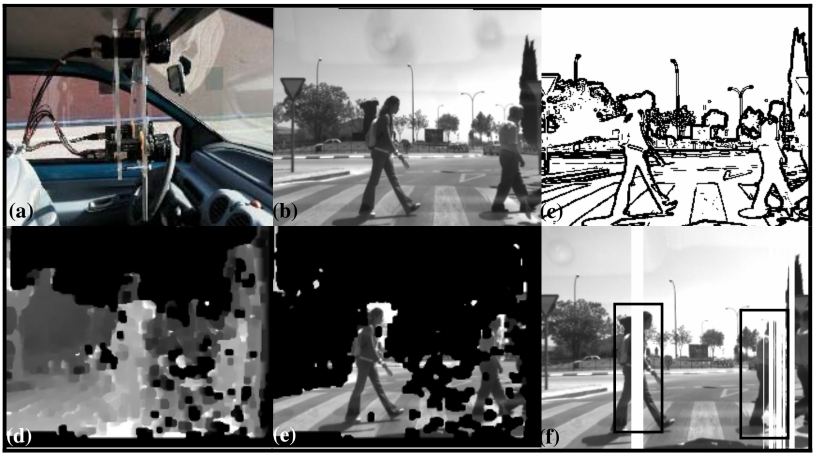


Fig. 1. (a) A detail of the stereo system. (b) A right image taken by the stereo camera. (c) Gradient image. (d) Disparity map (e) and the filtered image based on distance measures. (f) Both vertical symmetries and snakes initialization region enclosed by bounding boxes are shown.

2.2 Active Contour Model

Active contour models are proposed as energy minimization splines that, from an initial position, are deformed by external and internal forces, until they reach an equilibrium state. The major reason for their success is the possibility to integrate physical and topological knowledge into the segmentation process. Our approach follows the explicit contour representation proposed by Kass *et al.*[9], because it allows an efficient processing and its physical properties can be controlled in a very intuitive manner. In their seminal paper, Kass *et al* model a contour as a rubber band under the influence of image forces and elastic forces. Image forces are due to external energies associated to a potential field that attracts the snake. On the other hand, elastic forces counteract strong expansion and bending of the deformable model. They represent the internal energy of the contour as a weighted combination of membrane and thin plate energy. It is used to regularize the contour and hence to avoid strange effects. The evolution of the contour is governed by the minimization of both internal and external energies.

The internal energies used in this proposal extend the ones used by Williams and Shah [10]. Their formulae maintain the points in the snake more evenly spaced than Kass, so the natural tendency to shrink of the snakes is mitigated.

$$E^{*}_{\text{int}}\left(v(s)\right) = \alpha(s)\left(\overline{dist} - \left|\frac{dv}{ds}\right|^{2}\right) + \beta(s)\left|\frac{d^{2}v}{ds^{2}}\right|^{2} . \qquad (1)$$

In order to avoid shrinkage, a new internal force is included to control the shape of the deformable model. This regularizing force prevents the shrinking effect of the snake, as it is based on higher degrees of smoothness than the membrane and the thin-plate energy terms, which are based on the first and second derivatives respectively.

$$E_{\text{int}}\left(v(s)\right) = E^{*}_{\text{int}}\left(v(s)\right) + \theta(s)\left|\frac{d^{4}v}{ds^{4}}\right|^{2}. \tag{2}$$

This term is based on the fourth derivatives along the contour and it looks for segments presenting no change on its center of curvature, and therefore are prone to correspond to head and feet areas of a pedestrian. Once those segments are localized, the amount of stretching and bending of them is modified locally.

While the classic active contour model is non-adaptive with respect to the underlying image data, in this algorithm the elasticity and bending properties of the contour are related to the underlying image structure. Firstly, curvature of the model is calculated and depending on its value, the elasticity and bending weights are modified. In general, bending of the snake is not too much allowed. Next, for those segments in the curve that present a slight curvature, the new energy term is calculated. Therefore, the snake is constrained to deform in a particular way.

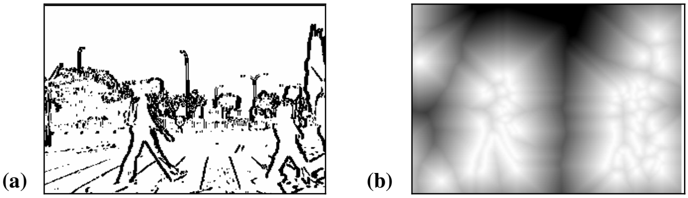


Fig. 2. (a) Vertical gradient and (b) its distance map

For the external forces, a new potential field which smoothly extends over a long distance is defined. So, the snake is affected not only by surrounding features. The fact that pedestrians have a strong vertical symmetry is exploded to construct the potential field. The same idea was used to decide where to put an active contour. Therefore, a distance map of the symmetry axes obtained from that stage is constructed. In order to avoid the snake shrinking to the axis, movement is allowed until it reaches a vertical edge. Besides this potential field, the image gradient (Fig. 1-c) and distances to vertical borders (Fig. 2-b) are also considered.

The deformable model proposed extends the greedy algorithm of Williams and Shah as it is a stable, fast and flexible optimization technique. This approach is adequate for non-rigid objects detection and tracking, performing contour extraction in sequential frames. Once the snake is initialized on an object contour in the first frame, it will automatically track the contour from frame to frame. This method requires small deformation and movement of an object between frames.

Some points in the snake are still prone to errors, like getting trapped into the shadow of the pedestrian (Fig. 3-d). Besides, if the external forces are not strong enough, the snake tends to shrink (Fig. 3-b and 3-e). These problems are a side-effect of the representation used. As the model is only evaluated at some discrete points, these have to be uniformly spaced. Otherwise, the elasticity, curvature and concavity terms are inaccurate. A possible solution could be using splines, as the model is evaluated not only at its control points, but also along the contour.

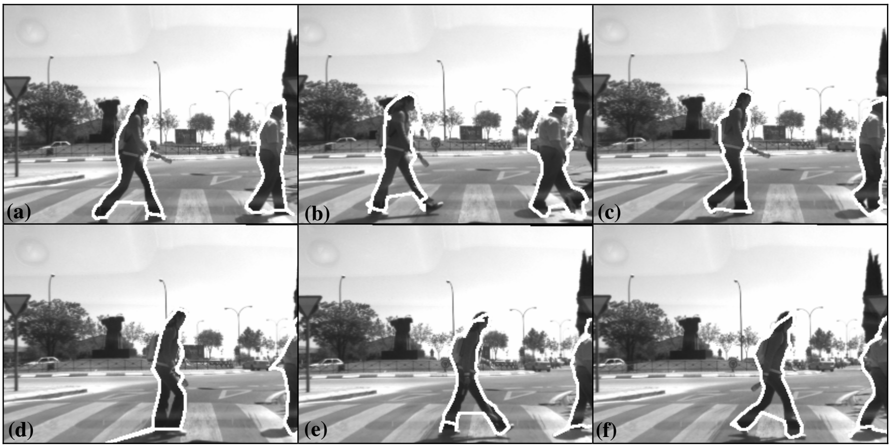


Fig. 3. From (a) to (f) a complete sequence of frames is shown

3 Conclusions and Results

A system based on computer vision for the detection of pedestrians has been presented. It is based on a deformable contour model using a parametric representation. The greedy algorithm is implemented to reach the minimum energy. The evolution of the contour is driven by a new potential based on distances to vertical symmetries and vertical borders. Besides, a regularization term is included in its internal energy, which aims to control the contour shape without producing any shrinkage.

The quality of the segmentation is improved by the information provided by the stereo module. Initial segmentation is performed in the images, filtering them with the data from disparity maps. Regions that are not at the desired distance are eliminated from the images, easing the active contour models initialization and the successive processing.

This algorithm has been tested on images taken by a stereo camera mounted on the IvvI (Intelligent Vehicle based on Visual Information) vehicle (Fig. 1-a), which is an experimentation platform for researching and developing Advance Driver Assistance System based on computer Vision. The pedestrian detection module is part of this ADAS.

Acknowledgements

This work was supported in part by the Spanish government under CICYT grant TRA2004-07441-C03-01.

References

1. McDonald, J., Markham, C.,McLoughlin, S.: Selected problems in automated vehicle guidance. Tech. Report NUIM/SS/2001/05 Signals and Systems Group, National University of Ireland (2001)
2. Gavrila D.M., Kunert M., Lages U.: A multi-sensor approach for the protection of vulnerable traffic participants-the PROTECTOR project. IEEE Instrumentation and Measurement Technology Conference, Vol.3. (2001) 2044-2048
3. Information Society Technologies for Transport and Mobility. Achievements and Ongoing Projects from the Fifth Framework Programme: Office for Official Publications of the European Communities, 2003. <http://europa.eu.int>
4. Papageorgiou, C., Evgeniou, T., Poggio, T.: A trainable pedestrian detection system. Proc. of Intelligent Vehicles (1998) 241-246
5. Gavrila, D.M.: Philomin, V.: Real-time object detection for "smart" vehicles. Proc. of IEEE Intl. Conf. On Computer Vision (1999) 87-93
6. Zhao, L., Thorpe, C.E.: Stereo- and neural network-based pedestrian detection. IEEE Transactions on Intelligent Transportation Systems, Vol. 1. (2000) 148-154
7. Broggi, A., Bertozzi, M., Fascioli, A., Sechi, M.: Shape-based pedestrian detection. IEEE Intelligent Vehicles Symposium (2000) 215-220
8. Meis, U., Oberländer, M., Ritter, W.: Reinforcing the reliability of pedestrian detection in far-infrared sensing. IEEE Intellignet Vehicles Symposium (2004) 779-783
9. Kass, M., Witkin, A., Terzopoulos D.: Snakes: Active Contour Models. Int. J. Comp. Vision, Vol. 1. (1988) 321-331
10. Williams, D.J., Shah, M.: A Fast Algorithm for Active Contours and Curvature Estimation. CVGIP: Image Understanding, Vol. 55. (1992) 14-26

Fast Road Sign Detection Using Hough Transform for Assisted Driving of Road Vehicles

Miguel Ángel García-Garrido, Miguel Ángel Sotelo, and Ernesto Martín-Gorostiza

Department of Electronics, University of Alcalá, Alcalá de Henares, Madrid, Spain
{garrido, michael, ernesto }@depeca.uah.es

Abstract. A system for real-time traffic sign detection is described in this paper. The system uses restricted Hough transform for circumferences in order to detect circular signs, and for straight lines for triangular ones. Some results obtained from a set of real road images captured under both normal and adverse weather conditions are presented as well in order to illustrate the robustness of the detection system. The average processing time is 30 ms per frame, what makes the system a good approach to work in real time conditions.

1 Introduction

Traffic sign detection and recognition has experimented increasing research interest in the last times. This is due to the importance of improving safety in road vehicles. Drivers sometimes miss signs because of distractions or lack of concentration. The two main causes of car-accidents in Spain are speed limit exceeding and distractions in general, so, aid to keeping the speed below the limit and avoiding distraction while driving are the two main targets in this work which is focussed on traffic sign detection for driving assistance. But the driver is kept in the control-loop, thus, our system will alert drivers, but will not actuate in order to control the vehicle. This can be done, for example, using acoustic warning if speed is over the limit, noticing the presence of a sign in a display, or by means of an audio signal (synthesized human voice) indicating the detection of a certain sign. Traffic-sign detection and recognition systems were born at the late 80's, but it has not been until recent times that real time-performing systems have been successfully achieved [1], [2], [3]. The most common method used for traffic signs detection is colour-segmentation. This method is based on the assumed fact that the wavelength coming from a sign, for instance the red-coloured edge of a speed-limit sign, does not change with changes in the intensity and the incidence-angle of the light onto the sign, if HSV or HIS spaces are used [4], [5]. The image of the camera is not, however, completely invariant under changes in the chromaticity of the received light, being these changes due to shades, adverse weather conditions, etc. Other authors use the information of scene-shapes for sign detection. Among these ones, some of them apply a template to an edge-image [1], [3]. The method consists in obtaining the distance-transform from an edge-image and doing a further matching with pre-selected templates, corresponding to those signals searched; these templates are hierarchically organized so that the number of comparisons is reduced, but this method implies nevertheless quite a high computational cost for a

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 543 – 548, 2005.
© Springer-Verlag Berlin Heidelberg 2005

real time system. In other works, Hough transform is used [10], or else a varied version of it [2], [7]. In these works the information about symmetry-properties of the objects is used [8]. Barnes and Zelinsky [2] show that by applying this method the system is fast enough to work in real time, but only with circular signs, being only specifically tested for 40 and 60 km per hour- speed-limit signs, while Loy and Barnes [7] have used a similar technique for triangular, square, and octagonal signs, but not working in real time.

This work can be included with those that use information from the shape of the objects in the scene, in order to reduce the number of possible sign candidates. From the four types of signs existing in the Spanish driving code, prohibition, obligation, warning, and informative, the two first except for the stop one are circular signs. Nevertheless this one can be also considered as circular for detection issues. Sign detection has been performed by using the Hough transform for circumferences [11], but with certain restrictions that make it very efficient for the detection of this kind of signs. For warning signs, which are triangular, Hough transform for straight lines has been chosen, as a triangular sign is formed by three straight lines cutting each other under 60º angles, in pairs. Informative signals are not considered in this work.

Finally, it must be noted that the system presented in this work focuses the search-area only in a certain pre-selected zone of the image, that one holding a higher probability of finding a traffic sign inside.

2 Contours Information

The method used for edge detection is Canny method [9]; this method preserves contours, what is very important for detecting traffic signs using shape information, because they are usually closed contours, as can be seen in figure 1.

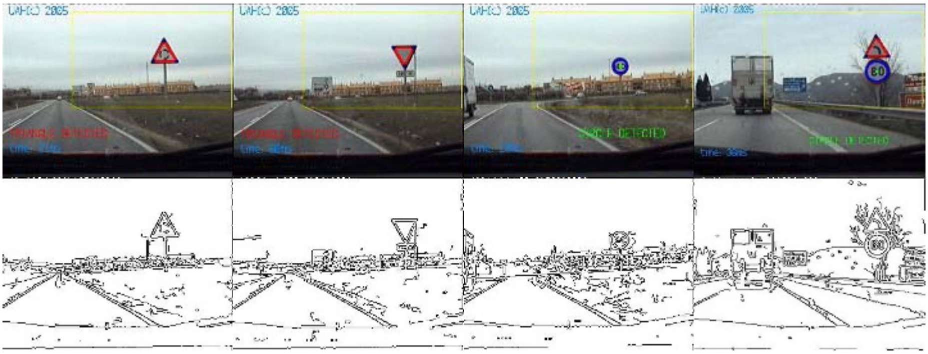


Fig. 1. Real images sequence, detected on the road, with the search area within each image outlined (square window), and Canny image used for contour-information search

Numerous implementations of edge-detection-systems based on Canny's idea have been developed. Canny described a method for generating a robust edge-detector. This method generates thin contours, which also avoids different contours from different objects joining together. Several tests, under different weather and illumination conditions have been taken, trying different threshold levels and being the results very similar in all cases. This shows that the value of Canny threshold levels is not critical for this application.

The contours obtained applying Canny method are codified using the ‘chain code’. The contours are accepted if they are closed contours, or almost closed contours. They must also fulfil a certain aspect-ratio constraint, showing similar width and height. circular traffic signs, including stop one, as well as triangular ones, meet these restrictions with high probability. Hough transform is only applied to accepted contours after filtered with the aforementioned restrictions, so that the computational time is reduced.

3 Hough Transform

The classical Hough algorithm can be easily extended to find any curves in an image that can be expressed analytically in the form $f(x, p) = 0$ [12]. Here, x is a point in the domain of the image and p is a parameter vector.

Hough transform for straight lines is applied in order to detect triangular signs. The aim is detecting three straight lines intersecting each other, forming a 60 degrees-angle. It must be observed that, as long as the number of straight lines intersecting each other might be very large if the Hough transform was applied to the whole image, more than the actual triangles existing in the image would be detected. Using Hough transform neither the beginning nor the end of a straight line is known, as a straight line defined by this transform is expressed in terms of the polar parameters θ and ρ:

$$x \cos(\theta) + y \sin(\theta) = \rho \tag{1}$$

In order to overcome this handicap in this work the strategy is to apply the Hough transform to every contour, one after the other. In this way, only those triangles existing actually in the image are detected, as shown in figure 2, reducing the computational time too.

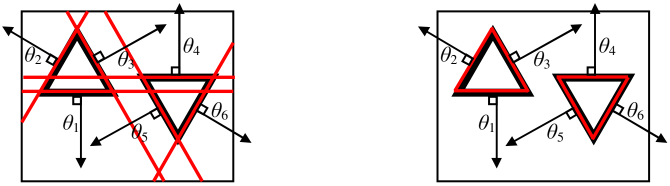


Fig. 2. Straight lines detected using Hough transform, applied to the whole image (left) applied to each contour, one by one (right)

Hough transform for circumferences is applied to detect circular signs, and the stop sign too. A circle in the xy-plane with center (χ, ψ) and radius ρ can be expressed as:

$$f(x, p) = (x - \chi)^2 + (y - \psi)^2 - \rho^2 = 0 \tag{2}$$

Where the parameter space, $p = (\chi, \psi, \rho)$, must be quantized. The accumulator matrix ‘a’ is the representation of the quantized parameter space. For circumference detection the accumulator 'a' will be a three-dimensional matrix with all entries initially set to 0. The entry a (χ_r, ψ_s, ρ_t) is incremented by 1 for every feature point (x_i,y_i) in the image-domain, contained in the circumference with centre (χ_r, ψ_s) and radius ρ_t as expressed in (3) where a precision margin ε for the radius ρ_t is introduced to compensate quantization error when digitizing the image [13]:

$$|(\chi_r - x_i)^2 + (\psi - y_i)^2 - \rho_t^{\,2}| < \varepsilon \tag{3}$$

For circular-objects detection the same criteria are followed as in the case of straight lines. Hough transform is applied contour by contour, so that those contours corresponding to other shapes but not signs do not affect the detection of the latter ones, as shown in figure 3.

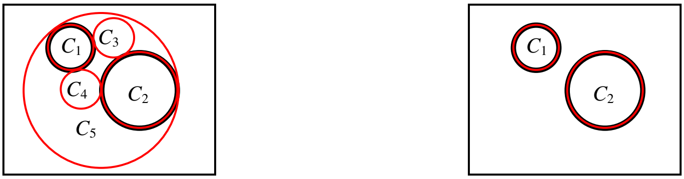


Fig. 3. Circumferences detected using Hough transform for circumferences, applied to the whole image (left) applied to each contour, one by one (right)

One important feature of a circular sign is that its centre and its centroid are in fact the same point. Making use of this property the centre is sought in a search-scope near the centroid. All these considerations make detection time to be very short, making the system able to work at a processing-speed between 5 and 50 frames per second, depending on the number of signs detected.

4 Results

The system works with one only camera mounted on the windscreen of the car, as shown in figure 4. Several tests have been conducted, placing the camera in different positions on the windscreen, and it has been concluded that the placement of the camera is not decisive, but orientation is, thus affecting the quality of detection. The best arrangement is to place the camera pointing towards the same direction and sense of the car so that signs are seen orthogonally to the motion-direction and thus suffering the least possible distortion. Should a circular sign be captured non-orthogonally by the camera, it would be seen as an ellipse in the image and would not be detected. However, Hough transform can be extended to ellipses, but it would be necessary to add two new parameters in the parameter-space with respect to the transform for circumference used in this work. This technique has been tested in fact, and it was noticed that the average processing time was 2 seconds, so the system could not operate in real time. It is important to realize that detecting circumferences but not ellipses is in fact a simplification of the method, but it does not imply a poorer performance at all. On the contrary, an elliptical shape in the image captured, if it happened to correspond to a sign, would be placed with high probability in another road with other direction, for instance in a crossroads. So, only those signs detected as circular are placed in our road in the egomotion direction.

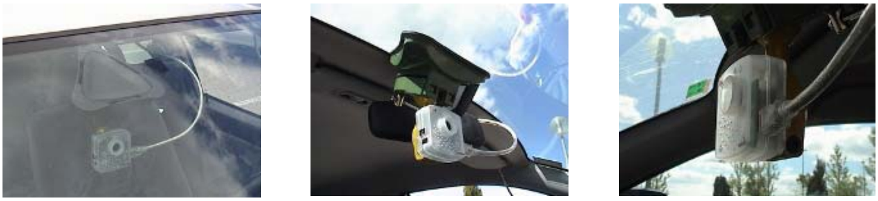


Fig. 4. One camera mounted on the windscreen of the car

The system has been empirically tested under severe adverse weather conditions, as it is depicted in figure 5, and the successful-detection ratio has not been affected. For every test made, the successfully-detected-sign percentage has been, over 99% with an average processing time of 30ms per frame.

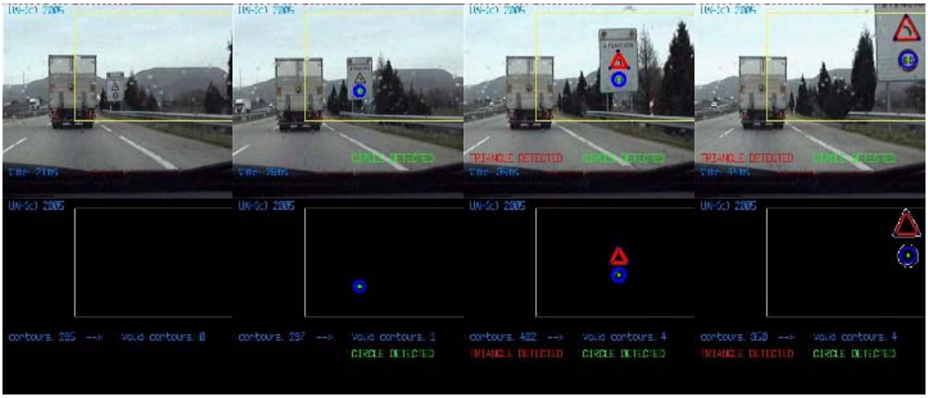

UN-(c) 2005
CIRCLE DETECTED
TRIANGLE DETECTED
contours: 295 ---> valid contours: 0
contours: 297 ---> valid contours: 1
contours: 482 ---> valid contours: 4
contours: 350 ---> valid contours: 4

Fig. 5. Sequence of real road images under adverse weather conditions where speed limit and triangular sign are detected

5 Conclusions

A real time-algorithm for traffic signs detection has been shown. The algorithm is able to detect any kind of signs but the informative ones, using a similar technique for all of them, making the algorithm very robust. Besides, the position of the camera in the car is not critical and it is fast enough so as to work in real time without any problems. Another important feature is that it shows the same good performance under adverse weather conditions, for example in a rainy day.

As future work, a kalman filter to make a continuous and soft tracking of detected signs until they are not present in the field of view will be added. By doing so, continuous detection of the same sign would be avoided. Another approach to be done is to implement the classification or recognition stage, which will be done with a neural network, the most widely used method for this purpose.

Acknowledgments

This work has been funded by Research Projects CICYT DPI2002-04064-05-04 (*Ministerio de Educación y Ciencia*) and FOM2002-002 (*Ministerio de Fomento, Spain*).

References

1. Gavrila, D.M.; Franke, U.; Wohler, C.; Gorzig, S. “Real time vision for intelligent vehicles,” Instrumentation & Measurement Magazine, IEEE Volume 4, Issue 2, June 2001 Page(s): 22-27.
2. Barnes, N.; Zelinsky, A. “Real-time radial symmetry for speed sign detection,” Intelligent Vehicles Symposium, 2004 IEEE, 14-17 June 2004 Page(s):566 – 571.
3. Gavrila, D.M.; Philomin, V. “Real-time object detection for “smart” vehicles,” Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, Volume 1, 20-27 Sept. 1999 Page(s):87 - 93 vol.1.
4. Priese L., Rehrmann V., Schian R., Lakmann R.,“Traffic Sign Recognition Based on Color Image," Evaluation IEEE Intelligent Vehicles Symposium’93, Tokyo, 1993.
5. De Micheli, E.; Prevete, R.; Piccioli, G.; Campani, M., "Color cues for traffic scene analysis," Intelligent Vehicles '95 Symposium., Proceedings of the25-26 Sept. 1995 Page(s):466 – 471.
6. C. Y. Fang, C. S. Fuh, S. W. Chen, and P. S. Yen, "A road sign recognition system based on dynamic visual model" in Proc IEEE Conf. on Computer Vision and Pattern Recognition, vol. 1, 2003, pp. 750-755.
7. Loy, G.; Barnes, N.;"Fast shape-based road sign detection for a driver assistance system," Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, Volume 1, 28 Sept.-2 Oct. 2004 Page(s):70 - 75 vol.1
8. Loy, G.; Zelinsky, A.;"Fast radial symmetry for detecting points of interest," Pattern Analysis and Machine Intelligence, IEEE Transactions on, Volume 25, Issue 8, Aug. 2003 Page(s):959 - 973
9. J. Canny. “A Computional approach to Edge-Detection,” IEEE Transactions on pattern Analysis and Machine Intelligence, vol 8, pp. 679-700, 1986.
10. P. Hough, “Method and means for recognizing complex patterns,” Dec. 18 1962. U.S. Patent 3,069,654.
11. R. Duda and P. Hart, “Use of the Hough transform to detect lines and curves in pictures,” Communications of the ACM, vol. 15, no. 1, pp. 11-15, 1972.
12. D. Ballard, “Generalizing the Hough transform to detect arbitrary shapes,” *Pattern Recognition*, vol. 13, no. 2, pp. 111-122, 1981.
13. S. D. Shapiro, “Properties of transforms for the detection of curves in noisy image,” *Computer Graphics and Image Processing*, vol. 8, pp. 219-236, 1978.

Advances in Robotics

Peter Kopacek

Intelligent Handling and Robotics – IHRT,
Vienna University of Technology, Austria
Favoritenstr. 9-11/3256, A-1040 Vienna, Austria
kopacek@ihrt.tuwien.ac.at

Abstract. The field of robotics is one of the most innovative in the last decade. We are moving now from conventional, unintelligent industrial robots to mobile, intelligent, cooperative robots. This new generation of robots opens a lot of new application fields. Some of them will be growing dramatically in the nearest future. Therefore in this paper the present state will be discussed, selected applications described and an outlook on future developments will be given.

1 Introduction

Conventional industrial robots from the late 70´s are now only a tool on the production level. One of the oldest dreams of the robotic community – intelligent, mobile, cooperative as well as humanoid robots – starts to become reality not only because of the rapid development of "external" sensors driven by micro- and nanotechnology.

External sensors (e.g. visual, auditive, force-torque…) combined with micro drives, embedded systems,… offer intelligent robots the possibility to see, hear, speak, feel, smell like humans. Compared with conventional, unintelligent, industrial robots, intelligent robots fulfil new, innovative tasks in new application areas.

There are three "starting" points for the development of intelligent robots: Conventional, stationary industrial robots; mobile, unintelligent platforms (robots) and walking machines.

Stationary industrial robots equipped with external sensors are used today for assembly and disassembly operations [1], fuelling cars, cleaning of buildings and airplanes, ... and have been the first "intelligent" robots.

Mobile platforms with external sensors are available since some years and cover a broad application field. The core of each robot is an intelligent mobile platform with an on-board PC. On this platform, various devices, like arms, grippers, transportation equipment, etc., can be attached. Communication between the „onboard PC" and the „supervisory PC" is carried out by radio-based networks - communication with the environment can be accomplished by voice, beep or bell.

Walking machines or mechanisms are well known since some decades. Usually they have 4 to 6 legs (multi-ped) and only in some cases 2 legs (biped) – walking on

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 549 – 558, 2005.
© Springer-Verlag Berlin Heidelberg 2005

two legs is from the viewpoint of control engineering a very complex (nonlinear) stability problem. Biped walking machines equipped with external sensors are the basis for “humanoid” robots.

In addition these intelligent robots – especially mobile platforms and humanoid robots - are able to work together on a common task in a cooperative way. The goals are so called “Multi Agent System – MAS”. A MAS consists of a distinct number of robots (agents), equipped with different arms, lifts, tools, gripping devices, ... and a host computer. The MAS has to carry out a whole task e.g. assemble a car. The host computer divides the whole task in a number of different subtasks (e.g. assembly of wheels, windows, brakes, ...) as long as all this subtasks can be carried out by at least one agent. The agents will fulfil their subtasks in a cooperative way until the whole task is solved.

In industry intelligent robots will work together with humans in a cooperative way on a common working place.

2 Application Examples

In the following some examples partially developed with and realized in small and medium-sized enterprises -SME´s will be shortly described and discussed.

2.1 Disassembly Cell for Printed Circuit Boards

The layout of the cell is shown in Fig. 1. In a manual feeding station the Printed Circuits Boards (PCBs) with a maximum size of 300 x 220 mm are attached on special work holding device.

The vision system has several tasks. It has

- to recognize the re-useable parts by means of a data base containing the data (kind, production company, assigned, dimensions),
- to detect the re-useable parts,
- to determine their position, size and the centre of inertia, and
- to classify the useable parts to be desoldered or removed from sockets.

The laser desoldering station consists of a cross table – two linear axes – controlled to reach every point (centre of inertia) on the PCB. The desoldering process is carried out by laser technology. The desoldered parts are put on a distinct area outside the laser from which they are removed by the industrial robot and to put into the appropriate magazines.

The third station is the removal station for socket parts. An industrial robot equipped with special grippers as well as external sensors carries out process. The robot removes these parts and puts them also in the right magazines.

In the heating and removal station the PCB`s were heated by 3 infrared elements until the desoldering temperature for each of the parts is reached. The parts are removed by a simple pneumatic or a controllable two finger gripper and putted in the storage devices.

A prototype of this disassembly cell is now in use since 3 years.

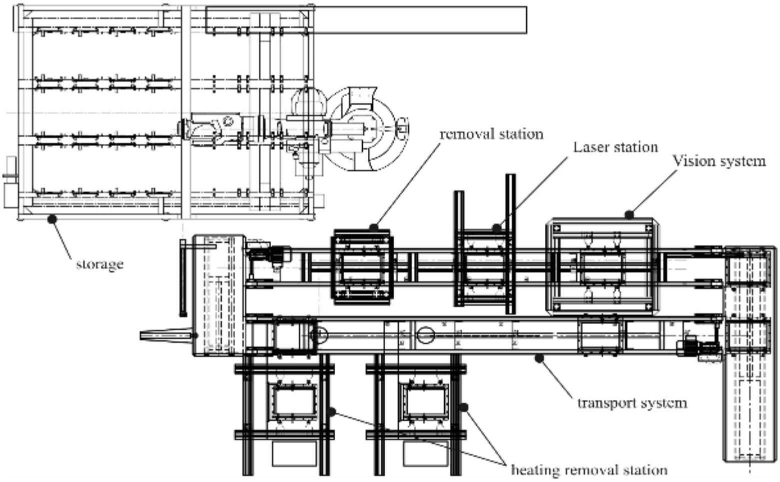


Fig. 1. Layout of the disassembly cell for PCB`s [2]

2.2 A Semiautomatized Disassembly Cell for Mobile Phones [3]

After a detailed analysis of used mobile phones concerning the parts as well as the assembly technology and tests for disassembly with the most frequent mobile phones the following concept for the disassembly cell was created (Fig. 2). It consists of five automated stations plus a manual feeding and removal station.

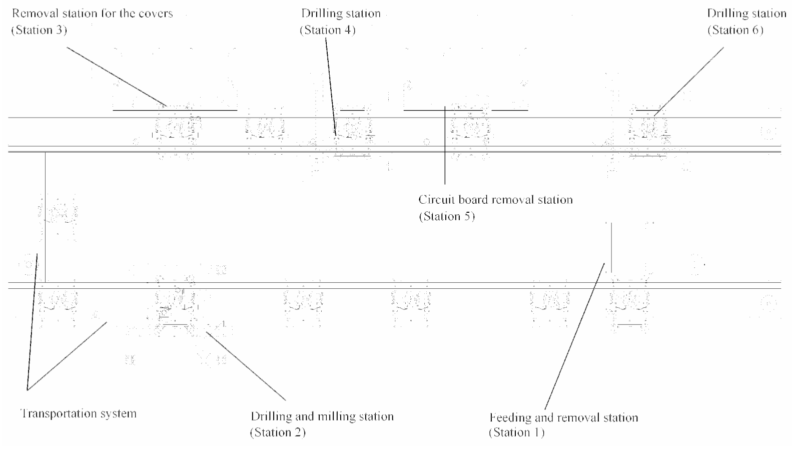


Fig. 2. Layout of the disassembly cell

For disassembly, the mobile phones were fixed on a pallet in a defined position. These pallets are moving around on a transportation system. According to the necessary disassembly operations the pallets with the mobile phones to be disassembled are stopped, lifted and fitted in the corresponding stations.

Before the mobile phone is fixed on a pallet the power supply will be removed and the type of the handy will be recognized by a barcode reader manually. Now the control computer knows exactly the type of the handy. The main dimensions of the handy are stored in a database of the host computer.

In the drilling and milling station (no. 2) the upper part of the handy will be cut off from the lower part and the screws – usually between 4 and 17 – are removed by a simple drilling mechanism. The dust content is removed by air from the pallet.

In the third station – the cover removal station – the cover as well as the keyboard of the handy will be removed by pneumatic sucks.

In the next station – drilling station; no. 4 – the screws which connect the printed circuit board to the lower part of the housing are removed.

In the printed circuit removal station various other parts will be removed from the handy.

Because some mobile phones have additional parts connected with the power part of the housing of the handy the remaining screws will be removed in the last drilling station – station 6. Finally the lower part of the handy will be removed in the fixing and removal station.

As a development of this semi-automated disassembly cell for used mobile phones some previous tests were necessary. For the milling in the drilling and milling station (no. 2) it was necessary to make tests with grinding wheels, with different saws and with milling devices. Finally a milling device was chosen as the right tool for this task.

Further extensive tests were carried out for the removal of the screws. From the literature there are very high sophisticated, complicated and therefore very expensive and heavy devices known. We found a very simple and very cheap method for the removal of the screws.

2.3 A ‘Tool Kit’ for Mobile Robots

The basis of a modular concept for mobile robots is the Mobile Robot Platform (MRP) which can be described as a multi-use mobile robot, developed in its basic configuration.

These platforms can be divided in some basic systems:

- Locomotion system
- Drive system
- Main control system
- Communication system

The mobile robot platform can be upgraded and modified by adding a number of peripheral systems and tools for the performance of different tasks or functions (Fig.3). There is a large variety of tools, which can be used.

Conventional tools (screw drivers, drilling tools, polishing tools, etc) are similar in regard to their function to conventional hand-held tools for manual operations. The difference is in their design, since they have to be fixed on the mobile robot platform, and actuation.

Special tools installed onboard of a mobile robot platform changes the same to a specialized mobile robot system. When special tools are lightweight constructed the manipulation system can be more flexible and with wider reach. Heavier tools cannot be very flexible. They need more rigid and strong manipulation systems. So there is often only one degree of freedom applied, and the other DOFs are realized by the mobility of the platform.

Installing a tool changing system enables the robot to achieve a wide variety of performable operations. Tool changing systems are normally placed at the end of a robot arm. They have to be light, simple and very reliable.

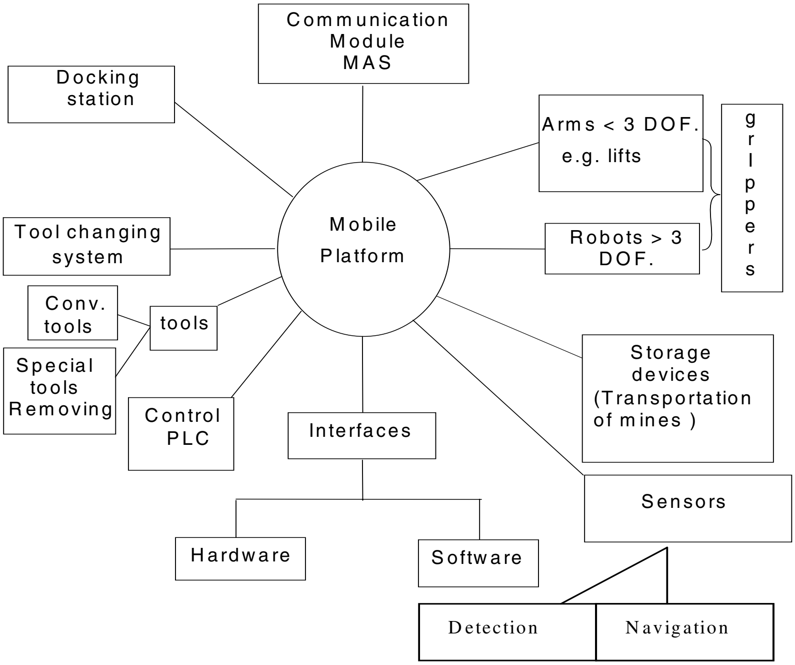


Fig. 3. Modular Robot System [7]

The basic configuration of each mobile robot platform has its integrated sensors. The navigation system makes excessive use of sensor for position determination and collision avoidance. But there are numerous possibilities to upgrade the system with additional sensors for some special applications or to extend its abilities.

In many mobile robot applications transportation is an important part of the overall task. To transport different items mobile robot platforms have to be upgraded with another type of peripheral devices: special storage systems or devices.

Although mobile robot platforms are normally equipped with a communication system it could be necessary to use some special communication systems. Especially in multi agent systems (MAS) where more robots act, cooperatively together communication is important.

2.4 A Tool Kit for Humanitarian Demining [5]

According to current estimates, more than 100.000.000 anti-personnel and other landmines have been laid in different parts of the world. A similar number exists in stockpiles and it is estimated that about two million new ones are being laid each year. According to recent estimates, mines and other unexploded ordnance are killing between 500 and 800 people, and maiming 2.000 others per month.

Landmines are usually very simple devices which are readily manufactured anywhere. There are two basic types of mines:

- anti-vehicle or anti-tank (AT) mines and
- anti-personnel (AP) mines.

AT mines are comparatively large (0.8 – 4 kg explosive), usually laid in unsealed roads or potholes, and detonate which a vehicle drives over one. They are typically activated by force (>100 kg), magnetic influence or remote control.

AP mines are much smaller (80-250g explosive, 7-15cm diameter) and are usually activated by force (3-20kg) or tripwires. There are over 700 known types with different designs and actuation mechanisms.

Hand-prodding is today the most reliable method of mine clearing, but it is very slow, and extremely dangerous. A person performing this type of clearing can normally perform only this task for twenty minutes before requiring a rest. This method clears one square meter of land in approximately 4 minutes.

Todays methods for destroying and removal are brutal force mechanical methods including ploughs, rakes, heavy rolls, flails mounted usually on tanks. The main problem with these methods is the contamination of the ground for 10 – 20 years. A better solution for the future is the use of demining robots.

A new approach is the use of robots in "Swarms". Swarms of robots can be connected; one is for searching, one for destroying and one for displacement. These three swarms consist of different types and numbers of robots. Many robots are searching and few or only one is necessary for destroying or displacing the mines.

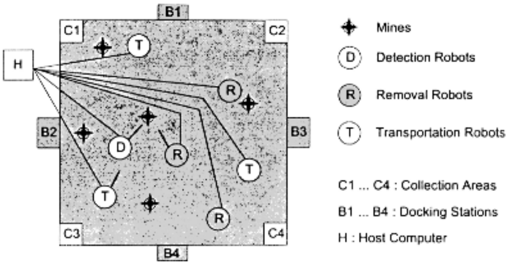


Fig. 4. Robot swarms for demining [5]

Robot swarms increase the number of robot applications in various areas where robots are already used today. Robot swarms are similar to – or a synonym for - 'Multi Agent Systems – MAS'. These systems are very well known in software engineering – "software agents" - since more than twenty years. In the last years there are more and more works related to "hardware agents" like robots.

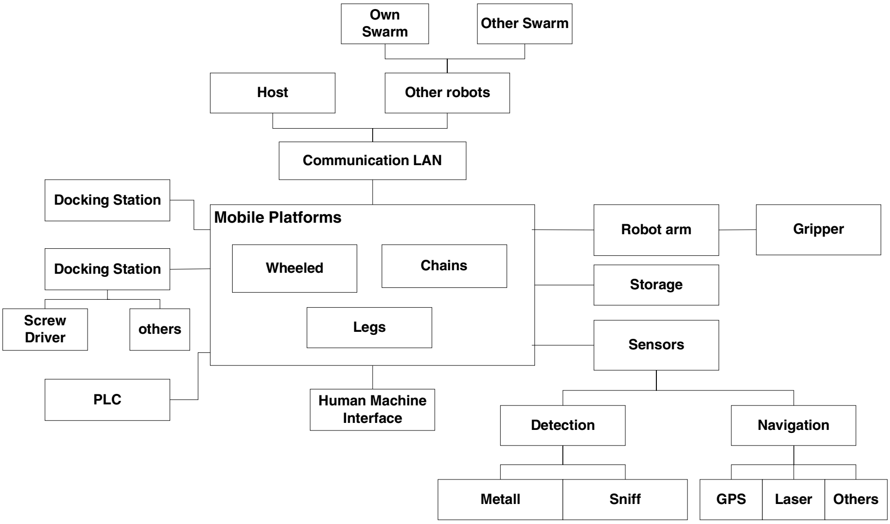


Fig. 5. "Tool Kit" for demining robots [5]

As mentioned before the use of modular robots is perfect for the design of task specific demining robots because of the similarities between the tasks. All three different types of robots can be realised by the toolkit of Fig. 5.

2.5 Roby-Run: A Mobile Mini Robot

One of the newest application areas of service robots is the field of entertainment, leisure and hobby because people have more and more free time. A new term "edutainment" – composed of two words, education and entertainment was created.

One example is robot soccer introduced with the purpose to develop intelligent cooperative multi-robot (agents) systems (MAS). From the scientific point of view a soccer robot is an intelligent, autonomous agent, carrying out tasks together with other agents in a cooperative, coordinated and communicative way. Robot soccer provides a good opportunity to implement and test MAS algorithms. Furthermore it is an excellent tool to make "High Tech" transparent to broader public by playing.

At our institute four robot soccer teams, three in the category MiroSot (Micro-Robot Soccer tournament) and one in the category NaroSot (Nano-Robot Soccer tournament) are used as a test bed for MAS and edutainment.

The size of playground (Fig. 6) bounded on all sides in category "MiroSot" is 150 x 130cm, 220 x 180cm, 280 x 220 cm or 440 x 280 cm depending on the number of the players.

A camera approximately 2m over the playground delivers pictures to the host computer. With information from colour patches on top of the robots, the vision software calculates the position and the orientation of the robots and the ball. Using this, the host computer generates motion commands according to the implemented game strategy and sends motion commands wireless to the robots.

It is strictly forbidden that the three human team members directly control the motion of their robots either with a joystick or by keyboard commands during the game. Only the host computer is responsible. The duration of a game is two times 5 (7.5) minutes with a half time break of 10 minutes.

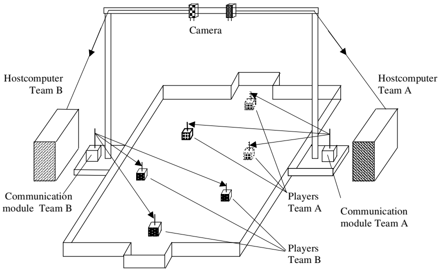


Fig. 6. Overall system of robot soccer [6]

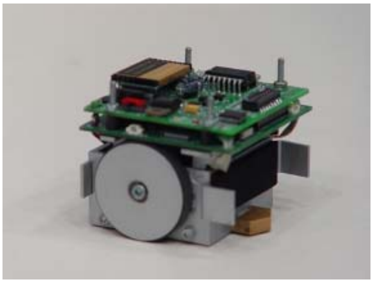


Fig. 7. The mobile mini robot “Roby Run”

2.6 Mobile Mini Robots for Space Applications: “Roby Space”

To get energy from the sun an approach is to set up nets with solar cells in the space and transmit the energy wireless to the earth by microwaves. For first tests a net (approximately 40 x 40 m) equipped with solar cells should be installed in outer space (~ 200 km above the earth). The main problem is the positioning of the solar cells on the net structure. For this task autonomous mobile robots could be used able to move (crawl) on this large quadratic mesh. The distance of the mesh wires is between 3 and 5cm; their thickness between 1 and 3mm.

The features of an autonomous mini robot for this purpose are:

- the maximum dimension 10x10x5cm
- light weight (less than 1 kg)
- simple mechanical construction,
- miniaturized electronics
- “low cost”
- independent of the mesh’s dimension (from 3 x 3cm to 5 x 5cm).
- on board power supply for approximately 10min

- equipped with a camera sending pictures to the earth
- wireless communication with the mother satellite by Bluetooth or similar
- free movement on the mesh
- mechanical and electronic robustness against low/high temperature, radiation, microgravity, vibration and shock during the flight in the rocket.

The main problem is the design of the moving and holding mechanism of the robot on the mesh. As a direct spin off from robot soccer two prototypes (Fig. 8) - Roby-Sandwich and Roby-Insect, based on “Roby Run” - were built and tested. Two tests – low temperature test at the 40 degree below zero and micro gravity tests- were already successfully done. At the 40 degree below zero Roby-Sandwich crawled on the net without any problem. In January 2005 these two robots were tested in the micro-gravity environment by means of parabolic flights in Japan.

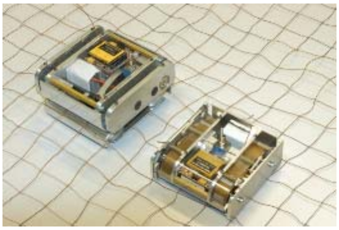


Fig. 8. Roby-Sandwich (Left) and Roby-Insect (Right)

3 The New Concept of a Humanoid Robot

The two legged, humanoid robots currently available can be divided in two categories:

“Professional” humanoid robots developed by large companies with a huge amount of research capacities. Examples are: the Honda robots (P1, P2, P3, ASIMO) – with the idea to assist humans in everyday working, the SONY robots (SDRX – 3,4,5) and “Qrio” – with the background to serve mostly for entertainment, leisure and hobby or in the future as personal robots. These robots are currently not available on the market not only because of the very high price.

“Research” humanoid robots: There a lot of such robots currently available or in the development stage e.g. approximately worldwide more than 500 University institutes and research centres are active in this field. The robots of this category a usually prototypes developed by computer scientists to implement methods of AI, image processing, theoretical scientists from mechanics implementing and testing walking mechanisms, control scientists to implement new control strategies, social scientists to implement human machine interfaces (HMI) for an efficient communication between humans and humanoid robots.

1. We are currently working on a humanoid, two legged robot called ARCHIE. The goal is to build up a humanoid robot situated just between these two worlds. Therefore Archie needs a head, a torso, two arms, two hands and two legs.

New is the control system realised by a network of processing nodes (distributed system), each consisting of relative simple and cheap microcontrollers with the necessary interface elements. According to the currently available technologies the main CPU is for example a PDA module, one processor for image processing and audio control and one microcontroller for each structural component, e.g.: a Basic Stamp from Parallax.

4 Summary and Outlook

In this paper some new, partially realized, applications of a new robot generation are described. In addition modern information technologies lead to loneliness of the humans (teleworking, telebanking, teleshopping, ...). Therefore service robots will become a real “partner” of humans in the nearest future. One dream of the scientists is the “personal” robot. In 5, 10 or 15 years everybody should have at least one of such a robot. Because the term personal robot is derived from personal computer the prices should be equal.

Acknowledgements

Parts of this work is supported by ESA under ESTEC/Contract No.18178/04/NL/MV-Furoshiki Net Mobility Concept

References

1. Kopacek, P., Noe, D.: Intelligent, flexible assembly and disassembly. In: Proceedings of the IFAC Workshop „Intelligent Assembly and Disassembly – IAD’01”, (2001) 1–10.
2. Kopacek, B., Kopacek, P: Semi-automatised Disassembly. Proceedings of the 10th International Workshop “Robotics in Alpe Adria Danube Region – RAAD`01”, Vienna, May 16. – 18. 2001, (2001) 363-370.
3. Kopacek, P., B. Kopacek: Robotized disassembly of mobile phones. In: Preprints of the IFAC Workshop „Intelligent assembly and disassembly – IAD’03”, Bucharest, (2003) 142-144.
4. Kopacek, P., Han, M.-W., Putz, B., Schierer, E., Würzl, M. : Robot soccer a first step to “edutainment” In Proceedings of RAAD’03, 12th International Workshop on Robotics in Alpe-Adria-Danube Region Cassino, Italy 2003
5. Kopacek, P.: Robots for humanitarian demining. In: M Voicu (eds.): Advances in Automatic Control, Kluwer, (2004) 159- 172.
6. Kopacek, P., Han, M.-W., Putz, B., Schierer, E., Würzl, M.: "A concept of a high-tech mobile minirobot for outer space"; in: "ESA SP-567 Abstracts of the International Conference on Solar Power from Space SPS'04 together with the International Conference on Wireless Power Transmission WPT5", Granada, Spain, (2004)132 - 133.
7. Shivarov, N.: A ‘Tool Kit’ for modular, intelligent, mobile robots, PhD. Thesis, Vienna University of Technology (2001)

Current and Future Trends and Challenges in Robot Soccer

Norman Weiss and Bernd Reusch

University of Dortmund, Chair Computer Science I,
Otto-Hahn-Str. 16, 44227 Dortmund, Germany
`{Norman.Weiss, Bernd.Reusch}@uni-dortmund.de`

Abstract. Robot soccer has evolved into a very dynamic and competitive field within the last few years. Many different robot soccer leagues now exist, the league most strongly dedicated to entertainment and edutainment is currently FIRA MiroSot. It has now reached 11 vs. 11 robots sized 7.5 x 7.5 x 7.5 cm on a 440 x 280 cm field and is therefore the first robot soccer league to physically play 11 vs. 11 games. This paper aims to provide a short status report and introduction into current problems and possible solutions within the challenging areas of FIRA MiroSot robot soccer created by having 22 robots on the field.

1 Introduction

Robot soccer provides excellent research and benchmarking opportunities in a diverse variety of fields, e.g. multi-agent systems, robot control, sensor fusion, intelligent control, communication, image processing, mechatronics and artificial intelligence.

Many different robot soccer leagues now exist. The scope ranges from truly tiny, centrally controlled robots in FIRA NaroSot (4.5 x 4.5 x 5 cm) to humanoid, fully autonomous robots in humanoid leagues (up to 180 cm in height). At the same time, the game play of the different leagues can be everywhere between highly autonomous and research-oriented to high-speed and entertainment-oriented.

The robot soccer league most strongly dedicated to entertainment and edutainment is currently FIRA MiroSot. In that league, the size of a robot is limited to 7.5 x 7.5 x 7.5 cm, which is very small considering the tasks it has to fulfil. The league started in 1996 with 3 vs. 3 robots playing on a field sized 150 x 130 cm. To get closer to “real” soccer, the number of players on the field slowly grew (5 vs. 5, 7 vs. 7) along with the field. It has now reached 11 vs. 11 robots on a 440 x 280 cm field.

During the early days of the league, the robots’ mechanical designs, their control and the vision systems were limiting factors. Since then, it has evolved into a very dynamic, high-speed game with robots reaching speeds of up to 4.0 m/s (14.4 km/h) during game play. Alongside with now refined strategies, this provides a very entertaining experience to spectators.

Due to the very small size of the robots, they must be supported by a host computer (usually an off-the-shelf PC) which receives a picture of the field from a camera mounted about 2.5 m above the field. The host is responsible for image processing and strategic decisions. It transmits – via a radio link – movement information to the robots on the field, which they execute, thereby closing the control cycle.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 559 – 564, 2005.
© Springer-Verlag Berlin Heidelberg 2005

While some problems robot soccer posed have been solved in the last few years [1], many still exist. At the same time, constant change in the league adds new problems – especially the change to 11 vs. 11 robots has posed many new challenges to the teams. This paper aims to provide a short status report and introduction into current problems and possible solutions within the challenging areas of FIRA MiroSot robot soccer created by having 22 robots on the field.

2 Robot Hardware

Since FIRA MiroSot is a "real life" competition, not only the quality of the software decides which team is competitive and which not. One major factor is the quality of the robot and the ability to control it precisely during the entire game. That is why, during the last few years, there has been a strong focus on the development of competitive FIRA MiroSot soccer robots.

First designs like the "robot cavalry" by the Korean company MicroAdventure were rather slow (at about 0.8 m/s net, approx. 3.0 km/h) [2] and still error-prone, but the quality of the robots rose quickly. The control quality rose as well, both regarding the robots themselves and the overall control cycle robot-camera-PC-radio-robot. The robot control mainly follows standard solutions of control engineering, but a number of dedicated mobile robot path planning algorithms have been adapted or developed for use in robot soccer, e.g. the purely reactive CMU algorithm [3], potential field based methods [4] or geometry based algorithms like limit-cycle navigation [5] or the S-curve algorithm [6].

Good MiroSot robots are now available commercially [7, 8] as well as from universities [9, 10]. Up until very recently, robot speeds of about 1.5 – 2.0 m/s (5.4 – 7.2 km/h) were common. The Austrian team of the Technical University of Vienna now introduced a much faster robot (4.0 m/s, 14.4 km/h) [9], which contributed strongly to their success during the World Championships 2005. This is why – after a short phase of consolidation – robot hardware design has again become an important point of focus within FIRA MiroSot.

3 Vision

The field the 11-vs.-11-league is played on measures 440 x 280 cm, the cameras are mounted at 2.5 m above the field. That means that strong fish eye lenses are necessary to get an image of the entire field and that the robots cover only a very small area within the image[1]. The fish eye effect is very undesirable, since it impedes exact calculation of the robots' positions. Raising the camera to a higher level is not possible since this would rule out too many locations for playing.
This leaves three solutions to the problem:

3.1 Optical Correction of the Fish Eye Effect

An optical correction can be ruled out in almost all cases. Most teams now use digital cameras like Sony DFW-V500 or Basler A311fc, mostly with standard C-Mount,

---

[1] Assuming a 720x576 pixel camera, every pixel covers an area of about 0.6 x 0.6 cm. The color team marker has a minimum size of 3.5 x 3.5 cm, i.e. only 5.7 x 5.7 pixels.

similar lens mountings or no mounting at all. For these cameras – to our knowledge – suitable lenses simply do not exist, and besides that it can be speculated that they would be quite expensive.

3.2 Correction of Image Distortions by Software

Obviously, the correction of image distortions induced by camera optics has been a long-standing problem outside of robot soccer [11], e.g. in areas like aerial photography. In contrast to other areas of application the correction algorithms for robot soccer have to perform in real time. Assuming a frame rate of 30 pictures a second (often even more), an algorithm cannot use more than a few milliseconds of computing time on an off-the-shelf PC.

The first approaches of many robot soccer teams therefore used simple linear corrections [12] mainly suitable for trapeze distortions and very limited fish eye distortions. In the meantime, work has progressed greatly on the subject with some robot soccer teams using very precise algorithms that can cope even with strong image distortions in real-time. Complete toolboxes are available [13, 14] implementing a two-step process based on the work of Zhang [15] and Heikkilä and Silvén [16]: They calculate intrinsic and extrinsic camera parameters (i.e. lens distortions and perspective correction) semi-automatically beforehand and create a look-up table for conversion. This way, image correction can be done efficiently under real-time conditions during the game by simply using the table.

With these algorithms it is feasible to play in the 11-vs.-11-league using one camera with a fish-eye lens. Nevertheless, the problem that the robots appear very small on the recorded image remains.

3.3 A Two-Camera Approach

Since the quality of the position and particularly the orientation calculations strongly depend on the number of pixels allotted to every colour, using one camera for the entire field results in a relatively low recognition quality. This is why some teams use a two-camera approach for the 11-vs.-11-league, where every camera records one half of the field [17, 18]. While the advantages are obvious, there are also disadvantages: The load on the host computer created by the vision system will almost double, as well as the setup time before the game, since two cameras have to be mounted over the field. Depending on the previous design of the vision software, the extension might also be very complex to implement if the processing steps during image processing have not been separated suitably within the software.

4 Strategy

Originally, only 6 robots (3 vs. 3) were playing during a FIRA MiroSot game. This put emphasis on control and vision problems, because the matter of strategy was deliberately kept simple. One goalkeeper and two robots actually playing the game did not allow for any refined strategic moves. This has strongly changed during the last few years. Strategy design will become much more difficult within FIRA MiroSot because of the recent extension to 22 robots on the field. Simple approaches as outlined above will not suffice and much more complex strategies will be needed that get much closer to “human” soccer. Complex team based behaviours like passes that require a highly sophisticated strategy and coordination will be of major advantage to the team mastering them.

For the first time it is possible to implement, test and compare robot soccer strategies similar to human soccer strategies on a “real” field instead of within simulation software. Currently, most teams still rely on strategies that evolved during the last few years. In an 11-vs.-11 game, this sometimes results in strategic moves like a 4-defender-chain, where 4 defending robots move exactly alike in the team’s own half because there is just a strategy implemented for one but not four independent defenders.

Current state of the art in FIRA MiroSot are (still) rule-based strategies [1], where a flexible design will be of advantage to adaptation to 11-vs.-11 [19]. Regarding future strategy design, FIRA MiroSot benefits from the fact that there is a Simulation League within FIRA. That league simulates MiroSot games[2]. The strategies here are mainly based on learning approaches, in majority reinforcement learning [20]. That means that there already is a set of options of useful 11-vs.-11 strategies, although there usefulness under “real” conditions must still be proven.

For example, strategies developed by learning methods (like evolution strategies, situational learning etc.) face a major difficulty in real life: The amount of learning (i.e. evolution steps, trial situations etc.) that can be done when using real robots is very limited compared to simulation environments. This is because simulation is neither bound to real time (i.e. can be much faster) nor by hardware limitations (e.g. power supply, wear and tear etc). At the same time, all data learned in simulation environments might not hold in “real” settings because of the inevitable differences between the simulation’s model and its counterpart in reality.

It can be said that the conditions under which FIRA MiroSot is played will not change anymore during the mid-term range. Certainly, 11-vs.-11 is the maximum number of robots that will play. As well, the field size will not grow, as even the current fields are hard to handle. One major change could be an extension of the field size but that will also mean that the field has to be soft instead of hard (currently usually wooden), at the same time it might as well lose its borders, as has been demonstrated by other leagues. Certainly, it can be said that as the playing conditions will not change in the short run and many teams now have a more and more sophisticated base of robot hardware and vision system, advances in the strategies will certainly be of high benefit to the teams mastering them.

5 Radio Link

All FIRA MiroSot robots are controlled by the host computer via a radio link. Currently, most teams use BiM series radio modules manufactured by the English company Radiometrix [7, 9, 10] or similar modules [1, 8]. These transmit in the 433 or 869 MHz ISM bands with specified gross data rates of up to 160 kbps. Unfortunately, due to various reasons, the net data rate will be around 35 kbps in this application, with the modules being half duplex only.

---

[2] This is a contrast to the RoboCup simulation league, which is not aimed at simulating a specific robot soccer league but is oriented more towards “human” soccer.

With 22 robots on the field playing robot soccer, the limits of the Radiometrix modules are now reached. New solutions like differently designed radio hardware transmitting in different frequency bands as well as updated and more stable protocols are needed. Unfortunately, commercially available standard solutions like WLAN and Bluetooth are not suitable for MiroSot robots. For WLAN, the main disadvantage is still the size of the available modules, despite recent advances due to WLAN integration into PDAs. For Bluetooth, it is mainly the somewhat limited net data rate in this application (although still much higher than with BiM modules) and the missing protocol suitability (like the limitation to 8 devices in a given standard Bluetooth net).

Certainly, it can be said that with current solutions no communication beyond the bare necessity of transmitting wheel speeds can be done, therefore one alternative shall be shortly introduced: The MiroSot team Dortmund Droids has developed a radio module operating in the licence-free 2.4 GHz frequency band with a data rate of 1 Mbps which is pin- and control-compatible to Radiometrix BiM modules (cf. fig. 1).

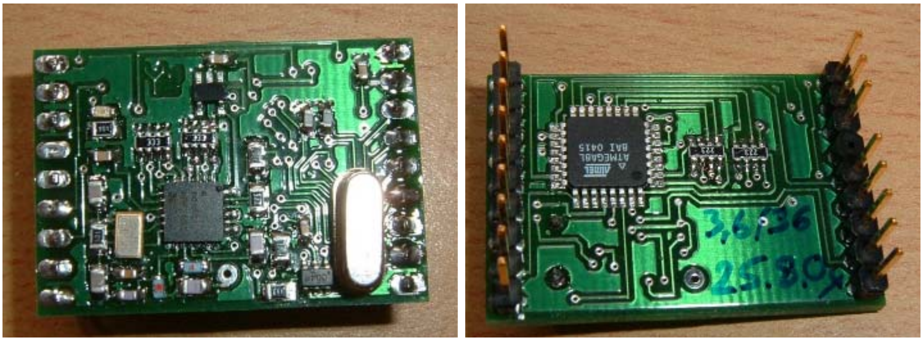


Fig. 1. Radio module (top view left, bottom view right)

The module is based on a single chip transceiver with a data rate of up to 1 Mbit/s. The chip – a nRF2401 built by Nordic – transmits on one of 125 channels. And with a short channel switching time of less than 200 μs, frequency hopping is possible. Payload generation and CRC generation/checking can be handled fully by the chip, which has two independent receivers, so it can receive from two channels simultaneously. Besides the chip only a crystal and a passive antenna matching network are needed. In order to establish compatibility to the Radiometrix BiM modules, the design includes a microcontroller that translates between the protocol used by the nRF2401 transceiver and the standard serial protocol used by the BiM modules.

References

1. Kim, J.-H., Kim, D.-H., Kim, Y.-J., Seow, K.-T.: Springer Tracts in Advanced Robotics, Vol. 11: Soccer Robotics. Springer Verlag, Berlin Heidelberg New York (2004)
2. Reusch, B. et al: Endbericht der Projektgruppe 340 – Roboterfußball. Internal Reports of the Department of Computer Science of the University of Dortmund, Dortmund, Germany (2000) (In German)

3. Veloso, M. et al: The CMUnited-98 Champion Small Robot Team. In: Asada, M., Kitano, H. (Eds.): RoboCup-98: Robot Soccer World Cup II. Lecture Notes in Computer Science, Vol. 1604. Springer-Verlag, Berlin Heidelberg New York (1999) 77-94
4. Kim, Y.-J., Kim, J.-K., Kwon, D.-S.: Evolutionary programming-based uni-vector field navigation method for fast mobile robots. IEEE Trans. On Systems, Man and Cybernetics, Part B 31(3) (2001) 450-458
5. Kim, D.-H., Kim, J.-H.: A real-time limit-cycle navigation method for fast mobile robots and its application to robot soccer. Robotics and Autonomous Systems 42(1) (2003) 17-30
6. Hildebrand, L., Reusch, B. et al: Path Planning For Mobile Robots Using The S-Curve Algorithm. Proc. FIRA World Congress 2003, Vienna, Austria (2003)
7. High Qualified Soccer Robot for Undergraduates & General Teams YSR-A. Yujin Robotics Co., Seoul, Korea (2005) http://www.edrobot.com/english/product/ysra.asp
8. Educational & Match Soccer Robot for Elementary & Middle Students VICTO. Yujin Robotics Co., Seoul, Korea (2005) http://www.edrobot.com/english/product/victo.asp
9. Putz, B.: Development of the new Soccer Robots "Roby-Speed" AND "Roby-Naro", Proc. CLAWAR/EURON ELH04, Vienna, Austria (2004)
10. Klute, T., Weiss, N., Schulz, S., Pfeifer, T.: A DSP-based Soccer Robot for FIRA MiroSot. Proc. 16th IFAC World Congress, Prague, Czech Republic (2005) (to be published)
11. Clarke, T.A., Fryer, J.G.: The Development of Camera Calibration Methods and Models. Photogrammetric Record, 16(91) (1998) 51-66
12. Simon, M., Behnke, S., Rojas, R.: Robust Real Time Color Tracking. In: Stone, P., Balch, T., Kraetzschmar, G. (Eds.): RoboCup 2000: Robot Soccer World Cup IV. Lecture Notes in Computer Science, Vol. 2019. Springer-Verlag, Berlin Heidelberg New York (2001) 239-248
13. Bouguet, J.-Y.: Camera Calibration Toolbox for Matlab. California Institute of Technology, Pasadena, CA, USA (2004)
http://www.vision.caltech.edu/~bouguetj/calib_doc/index.html
14. Open Source Computer Vision Library. Intel Corp., Santa Clara, CA, USA (2004) http://www.intel.com/research/mrl/research/opencv/index.htm
15. Zhang, Z.-Y.: A Flexible New Technique for Camera Calibration. PAMI 22(11) (2000) 1330-1334.
16. Heikkila, J., Silven, O.: A four-step camera calibration procedure with implicit image correction. Proc. CVPR97, San Juan, Puerto Rico (1997) 1106-1112
17. Weiss, N., Jesse, N.: Towards Local Vision in Centralized Robot Soccer Leagues: A Robust And Flexible Vision System Also Allowing Varying Degrees Of Robot Autonomy. Proc. FIRA World Congress 2004, Busan, Korea (2004)
18. Weiss, N., Hildebrand, L.: An exemplary Robot Soccer Vision System. Proc. CLAWAR/EURON ELH04, Vienna, Austria (2004)
19. Hildebrand, L., Michalski, C., Valentin, H., Wickrath, M.: Strategy Implementation For Mobile Robots Using The Pipes & Filters Architecture. Proc. FIRA World Congress 2003, Vienna, Austria (2003)
20. Zhang, Y.-D, Min, F.: Application of Reinforcement Learning Based on Artificial Neural network to Robot Soccer. Proc. FIRA World Congress 2004, Busan, Korea (2004)

Strategy and Communication in Robotic Soccer Game

Bobumil Horák[1], Marek Obitko[2], Jan Smid[3], and Václav Snášel[1]

[1] Department of Computer Science, FEI, VŠB,
Technical Univerzity Ostrava,
17. listopadu 15, 708 33, Ostrava-Poruba, Czech Republic
{bohumil.horak, Vaclav.snasel}@vsb.cz
[2] Gerstner Laboratory, Department of Cybernetics,
Faculty of Electrical Engineering, Czech Technical University,
Technicka 2, 166 27 Prague, Czech Republic
obitko@labe.felk.cvut.cz
[3] Computer Science Department, Morgan State University, Baltimore MD, USA
jsmid@jewel.morgan.edu

Abstract. We describe the key components of our game strategy for robot soccer [3] in detail. The game can be represented as a trajectory in so-called the virtual grid. The virtual grid generally allows us to reduce data volume for easy description of player motion and subsequently for controlling the game or for learning game strategies. The natural coordinate system is provided be accurate optical sensing of the subject position using lens optical transformations and the CCD camera. This natural coordinate system can be easily mapped to a virtual grid.

1 Introduction

In robot soccer, the game situation in the playground is typically read in terms of the robots postures and the balls position. Using real-time information of this dynamically changing game situation, the host computer program of a command-based robot soccer team would need to continually decide the role and action to take of each team robot, and to direct each robot to perform a selected action. The purpose, of course, is to get the team robots to exhibit some artificial form of cooperation, manifested by their coordinated movements, ball passing, running into proper postures and ball shooting in the goal-ward direction as often as the opportunity arises in the course of the match.

The game can be represented as a trajectory in what we call the virtual grid. The virtual grid generally allows us to reduce data volume for easy description of player motion and subsequently for controlling the game or for learning game strategies. The natural coordinate system is provided by accurate optical sensing of the subject position using lens for optical transformations and the CCD camera. This natural coordinate system can be easily mapped to a virtual grid. A sample picture before processing is shown in the figure 1. The data volume of the description using the virtual grid is obviously smaller than the description using natural coordinates. The exact values depend on the frequency of samples, i.e. on the used CCD camera (typically 25-75 fps) and on the maximal velocity of the mobile robot movement at game field (typically up to 2,5m/s). The dimensions of the primary virtual grid are determined by the possible distance of

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 565–570, 2005.
© Springer-Verlag Berlin Heidelberg 2005

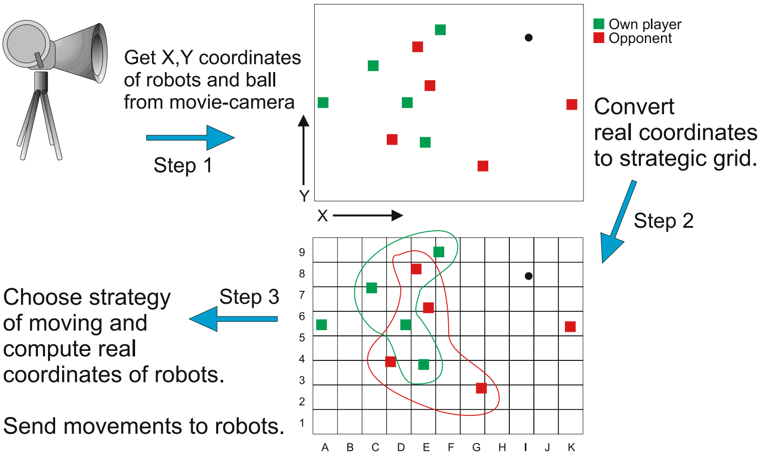

Get X,Y coordinates of robots and ball from movie-camera
Step 1
Y
X
🟩 Own player
🟥 Opponent
Convert real coordinates to strategic grid.
Step 2
Choose strategy of moving and compute real coordinates of robots.
Step 3
Send movements to robots.
9
8
7
6
5
4
3
2
1
A B C D E F G H I J K

Fig. 1. Alphanumeric representation of robot position and movement using virtual grid

the robot position in two subsequent frames from the CCD camera. The primary virtual grid can be divided to (2, 4, 8, . . .) parts, which creates *secondary* virtual grid.

Using the virtual grid, it is possible to describe the position of the robot using an alphanumeric description see [1,2].

The tercial grid – emphstrategy grid – depends on the partition of the game field (the left-right wing, the central field, and transversely the attack-defence field and the central field). In the discrete frame samples it is possible to study movements and movement strategies of the robot.

2 Strategy

The game strategy can be dynamically changed based on the game progress (i.e. the history and current position of the players and the ball).

Strategy application for one movement of players is computed in following steps:

1. Get coordinates of players and ball from camera
2. Convert coordinates of players into strategic grid
3. Convert ball and opponents' positions into virtual and strategic grids
4. Choose goalkeeper and attacker, exclude them from strategy and calculate their exact positions.
5. Detect strategic rule from opponents' and ball positions
6. Convert movement from strategic grid to physical coordinates
7. Send movements to robots

The game progress can be divided in time into the following three ground playing classes (GPC):

- GPC of game opening (GPCO)
- GPC of movements in game site (GPCS)
- GPC of game end (GPCE)

The game progress, especially in the GPCS class, can be also divided into the following two games playing situations (GPS):

- GPS of attack (GPSA). The interactions of simple behaviours cause the robots to fall into a V-formation where the ball is in motion roughly towards the opponents goal.
- GPS of defence (GPSD). When the ball is not moving roughly towards the opponents goal, the robots move around it to form an effective barrier and to be in a good position for recovery.

Each GPC has its own movement rules. The classes GPCO and GPCE consist of finite number of possible movements that are determined by initial positions of players and the ball. The class GPCS has virtually unlimited number of possible movements (see Fig. 2). The movements are determined by the current game situation (GPS) and by the appropriate global game strategy (in next GGS). The movement of the particular robot is determined by the current game class and situation, and also by the robot role. For example, the goalkeepers task is to prevent the opponent to score a goal. His movements are in most cases limited along the goalmouth near of goal line. The preferred movements are in goal line direction. The preference of these movements comes from the particular GGS, where the goalkeeper prevents to score a goal in the way of moving in the position between the central goal point and the ball (or the expected ball position). The preference of other movement directions is created using GPSA, where the movements of goalkeeper secure kicking the ball from the defence zone.

Each strategy is stored in one file and currently consists of about 50 basic rules. Furthermore the file contains following metadata (see Fig.3) :

- Information about the name of strategy (e.g. *.Strategy "S-1"*),
- the algorithm to strategy choosing (e.g. *.Algorithm 2*),

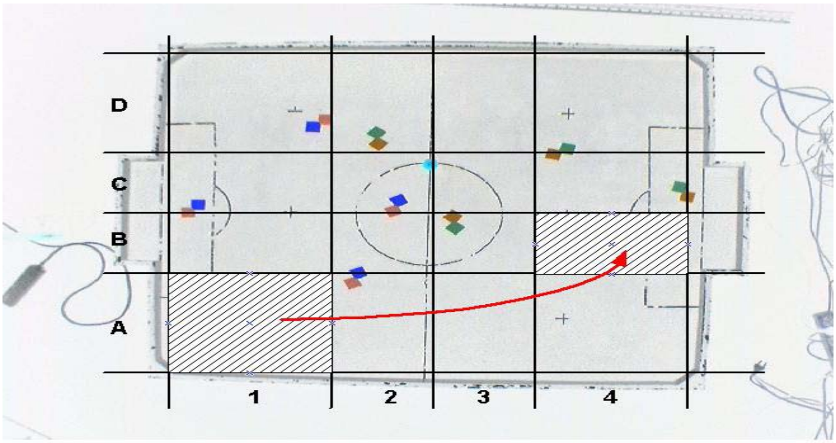


Fig. 2. Example of strategy movements

```
.Strategy "S-1"
.Algorithm 2
.Author "Vaclav Snasel"
.Date "1.2.2005"
.Size 11x9

.Rule 1 "Attack1"
.Mine a4 b3 c1 d1 d4
.Opponent a2 b3 c2 d3 d4
.Ball a1
.Move a4 b3 c2 d3 d4

.Rule 2 "Attack2"
.Mine a4 b3 c11 d1 c10
.Opponent a2 b3 c2 d3 a1
.Ball a1
.Move a4 b3 c2 d1 a11
⋮
```

Fig. 3. Sample of strategy notation with 9×11 play field

- the author responsible for current strategy (e.g. *.Author ”Vaclav Snasel”*),
- the date of last modification (e.g. *.Date ”1.2.2005”*),
- the size of strategic grid (e.g. *.Size 11x9*),
- strategic rules

Each rule consists of five records:

- The rule ID and description (e.g. *.Rule 1 ”Attack1”*),
- the coordinates of our players in strategic grid (e.g. *.Mine a4 b3 c1 d1 d4*),
- the coordinates of opponent's players in strategic or virtual grid (e.g. *.Opponent a2 b3 c2 d3 d4*),
- the ball coordinates in virtual or strategic grid (e.g. *.Ball a1*)
- strategic or virtual grid positions of the move (*.Move a4 b3 c2 d3 d4*).

In current algorithm, the *.Mine* coordinates are not important for movement rule selection, but they can be used in the future. We believe, that the file system for strategies is an advantage. From observation of opponent's strategy a new set of rules can be written, without necessity of program code modification. Furthermore, there is a possibility of automatic strategy (movement) extraction from running game.

There exist two main criteria in the Strategy selection process (see Fig. 4). The selection depends on opponents' coordinates and ball position. The strategy file contains rules, describing three possible formations suggesting danger of current game situation. The opponent's team could be in offensive, neutral or defensive formations. Furthermore, we need to weigh up the ball position risk. Generally, opponent is not dangerous if the ball is near his goal. The chosen rule has minimal strategic grid distance from current configuration of players and ball.

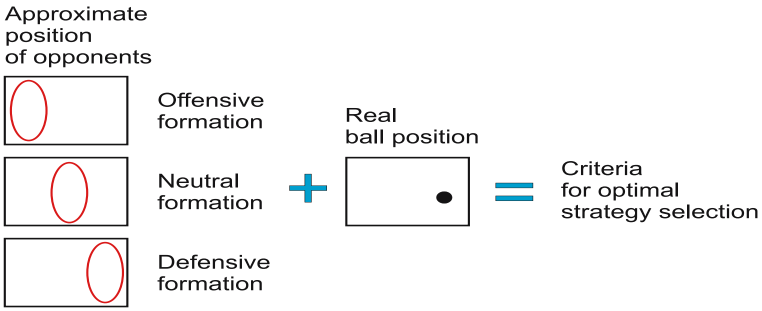


Fig. 4. Main criteria for strategy selection

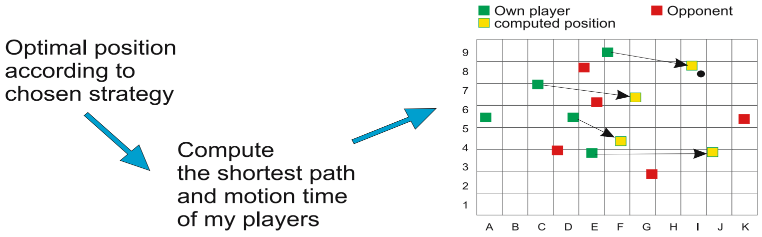

Optimal position according to chosen strategy
Compute the shortest path and motion time of my players
Own player
computed position
Opponent
9
8
7
6
5
4
3
2
1
A
B
C
D
E
F
G
H
I
J
K

Fig. 5. Final steps of the control process

Optimal movements of our robots are calculated by applying minimal distance from strategic grid position and rotation penalty . The goalkeeper and attacking player, whose distance is closest to the ball are excluded from strategic movement and their new position are calculated in exact coordinates (see Fig. 5).

3 Conclusion

The main goal of the control system is to enable immediate response in the real time. The system response should be shorter than time between two frames from camera [1]. When the time response of the algorithm exceeds this difference the control quality deteriorates. The method we described provides fast control. This is achieved by using rules that are fast to process. We have described a method of game and strategy representation.

In future work we want to use observation for learning game strategy [4,5]. Our goal is to learn an abstract strategy. The main steps of the learning process are:

- Transformation of observations into virtual grids.
- Transformation of observations into strategy grids.
- Learning a strategy based on the observed transitions between the strategies grids.

We believe that the possibility of learning the game strategy that leads to a fast control is critical for success of the robotic soccer players. Like in chess playing programs, the database of game strategies along with the indication of their success can be stored in the database and can be used for subsequent matches.

Acknowledgements

The Czech Academy of Sciences partially supported this project within the grant Strategic control of the systems with multiagents No.1ET101940418 .

References

1. Bernatík, R., Horák, B., Kovář, P.: Quick image recognize algorithms. In: Proceeding International workshop Robot-Multi-Agent-Systems R-MAS 2001. VSB Ostrava 2001, Czech Republic, p. 53-58
2. Horák, B., Obitko, M., Smid, J., Snášel, V.: Communication in Robotic Soccer Game. The 2004 International Conference on Communications in Computing CIC 2004/PSMP: June 21-24, 2004, Las Vegas, Nevada, USA, CSREA Press, p. 295-301
3. Kim, J., Kim, D., Kim, Y., Seow, K.: Soccer Robotics (Springer Tracts in Advanced Robotics), Springer-Verlag, 2004
4. Obitko, M., Snasel, V.: Ontology Repository in Multi-Agent System. IASTED, International Conference on Artificial Intelligence and Applications (AIA 2004), Innsbruck, Austria, 2004
5. Smid, J., Obitko, M., Snášel, V. (2004): Communicating Agents and Property-Based Types versus Objects. Sofsem MatfyzPress 2004
6. Srovnal, V., Pavliska, A.: Robot Control Using UML and Multi-agent System. In: Proceeding 6th World Multiconference SCI 2002. Orlando, Florida, USA, p. 306-311

Rete Algorithm Applied to Robotic Soccer

M. Palomo[1], F.J. Martín-Mateos[2], and J.A. Alonso[2]

[1] Department of Computer Languages and Systems, University of Cádiz,
Escuela Superior de Ingeniería, C/ Chile, s/n. 11003 Cádiz, Spain
`manuel.palomo@uca.es`
[2] Computational Logic Group,
Dept. of Computer Science and Artificial Intelligence, University of Seville,
E.T.S.I. Informática, Avda. Reina Mercedes, s/n. 41012 Sevilla, Spain
{`fmartin, jalonso`}`@cs.us.es`

Abstract. This article is a first approach to the use of *Rete* algorithm to design a team of robotic soccer playing agents for *Robocup Soccer Server*. Rete algorithm is widely used to design rule-based expert systems. Robocup Soccer Server is a system that simulates 2D robotic soccer matches. The paper presents an architecture based on *CM United* team architecture for Robocup Soccer Server simulation system. It generalizes the low-level information received by the agent as high-level soccer concepts. This way it can take advantage of expert system design techniques.

1 Introduction

Robotic soccer is one of the most interesting examples of multi-agent systems. In this environment, agents must be able to perform as a team to get a common long-term goal. They have to manage themselves in a real-time, nondeterministic, partially-known world. All this facing a team whose goal is the opposite (both teams can't fulfil their goals at the same time.) These features are common to other problems like hospital or factory maintenance, search and rescue missions, multi-spacecraft space missions, etc.

This paper is focused on simulated robotic soccer. The simulation system allows to deal with high-level problems instead of spending time with low-level details. Scientists can apply automated reasoning techniques (off-line and online.) It also offers the advantages of software utilities: recording matches (as video or log files) and replaying them later, playing matches in any moment just with a computer (if the implementations of the teams are available), building teams with exactly the same resources, etc. Anyway, under some constraints, direct implementation of simulated soccer algorithms in real robots is possible [2].

2 Foundations

This paper is a first approach that shows how expert system design techniques (in particular, *Rete* algorithm) can be used to design a soccer playing agent for *Robocup Soccer Server*.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 571–576, 2005.
© Springer-Verlag Berlin Heidelberg 2005

2.1 Expert Systems

Our proposal is based on the use of Rete algorithm [1] to design a rule-based expert system. Expert systems design is a branch of artificial intelligence specialized in the development of systems simulating expert human decision skills. In particular, rule-based expert systems are composed of a set of facts (knowledge base) and a set of if-then statements (rules.) Facts represent information known (or believed) about the world. And rules describe how that information could change. The set of rules are fixed during all the execution of an expert system, but facts usually change. Every rule has two parts: the antecedent (a set of conditions about facts) and the consequent (a set of actions that modify the knowledge base adding or deleting facts.) One of the key features of rule-based expert systems is the possibility of increasing the knowledge (adding new rules) without losing the information in the knowledge base.

Expert systems work in an infinite loop. In each cycle they check the antecedent of every rule. If all the conditions of one of them are true then the rule is activated and placed in the *agenda*. The agenda is the list of all rules which have their conditions satisfied and have not been executed yet. After checking all the rules, one of the activated ones (that are in the agenda) is selected to be triggered. The selection method is called *conflict resolution strategy*, and depends on the implementation of the system. There are several conflict resolution strategies, such as assigning a priority to every rule, choosing the most recent activated, the least frequently triggered, etc. Each one of these strategies has both advantages and drawbacks.

In every cycle of simulation the system checks the rules seeking for new activations, as the triggered rule could have changed the facts in the knowledge base. A direct implementation of an algorithm checking all the conditions of every rules would be too inefficient ($O(R \cdot F^P)$ being R the number of rules, F the number of rules in the knowledge base and P the average of conditions per rule.)

Rete algorithm takes advantage of two facts (as pointed in [5]). First, most of the facts in an expert system don't change from one cycle to another. Thus, it doesn't check all the rules every cycle, but it remembers past facts test results, so only new facts are tested. And second, as several rules can share part of their antecedents a network is created to minimize the number of tests to be made in each cycle. Figure 1 shows an example of optimization of two rules. These advantages produce a more efficient algorithm on the average $O(R \cdot F \cdot P)$.

2.2 Robocup Soccer Server

Robocup Soccer Server [3] is a system that simulates robotic soccer matches in a 2D field. It's supported by the *Robocup Federation* [4] and used in all its competitions.

The code is distributed under the GNU GPL license, and it works with a client-server architecture. In each match there is one server simulating everything concerning the match, and 22 clients (11 for each team), each one controlling

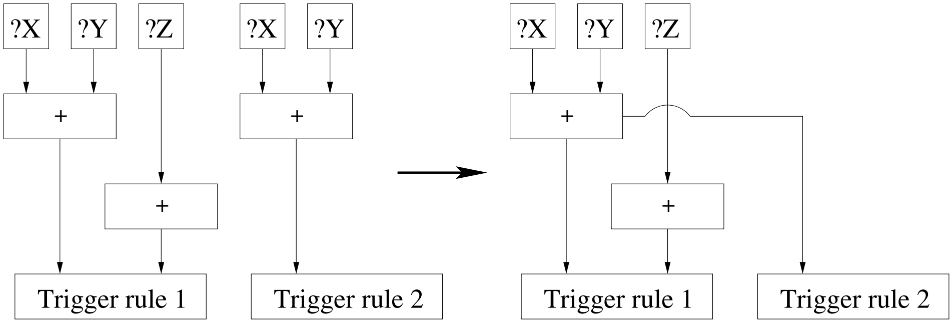


Fig. 1. Rule optimization creating a network sharing facts and an inner condition

one player. Clients are autonomous, and they can only communicate with mates through the server, sending messages in a standard language through an UDP socket. So clients can be programmed in any hardware, any operating system and any programming language as long as it implements UDP socket support. The server implements a low bandwidth, unreliable, communication channel.

The server simulates the world in steps of 100 ms. It accepts one action (like kick or turn) from each client every step, and simulates them all (applying real physics and soccer rules like decay, off-side rule, etc.) The simulation is non-deterministic and all the actions are affected by some noise.

Part of the result of this simulation is sent to every client in each cycle. That information is of three kinds: aural (what a player hears), visual (what a player sees) and physical (what a player knows about himself, like stamina, speed and so.) Every player only receives the information it senses: messages heard, objects seen and physical information.

3 Proposed Architecture

Our proposal is based on the architecture presented by Peter Stone in [2] for *CM United* team. It, basically, uses some static information common to all the players in a team (called *locker-room agreements*) and the information received from the world (partial information) to update the internal state of the agent (Fig. 2.) With those internal beliefs a directed acyclic graph (known as *external behaviors*) selects the action to be made (an example is shown in Fig. 3.)

We propose two modifications to this model:

3.1 Generalization of the Information

In the CM United architecture all the information received from the world is stored as low-level facts (like positions of players, speed of the ball, etc.) But it can be generalized as high-level information relative to objects in the field, in such a way that small changes in the world won't lead to changes in the

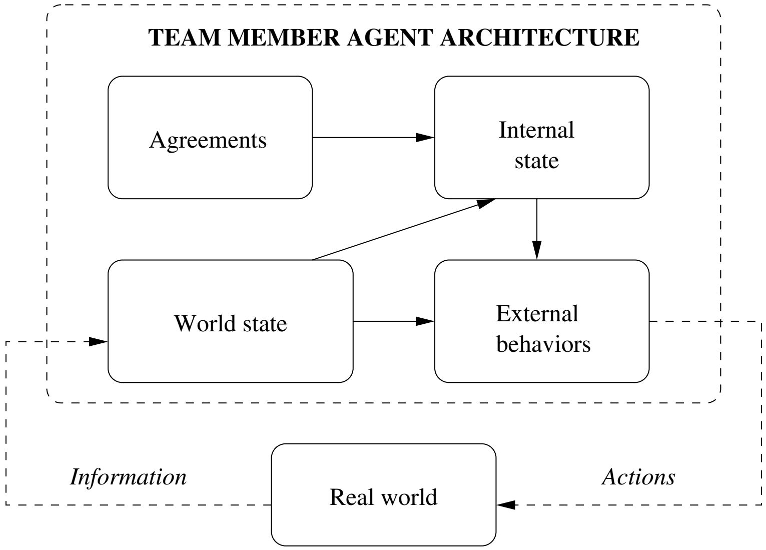


Fig. 2. Simplified internal architecture of an agent

high-level perception of it. Of course, this generalization must be according to soccer concepts: for example, a set of positions of players can be generalized in an "off-side" fact.

This generalization allows the agent to focus on soccer concepts instead of coordinates in a field. And these concepts will only change when an important event (from a soccer point of view) happens.

3.2 Use of a Rule-Based Expert System

The second modification proposed consists in replacing the directed acyclic graph that chooses the action to be performed in the CM United architecture by a set of rules managed with Rete algorithm. In this case the antecedents of the rules must only check high-level properties (so the algorithm won't have to be re-testing a lot of rules every cycle), and the consequent (that indicates the action to be made) uses low-level facts in order to calculate the best action according to accurate information. According to the off-side example, if "off-side" fact exists in the knowledge base, rules as "pass forward" can't be activated. And a slight movement of the player to another off-side position won't lead to the activation of any of those rules. But if the movement avoids off-side, the rules could be activated.

The most suitable conflict resolution strategy is priority-based: assigning a priority value to every rule and selected the activated rule with the highest priority. In the case that several activated rules have the highest priority value, a random selection is performed. This way the agents will be able to take good

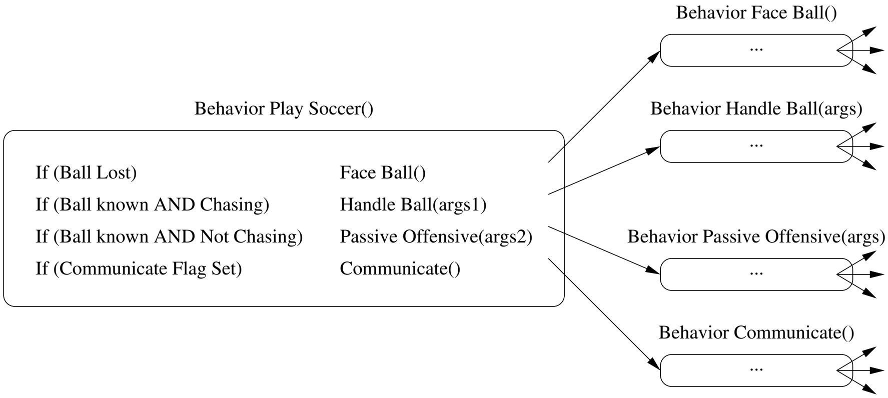


Fig. 3. Example of external behaviors as implemented in [2]

actions (as they have higher priority.) But they won't behave always the same, as in the same circumstances they could take different rules with the same priority.

3.3 Advantages

The modified architecture provides some very interesting advantages:

1. The team can be easily debugged. With a log file of a match (and some help from an engineer) a human expert could check if the concepts generalized in each moment and the actions taken are right, and improve the team. This task is not something obvious in other mathematical approaches to the design of playing agents.
2. Increasing or modifying the system knowledge is as simple as adding or editing rules.
3. Rete algorithm is widely used (there are several free implementations available) and has proved to be very efficient.
4. As Rete minimizes the number of tests to be made in each cycle, the number of rules could be high.
5. Our proposal defines an architecture, not a team. Different concept generalization and different rules design absolutely different teams with absolutely different behavior.

4 Conclusions and Future Work

Robocup Soccer Server is a very interesting test-bed for soccer playing multiagent systems. This paper is a first approach that shows how expert system design techniques can be applied to design a soccer playing agent for Robocup Soccer Server. One of the main contributions is the generalization of the low-level information received as soccer concepts. With those soccer concepts an expert human could program the agent with simple if-then rules. The second contribution is the introduction of Rete algorithm in the architecture used by the CM United team. That way the rules could be easily processed by the agent.

Anyway there are several details that have to be precised to finish this work:

The use of communication protocols is very important for two reasons. First agents can help each other sharing information known about the world, and second, they need to synchronize periodically (in case they dynamically change the tactic of the team or the roles played by each agent in the tactic.) So the communication protocol should be defined carefully. There are three possibilities to do it: agents can use it on their own automatically (without human control), it can be controlled only by rules, or an hybrid method could be implemented.

The inclusion of *set-plays* can be very interesting. A set-play is a multi-agent plan fired by some condition. Their main advantage is that during their execution agents know where mates must be, so they can act faster and more accurately. An interface to define set-plays would consist of three parts: definition of the activation conditions, specification of the actions to be taken during its execution and definition of the conditions to cease it.

Robocup Soccer Server allows the use of an on-line coach in each team during the match to assist players. It's a privileged agent that receives all the information of the environment and can send messages to players periodically. It works as a soccer coach: analyzing the game and the opponent, and sending information to players. It could be implemented in lots of different ways, as long as it send the players interesting facts for them to play better.

Finally we are developing a batch file to test teams. It will be used to test our final architecture (implementing different sets of rules and generalizations). The batch file will program as many matches as desired versus some of the top-level teams in previous *Robocup World Championships* and will collect the results.

References

1. Forgy, C. Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem. *The Journal of Artificial Intelligence*, Vol. 19, 1982, pp. 17-37.
2. Stone, P. *Layered Learning in Multi-agent Systems*. PhD thesis, School of Computer Science, Carnegie Mellon University, 1998.
3. Robocup team. Soccer Server System. *http://sserver.sourceforge.net.*
4. Robocup Federation home page. *http://www.robocup.org.*
5. Giarratano, J. & Riley, G. *Expert systems: principles and programming*, third edition. PWS Publishing Company, 1998.

Towards a Biomathematical Model of Intentional Autonomous Multiagent Systems

J. Pfalzgraf and B. Mitterauer

Department of Computer Science,
Institute of Forensic Neuropsychiatry, University of Salzburg
{jochen.pfalzgraf, bernhard.mitterauer}@sbg.ac.at

Abstract. The objective of this contribution is to establish a 1-1 correspondence between an existing biological model of so-called tripartite synapses in biological brains and a corresponding autonomous multiagent system (MAS) in an unexplored environment. The logical part of the mMAS model is based on the concept of logical fiberings - systems of distributed logics for MAS. Two important notions, intention and rejection, will be subject of the development of a suitable mathematical formalization - the notion of space- and time dependent logical formulas will play a basic role. Our new general model for MAS is based on category theory and general categorical semantics. Of further basic interest in MAS is the issue of learning with respect to artificial neural network applications to agent systems and robotics. According to the cybernetic principle of feasibility, it is in the center of our work to achieve implementations of autonomous robots based on the proposed biomimetic MAS model.

1 Introduction

In this contribution we give a brief account of some existing work, work in progress and intended future work. Our overview deals with the following topics and aspects. A first objective of our work is to establish a 1-1 correspondence between an existing biological model of so-called tripartite synapses in biological brains and a corresponding autonomous multiagent system (MAS) in an unexplored environment. This correspondence can be formalized as a bijective mapping from the biological system TSy to a mathematical model of the corresponding multiagent system mMAS. The direction TSy to mMAS is the mathematization of the biological model and the reverse direction mMAS to TSy can be interpreted as the derivation of a biotechnical model of the underlying MAS. The logical part of the mMAS model is based on logical fiberings - systems of distributed logics for MAS. Two important notions, intention and rejection, will be subject of the development of a suitable mathematical formalization. To this end we can resort to existing work on logical fiberings for cooperating robots and agents which deals with the notion of space- and time dependent logical formulas. This arises in what is called generic modeling principle for a constructive description of scenarios of mobile autonomous agents (robots) where the local logical

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 577–583, 2005.
© Springer-Verlag Berlin Heidelberg 2005

state space of an agent is modeled variably in dependence of the corresponding movements. Concerning questions of automated reasoning and deduction in the logical state spaces of MAS and cooperating robots we can apply methods from symbolic computation and computer algebra in the framework of logical fiberings and many-valued logic. Our new general model for MAS is based on category theory and general categorical semantics. Of further basic interest in MAS is the issue of learning with respect to artificial neural network applications to agent systems and robotics. We use a new geometric, categorical modeling approach where learning can be formally interpreted in terms of morphisms. According to the cybernetic principle of feasibility, it is in the center of our work to achieve implementations of autonomous robots based on the proposed biomimetic MAS model.

2 On Multiagent Systems (MAS) Modeling

From a general point of view we would like to state that many systems in real world have the nature of a MAS - possibly many units ("agents") communicate, cooperate, interact to perform a mission. A physicist might tend to speak about many particle systems with interaction. The textbook [15] provides a good introduction to MAS. There is a remark that so far there is no general, unique (commonly used) definition of a MAS.

At the conference InterSymp-2004 in Baden-Baden, we presented first steps towards a unifying mathematical modeling approach for MAS based on category theory (CAT) (cf. [13]), it shall provide a broad basis of future work. In particular, it is our hope to be able to exploit categorical construction principles for MAS scenario constructions. Subsequently, we briefly recall some basic notions from the beginning of that work.

Category Theory (CAT) is a very abstract and unifying mathematical language with constructive elements. It generalizes and unifies basic notions and operations of many mathematical disciplines on a common linguistic basis in an "economic" way. CAT has a broad area of applications in mathematics, computer science, and artificial intelligence.

A *Category* consists of *Objects* A,B,.... and *Morphisms (Arrows)* $A \xrightarrow{f} B$,...., $X \xrightarrow{g} Y$, a *Composition Operation* on morphisms, i.e. for $f : A \to B$ and $g : B \to C$ there is a morphism $g \circ f : A \to C$, the composition of f and g. The composition of morphisms is *associative*, that is $h \circ (g \circ f) = (h \circ g) \circ f$ holds for composable morphisms. For every object A there is the *identity morphism* id_A fullfilling the propreties: $f \circ id_A = f$ and $id_B \circ f = f$, for all $f : A \to B$. We underline here that the arrow notation in CAT is of elementary importance. Well-known classical examples of categories in mathematics are, among others, **SET**, **GROUP**, topological spaces, vector spaces, etc.. For our work with CAT it is of elementary importance that all kinds of relational structures and arrow diagrams can be interpreted categorically (this has been pointed out first in [8]).

Introduction of the Category "MAS" - first step: *Objects* are AGENTS, where we use "Typed Agents" - Types of Agents are described by properties.

Morphisms (Arrows) are RELATIONS between agents, where we use "Typed Morphisms" - Types of Relations are specified.

Communication between agents is modeled via morphisms (*typed arrows*) between agents, specifying a corresponding *"communication type"*.

A *Subsystem* of a MAS - i.e. a subgroup of (cooperating) agents can be interpreted categorically as a corresponding *Subcategory.*

3 The Tripartite Synapse Model

According to the prevailing view, chemical synaptic transmission exclusively involves bipartite synapses consisting of presynaptic und popstsynaptic components and a synaptic cleft, in which a presynaptically released neurotransmitter binds to cognate receptors in the postsynaptic cell. However, there is a new wave of information suggesting that glia, especially astrocytes, are intimately involved in the active control of neuronal activity and synaptic transmission.

Model of a Cholingergic Tripartite Synapse: Smit and coworkers (cf. [14]) proposed a model of a cholinergic tripartite synapse that might turn out to be a milestone for our understanding of the glial-neuronal interaction. But first let us shortly describe this type of tripartite synapse These authors identified a glia-derived soluble acetylcholine-binding protein (AChBP), which is a naturally occurring analogue of the ligand-binding domains of the nicotinic acetylcholine receptors (nAChRs). Like the nAChRs, it assembles into a heptamer with ligand-binding characteristics typical of a nicotinic receptor. Presynaptic releases of acetylcholine induce the secretion of AChBP through the glial secretory pathway, and once in the synaptic cleft, it acts as a molecular decoy, binding the transmitter und reducing its availability at the synapse.

This model, which focuses on the role of AChBP in neurotransmission, suggests that there is a basal level of AChBP in the synaptic cleft, maintained by continuous release from the synaptic glial cells. Under conditions of active presynaptic transmitter release, high millimolar concentrations of free ACh will probably activate both postsynaptic receptors and nAChRs on the synaptic glial cells, which would enhance the release of AChBP, thus increasing its concentration in the synaptic cleft. This may either diminish or terminate the ongoing ACh response or raise the concentration of basal AChBP to the extent that subsequent responses to ACh are decreased.

Biocybernetic Model of a Tripartite Synapse: A simple biocybernetic model of a tripartite synapse could be helpful for interpreting an elementary reflection mechanism. Generally, a living system like man is endowed with intentional programs (hunger, desires, etc.) that strive for realization in the environment (cf. [2]). This intentional relationship of a living system with its environment can be described as an elementary behavioral cycle (cf. [4]). Information from the environment actualizes an intentional program. If a living system is able to find appropriate objects for realizing a specific intentional program in the environment, then the cycle is closed, comparable to an experience. A negative feedback mechanism breaks off the information processing.

Such elementary behavioral cycles may also control the information processing in tripartite synapses. The production of neurotransmitters in the presynapse can be interpreted as "environmental information" stimulating the expression of glial binding protein (GBPs) in an astrocyte. GBPs may embody an "intentional program" that is ready for occupancy by an appropriate neurotransmitter. If an appropriate occupancy occurs ("realization of an intentional program"), the glial system negatively feeds back this "experience" to the presynapse. In parallel, this synaptic experience is transmitted to other cells in the glial-neuronal networks by occupancy of postsynaptic receptors ("information transmission"). Now, the cycle can start again.

But what makes astrocytes so intentional ? In a series of papers, we have hypothesized that the glial system has a spatio-temporal boundary-setting function in its interaction with the neuronal system (cf. [7,3,4,5,6]). With respect to a tripartite synapse, this would mean that astrocytes control synaptic information processing by setting temporal boundaries dependent on the occupancy of GBPs. In that case, GBPs embodies an essential parameter of synaptic information processing.

4 On MAS, Logical Fiberings, and Rejection

The concept of logical fiberings was introduced by the first author about 15 years ago, inspired by a project on Gotthard Günther's polycontextural logic (cf. [1]). The second author knew G.Günther very well and he calls him a philosopher of cybernetics. Recent brief introductions and summaries of the basic notions and notation of logical fiberings can be found in [12,10]. Logical fiberings are systems of distributed logics - a fiber is a "local logic" over a base point (index) of the base space of the whole fiber bundle. For the convenience of the reader we recall only a few mathematical notions below.

A general (abstract) fibering (fiber bundle) (E, π, B) consists of a base space B, total space E and (projection) map $\pi : E \to B$. For an Element $b \in B$ the preimage set $\pi^{-1}(b) = \{x \in E \mid \pi(x) = b\}$ is called the fiber over b. A *Logical Fibering (LogFib)* is a fiber bundle (E, π, I) where every fiber L_i over $i \in I$ is a logical space: $\pi^{-1}(i) = L_i$ (*"local logic"*), usually a classical 2-valued logic (local logical system over base point (index) i with local truth values $\Omega_i = \{T_i, F_i\}$). Technically possible: "Mixing" of Logics". Very shortly expressed we can say *LogFib = system of distributed logics* over index set (base space) I.
"Logical Communication" between fibers - by fiberwise morphisms (e.g algebra homomorphisms in case of Boolean algebras as fibers) - corresponds to "transportation" of logical information (e.g. formulas) between fibers.

For logical modeling of MAS we assign a logical fibering to a multiagent system in such a way that every agent "i" has its own local logic (fiber) L_i, then the collection of all fibers will be interpreted as the global logical state space E of the MAS. All this data and structure together forms the logical fibering (fiber bundle) of the whole MAS.

In [9], besides a brief introduction to logical fiberings and fibered logical controllers for robots, the notion of *space- and time dependent (more generally, base point dependend) formulas* is briefly sketched. An earlier introduction can be found in an article on logical modeling of robotics scenarios in [8] (cf. also [10]). The basic idea and motivation for introducing this concept can be described as follows. Considering cooperating agents (e.g. robots) which are mobile, their individual logical state space can vary in dependence of space and time - this seems to be a natural point of view. Thus it is realistic to model logical formulas having evaluation behavior depending on an index set (base space in logical fiberings), e.g. this can be a coordinate system of a certain workspace or time intervals. A mathematical approach has been introduced making formulas evaluable or not ("switched on" or "switched off") depending on such a base space. In various interesting discussions we found out that this approach is naturally suited to formalize mathematically the concept of *rejection* applied by agents having intentions. An intention can have to be rejected by an agent depending on his mission and the environment, among others. As a short remark, it seems to be reasonable to formalize in a comparable way the notion of *intention* on basis of a set of executable actions that an agent is able to perform.

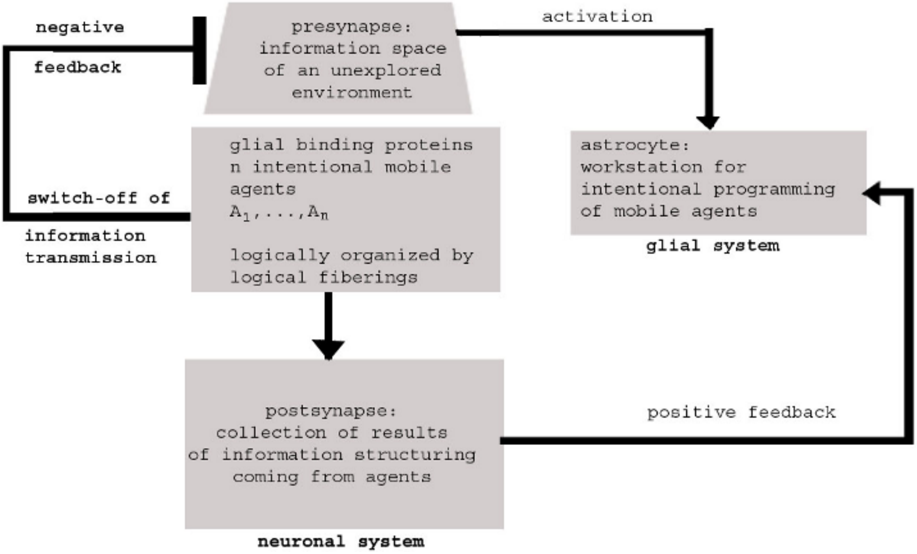


Fig. 1. The 1-1-Correspondence

Concluding, we point out the significance of the use and deployment of *connectionist networks* and the instance of neural network *learning* for modeling learning agents and learning in MAS, in general, including the important aspect of self-organization. A general mathematical approach for modeling neural network structures and learning based on category theory and the notion of a geometric network is introduced in [11] - very much interesting and challenging work in these directions can still be done.

5 The 1-1-Correspondence

Shortly speaking, between presynapse and postsynapse a process takes place that can be interpreted in terms of a multiagent system. More details follow subsequently. The principle of information structuring via mobile (artificial, technical) agents $A_1, ..., A_n$: Agents explore the so far unexplored environment driven by programs that represent intentions of human agents. The ultimate goal of these human agents is to verify the feasibility of the intentional programs implemented in the system of artificial mobile agents. Mathematically and logically the agent sy¡stem $A_1, ..., A_n$ is modeled on basis of the concept of logical fiberings.

The role of rejection: 1st, if an agent is incapable to retrieve processable information, then its functionality is switched off by self-rejection within the system mMAS. Processable is defined as information appropriate to verify the intentional program. 2nd, a successful information processing by an artificial agent results in a break-off of further information processing, coming from the environment, such that a finite information structuring is possible in a period of operating time.

Future projects and prospects of work: Future robotics research will be increasingly based on biomimetic technical systems. This will be of great relevance for future service robots construction. In our opinion, the big challenge we are faced with is to implement "a touch of subjectivity" according to the principles we proposed, intentional programming and operating with rejection operators.

References

1. G. Günther. Cybernetic Ontology and Transjunctional Operations. *Biological Computer Lab. Publ., Vol.68 (Urbana, Ill.)*, publ. in Self-organizing Systems 1962, Spartan Books, Washington, D.C., pp. 313-392, 1962.
2. W. McCulloch and A. Iberall. The organizing principle of complex living systems. *Transactions of the ASME*, 6:290–294, 1969.
3. B. Mitterauer. An interdisciplinary approach towards a theory of consciousness. *BioSystems*, 45:99–121, 1998.
4. B. Mitterauer. Some principles for conscious robots. *Journal of Intelligent Systems*, 10:27–56, 2000.
5. B. Mitterauer. The loss of self-boundaries: towards a neuromolecular theory of schizophrenia. *BioSystems*, 72:209–215, 2003.
6. B. Mitterauer and C. Kopp. The self-composing brain: Towards a glial-neuronal brain theory. *Brain and Cognition*, 51:357–367, 2003.
7. B. Mitterauer, H. Leitgeb, and H. Reitboeck. The neuro-glial synchronization hypothesis. *Recent Research Development in Biological Cybernetics*, 1:137–155, 1996.

8. J. Pfalzgraf. On a general notion of a hull. In *Automated Practical Reasoning, J.Pfalzgraf and D.Wang (eds.). Texts and Monographs in Symbolic Computation, Springer Verlag Wien, New York*, 1994.
9. J. Pfalzgraf. On geometric and topological reasoning in robotics. *Annals of Mathematics and Artificial Intelligence*, 19:279–318, 1997.
10. J. Pfalzgraf. The concept of logical fiberings and fibered logical controllers. In *Proceedings CASYS'2000, August 7-12, 2000, Liège, Belgium. American Institute of Physics, AIP Conference Proceedings, Vol.573 (2001), pp.683-693. D.M.Dubois (Ed.)*, 2000.
11. J. Pfalzgraf. Modeling connectionist networks: categorical, geometric aspects (towards 'homomorphic learning'). In *Proceedings CASYS'2003, August 11-16, 2003, Liège, Belgium. American Institute of Physics, AIP Conference Proceedings, Vol.718 (2004). D.M.Dubois (Ed.)*, 2004.
12. J. Pfalzgraf. On Logical Fiberings and Automated Deduction in Many-valued Logics Using Gröbner Bases. *Revista Real Academia de Ciencias, Serie A de Matemáticas, RACSAM*, 98, 2004.
13. J. Pfalzgraf. On categorical and logical modeling in multiagent systems. In *Anticipative and Predictive Models in Systems Science, Vol.1, pp.93-98. George E.Lasker and Daniel Dubois (Eds.), ISBN 1894613-49-X, published by the IIAS, Windsor, Ontario, Canada*, 2005.
14. A. Smit, N. Syed, and D. Schaap. A glial-derived acetylcholin-binding protein that modulates synaptic transmission. *Nature*, 411:261–268, 2001.
15. M. Wooldridge. An Introduction to Multiagent Systems. John Wiley & Sons Ltd, 2002.

A Controller Network for a Humanoid Robot

Peter Kopacek, Edmund Schierer, and Markus Wuerzl

Vienna University of Technology,Institute for Mechanics and Mechatronics,
Department of Intelligent Handling and Robotics, E325/A6,
Favoritenstrasse, 9 – 11, 1040, Vienna
Tel.: +43 1 58801 31815, FAX: +43 1 58801 31899
{`kopacek, schierer, wuerzl`}`@ihrt.tuwien.ac.at`

Abstract. In this paper we describe the design and construction of a distributed controller network for the humanoid robot "Archie". The mechanical design of the robot incorporates 6 DOF per leg, 6 DOF per arm and 3 DOF for the torso and the head respectively. The network consists of 3 subnetworks, each including a set of dedicated processing nodes interconnected by a CAN bus. This partitioning into levels of competence guarentees failure tolerance and reliability and minimizes communication overhead even on heavy load conditions.

1 Introduction

Humanoid robots presently are one of the most exciting, most challenging and most attractive research targets in the field of mobile robotics. It covers not only mechanical engineering, but also includes problems from the field of electrical engineering, electronics, computer science, artificial intelligence and more often some domains of psychology.

Mechanical engineering solves questions of design and construction, mechanical loads and properties of used materials whilst electrical engineering gives answers to problems like drive systems and components, power supply and interfacing high power devices to processors and controllers. Computer science on the other hand provides algorithms for motion control, feedback control systems, pattern recognition, image processing and so forth. Artificial intelligence as a special application in the field of computer science may be used for high-grade processing of various sensory inputs and environmental influences to perform a more sophisticated and convincingly behaviour.

One of the keyroles in such a complex system like a humanoid robot is the motion control and balancing system to guarentee smooth and life-like trajectories.

2 Robot Specification

The basic design of a humanoid robot follows the "well proven design" of the human body. That is to say, it has 2 legs, 2 arms, a torso and a head in well-balanced proportions. Altough it is presently infeasible to implement the high

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 584–589, 2005.
© Springer-Verlag Berlin Heidelberg 2005

number of DOF of a human being in a robot, there *is* a certain limit a humanoid must have to satisfy fundamental requirements concerning motion and balancing. A robot like HRP-2 [1] (Fig.1) or PINO [2] (Fig.2) has typically about 30 DOF what is in our considered opinion the minimum for a lifelike and natural locomotion.

It must be tall enough to operate without restrictions of any kind in environments designed for humans. That is, it must not be smaller than 120 cm. As mentioned above a robot generally should have anthropomorphic properties in size, proportions and weight. Just the legs should have a slim design to allow one leg to be put in front of the other (imagine a robot walking on a narrow plank).

Furthermore it has to be robust and reliable, exceptionally in a research environment, where control algorithms are developed and tested on this robot.

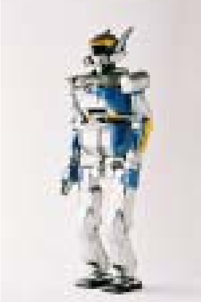


Fig. 1. Robot HRP-2

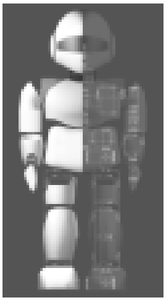


Fig. 2. Robot HRP-2

3 Mechanical Design Considerations

When designing a robot various mechanical constraints have to be incorporated. The most important are:

- Size of the robot
- Number of DOF
- Type of actuators (electric motors, hydraulics, pneumatics, ...)
- Weight of the robot

Our robot is 120 cm tall, has a total number of 36 DOF, is driven by DC motors coupled to harmonic drives and weights approximately 20 kg.

4 Drive System, Gears, Sensors and Control System

Basically there are more or less 3 types of actuators available: hydraulics, pneumatics and electric motors. Hydraulic has its advantages, but due to its heavy weight it is unusable in such a robot. A pneumatic actuator, on the other hand, is light weight, but its power to weight ratio is insufficient and a compressed air machine is too bulky and heavy.

A robot definitely needs information about the outer world but sensorial inputs regarding its internal states is of particular importance (e.g. capacity of batteries, joint positions, orientation and acceleration of the body).

4.1 Sensors

Our robot incorporates both internal and external sensors. Whereas the need for external sensors is obvious the additional demand for internal sensors is not intuitive plausible. But for a precise locomotion he definitely needs information of current joint positions and velocities, torques, contact forces to ground, temperature conditions inside its structural components and so on. Another important data is the state of charge of the battery, which shall prevent the robot being lost due to "low battery" far away from its charging station. A fact to be reckoned with is self-collision. Without any information about joint angles and position it is impossible for a robot to move its limbs in a way that they do not collide with other parts of the body.

4.2 Control System

Each motor drive combination is controlled by its own local processor. This enables us to use cheap, small and dedicated microcontrollers. If there are increasing requirements in the future it is quite simple to replace the controller with a better one without any influence to the remaining network. The algorithm used in such a single node is a standard PID with special consideration of the coefficients of friction. The master control processor [Fig.3] propagates a complete parameter set for each joint controller each 5 ms. Simulations has proven this time interval to satisfy almost any of the conditions for locomotion and balancing such a robot.

5 Controller Network

Figure 3 shows an overview of our approach of a distributed controller network for a humanoid robot. Each box shown is a single processing node dedicated to a special task. The bus that interconnects the nodes is a standard fieldbus, in our case CAN [3].

5.1 Layered Structure of the Network

The network (Fig. 3) forms a layered structure to minimize communication overhead and heavy load conditions. We decided to limit the number of levels to three which seems to be a good compromise. Level 1 performs all the high-grade processing like gait and trajectory generation, balance control, vision and image processing as well as speech recognition and synthesis. The limbs of the robot are controlled by their appropriate LCPs which form the level 2 subnetwork. Level 3 includes all the single joint controllers with their local sensors.

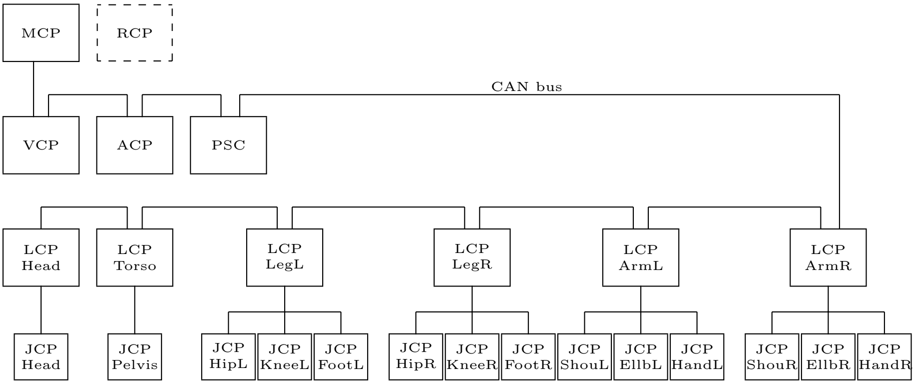


Fig. 3. Controller network architecture of the humanoid robot "Archie"

5.2 The Processing Nodes

The Master Control Processor (MCP). Based on external optical or tactile sensor data the MCP calculates every 5 ms all the trajectories for the several joints of the robot. The outcome of this process is then propagated over the network to the corresponding LCPs.

The Radio Control Processor (RCP). The RCP is a small radio module used for remote control and maintenance. It has a standard serial interface and is directly connected to the MCP

The Vision Control Processor (VCP). Stereo vision and image processing with its high demands on computing power requires a very powerful processor. The VCP is a DIMM-PC based on ARM technology with 500 MHz, 64 MB RAM and 16 MB Flash. 2 commercially available CMUcam2 camera modules are used as "eyes".

The Audio Control Processor (ACP). The ACP currently only supports speech *output* via a standard phonem synthesizer (SSI263) controlled by a quite simple Basic Stamp 2 micro controller.

The Proprioceptive Sensor Control (PSC). The PSC subsystem incorporates all the sensors for measuring internal states like body acceleration, orientation and posture, ambient temperature and charge condition of the battery. These informations are fed into the MCP and used to retain stability. Due to the complex mathematical operations for calculating the robot's center of mass depending on the joint positions and arrangement of the several limbs a second PSC could be added.

The Limb Control Processor (LCP). The LCP receives global data sets which specify the future positions the several joints have to reach. It checks these data for validity, that is mainly for self collision based on the position of all relevant limbs and transmits these data to its associated JCPs. In case of a malfunction of the level 1 network all the LCPs can run in emergency mode. In this mode the robot can continue walking with some restrictions in smoothness and walking speed.

The Joint Control Processor (JCP). The JCPs perform the control of the motors and sensors applied to their joints. Though there are 3 different types of joints depending on the DOF we developed a very small (36 mm by 36 mm) universal controller board for the motor control application (fig.4).

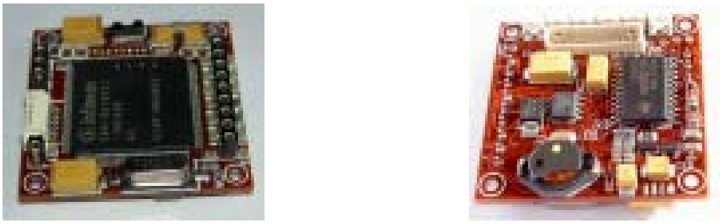


Fig. 4. The motor controller board (left: upper side, rigth: lower side)

5.3 The CAN Bus

The CAN bus is a widely used fieldbus especially designed for automotive applications with their stringent requirements concerning pollution control and reduced fuel consumption.
Some of the most important features of CAN are

- high-integrity serial data communications bus for real-time applications
- data rates of up to 1 MBits/s
- excellent error detection and confinement capabilities
- multi-master capabilities

Another advantage is the availability of highly integrated, powerful controllers with integrated CAN interface(s) and good software support even with public domain compilers.

6 Design Philosophy

The philosophy behind designing such a distributed network is to improve the reliability of the control system for a humanoid robot. Loss of reliability of the computing system is intolerable even for humanoid robots acting in environments populated by humans. A minimum functionality can only be guarenteed by observing the below-mentioned requirements.

Modular Design. Instead of a single but powerful processor which performs all the necessary functions in one, split it up into several dedicated hardware modules and distribute the processes to these specialized units. If one or more fails (down to a certain limit) the remaining nodes can continue working. If there is need for more computing power or adding new sensors (e.g. a compass, a laser range scanner, force and torque sensors, ...), it is quite easy to integrate additional units.

Keep it cheap and simple. High-performance computing systems are expensive. Dividing such a single system into cheaper parts can remarkably decrease the overall system costs. On the other hand it is often easier to implement a dedicated piece of software on its own hardware than writing it all together on one system and considering all the problems which may occur on parallel processing.

7 Conclusion

This paper presented the main components of our humanoid robot "Archie". The focus was on the distributed controller network whereas the mechanical design was only shortly discussed (as far as it was necessary to understand its influence on the electronic part). It illustrated the considerations, strategies and tailor-made solutions to some problems described in the beginning. The major design constraints for a humanoid robot pointed out are first of all its size, degrees of freedom and type(s) of actuators. Type and number of sensors are also of special interest as well as the type of the control system. Last but not least the overall performance of a robot is definitely determined by the type of the processing subsystems and the architecture of the network they are linked with. Our approach is a high number of low cost, yet powerful microcontrollers loosely coupled by a fieldbus organized in 3 layers of competence. Simulation results showed, that this decentralized control system is able to continue its processing even if some nodes fail.

References

1. Humanoid Robotics Project, http://www.plyojump.com/hrp.html
2. Open PINO Platform, http://www.symbio.jst.go.jp/PINO/index.html
3. The Robert Bosch CAN web site, http://www.can.bosch.com
4. BASIC Stamp Microcontrollers, http://www.parallax.com/

Programming by Integration in Robotics*

José L. Fernández-Pérez, Antonio C. Domínguez-Brito, Daniel Hernández-Sosa,
and Jorge Cabrera-Gámez

IUSIANI - Universidad de Las Palmas de Gran Canaria (ULPGC), Spain
{jfernandez, adominguez, dhernandez}@iusiani.ulpgc.es
jcabrera@dis.ulpgc.es

Abstract. This document presents the first operating version of CoolBOT, a component oriented software framework for programming robotic systems. CoolBOT has been designed having in mind the idea of programming by integrating software components, in order to reduce the developing effort typically invested when programming robots. CoolBOT also fosters some interesting features, such as asynchronous execution, asynchronous inter communication, data-flow-driven processing, and cognizant failure systems. A simple demonstrator illustrates the benefits of using the proposed approach.

1 Introduction

Developing and integrating software for controlling robotic systems is costly due to the complexity inherent in these systems. There is a need for tools that permit a reduction in the programming effort, aiming at the generation of modular and robust applications, and promoting software reuse. The techniques which are of common use today in other areas are not adequate to deal with the complexity associated with these systems [1]. Some authors [2][3] have already made similar considerations working on generic programming tools in robotics, such as frameworks, which are neutral in terms of control and system architecture, the contribution presented in this document should be situated following this approach.

In the following sections we will introduce *CoolBOT*, a component-oriented software framework aimed to programming robotic systems based on a *port automata model* [4] that fosters controllability and observability of software components. Thus, in the next section, Sect. 2, a short introduction to the framework will be given, where their main concepts and abstractions will be briefly explained. Next, in Sect. 3 a simple demonstrator is commented in some detail, and finally, in Sect. 4 we will comment some of the conclusions we have drawn from this work.

---

* This work has been partially supported by the research project *PI2003/160* funded by the Autonomous Government of Canary Islands (Gobierno de Canarias - Consejería de Educación, Cultura y Deportes, Spain), and by the ULPGC research projects *UNI2004/11* and *UNI2004/25*.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 590–595, 2005.
© Springer-Verlag Berlin Heidelberg 2005

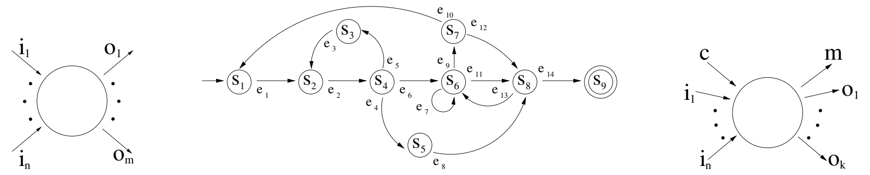
2 CoolBOT

CoolBOT [5] is a C++ component-oriented framework for programming robotic systems that allows designing systems in terms of composition and integration of software components. Each *software component* [6] is an independent execution unit which provides a given functionality, hidden behind an external interface specifying clearly which data it needs and which data it produces. Components, once defined and built, may be instantiated, integrated and used as many times as needed in other systems.

In CoolBOT, components are modelled as *Port Automata* [4]. This concept establishes a clear distinction between the internal functionality of an active entity, an automaton, and its external interface, sets of input and output ports. Fig. 1(a) displays the external view of a component where the component itself is represented by a circle, input ports, i_i, by the arrows oriented towards the circle, and output ports, o_i, by arrows oriented outwards. Fig. 1(b) depicts an example of the internal view of a component, concretely the automaton that models it, where circles are states of the automaton, and arrows, transitions between states. Transitions are triggered by events, e_i, caused either by incoming data through an input port, or by an internal condition, or by a combination of both. Double circles indicate automaton final states. CoolBOT components interact and inter communicate each other by means of *port connections* established among their input and output ports. Data are transmitted through port connections in discrete units called *port packets. Port packets* are also classified by their type, and usually each input and output port can only accept a specific set of port packet types.

CoolBOT introduces two kinds of variables as facilities in order to support the monitoring and control of components: *observable variables*, that represent features of components that should be of interest from outside in terms of control, or just for observability and monitoring purposes; and *controllable variables*, which represent aspects of components which can be modified from outside, in order to be able to control the internal behavior of a component. Additionally, to guarantee external observation and control, CoolBOT components provide by default two important ports: the *control* port and the *monitoring* port, both depicted in Fig. 1(c). The *monitoring* port: which is a public output port by means of which component *observable variables* are published; and the *control* port, that is a public input port through which component *controllable variables* are modified and updated. Fig. 2(a) illustrates graphically a typical execution control loop for a component using these ports where there is another component as external supervisor.

Internally all components are modelled using the same default state automaton, the *default automaton*, shown in Fig. 2(b), that contains all possible control paths that a component may follow. In the figure, the transitions that rule the automaton are labelled to indicate the event that triggers each one, some of them correspond to internal events: *ok*, *exception*, *attempt*, *last attempt* and *finish.* The other ones indicate default controllable variable changes: ns_r, ns_{re}, ns_s, ns_d, *np*, and *nex.* Subscripts in ns_i indicate which state has been commanded: r (*running* state), re (*ready* state), s (*suspended* state), and d (*dead* state). Event



(a) External view.

(b) Internal view.

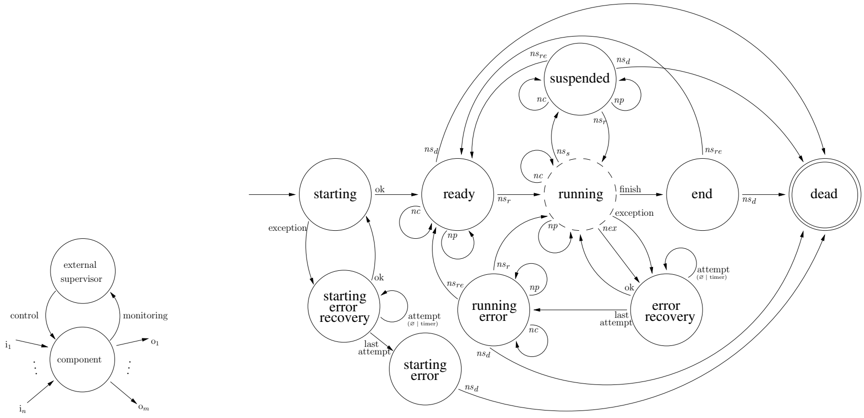
(c) The *control* port, *c*, and the *monitoring* port, *m*.

Fig. 1. Component fundamentals

np happens when an external supervisor forces a priority change, and event *nex* occurs when it provokes the occurrence of an exeption.

The *default automaton* is said to be “controllable” because it can be brought externally in finite time by means of the *control* port to any of the controllable states of the automaton, which are: *ready*, *running*, *suspended* and *dead*. The rest of states are reachable only internally, and from them, a transition to one of the controllable states can be forced externally. Having a look to Fig. 2(b) we can see how CoolBOT components evolve along their execution time. Basically, the *default automaton* organize the life of a component in several phases which correspond to different states: *starting*, devised for initial resource allocation; *ready*, the component is ready for a task execution; *running*, here the component executes its specific task; *suspended*, execution has been suspended temporally; *end*, a task execution has just been completed. Furthermore, there are two pair of states conceived for handling faulty situations during execution which are part of the support CoolBOT provides for error and exception handling. One of them devised to face errors during resource allocation (*starting error recovery* and *starting error* states), and the other one dedicated to deal with errors during task execution (*error recovery* and *running error* states). Moreover, exceptions constitute a useful concept present in numerous programming languages (C++, Java, etc.) to separate error handling from the normal flow of instructions in a program. Importing this concept of exception, a CoolBOT component may define a list of *component exceptions* to signal and handle erroneous, exceptional or abnormal situations during execution.

Analogously to modern operating systems that provide IPC (**I**nter **P**rocess **C**ommunications) mechanisms to inter communicate processes, CoolBOT provides ***I****nter* ***C****omponent* ***C****ommunications* or *ICC* mechanisms to allow components to interact and communicate among them. CoolBOT *ICC* mechanisms are carried out by means of input ports, output ports, and ports connections. There are several types of output and input ports supported by CoolBOT which combined adequately implement different protocols of interaction between components. Specifically the framework offers the following protocols: a protocol for event signaling, an active sender/passive receiver protocol, a passive sender/active receiver protocol, a protocol for sharing memory between components, a protocol for connections transporting packets of multiple types, a sending-with-priority protocol and a request/answer protocol.



(a) A typical component control loop.

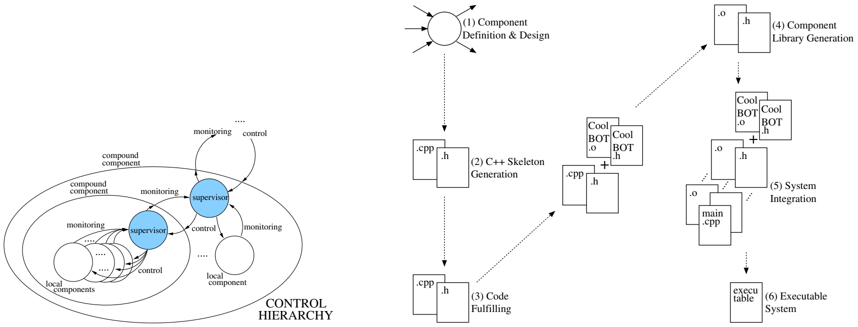
(b) The Default Automaton.

Fig. 2. CoolBOT fundamentals

CoolBOT components are classified into two kinds: *atomic* and *compound* components. *Atomic* components have been mainly devised in order to abstract low level hardware layers to control sensors and/or effectors; to interface and/or to wrap third party software and libraries; and to implement generic algorithms. *Compound* components are compositions of instances of several components which can be either atomic or compound. The functionality of a compound component resides in its *supervisor*, depicted in Fig. 3(a), which controls and observes the execution of its *local* components through the control and monitoring ports present in all of them. The *supervisor* of a *compound* component concentrates the control flow of a composition of components, and in the same way that in *atomic* components, it follows the control graph defined by the *default automaton* of Fig. 2(b). All in all, compound components use the functionality of instances of another atomic or compound components to implement its own functionality. Moreover, they, in turn, can be integrated and composed hierarchically with other components to form new compound components.

2.1 Development Process

The process of developing CoolBOT components and systems is resumed on Fig. 3(b) in six steps. *(1) Definition and Design*: in this step the component is completely defined and designed. This comprises deciding if it is atomic or not, functionality – user automaton–, thread use, resources, output and input ports, port packets, observable and controllable variables, exceptions, timers and watchdogs. *(2) Skeleton Generation*: There is already a small set of developed components, and component examples in the form of C++ classes illustrating the most common patterns of use. It is possible to start from one of them as skeleton, or generate a new one from a component skeleton description language by means of a compiler. *(3) Code Fulfilling*: Using the component´s skeleton



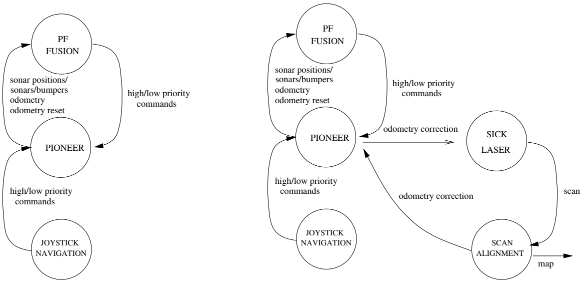
(a) Compound components. (b) The development process.

Fig. 3. CoolBOT fundamentals

obtained in the previous step we complete the component fulfilling its code. *(4) Library Generation*: Then the component is compiled obtaining a library. *(5) System Integration*: Next the component may be integrated in a system alone or with other components. *(6) System Generation*: And finally, the system gets compiled and an executable system is obtained. With it, we can already test the whole integration with our component.

3 A Simple Demonstrator

CoolBOT has been conceived to promote integrability, incremental design and robustness of software developments in robotics. In this section, a simple demonstrator will be outlined to illustrate how such principles manifest in systems built using CoolBOT. The first level of this simple demonstrator is shown in Fig. 4(a) and it is made up of four components: the *Pioneer* which encapsulates the set of sensors and effectors provided by an ActivMedia Robotics Pioneer robot; the *PF Fusion* that is a potential field fuser; the *Strategic PF* component that transforms high level movement commands into combinations of potential fields; and finally, the *Joystick Navigation* component which allows controlling the robot



(a) The avoiding level. (b) The whole system.

Fig. 4. A two level system

using a joystick. The integration shown in the figure makes the robot to avoid obstacles while executing a high level movement command like, for example, going to a specific destination point. The second and last level of our demostrator is depicted in Fig. 4(b). Note that the systems adds two new components, the *Sick Laser* which controls a Sick laser range finder and *Scan Alignment* that performs self-localization using a SLAM (Simultaneous Localization And Mapping) algorithm [7][8].

4 Conclusions

This document describes briefly a first operating version of CoolBOT, a component-oriented C++ programming framework supported under GNU/Linux and Microsoft Windows that favors a programming methodology for robotic systems that fosters software integration, concurrency and parallelism, asynchronous execution, asynchronous inter communication and data-flow-driven processing. The framework also promotes a uniform approach for handling faulty situations.

References

1. Kortenkamp, D., Schultz, A.C.: Integrating robotics research. Autonomous Robots **6** (1999) 243–245
2. Fleury, S., Herrb, M., Chatila, R.: G^{en}oM: A Tool for the Specification and the Implementation of Operating Modules in a Distributed Robot Architecture. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Grenoble, Francia (1997) 842–848
3. Schlegel, C., Wörz, R.: Interfacing Different Layers of a Multilayer Architecture for Sensorimotor Systems using the Object Oriented Framework SmartSoft. Third European Workshop on Advanced Mobile Robots - Eurobot´99. Zürich, Switzerland (1999)
4. Stewart, D.B., Volpe, R.A., Khosla, P.: Design of dynamically reconfigurable real-time software using port-based objects. IEEE Transactions on Software Engineering **23** (1997) 759–776
5. Domínguez-Brito, A.C., Hernández-Sosa, D., Isern-González, J., Cabrera-Gámez, J.: Integrating robotics software. IEEE International Conference on Robotics and Automation, New Orleans, USA (2004)
6. Szyperski, C.: Component Software: Beyond Object-Oriented Programming. Addison-Wesley (1999)
7. Lu, F., Milios, E.: Robot pose estimation in unknown environments by matching 2d range scans. Proc. IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognition, Seattle, USA (1994)
8. Lu, F., Milios, E.: Globally consistent range scan alignment for environment mapping. Autonomous Robots **4** (1997) 333–349

A Mathematical Formalism for the Evaluation of C-Space for Redundant Robots

Roberto Therón, Vidal Moreno, Belén Curto, and Francisco J. Blanco

Departamento de Informática y Automática,
Facultad de Ciencias - Universidad de Salamanca,
Plaza de la Merced, s/n 37008, Salamanca, Spain
`theron@usal.es`, `{vmoreno@, bcurto, jblanco}@abedul.usal.es`

Abstract. This paper presents a new general method for obstacle representation in the configuration space (C-space) for redundant robots. The method is based on the analytical deconstruction of the C-space, i.e., the separated evaluation of the C-space portion contributed by the collisions of each link in the kinematic chain. The systematic application of a simple convolution of two functions describing each link and the workspace, respectively, is applied. In order to do that, the transformation of the workspace among reference systems located at one point of each link is needed; in this step a well-known and sound method is used.

1 Introduction

In this paper the explicit representation of obstacles in the Configuration Space (C-space) of redundant articulated robots is adressed. This representation is widely used in robotics in many tasks that involve the obstacle-avoidance problem as for example path planning or the optimization of robot paths.

The C-space concept [1] is very useful since the robot is representated by a single point at this manifold. The advent of new methods that enable simpler and faster evaluations is a challenge. Consequently, Kavraki [2] suggested the use of the convolution of the obstacles and a mobile robot and the application of the Convolution Theorem to evaluate the discrete C-space. Although, it is doubtless a great advance on the way to optimize the computation time for the C-space evaluation, only 2D mobile robots were considered, thus lacking in generality.

While a lot of effort has been done dealing with mobile robots, only a few works [3][4][5][6] are concerned with articulated manipulators. In [7] a general method is proposed, for many types of structures that include both mobile and articulated, where the C-obstacles evaluation is established as the convolution product of two functions that represent the robot and the obstacles.

Although this is a big step, the method is not general enough, since some structures —such as redundant robots— can not be considered. A further step can be taken by an analysis of the proposed general method as applied to articulated robots, since changing the reference system for each link would provide a new method that simplifies computation, thus reducing both the memory needs and the computation times [8].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 596–601, 2005.
© Springer-Verlag Berlin Heidelberg 2005

2 Evaluating C-Obstacles as a Convolution

In this section, the method proposed by Curto in [7] is revised, as it is the basis for the method presented in this paper.

The representation of the C-obstacles is proposed based on the integral of the product of two functions: one that represents the robot A and another one that represents the obstacles in the workspace, B. W will designate the workspace and C the C-space. Thus,

Definition 1. *Let $A : C \times W \rightarrow R$ be the function defined by*

$$A(q,x) = \begin{cases} 1 \text{ if } x \in \mathbf{A}(q) \\ 0 \text{ if } x \notin \mathbf{A}(q) \end{cases} \tag{1}$$

where $\mathbf{A}(q)$ is the subset of W that represents the robot at the configuration q.

Definition 2. *Let $B : W \rightarrow R$ be the function defined by*

$$B(x) = \begin{cases} 1 \text{ if } x \in \mathbf{B} \\ 0 \text{ if } x \notin \mathbf{B} \end{cases} \tag{2}$$

where $\mathbf{B}$ is the subset of W formed by the obstacles.

Using both A and B, a new definition for calculating C-obstacles is proposed:

Definition 3. *Let $CB : C \rightarrow R$ be the function defined by*

$$CB(q) = \int A(q,x)B(x)dx \quad \forall q \in C, \quad \forall x \in W \tag{3}$$

The region $\mathbf{C}\mathrm{B}_{\mathbf{f}}$ is defined as the subset of C that verifies

$$\mathbf{C}\mathrm{B}_{\mathbf{f}} = \{q \in C / CB(q) > 0\} \tag{4}$$

The previous expressions were defined without considering any specific parameterization of W and C.

Now, a representation of W and C is given by selecting two frames F_W and F_A for the workspace and for the robot, respectively, where F_W is fixed and F_A is attached to the robot. In this way, a point $x \in W$ is given by $(x_1, x_2, \cdots, x_n)$ where n is the workspace dimension, and a configuration $q \in C$ is represented by $(q_1, q_2, \cdots, q_m)$ that specify the position and orientation of F_A respect to F_W, where m is the dimension of C. Thus, the expression (3) becomes

$$CB(q_1, \cdots, q_m) = \int A(q_1, \cdots, q_m, x_1, \cdots, x_n)B(x_1, \cdots, x_n)dx_1 \cdots dx_n \tag{5}$$

3 Superposition Principle of C-Obstacles

In this paper, an articulated robot is considered as a kinematic chain. In this way, a robot $\mathbf{A}$ is viewed as a set of r rigid objects. The kinematics of this chain, i.e., the movement restrictions imposed by the joint to each element, $\mathbf{A}_i$ —the degrees of freedom, DOFs—, would determine some regions of the C-space.

This principle is the basis of the evaluation of the C-space for robots that consist of several elements connected by means of different types of joints.

Considering that a robot consists of r rigid objects, the resulting C-obstacles will follow the Superposition Principle:

Theorem 1. *Let* $\mathbf{A}$ *be an articulated robot formed by* r *elements* $\mathbf{A}_1, \ldots, \mathbf{A}_r$. *If* $\mathbf{CB}_1, \ldots, \mathbf{CB}_r$ *are, respectively, the C-obstacle regions for the* $\mathbf{A}_1, \ldots, \mathbf{A}_r$ *elements in the space where the obstacle* $\mathbf{B}$ *is projected, then, the C-obstacle* $\mathbf{CB}$ *due to* $\mathbf{B}$ *for the robot* $\mathbf{A}$ *can be obtained as*

$$\mathbf{CB} = \bigcup_{k=1}^{r} \mathbf{CB}_k \tag{6}$$

The expression (6) reflects the fact that the union of these subsets equals the configuration space for $\mathbf{A}$. The idea of C-obstacles superposition is the key principle that enables the deconstruction approach.

4 The Deconstruction Method

The Deconstruction method tries to independently evaluate portions of the C-space in order to find the C-obstacles due to each link in the kinematic chain.

4.1 Applicating the Superposition Principle

Taking into account (6), the calculation of $\mathbf{CB}$ for a robot $\mathbf{A}$, a kinematic chain of r links, is done through the union of all the $\mathbf{CB}_k$ related to each of the elements of the robot. The computation of every C-obstacle region must be done through the evaluation of the associated CB_k functions.

$$CB_k(q_{1_k}, \cdots, q_{s_k}),\ \forall k \in \{1, \ldots, r\} \tag{7}$$

with $\{q_{1_k}, \cdots, q_{s_k}\} \subseteq \{q_1, \cdots, q_m\}$, where $\{q_1, \cdots, q_m\}$ are the DOFs associated to the robot $\mathbf{A}$. That is, for the *k-th* element only the subset of configuration variables associated to it are considered, and, analogously to (5), each of the $CB_k(q_{1_k}, \cdots, q_{s_k})$ functions is evaluated as follows

$$\int A_k(q_{1_k}, \cdots, q_{s_k}, x_1, \cdots, x_n) B(x_1, \cdots, x_n) dx_1 \cdots dx_n \tag{8}$$

4.2 Choosing the Frames

When solving the integral (8), the function $A_k(q_{1_k}, \cdots, q_{s_k}, x_1, \cdots, x_n)$, representing the robot, is difficult to evaluate, due to its dependency on all of the DOFs related to itself and to the previous links in the chain. Thus, we will try to reduce this difficulty by choosing the proper frames.

In order to do that, let's consider the robot formed by the kinematic chain of figure 1. As one can see, following the Denavit-Hartenberg method [9], a frame is associated with each link, placing the origin at the end of the link; the orientation of axes depends on the position and orientation of the link.

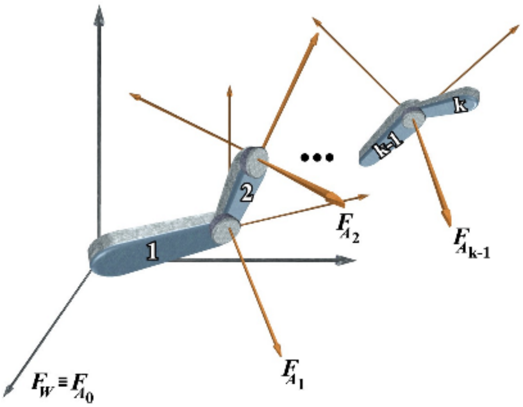


Fig. 1. Frames in the kinematic chain of an articulated robot

Following the Denavit-Hartenberg procedure, the Deconstruction method proposes to use the frame determined by the previous link for the *k-th* element. Thus, for link 1 the frame F_{A_0} —which coincides with the workspace frame, F_W— is used; similarly, for the *k-th* link, frame $F_{A_{k-1}}$ will be used (figure 1).

Now, if we have a look to $A_k(q_{1_k}, \cdots, q_{s_k}, x_1, \cdots, x_n)$, the expression we are evaluating, it can be written as follows

$$A_k(\underbrace{q_{1_k}, \cdots, q_{u_k}}_{DOF_{(1,\ldots,k-1)}}, \underbrace{q_{(u+1)_k}, \cdots, q_{s_k}}_{DOF_k}, x_1, \cdots, x_n) \tag{9}$$

where $\{q_{1_k}, \cdots, q_{u_k}\}$ are the degrees of freedom associated to the elements preceding the k-th element the *k-th* element, whose DOFs are $\left\{q_{(u+1)_k}, \cdots, q_{s_k}\right\}$.

At this point, the position and orientation of the element $\mathbf{A}_k$ is expressed related to the frame F_{A_0}. The position, just like the frame $F_{A_{k-1}}$, is determined by the associated degrees of freedom of the previous links in the chain, that is to say, some of the parameters related to each $\mathbf{A}_i$ —previous elements– in that subchain, (a_i, α_i, d_i and θ_i, the Denavit-Hartenberg parameters).

Thus, if the position and orientation of the element $\mathbf{A}_k$ are expressed taking as origin the frame $F_{A_{k-1}}$, its evaluation will be much simpler. An homogeneous transformation $\mathbf{T}$ is needed to perform this operation.

Definition 4. *Let* ${}_0^{k-1}\mathbf{T}$ *be the transformation that permits us to move the frame* F_{A_0} *to such point that it will coincide with* $F_{A_{k\quad 1}}$.

It is important to point out that this homogeneous transformation depends on the configuration parameters related to the previous elements in the chain, that is to say, ${}_0^{k-1}\mathbf{T} = f(q_{1_k}, \cdots, q_{u_k})$. At this point, the position and orientation of the link $\mathbf{A}_k$, expressed related to the frame $F_{A_{k\quad 1}}$, will only depend of its associated degrees of freedom, that is, $\{q_{(u+1)_k}, \cdots, q_{s_k}\}$.

However, this homogeneous transformation has a consequence: it will be necessary to express the workspace as a function of the new frame, $F_{A_{k\quad 1}}$:

$$B'(x'_1, \cdots, x'_n) =_0^{k-1} \mathbf{T}B(x_1, \cdots, x_n) \tag{10}$$

In this way, the evaluation of (9) is equivalent to the following one

$$A'_k(q_{(u+1)_k}, \cdots, q_{s_k}, x'_1, \cdots, x'_n) \tag{11}$$

Finally, (8), which is used to calculate the C-obstacle portion pertaining to the element $\mathbf{A}_k$, becomes

$$\int A'_k(q_{(u+1)_k}, \cdots, q_{s_k}, x'_1, \cdots, x'_n)B'(x'_1, \cdots, x'_n)dx'_1 \cdots dx'_n \tag{12}$$

Now, after the proper frame is chosen, as it can be seen in (12), it is possible to study individually each one of the links.

4.3 Choosing the Coordinate Functions

Kavraki [2] and Curto [7] propose the simplification of the C-space calculation by the use of the Convolution theorem (and the Fast Fourieer Transform). We shall now expose how this is applicable inside the new proposed formalism by means of the introduction of a coordinate functions change.

As demonstrated in [7], it is sufficient to choose the proper coordinate functions, $(\xi_1, \cdots, \xi_n)$, that will permit to find one or more relationships between some of the configuration variables and some of the coordinate functions, which will allow to find the convolution.

Thus, a new function, $\bar{A}'_k$, is introduced; the idea is to find a simpler functional dependency in function A'_k, in such a way that element $\mathbf{A}_k$ becomes independent of a subset of $\{q_{(u+1)_{\underline{k}}}, \cdots, q_{s_k}\}$, depending only on $\{q_{(v+1)_k}, \cdots, q_{s_k}\}$.

Having this new function $\bar{A}'_k$, (12) will be defined as

$$\int longA\ longB\ d\xi_1 \cdots d\xi_n$$

$$\begin{aligned} &longA{=}\bar{A}_k(0,\cdots,0,q_{(v+1)_k},\cdots,q_{s_k},\xi_1{-}q_{(u+1)_k},\cdots,\xi_v{-}q_{v_k},\xi_{(v+1)_k},\cdots,\xi_n) \\ &longB{=}B\ (\xi_1,\cdots,\xi_n) \end{aligned} \tag{13}$$

which leads to a function $\bar{A}'_k$ that depends only on $\{q_{(v+1)_k}, \cdots, q_{s_k}\}$. Now, for variables $\{q_{(u+1)_k}, \cdots, q_{v_k}\}$ the following convolution product appears.

$$\int (\bar{A}_{k(0,\quad ,0,q_{(v+1)_k},\quad ,q_{s_k})} * B)_{(\xi_1,\quad ,\xi_{v_k})}(\xi_{(v+1)_k},\cdots,\xi_n)\, d\xi_{(v+1)_k}\cdots d\xi_n \tag{14}$$

where subindices $(\xi_1, \cdots, \xi_{v_k})$ denote that the convolution product is calculated for all of the values of these variables.

5 Conclusion

In this paper, a mathematical formalism for the Deconstruction method is proposed. This approach permits the simplification of the C-space evaluating process by means of the application of a simple and repetitive operation for each link in the kinematic chain, being valid for redundant robots. This method can naturally face the evaluation of C-spaces of many dimensions, since only non-colliding configurations are considered for the evaluation of the following links.

References

1. Lozano-Pérez, T.: Spatial planning: A configuration space approach. IEEE Transactions on Computers **32** (1983) 108–120
2. Kavraki, L.E.: Computation of configuration space obstacles using the fast fourier transform. IEEE Tr. on Robotics and Automation **11** (1995) 408–413
3. Maciejewski, A.A., Fox, J.J.: Path planning and the topology of configuration space. IEEE Tr. on Robotics and Automation **9** (1993)
4. Newman, W., Branicky, M.: Real-time configuration space transforms for obstacle avoidance. The International Journal of Robotics Research **6** (1991)
5. Lozano-Pérez, T.: A simple motion-planning algorithm for general robot manipulators. IEEE Journal of Robotics and Automation **3** (1987) 224–238
6. Lozano-Pérez, T., P.O'Donnell: Parallel robot motion planning. In: Proc. of the IEEE Int. Conf. on Robotics and Automation. (1991) 1000–1007
7. Curto, B., Moreno, V., Blanco, F.J.: A general method for c-space evaluation and its application to articulated robots. IEEE Transactions on Robotics and Automation **18** (2002) 24–31
8. Therón, R., Moreno, V., Curto, B., Blanco, F.J.: Assessing methods for the evaluation of the configuration space for planar revolute manipulators. Journal of Computational Methods in Science and Engineering **4** (2004) 149–156
9. Denavit, J., Hartenberg, R.S.: A kinematic notation for lower-pair mechanisms on matrices. Journal of Applied Mathematics (1955) 215–221

Global Modal Logics for Multiagent Systems: A Logical Fibering Approach

Johann Edtmayr

University of Salzburg, Department of Computer Science,
Jakob Haringer Str. 2, A-5020 Salzburg, Austria
johann.edtmayr@cs.uni-salzburg.at

Abstract. It has become customary to use epistemic modal logic for the formal study of knowledge and beliefs of agents. Based on the concept of logical fibering, which is briefly summarized, we present so-called fibered global modal logics for the logical modeling of multiagent systems (MAS). Considering a simple multi-robot scenario we show that our global logical models allow to study the knowledge and beliefs of the agents, taking into account the communication between the agents.

1 Introduction

The problem of combining logics (or logical systems) is a growing area of interest in modern logic as shown among other things in [5] and [2]. In recent years logic is used more and more to formalize complex problems in artificial intelligence, software engineering, and computer science as a whole. In particular the formal modeling of systems of cooperating agents is of general interest, because in many studies of complex systems, especially distributed systems, a multiagent model is used. The formal modeling of systems in this area usually requires combined systems of logics.

A very flexible approach for combining logics is the concept of logical fiberings which was originally introduced by J.Pfalzgraf in [5]. This concept provides a framework for the construction of complex logical models as follows: Several logical systems called fibers, or local subsystems, are arranged (modeled) over the points of a base space. The base space of a logical fibering can carry an additional structure, for example a communication network, which represents some kind of interaction or communication between the local subsystems. J.Pfalzgraf and coworkers have shown among other things in [8], [10], and [7] that this modeling language is especially suitable to associate a system of distributed logics to a multiagent system (MAS). In modeling a MAS in this way each individual agent obtains its own logic (local logical state space). The whole logical model can be formed by putting together all local fibers thus yielding the corresponding logical fibering, which presents the global logical state space of the MAS.

For the designer of an autonomous artificial agent the relationship between knowledge and actions of the agent are of great interest. It has become customary to use epistemic modal logic for the formal study of knowledge and beliefs of

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 602–607, 2005.
© Springer-Verlag Berlin Heidelberg 2005

such systems. So far the concept of logical fiberings was not used in conjunction with modal logics and possible world semantics. A closer study of systems of distributed modal logics, in terms of fibered structures, for the formal modeling of multiagent systems seems to be reasonable (cf. [9]). Thus, our research has been aimed at testing whether it is possible to build a logical fibering, which represents a *global* modal logic modeling a multiagent system, consisting of the *local* modal logics (fibers) assigned to the agents. It has turned out that such a logical fibering results in a formal model of a MAS, which allows to study the knowledge and beliefs of agents, taking into account the *communication* between the agents. To show first results of our research we consider a simple task (manufacturing process) which is performed by three cooperating robots.

2 A Brief Introduction to Logical Fiberings

The concept of logical fiberings was developed on the basis of the concept of polycontextural logics (PCL) and the mathematical theory of fiber bundles (cf.[5]). The polycontextural logics have come from the work of Gotthard Günther on philosophy and cybernetics. This so-called "transclassical logic" is a distribution of classical (2-valued) logics in which the individual logical subsystems are enabled to interplay with each other. One main argument of experts in PCL is that "transclassical logic" is a suitable logical basis for modeling (living) communicating systems. For more literature on PCL we refer to [4] and [3].

The basic structure of fiber bundles is given by the definition of a so-called abstract fibering. A general abstract fibering (fiber bundle) is a triple $\xi = (E, \pi, B)$ consisting of a so-called base space B and a total space E. Both spaces are connected by the projection map $\pi : E \longrightarrow B$. The fiber over a point $b \in B$ is the preimage set $\pi^{-1}(b) = \{x \in E \mid \pi(x) = b\} =: E_b$. The total space E is the disjoint union (coproduct) of the E_b, denoted by $E = \coprod_{b \in B} E_b$.

If logics (logical languages) are used as fibers over the points of B the corresponding fibering is called a logical fibering. In this context the base space is formally interpreted as an index set I ($B = I$). A fiber $\pi^{-1}(i)$ over a point $i \in I$ of the base space is a local logic. Thus, a logical fibering is a system of logics distributed over a corresponding index set I. Actually, it has turned out that polycontextural logics are formally presentable as a specific class of logical fiberings (cf. [5]). Altogether, this suggests the usage of logical fiberings for the logical modeling of groups of cooperating and communicating agents (i.e. multiagent systems). In modeling a multiagent system in this way, each agent obtains its own local logic. In section 4 the construction process of a concrete logical fibering will be treated in great detail.

3 On the Use of Modal Logics for Agents

Robots are equipped with sensors for the purpose of acquiring information about their environment. Due to sensory limitations a robot is usually not able to perceive the environment completely. Thus, due to these sensory limitations, multiple world states are perceived as the same state. Consequently, the individual sensory capabilities of a robot $\mathcal{A}_i$ induces an *equivalence relation* R_i over the set S of states (worlds) defined as follows.

$$R_i = \{(s_x, s_y) \in S \times S \mid \mathcal{A}_i \text{ cannot distinguish between } s_x \text{ and } s_y\} \ . \tag{1}$$

Obviously, robot $\mathcal{A}_i$ cannot distinguish between any two members of the same equivalence class, only between elements of the quotient space S/R_i. Intuitively we can say that a robot knows a fact if this fact is true in all states of the equivalence class $[s_r]_i$ containing the real state $s_r \in S$ of the environment.

For the formal study of knowledge and beliefs it has become customary to use epistemic modal logic. In modal logic so called *possible world models* are used for semantical considerations. A possible world model is formally defined as a triple $\mathcal{M} = (S, R, v)$ consisting of a set S of states (worlds), a binary relation R on S, also called accessibility relation, and a valuation function v. The accessibility relation R specifies which worlds are considered possible (or accessible) relative to other worlds. The valuation function $v : S \times \Phi \longrightarrow \{T, F\}$ describes each world of S by the assignment of truth values to the atomic propositions of a corresponding set Φ. In modal logic, on the syntactic level, it is customary to use the the symbol $\Box$ to express that a formula φ is necessarily true, or known by an agent. Furthermore, we use $\Box_i\varphi$ to express that an agent $\mathcal{A}_i$ *knows* φ. The following definition taken from modal logic literature determines if a formula $\Box\varphi$ holds at a world $s_x \in S$ of a possible world model $\mathcal{M}$. $(\mathcal{M}, s_x) \models \Box\varphi \Leftrightarrow (\mathcal{M}, s_y) \models \varphi$ for all s_y such that $(s_x, s_y) \in R$.

Now let $\mathcal{M}$ be the possible world model where the accessibility relation is R_i of (1), and let S be the states of the environment in which the corresponding robot $\mathcal{A}_i$ is situated. In this case $(\mathcal{M}, s_x) \models \Box\varphi$ means that φ holds at every state of $[s_x]_i$. According to the informal considerations in the first part of this section, this means that in environment state s_x, robot $\mathcal{A}_i$ *knows* φ.

4 Global Modal Logics for MAS

For the logical modeling of groups of cooperating and communicating agents we use the concept of logical fiberings in conjunction with modal logic and possible world semantics. This results in global logical systems consisting of several local modal logics assigned to the agents. We introduce the basic concept of our global modal logics considering a very simple example scenario consisting of three cooperating robots performing a manufacturing task. The task consists of positioning a bolt on top of a steel-plate. Afterwards the two work pieces have to be welded together. The work-sharing between the three robots is as follows: The welding robot ($\mathcal{R}_0$) has sensors to perceive the existence of the work pieces in its working area. According to this sensory information it requests missing work pieces from the robots $\mathcal{R}_1$ and $\mathcal{R}_2$ respectively. $\mathcal{R}_1$ provides (on request) the steel-plate and positions it properly within the working area of the welding robot. On request, $\mathcal{R}_2$ puts the bolt on top of the steel plate, which is already well positioned in the working area of $\mathcal{R}_0$. If the two work pieces are positioned properly in the working area of $\mathcal{R}_0$ it finally starts the welding process. It is important to mention here that the robots $\mathcal{R}_1$ and $\mathcal{R}_2$ have no sensors at all. They get their information about the actual state of the environment solely from $\mathcal{R}_0$ through communication.

The concept of logical fiberings provides us with the framework for the construction of a global logical model (logical fibering) for our simple robot scenario. In the beginning, the corresponding base space of the logical fibering is simply the set $I = \{0, 1, 2\}$, consisting of the indices assigned to the robots. For the construction of the local logical subsystems (fibers) over the points of the base space I we use a propositional modal language. More precisely, in order to describe the different (relevant) states of the robot scenario we utilize the set $\Phi = \{plate\_1, bolt\_2, plate\_0, bolt\_0, welded\}$ of primitive propositions. These primitive propositions stand for basic facts about the world. For instance, *plate_0* stands for *"the steel plate is positioned in the working area of $\mathcal{R}_0$"* and *welded* stands for *"the steel plate and the bolt are welded together"*. We use the primitive propositions in Φ and form more complicated formulas by closing off under $\neg, \vee, \wedge$, and $\Box$. This yields a propositional modal language L in terms of a set of formulas. In order to define the logical fibering for the robot scenario we set the total space $E = I \times L$, and we use the first projection as the projection map $\pi : E \longrightarrow I$. Thus, for some $i \in I$, $\pi^{-1}(i) = \{i\} \times L$ is the propositional modal language L_i (fiber) assigned to robot i (we use $\Box_i\varphi$ to denote $(i, \Box\varphi) \in \pi^{-1}(i)$).

Our model of communication is based on the following idea: If an agent acquires additional information - such as hearing from a reliable source that fact φ is true - then he would no longer consider possible any of the worlds in which φ is false (cf. [1]). Thus, communication influences the set of worlds an agent considers possible, and consequently also his knowledge (cf. section 3). In order to incorporate communication into our global logical models (logical fiberings) of multiagent systems we establish special communication networks in the base spaces of the fiberings. We use these networks to determine the sets of worlds the *communicating* agents consider possible given a certain actual world. The communication network of our example scenario is shown in figure 1. The labels of an edge in such a communication network determine both the situations in which the sending agent communicates, in the form of a set of *possible* worlds, and the content of the communication, in the form of a propositional modal formula. We define the worlds (states) in which an agent $\mathcal{A}_i$ communicates in using a map $\sigma_i : S \longrightarrow \mathcal{P}(S)$, which maps each world $s_x \in S$ to a corresponding subset $\sigma_i(s_x) \subseteq S$ in the power set $\mathcal{P}(S)$. Under consideration of the whole

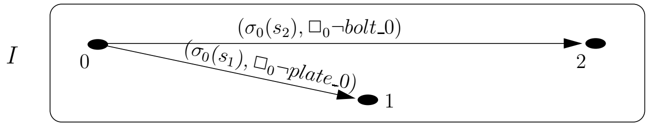

I
$(\sigma_0(s_2), \Box_0\neg bolt\_0)$
$(\sigma_0(s_1), \Box_0\neg plate\_0)$
0
1
2

Fig. 1. Schematic diagram of the communication network in the base space I

communication in the system, this map yields for any actual world s_x the set of worlds $\sigma_i(s_x) \in \mathcal{P}(S)$ agent $\mathcal{A}_i$ considers possible. It is important to mention here that if an agent communicates a formula φ in some state s_x he sends this message within all states $s_y \in \sigma_i(s_x)$, because he is not able to tell the difference between s_x and any other state $s_y \in \sigma_i(s_x)$. For our simple robot scenario the following definition of the functions σ_i, $i \in I$, is applicable.

$$\sigma_i(s_x) := \begin{cases} \{s_y \in S \mid s_x \equiv_i s_y\} = [s_x]_i & \text{if } \delta(i)^+ = 0 \\ \\ [s_x]_i \quad \setminus \quad \begin{array}{l} \{s_y \in S \mid \exists \text{ a labeling } (\sigma_j(s_z), \Box\varphi) \\ \text{of an edge } e \text{ with } \tau(e) = i \text{ such that} \\ s_x \in \sigma_j(s_z) \wedge \mathit{eval}_i(\varphi, s_y) = F\} \end{array} & \mathit{else}\ . \end{cases} \tag{2}$$

Informally, this definition says that if on the one hand a vertex i of the communication network has no incoming edges (i.e. $\delta(i)^+ = 0$), robot $\mathcal{R}_i$ never receives messages from other robots. Thus, in every state, its knowledge is determined solely by its sensory information. If on the other hand a vertex i is the end point of one or more edges e (i.e. $\exists e : \ \tau(e) = i$), the worlds robot $\mathcal{R}_i$ considers possible in a given actual world s_x, and consequently its knowledge, is determined by both its sensory information and the information it possibly receives from other robots in the actual world s_x.

Now let s_1 be the world in which neither the steel plate nor the bolt is present in the working area of $\mathcal{R}_0$ (the welding robot). Furthermore, let s_2 be the world in which the steel plate is already well positioned in the working area of $\mathcal{R}_0$, but the bolt is not yet present. Formally, we describe these worlds in defining the valuation function v in s_1 and s_2 for all atomic propositions of Φ in an appropriate manner. For instance, we set $v(s_1, \mathit{plate\_1}) = T$, $v(s_1, \mathit{plate\_0}) = F$, $v(s_1, \mathit{bolt\_2}) = T$, $v(s_1, \mathit{bolt\_0}) = F$, $v(s_1, \mathit{welded}) = F$. The other worlds (states) of the manufacturing process are described in an analogous way.

With the global logical model constructed above it is possible to consider the whole robot scenario (manufacturing process) formally as follows. The sensory capability of $\mathcal{R}_0$ induces a partitioning of S into disjoint subsets (cf. figure 2). That is, $\mathcal{R}_0$ can only distinguish between worlds of different subsets of S. The robots $\mathcal{R}_1$ and $\mathcal{R}_2$ have no sensors at all. Consequently, without getting information from $\mathcal{R}_0$ they are unable to distinguish any two worlds of S. Now let us look at the knowledge ascribed to the robots by our global logical model, which takes into account the communication between the robots. In the initial world s_1 the function σ_0 (assigned to $\mathcal{R}_0$) yields $\sigma_0(s_1) = \{s_1\}$. That is, in world s_1 robot $\mathcal{R}_0$ considers only the world s_1 possible. Because of $\neg \mathit{plate\_0} = T$ in s_1 and $\sigma_0(s_1) = \{s_1\}$, the formula $\Box_0 \neg \mathit{plate\_0}$ is true (T) in s_1. That is, in our model $\mathcal{R}_0$ knows $\neg \mathit{plate\_0}$ in world s_1. According to the communication network (cf. figure 1) $\mathcal{R}_0$ communicates this knowledge to $\mathcal{R}_1$ in $\sigma_0(s_1) = \{s_1\}$. Consequently, under this conditions $\sigma_1(s_1) = \{s_1, s_5\}$ is the set of worlds $\mathcal{R}_1$ considers possible (cf. figure 2), if we assume that s_5 is the world described by $v(s_5, \mathit{plate\_1}) = T$, $v(s_5, \mathit{plate\_0}) = F$, $v(s_5, \mathit{bolt\_2}) = F$, $v(s_5, \mathit{bolt\_0}) = T$, $v(s_5, \mathit{welded}) = F$.

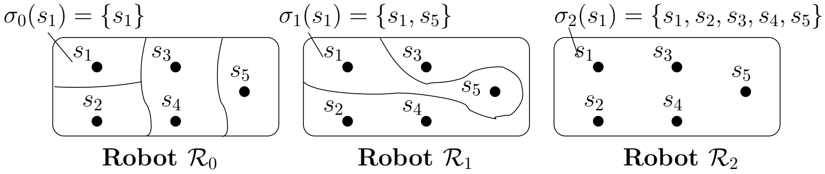

$\sigma_0(s_1) = \{s_1\}$
$\sigma_1(s_1) = \{s_1, s_5\}$
$\sigma_2(s_1) = \{s_1, s_2, s_3, s_4, s_5\}$
s_1
s_2
s_3
s_4
s_5
Robot $\mathcal{R}_0$
Robot $\mathcal{R}_1$
Robot $\mathcal{R}_2$

Fig. 2. Sets of worlds $\sigma_i(s_1) \subseteq S$, $i \in I$, the robots consider possible in s_1

From this it follows that in world s_1 robot $\mathcal{R}_1$ also *knows* $\neg$*plate*_0, because this formula is true in all worlds of $\sigma_1(s_1)$. This new knowledge of $\mathcal{R}_1$, which triggers it to provide $\mathcal{R}_0$ with the steel plate, arises from a temporary partitioning of S in world s_1 caused by communication. The communication in other states of S takes place in an analogous manner.

5 Concluding Remarks and Prospects

The usage of fibered structures in conjunction with modal logics in order to construct *global modal logics* for MAS is a further development of the logical fibering approach. In our future work we intend to study, among other things, MAS with more complex communication structures. The approach of treating a classical deduction problem as an ideal membership problem was already used for m-valued logics parallelized by logical fiberings (cf. [6]). We intend to analyse if deduction problems in global modal logics can be treated in a similar way.

References

1. R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardin. *Reasoning About Knowledge.* MIT Press, 1995.
2. Dov M. Gabbay. *Fibring Logics.* Oxford University Press, 1999.
3. G. Günther. Cybernetic Ontologie and Transjunctional Operations. *Biological Computer Lab. Publ., Vol. 68 (Urbana, Ill.)*, pages 313–392, 1962.
4. G. Günther. *Beiträge zur Grundlegung einer operationsfähigen Dialektik, 3 Volumes.* Felix Meiner Verlag, Hamburg, 1980.
5. J. Pfalzgraf. *Logical Fiberings and Polycontextural Systems.* 1991. In: Proc. Fundamentals of Artificial Intelligence Research, Jorrand Ph., Kelemen J. (Eds.). Lecture Notes in Computer Science 535 (subseries LNAI).
6. J. Pfalzgraf. On logical fiberings and automated deduction in many-valued logics using Gröbner Bases. *Revista Real Academia de Ciencias, RACSAM*, 98, 2004.
7. J. Pfalzgraf and J. Edtmayr. The concept of logical fiberings: Distributed logics for multiagent systems. *Cybernetics and Systems (EMCSR 04)*, 1:58–63, 2004.
8. J. Pfalzgraf, V. Sofronie, and K. Stokkermans. On semantics for cooperative agents scenarios. *Cybernetics and Systems (EMCSR 96)*, 1:201–206, 1996.
9. T. Porter. Geometric aspects of multiagent systems. *Electronic Notes in Theoretical Computer Science 81*, 2003.
10. D. Wang and J. Pfalzgraf. *Automated Practical Reasoning.* Springer-Verlag, 1995.

Improved Non-standard Discretization Methods for Nonlinear Dynamical Control Systems

Jesús Rodríguez-Millán, Carla González, and Anna Patete

Universidad de Los Andes - Facultad de Ingeniería - Dept. Sistemas de Control,
Apartado 11 - La Hechicera, Mérida 5101-A, Venezuela
Fax: 58-274-2402847
`{jrmillan, carlag, patete}@ula.ve`

Abstract. In this paper we describe two modified versions of the Euler-Picard and Euler-Taylor-Picard discretization methods for nonlinear dynamical control systems. We use an upper bound for the absolute difference of each pair of consecutive Picard iterations, in the first case to control the number of Picard iterations, and in the second case to control the sampling frequency, while keeping the maximum allowed number of Picard iterations fixed. These non-standard discretization methods are used to support the construction of computer animated mimics of nonlinear dynamical control systems.

1 Introduction

The discretization of a linear dynamical control systems (LDCS)

$$\dot{x} = A\,x + B\,u,\ \ x(0) = x_0,\ \ \ y = C\,x + D\,u, \tag{1}$$

where $x \in \Re^n$, $u \in \Re$, and $y \in \Re$, always assumes the standard form

$$x((k+1)T) = A_{dm}\,x(kT) + B_{dm}\,u(kT),\quad y(kT) = Cx(kT) + D\,u(kT) \tag{2}$$

where $x(0) = x_0$, $x(kT) \in \Re^n$, $u(kT) \in \Re$, $y(kT) \in \Re$, and the matrices A_{dm}, B_{dm}, depend on the discretization method [2], indicated by the second suffix *m.* In the *approximate discretization method,* we discretize the ordinary differential equation in (1) by substituting the first-order time derivative by its first-order backward difference, to obtain $A_{da} = (I - TA)^{-1}$, and $B_{da} = T(I - TA)^{-1}B$. In the *exact discretization method* we first apply zero-order hold [2] to the control signal *u*, and then sample the exact trajectory of the system at $t = kT$, to obtain that $A_{de} = \mathrm{Exp}(AT)$, and $B_{de} = \int_0^T \mathrm{Exp}[As]ds B$. The *jth-degree truncated discretization method* is based on truncating the Taylor-series expansion of Exp[At] at its jth-degree term to get $A_{dt} = \sum_{i=0}^{j} \frac{T^i}{i!}A^i$, and $B_{dt} = \sum_{i=0}^{j-1} \frac{T^{i+1}}{(i+1)!}A^i B$ [2].

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 608 – 613, 2005.
© Springer-Verlag Berlin Heidelberg 2005

2 Euler Discretization of Nonlinear Dynamical Systems

By a *nth-order continuous-time nonlinear dynamical control system* (NLDCS) we will mean a pair of equations

$$\dot{x} = f(x, u), \;\; x(0) = x_0, \;\; y = h(x, u)), \tag{3}$$

where $x \in \Re^n$, $u \in \Re$, $y \in \Re$, and the vector-field *f* and the output-function *h* are as smooth as needed. To discretize the NLDCS (1) means to construct a *discrete-time nonlinear dynamical control system*

$$\Phi((k+1)T) = F(\Phi(kT),u(kT)), \Phi(0) = x_0, \Phi(kT) \in \Re^n, u(kT) \in \Re, \tag{4-a}$$

$$\Psi(kT) = H(\Phi(kT),u(kT)), \Psi(kT) \in \Re, \tag{4-b}$$

where the functions *F* and *H* are obtained from *f* and *h*, respectively, according to a prescribed discretization method. Even though we think of mimics of NLDCS like attractive computer animated graphical objects, mathematically speaking mimics are nth-order discrete-time NLDCS like (4).

Backward approximate discretization is not extendable to NLDCS because of invertibility problems. This limitation may be mathematically overcome using forward approximate discretization, based on approximating first-order time derivatives by their forward first-order time differences. Yet, whenever possible engineers avoid this approach not to contradict the causality principle. Forward approximate discretization supports the construction of Euler polygons like approximated solutions of NLDCS [1], wherefore forward approximate discretization of NLDCS will be called the *Euler discretization method*. This is the default method used in practice to discretize NLDCS [6, 8], and we will metaphorically describe it as *periodic sampling plus linear interpolation*. The Euler-discretization of the NLDCS (3) is given by

$$\Phi(kT) = \Phi((k-1)T) + T\, f(\Phi((k-1)T),u((k-1)T)), \Phi(0) = x_0 \tag{5-a}$$

$$\Psi(kT) = h(\Phi(kT),u(kT)) \tag{5-b}$$

where, $\Phi(kT) \in \Re^n$, $u(kT) \in \Re$, and $\Psi(kT) \in \Re$.

3 Euler-Picard Discretization of Nonlinear Dynamical Systems

Given that NLDCS are in general not explicitly solvable, the exact discretization method is neither extendable to them. Yet, in [8] we proposed the *Euler-Picard discretization method* (EPDM), whose spirit might be paraphrased as *global periodic Euler-like sampling plus Picard-like interpolation*. The ith iterated discrete-time *Euler-Picard trajectory* Ψ_i of period T [8],

$$\Psi_i(0) = x_0, \;\; \Psi_i(kT) = x_0 + \sum_{j=1}^{k} \int_{(j-1)T}^{jT} f(\phi_{i-1}^{k}(s),u(s))ds \tag{6}$$

constructed using the EPDM, would closely correspond to the sampling of the trajectories of the NLDCS (1), as the number of Picard iterations i → ∞, and the period $T \to 0$.

It is well known that the local fitting properties of the Picard iteration method improved with the number of computed iterations. On the other hand, the computation time of Euler-Picard trajectories Y_i in (6) obviously depends on the number of Picard iterations, which is the same on all intervals $[(j-1)T, jT]$, $1 \leq j \leq k$, of the flying interval $[0, kT]$. Practice also shows, that for fixed values of T and i, the fitting properties of the Picard iteration method strongly depends on how fast the trajectories of the NLDCS evolve. So, it would be reasonable to expect that within the context of NLDCS, we would need to compute less Picard iterations during the steady-state than during the transient-state, to satisfy a uniform numerical fitting criterion.

4 Euler-Taylor-Picard Discretization of Nonlinear Systems

Concerning the fitting properties of the periodic non-standard discretization methods proposed in [8], the EPDM is the best. Yet, because of the computation of the integrals involved, it is also prohibitively expensive for NLDCS with complex nonlinearities. To reduce the computation time and the unpredictable impact of arbitrary nonlinearities, we introduced the *Euler-Taylor-Picard discretization method* (ETPDM). ETPDM may also be paraphrased as periodic sampling plus Picard interpolation, but is not applied to the NLDCS (3), but to its jth-degree Taylor polynomial approximation

$$\dot{x} = F(x, u, j) = \sum_{i=0}^{j} \alpha_i(x_0, u)\,(x - x_0)^j,\ x(0) = x_0. \tag{7}$$

The associated ETP discrete-time trajectories are

$$\Psi(j, i, 0) = x_0,\ \Psi(j, i, k, T) = x_0 + \sum_{r=1}^{k} \int_{(r-1)T}^{rT} F(\Phi_{i-1}^{k}(s), u(s), j)ds \quad . \tag{8}$$

The ETPDM is also a one-step method, whose computation time depends on the sampling period T and the number of Picard iterations, but also on the degree of the Taylor polynomial expansion of the vector field f.

5 Non-standard Discretization Methods with Varying Number of Picard Iterations

The Picard successive approximations for the NLDCS (3), namely,

$$\Psi_i(0) = x_0,\quad \Psi_i(jT) = \Psi_i((j-1)T) + \int_{(j-1)T}^{jT} f(\phi_{i-1}^{j}(s), u((j-1)T))ds, \tag{9}$$

where *u((j-1)T)* indicates the sampling and zero-order holding operations on the control signal *u*, converge absolutely and uniformly on [(j-1)T, jT], $1 \le j \le m$, provided $T > 0$ is small enough [1]. Hence, for T small enough and $1 \le j \le m$, the real i-sequences $\Psi_i(jT))$ are Cauchy sequences, and therefore $|\Psi_{i+1}(jT) - \Psi_i(jT)| \to 0$, as $i \to \infty$. Moreover, an upper bound for the error in approximating the exact solution Ψ of the NLDCS (3) by its ith Picard approximation maybe easily computed [1], and we could, in principle, use this error criterion to estimate the number of Picard iterations needed to satisfy a prescribed maximum approximation error.

That the Picard successive approximations method only holds locally has important computational consequences. In particular, for any two different sampling intervals $I_j = [(j-1)T, jT]$ and $I_k = [(k-1)T, kT]$, and any two initial conditions $x((j-1)T) = x_j$ and $x((k-1)T) = x_k$, local Lipschitz constants L_1 and L_2 around $((j-1)T, x_j)$ and $((j-1)T, x_k)$ may also be different, hence we would need to compute different numbers of Picard iterations on I_j and I_k to keep the error in approximating the exact solution of the NLDCS (3) around $((j-1)T, x_j)$ and $((j-1)T, x_k)$ bounded by a common upper bound. Yet, instead of calculating the number of Picard iterations required to fit an upper bound for the approximating error at every single sampling interval, for implementation purposes we use a different and easier to implement approach. So, let us define the error function *e(i, j)* as $e(i, j) = |\Psi_{i+1}(jT) - \Psi_i(jT)|$, where its arguments *i* and *j* stands up for the number of computed Picard iteration, and the sampling interval [(j-1)T, jT], respectively. Now we use the error function e(i, j) to control the number of Picard iterations we compute on each sampling interval [(j-1)T, jT]. The computation algorithm goes as follows.

Let us assume the EPDM as default discretization method, and let $E > 0$ be the prescribed upper bound the error function e(i, j) must satisfy on the whole flying interval [0, mT]. Starting with the first sampling interval [0, T], let us compute the first two Picard iterations [8], and evaluate

$$e(1,1) = \left|\phi_2^1(T) - \phi_1^1(T)\right| = \left|\int_0^T [f(\phi_1^1(s), u(0)) - f(\phi_0^1(s), u(0))]ds\right|. \tag{10}$$

If $e(1,1) > E$, compute $\phi_3^1(T)$ and $e(2, 1) = \left|\phi_3^1(T) - \phi_2^1(T)\right|$. If $e(2,1) > E$, compute $\phi_4^1(T)$ and $e(3, 1) = \left|\phi_4^1(T) - \phi_3^1(T)\right|$, and so on, until $e(j, 1) < E$. Once $e(j, 1) < E$, take $\Psi(T) = \phi_{j+1}^1(T) = \phi_0^2(T)$, proceed to the next sampling interval [T, 2T], and compute

$$e(1,2) = \left|\phi_2^2(T) - \phi_1^2(T)\right| = \left|\int_0^T [f(\phi_1^2(s), u(0)) - f(\phi_0^2(s), u(0))]ds\right|. \tag{11}$$

If $e(1,2) > E$, compute the successive e(i,2), $2 \le i \le k$, until $e(k, 2) < E$. Then, take $\Psi(2T) = \phi_{k+1}^2(2T) = \phi_0^3(2T)$, proceed to the next sampling interval [2T, 3T], and compute e(1, 3), and so on, until you reach the last sampling interval [(m-1)T, mT].

Even though the argumentation above was formulated by default for the EPDM, it equally holds for the ETPDM. The modified one-step EPDM and ETPDM with varying number of Picard iterations will be denoted by VP-EPDM and VP-ETPDM.

6 Non-standard Discretization Methods with Varying Sampling Frequency

Choosing a discretization method is a typical optimization problem, where we face a trade off between fitting properties, computation time, and complexity, among other considerations [3]. Thus, the EPDM has the better fitting properties, but it maybe too slow depending on what kind of nonlinearities are involved. The EDM in the fastest one, but its accuracy is not good enough to track transient-state dynamics. With the ETPDM we improve computation times, but we lose qualitative sources of information, etc. Yet, we have empirically learned that concerning the non-standard discretization methods proposed in [8] it is always cheaper to increase the sampling frequency and keeping the number or Picard iterations low, than to keep the sampling frequency low and computing a higher number of Picard iterations. This remark led us to proposed a second variation of both the EPDM and the ETPDM, consisting of fixing not only an upper bound for the error in approximating the exact solution Ψ of the NLDCS (3) by its ith Picard approximation, but also a maximum to the number of Picard iterations allowed to be computed. This modified non-standard discretization methods will be identify as VF-EPDM and VF-ETPDM, where VF means variable sampling frequency.

Let us assume the EPDM by default, and let $E > 0$ be the prescribed upper bound for the error function $e(i, j)$ on the whole flying interval $[0, mT]$. Let N be the maximum number of allowed Picard iterations. Starting with the first sampling interval $[0, T]$, keep on computing the Picard iterations $\phi_i^1(T)$ and the error function $e(i, 1)$, $i = 0, 1, 2, \ldots$, until either $e(i, 1) < E$ for some $i < N$, in which case we proceed to the second sampling interval $[T, 2T]$ and repeat the procedure, or $i = N$ and $e(N, 1) > E$, in which case we go back, split the first sampling interval $[0, T]$ in two subintervals $[0, \frac{T}{2}]$ and $[\frac{T}{2}, T]$, and restart the procedure on the new first sampling interval $[0, \frac{T}{2}]$.

From the description of the algorithm supporting the VF-EPDM it transpires that this method contains an intrinsic procedure to search for an appropriate sampling frequency for the NLDCS (3), for which a prescribed approximating error criterion holds. By the time being we implement the VF-EPDM as a one-step discretization method, what works very well for systems whose fastest dynamics occur at the beginning of the transient-state. Regarding the computation time, this high mono sampling frequency implementation is not optimal, because in many cases it would be very convenient to use a higher sampling frequency during the transient-state than during the steady-state. Mathematically speaking it is completely natural to think about a varying sampling frequency discretization method, but an eventual physical implementation would then also require of an identical varying frequency sampling of, among others, the control signal u. This has to be evaluated in future works.

7 Remarks, Perspective and Future Work

Because of the constrains imposed by space, we can not go in this paper into many details regarding, for instance, the advantages and disadvantages of each one of the proposed discretization methods, the discussion of the results obtained in the case studies we have worked out, the libraries of functions in Mathematica supporting the implementation of the proposed discretization methods, etc. These and other details may be found in [3], which will be available upon request.

A lot of work remains to be done on improving the non-standard discretization methods for NLDCS proposed both in [8] and in this paper, and on given them a sound mathematical foundation. A particularly nice and important class of NLDCS is the set of NLDCS which are linear with respect to the control signal. To particularize the proposed discretization methods to this class of systems might lead to faster algorithms for this particular class of systems. We may probably also speed up all the proposed discretization method using an integrated symbolic-numeric implementation, instead of the purely symbolical ones we already have. We have not yet tried the proposed discretization algorithms on laboratory plants.

References

1. Coddington, E. and Levinson, N., Theory of Ordinary Differential Equations, Tata McGraw-Hill Publishing Co., New Delhi 1977.
2. Isermann, R., Digital Control Systems, Volume 1, Fundamentals, Deterministic Control, Springer-Verlag, Berlin, 1989.
3. Patete, A., Visualisación de Dinámicas No Lineales Mediante Mímicos en Mathematica, M.Sc. Dissertation, Graduate Program on Control and Automation, Universidad de Los Andes, Mérida, 2004.
4. Rodríguez-Millán, J., Analysis and Design of Nonlinear Dynamical Systems Using *Mathematica*, Nonlinear Analysis, Theory, Methods and Applications, Volume 30, Number 6, 3795-3805, 1997.
5. Rodríguez-Millán, J., NLcontrol: a Symbolic Computation Toolbox for Nonlinear Control by Extended Linearization, Proceedings of the 1998 American Control Conference, Volume 3, 1390-1394, IEEE, Philadelphia, 1998.
6. Rodríguez-Millán, J. Control de Procesos, Capítulo 2: Sistemas Dinámicos a Tiempo Discreto, Cuadernos de Sistemas Dinámicos, Grupo de Sistemas Dinámicos, Facultad de Ingeniería, Universidad de Los Andes, Mérida, 2004.
7. Rodríguez-Millán, J. and Cardillo, J., Symbolic Computing Aided Design of Nonlinear PID Controllers by Extended Linearization, In F. Pichler, R. Moreno-Díaz and R. Albrecht (Editors), Computer Aided Systems Theory, Lecture Notes in Computer Sciences 1030, pp. 253-270, Springer-Verlag, Berlin, 1996.
8. Rodríguez-Millán, J. and González, C., Three *Mathematica* Supported Proposals for the Discretization of Nonlinear Control Systems, to appear in the Proceedings of the 4th WCNA-2004, Orlando, July, 2004.
9. Sastry, Sh., Nonlinear Systems, Analysis, Stability and Control, Springer-Verlag, New York, 1999

Hierarchical Control of a Distributed Solar Collector Field

Manuel Berenguel[1], Cristina M. Cirre[1,4], Ryszard Klempous[2],
Henryk Maciejewski[2], Maciej Nikodem[2], Jan Nikodem[2],
Imre Rudas[3], and Loreto Valenzuela[4]

[1] Universidad de Almería, Dept. Lenguajes y Computación,
Ctra. Sacramento s/n, 04120, Almería, Spain
beren@ual.es
http://aer.ual.es/
[2] Wroclaw University of Technology, Institute of Engineering,
Cybernetics, 11/17 Janiszewskiego Street, 50-372 Wroclaw, Poland
[3] Budapest Polytechnic, H-1034 Budapest, Doberdo u. 6, Hungary
[4] CIEMAT, Plataforma Solar de Almería,
Ctra. Senes s/n, E-04200, Almería, Spain

Abstract. This article presents a hierarchical control structure aimed at optimizing the electricity production process in solar power plants with distributed collectors. In these systems, a fluid is heated using the energy provided by the solar irradiation until a desired outlet temperature range is achieved, despite of the effect of disturbances (mainly radiation and inlet temperature),using as manipulated variable the fluid flow. The heated fluid is then used for feeding a heat exchanger where steam is produced for electricity generation using a turbine. Nonlinear models are used in the design of the different layers of the control architecture.

1 Introduction

The objective of the work presented here is to optimize the electricity production process in solar power plants with a distributed collector system (DCS). The parabolic-trough solar field ACUREX used as a test-bed in this work is a facility belonging to the Plataforma Solar de Almería (PSA) (Southern of Spain). The solar field consists of 480 modules of collectors, distributed in 10 parallel rows. Each collector is made up of a reflecting parabolic surface that concentrates the direct solar irradiation in a pipe placed in the parabola focal line. The solar ray concentration in the pipe allows to heat the fluid circulating through inside. The collectors are oriented from the east to the west and have one axis (elevation) solar tracking system to guarantee the continuous concentration of the solar irradiation in the absorber pipe (figure 1). The heat transfer fluid used is thermal oil that can be heated up to 300^{o}C maximum. The oil is inside a thermal storage tank and it is extracted from the bottom by means of a pump to be heated in the field. The hot oil is returned to the top (process oil) or to the bottom of the tank (recirculation until get a nominal working point) by using a three-way

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 614–620, 2005.
© Springer-Verlag Berlin Heidelberg 2005

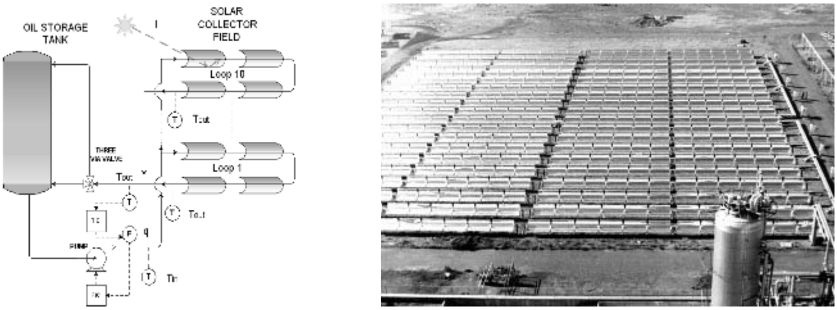


Fig. 1. Diagram and picture of the ACUREX field

valve placed at the outlet of the solar field. The oil properties permit stratified energy storage according to its density. The hot oil, that is taken from the top of the tank, can be used in an electricity generation process or for desalination, so it is convenient to avoid fluctuations at the outlet of the solar field. A complete description can be found in [1].

A hierarchical control architecture is proposed in this paper trying to optimize the electricity production process. This problem involves systems with different dynamical behavior and time scales, so that a typical control solution consists in using a multilayer hierarchical architecture, where the control of an objective is split into algorithms or layers, each of which acts at different time intervals in which the dynamic optimisation horizon has been divided [2].

2 Hierarchical Control Architecture

The hierarchical control architecture proposed in this paper, following the ideas in [3] and [4], is shown in figure 2. It is composed of the following layers:

1. The low level control problem, consisting in optimizing outlet temperature setpoint tracking and disturbance rejection. This level is usually implemented by means of classical, optimal feedback-feedforward control, or nonlinear control (in this case, a feedback linearization technique is applied).
2. The setpoint optimization problem, characterized by the need of obtaining adequate temperature setpoints (to maximize energy production) taking into account the operating conditions of the plant and constraints.
3. The daily operation problem, where the operating hours are defined.
4. The production planing problem, to schedule the number of days and operating hours a week the plant must be optimally operated.

The implementation of the hierarchical control architecture is based on first principles based physical models of the solar plant. In order to optimize the electricity production problem, there are some optimization problems involved (optimizing the outlet temperature regulation, maximize the energy accumulated in the oil storage tank, etc.) looking for the maximum efficiency of obtaining energy, that have to fulfill some constraints resulting from physical and technical

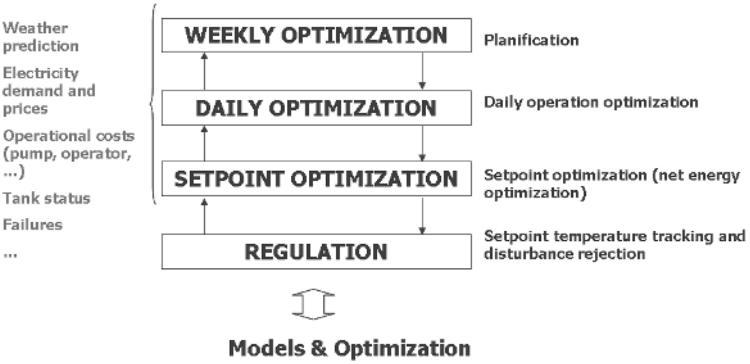

Weather prediction
Electricity demand and prices
Operational costs (pump, operator, ...)
Tank status
Failures
...
WEEKLY OPTIMIZATION
DAILY OPTIMIZATION
SETPOINT OPTIMIZATION
REGULATION
Planification
Daily operation optimization
Setpoint optimization (net energy optimization)
Setpoint temperature tracking and disturbance rejection
Models & Optimization

Fig. 2. Hierarchical control architecture

restrictions. It results in a nonlinear dynamic optimization problem with nonlinear constraints. Required simplifications lead us from dynamic to quasi static, distributed to concentrated or nonlinear to locally linear models.

2.1 Modeling and Simulation Approaches

In the development of the hierarchical control architecture, fundamentals models of DCS have been used for simulation, control and setpoint optimization purposes. The dynamics of the DCS are described by the following system of partial differential equations describing the energy balance [1]:

$$\rho_m c_m A_m \frac{\partial T_m}{\partial t}(t,x) = \alpha G I(t) - H_l G(T_m(t,x) - T_a(t)) - L H_t(T_m(t,x) - T_f(t,x)) \quad (1)$$

$$\rho_f c_f A_f \frac{\partial T_f}{\partial t}(t,x) + \rho_f c_f q(t) \frac{\partial T_f}{\partial x}(t,x) = L H_t(T_m(t,x) - T_f(t,x)) \quad (2)$$

where the subindex m refers to the metal and that of f to the fluid and all the parameters and variables are: t: time, x: space; ρ: density; c: specific heat capacity; A: cross-sectional area; $T(t,x)$: temperature; $q(t)$: oil pump volumetric flow rate; $I(t)$: corrected solar radiation, dependent on solar hour and date; α: mirror optical efficiency; G: mirror aperture; $T_a(t)$: ambient temperature; H_l: global coefficient of thermal losses; H_t: coefficient of metal-fluid transmission; L: inner diameter of the pipe line; l: tube length; $T_{in}(t)$: inlet oil temperature; $T_{out}(t)$: outlet oil temperature. Boundary conditions are $T(t,0) = T_{in}(t)$, $T_{out}(t) = T(t,l)$. Models (1) and (2) have been used to develop a simulator of the solar plant [5], and also for optimization and control purposes, where also static and lumped parameters versions have been obtained [6]. In this last case, the simplified equations are given by:

$$\rho(T_m)c(T_m)A\frac{\partial T_{out}}{\partial t} = \alpha G I - \rho(T_m)c(T_m)Aq\frac{T_{out} - T_{in}}{L} - \frac{\hat{H}_l(T_a, T_m)}{L_t} \quad (3)$$

where $T_m = (T_{out} + T_{in})/2$, L_t is the total solar field length and $\hat{H}_l$ is a corrected thermal losses function. The expressions of temperature-dependent variables and the values of the parameters in (3) can be found in [6]. This model has been used for control purposes within feedforward and feedback linearization control schemes. The feedforward controllers have been extensively explained in [1] and are based on a static version of equation (3), taking into account values of solar radiation and inlet oil temperature and helping to linearize the behavior of the controlled system. These feedforward controllers can be placed in series or in parallel with the feedback controller [1], with is treated in the next section.

2.2 Control Layer

The control problem of a solar collector field is to keep the outlet temperature of the field near a desired level (temperature set point or reference). This value should match the inlet conditions to feed a turbine (around 285^{o}C). Moreover, in order to avoid stress in the material of the absorber pipes, the outlet temperature should not be over 80^{o}C of the inlet temperature. The plant is exposed to non manipulated disturbances: inlet temperature changes (because of the stratification inside the tank), ambient temperature variations and solar irradiation changes due to the daily cycle or passing clouds causing fast variations in the outlet oil temperature. The manipulated signal is the velocity (or flow) of the oil propelled by the pump, that is constrained in the range $2 \cdot 10^{-3} - 12 \cdot 10^{-3}$ m^3 s^{-1}. The nonlinearities that characterize this plant increase the difficulty to operate it. According to the input, output and disturbance values, the characteristic gains, time delays and time constants change. The more difficult stage of the plant operation is the start up. In this control layer any control strategy can be used. Several examples of advanced control schemes can be found in [1]. Using the model in (3), classical feedback linearization techniques and feedback linearizing predictive control schemes are being developed. This level typically requires predicting the evolution of the disturbances in an horizon of about 30 minutes.

Under several simplifications and summarizing the design for saving space, the system in equation (3) can be described by the following equation:

$$\dot{x} = f(x) + b(x)u \rightarrow u = \frac{v - f(x)}{b(x)} \tag{4}$$

where u is the control signal (oil flow) and a nonlinear mapping has been used to transform the system into a linear one, with the condition that $b(x)$ can not be equal to zero (no thermal inversion occurs), v being the virtual control signal. With this nonlinear mapping the transformed system becomes an integrator and can be controlled using any control scheme. One example of the application of this technique is shown in [6].

The same idea can be used within a model predictive control framework, following the ideas in [7]. Figure 3 summarize the underlying ideas. In discrete time, the system can be represented by:

$$x(k+1) = f(x(k), u(k)); y(k) = h(x(k)); f(0,0) = h(0) = 0 \tag{5}$$

$$x(k+1) = Ax(k) + \underbrace{Fx(k)u(k) + \gamma(w(k))u(k) + \Gamma(w(k))}_{v(k)} \tag{6}$$

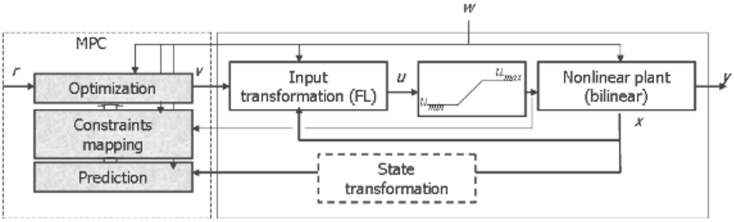


Fig. 3. Model predictive feedback linearizing control

$v(k)$ being again the virtual control signal and coefficients of matrices A, F, γ and Γ can be considered temperature dependent for nonlinear control purposes, belonging to a set of values for robust control purposes or constant for bilinear control issues. When using this nonlinear mapping within a model predictive control framework, the most interesting elements that have to be considered and that are object of research at present are [7]: mechanisms for mapping real constraints to virtual control signal constraints and convexity analysis. (notice that in this case the virtual control signal constraints depend on the state of the system and the disturbances, so that, even under conservative approaches, unfeasibility problems may arise), closed loop stability, and inclusion of the influence of measurable disturbances and uncertainty.

2.3 Setpoint Optimization Layer

The role of the setpoint optimizer is to find the most adequate reference temperature level taking into account the disturbance values (solar radiation, inlet oil temperature, ambient temperature, etc.) and minimizing energy consumption by the elements of the plant (e.g. the pump). If the heat losses are negligible, the optimal temperature setpoint will be near the maximum temperature achievable under the actual operating conditions and taking into account the constraints affecting the plant operation (mainly contraints in oil flow rate and maximum temperature and temperature difference). A static version of model (1) and (2) is used in this layer. This level also requires predicting the evolution of the disturbances in an horizon of about 2 hours and has demonstrated to be a valuable tool during the start-up of the operation. An example of the application of the setpoint optimization can be found in [8].

2.4 Daily and Seasonal Operation Optimization

The objective of this layer is to determine the time when the operation has to be started and finished each day, according to the weather predictions, requirements of the electrical network, electricity demand and prices, operational costs (pump, operators, etc.), ambient conditions, storage tank status, failures, etc. Several computer tools are being developed to help performing this daily and seasonal operation optimization. A data mining approach [9] has been developed aiming at discovering relationships and patterns in data so that predictions can be performed and used in this layer (e.g. failures, operating conditions, etc.). In order to predict the evolution of disturbances both for setpoint optimization and daily and seasonal optimization, another software tool initially developed within the framework of greenhouse crop production has been used [4]. The basic applied ideas are the following: (1) obtain the weather forecast for the next four days from the National Institute of Meteorology; (2)based on this information, assign a value for each variable (minimum temperature and temperature, and type of day as function of the radiation or cloudiness); search the parameterised historical database looking for four day with the closest values to those of the previous paragraph; (4) the long term weather forecast is determined by selecting four days and the consecutive following days until completing the horizon.

3 Conclusions

A hierarchical control architecture has been presented aimed at optimizing electricity generation in solar power plants with distributed collectors. The layers of the architecture have been briefly explained, providing references to related works. The bottom layers, in which basic temperature output regulation and setpoint optimization is carried out, are well-known and many control algorithms have been developed. The effort is nowadays focused on developing those elements required to perform daily and seasonal optimization, which is a hard issue due to the difficulties found in long-term prediction of weather and electricity demands. Several tools are being developed to help solving these problems.

Acknowledgements

This work has been financed by CICYT-FEDER funds (projects DPI2002-04375-C03 and DPI2004-07444-C04-04) and performed within the scope of the specific collaboration agreement between the CIEMAT and the Automatic Control, Electronics and Robotics (TEP197) research group of the Universidad de Almería titled ”Development of control systems and tools for thermosolar plants”. The authors would like to acknowledge Prof. F. Rodríguez for his help.

References

1. Camacho, E.F., Berenguel, M., Rubio, F.R.: Advanced control of solar plants. Springer-Verlag, London, UK (1997)
2. Findeisen, W., Bailey, F.N., Brdys, M., Malinowski, K., Tatjewski, P., Wozniak, A.: Control and coordination in hierarchical systems. John Wiley & Sons, USA (1980)
3. Cirre, C.M.: Hierarchical control of distributed solar collector fields. PhD. Thesis (in preparation). University of Almera, SPAIN (2005)
4. Rodríguez, F., Berenguel, M., Arahal, M.R.: A hierarchical control system for maximizing profit in greenhouse crop production. ECC2003, Cambridge, UK (2003).

5. Berenguel, M., Camacho, E.F., Rubio, F.R.: Simulation software package for the ACUREX field. Internal report ESI Sevilla. http://www.esi.us.es/rubio (1993)
6. Cirre, C.M., Valenzuela, L., Berenguel, M., Camacho, E.F.: Feedback linearization control for a distributed solar collector field. 16th IFAC World Congress, Praha (2005)
7. Kurtz, M.J., Henson, M.: Feedback linearising control of discrete-time nonlinear systems with input contraints. Int. J. Control, **70**(4) (1998) 603–616
8. Cirre, C.M., Valenzuela, L., Berenguel, M., Camacho, E.F.: Control de plantas solares con generación automática de consignas (in Spanish). RIAI, **1** (2004) 46–52.
9. Berenguel, M., Klempous, R., Maciejewski, H., Nikodem, J., Nikodem, M., Valenzuela, L.: Data Analysis of a Distributed Solar Collector Field Computer Aided Systems Theory - EUROCAST 2005 (2005)

Explanatory Analysis of Data from a Distributed Solar Collector Field

Manuel Berenguel[1], Ryszard Klempous[2], Henryk Maciejewski[2],
Jan Nikodem[2], Maciej Nikodem[2], and Loreto Valenzuela[3]

[1] University of Almeria, Spain
[2] Wroclaw University of Technology, Poland
[3] CIEMAT, Plataforma Solar de Almeria, Spain

Abstract. This work is devoted to application of explanatory data analysis in the field of solar plant control and monitoring. Data analysis tasks are discussed that allow discovering useful plant knowledge based on historical data obtained from plant control and monitoring systems. Approaches discussed include both OLAP analysis of plant monitoring data (i.e. multidimensional analysis based on a data warehouse), as well as data mining on plant monitoring data. Some of the data analysis tasks discussed have been realized based on monitoring data from distributed solar collector fields at Plataforma Solar de Almeria in Spain.

1 Introduction

Monitoring system at a complex industrial facility such as a distributed solar collector field gathers vast quantities of data over time. For instance, state monitoring system of the Direct Steam Generation (DSG) facility at the Plataforma Solar de Almeria in Southern Spain generates about 700 monitoring parameters (sensor readouts) every 5 seconds. The parameters refer to inlet and outlet temperatures measured for individual solar collectors, injection water temperature and outlet steam temperature, pressure levels measured at several points of the facility (pumps, tanks, collectors, injectors, deareator, etc.), water and steam flow rates, environmental conditions (solar radiation for individual collectors, ambient temperature, wind speed and direction). Also the complete vector of set point order values sent by the control system to collectors, valves and pumps is registered, as well as position values and status values of collectors, valves and pumps and other elements of the distributed facility. Similar monitoring system is implemented at the ACUREX field at the Plataforma Solar - a facility composed of parabolic collectors with oil used as heat transfer medium. By systematic explanatory analysis of the monitoring data gathered, interesting relationships can be discovered pertaining to the plant operation at different environmental conditions or to scenarios leading to failures. For example, by looking for monitoring parameters with strong outliers, one may detect sensor misbehavior periods, or by detecting patterns in monitoring historical data preceding points in time of plant abnormal condition, one may build knowledge to

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 621–626, 2005.
© Springer-Verlag Berlin Heidelberg 2005

predict abnormal conditions in advance. Similar tasks to extract useful knowledge from industrial plant database explanatory analysis have been reported in literature. For example, Singhal and Seborg ([10] and [11]) presented ideas on condition monitoring of electrical power plant equipment based on data analysis, and Wehenkel et al. (see [13]) discussed detection of abnormal plant operation based on analysis of historical data. Other interesting applications are given in [2], [4], [3], [12] and [5]. The explanatory data analysis discussed in this work follows two approaches: (a) multidimensional analysis along the OLAP (OnLine Analytical Processing) paradigm and (b) data mining based approach. These two will be characterized in the subsequent sections.

2 OLAP Approach to Analysis of Data from Collector Field

OLAP (online analytical processing) approach to explanatory data analysis requires that data to be analyzed is stored in a data warehouse built from raw monitoring data (data preprocessing stage consists in data integration, cleaning and appropriate indexing for query efficiency). Based on such preprocessed data, users can construct multidimensional queries, e.g., in the form of a dependent variable as a function of one or two independent variables, under specified values of parameter variables. Data analysis is performed using a Web based tool through which users define their multidimensional queries. The Web-based front end passes the query to the data warehouse server for processing; query results are then returned to users' web browsers. This approach offers several advantages over the SQL-query-raw-data approach. These include:

- or analysis is preprocessed prior to analysis that allows for integration of different sources and assurance of data quality,
- for large data volumes, analysis time can be reduced due to efficient data storage from the point of view of data querying, e.g. using denormalized database schemes which require fewer table joins, known in data warehousing technologies as ROLAP (star- or snowflake schemes),
- analysis does not require any programming skills - can be realized solely with HTML form-type interface,
- analysis does not require any knowledge about data organization in the monitoring databases (table names, variable names, data types etc.).

The pilot version of the system, built using SAS software tools is available through the Web page: http://rush.ict.pwr.wroc.pl/ac (user: acurex, passwd: manolo).

3 Data Mining on Monitoring Data from Collector Field

The application of data-mining technologies can be useful in the context of solar plant control and monitoring for searching patterns or studying different features within the following fields.

3.1 Study of Structural Features of the Plant

Structural features of the plant can be analyzed in the context of:

- System degradation - for instance, using data of temperature, can be studied if the oil properties are deteriorating with time (using the equations provided by the provider). It is probable that some off-line measurements of oil density should be necessary. The same could be applied to mirrors reflectivity, being necessary to include data of mirrors cleaning, repairing, etc. within the database.
- Detection of constructive features of the plant - under the same environmental and input conditions (input flow, solar radiation, inlet oil temperature and ambient temperature), analysis of the main differences in performance of the 10 loops can be of interest for control purposes. E.g., if data reveals some specific deterministic features in behavior of some of the rows (for instance, one row has always several Celsius degrees more than another row), then this information could be used for control purposes (for instance, for set point generation).

3.2 Dynamic Performance, Modeling and Control Design

Specific studies that can be defined in this group are:

- Study of the influence of environmental variables in the states of the system. In principle, this can be realized using the explanatory data analysis approach presented in the previous section. It should be of interest to see and calculate which is the relative weight of solar radiation, inlet temperature and ambient temperature on outlet temperature, to compare these statistical data with existing physical models of the field.
- Study of static gains of the system from the energy point of view in the face of different exogenous signal profiles. Study of maximum and minimum temperature increments.
- Study of the temperature profiles during the night. This could serve to characterize night losses and to predict the temperature profile within the field and storage tank to select the most adequate starting operation set point profile.
- Detection of zones of "massive operation": it should be important to determine those operating points in which the plant usually operates. This can be interesting both from the modeling and control viewpoints. From the modeling viewpoint, it may be interesting to put more emphasis and modeling effort in obtaining more reliable models for these zones, and only approximated models for other zones in which operation is not usual (moreover, as operation is not usual, poor data streams are often available for model calibration). From the control viewpoint, there are several control strategies (as fuzzy logic control) that use the concept of "control surface" - e.g., work by Rubio et al. ([9]), or Berenguel et al. ([1]). The building of this control

surface could be a time-consuming task if many operating points are selected and extrapolation is to be avoided. So, the controller can be designed more carefully for those zones of massive operation and conservative control signals can be applied to outer zones.

- Controller design: examples could be for fuzzy controllers, feed-forward controllers, artificial neural networks controllers, etc. It should also be of interest to use data of manual operation by expert controllers to compare behavior with automatic controllers and to use these data for expert control design.
- Oscillatory behavior: under which operating conditions (independent of the control algorithm) the system presents oscillatory behavior or very slow behavior.
- Detection of conditions leading to antiresonance behavior: resonances are characteristic of this kind of installation ([7]). The antiresonance modes are defined by frequencies at which the magnitude of the frequency response of the system changes abruptly (when exciting the system with a signal with principal frequency components corresponding to those of the antiresonance modes, variations at the system output are very small). It can be useful to discover this in the data by using several indexes: detect if the oil flow or the solar radiation has a large frequency content within the "dangerous" band (fast Fourier transform should be applied to the data) and see if the output temperature suffers from oscillations and the spatial temperature distribution within the collectors is not monotonous. The excitation of the resonance modes depends on the operation point.
- Detection of plant/model mismatch. If output of different models for the same environmental conditions is entered in the database, plant/model mismatch can be studied.

3.3 Disturbance Prediction

Application of data mining to help build disturbance models is of great interest. When using optimal/predictive/receding horizon control strategies, it is necessary to have a model of disturbances in a time window of about one hour. One option to try is to use a "lazy man" weather approach by using the actual measured value of the disturbance as that in the following hour, or using a clear-day solar radiation prediction. Another approach can be to use the previous day data as prediction, what is also typical in greenhouse climate control. In other applications (adopted e.g., in greenhouse climate control or in the case of prediction of electrical demand), the entire database is searched for a pattern of three days which weather is quite similar of the last three days of this year and then use the following day (fourth day) as a prediction for the current weather, Rodriguez et al. (see [8]). This is because some people think that weather patterns tend to be repeated but in different days each year. For instance, if you have rain during three days, there is a large probability to have a non-raining day the fourth one and so on. So, the explanatory data analysis/data mining approach can be used to search for this kind of patterns to select the appropriate day to be used for disturbance (mainly solar radiation and outlet temperature) prediction purposes.

3.4 Fault Prediction and Tolerance and Diagnosis of Abnormal Plant Operation

Specific studies of interest in this group are: Detection of outliers in the rough data: this may indicate having wrong sensor data. Prediction of abnormal plant situations. By searching patterns in the database trying to detect abnormal plant situations and/or faults (plant to de-steer, pump fail, one collector row fail, strong oscillatory behaviors, etc.), data mining/explanatory data analysis can be used to develop a supervisory system able to help the operators avoid the system entering in these dangerous situations in advance. Performance Evaluation Specific studies in this group are:

- Model evaluation: comparing the output of several models (linear and non-linear) with real plant data to select which models are more suitable for different operating points or operating conditions. This could be implemented by injecting the output of different simulation and control models in the database using the environmental data corresponding with the real plant output.
- Control evaluation: as partly illustrated by results in the previous section, evaluation of different control algorithms could be done on-line in terms of some performance indexes, mainly related with the error integrals (as ISE, IAE, ITAE). After including information about what control algorithms were tested on specific dates/time periods, one can investigate what the data says about performance of the algorithms under similar environmental conditions. The tool could be also used as a help to establish set-point policies depending on operating and even economical conditions. For this type of analyses, more information than shown in the examples above has to be included in the database (economical, turbine related, losses, etc.).

3.5 Training Purposes

The explanatory data analysis approach / tool can be of interest to train non-expert operators for future solar plants, using different situations obtained from data mining (for instance, we can characterize abnormal plant situations - see Fault prediction - and ask the operator what should he/she do under these circumstances and what was done by an expert operator or an automatic control system.

Acknowledgements. This work has been financed by CICYT-FEDER funds (projects DPI2002-04375-C03 and DPI2004-07444-C04-04) and by the Consejeria de Innovacion, Ciencia y Empresa de la Junta de Andalucia. This work has been also performed within the scope of the specific collaboration agreement between the Plataforma Solar de Almeria and the Automatic Control, Electronics and Robotics (TEP197) research group of the Universidad de Almeria titled "Development of control systems and tools for thermosolar plants". The authors would like to acknowledge Prof. F. Rodriguez for his help in the development of the hierarchical control architecture.

References

1. Berenguel, M., Camacho, E.F., Rubio, F.R., Luk, P.C.K.: Incremental Fuzzy PI Control of a Solar Power Plant. IEE Proceedings - Control Theory and Applications (Part D) **144** (1997) 596-604
2. Booth, C.; McDonald, J. R. ; McArthur, S. D. J.: Forecasting and Prediction Applications in the Field of Power Engineering. Journal of Intelligent and Robotic Systems **31(1/3)** (2001)
3. Cser L., et al,: Data Mining and State Monitoring in Hot Rolling. IEEE (1999)
4. Hou, Tung-Hsu (Tony); Liu, Wang-Lin; Lin, Li: Intelligent remote monitoring and diagnosis of manufacturing processes using an integrated approach of neural networks and rough sets. Journal of Intelligent Manufacturing **14(2)** (2003)
5. Lian-Yin Zhai, Li-Pheng Khoo, Sai-Cheong Fok: Feature extraction using rough set theory and genetic algorithms- an application for the simplification of product quality evaluation. Computers and Industrial Engineering **43** (2002)
6. Liao S.: Knowledge management technologies and applications - literature review from 1995 to 2002. Expert Systems with Applications (2003)
7. Meaburn, A., Hughes, F.M.: Resonance characteristics of distributed solar collector fields. Solar Energy, **51(3)** (1993) 215–221
8. Rodrguez, F., Berenguel, M., Arahal, M.: A hierarchical control system for maximizing profit in greenhouse crop production. European Control Conference ECC03, Cambridge, UK, (2003)
9. Rubio, F.R., Berenguel, M., Camacho, E.F.: Fuzzy Logic Control of a Solar Power Plant. IEEE Transactions on Fuzzy Systems **3** (1995) 459–468
10. Singhal A., Seborg D.E.: Matching Patterns from Historical Data Using PCA and Distance Similarity Factors. Proceedings of the American Control Conference, Arlington (2001)
11. Singhal A., Seborg D.E.: Pattern Matching in Historical Batch Data Using PCA. IEEE Control Systems Magazine (2002)
12. Steele J.A., McDonald J.R., D'Arcy C.: Knowledge Discovery in Databases: Applications the Electrical Power Engineering Domain. IEE (1998)
13. Wehenkel L., Lebrevelec C., Trotignon M., Batut J.: Probabilitic design of power-system special stability controls. Control Engineering Practice (1999)

Author Index

Abalde, Carlos 121
Aguila, Isabel M. del 217
Alemán-Flores, Miguel 467
Alemán-Flores, Patricia 467
Alonso, José A. 571
Alonso, Javier 235
Alonso, Jesús B. 379
Alonso, Miguel A. 169
Álvarez-León, Luis 467
Amo, I.J. García del 199
Aransay, Jesús 159
Argüelles, Luís 33
Armingol, J. Ma 537
Astola, Jaakko 291

Bailón, Antonio 189
Ballarin, Clemens 159
Barcala, Fco. Mario 91
Barreiro, Álvaro 254
Batista, B. Melián 199
Berenguel, Manuel 614, 621
Bergasa, Luis Miguel 507
Bernardos, Pilar 101
Biernat, Janusz 327
Blanco, Antonio 69
Blanco, Francisco J. 596
Blanco, Roi 254
Blasio, Gabriel de 483
Borrego-Díaz, Joaquín 247
Bosch, Alfonso 189, 269
Brugos, José Antonio López 337
Bubnicki, Zdzislaw 38
Buonocore, Aniello 426

Cárdenes, Rubén 392
Caballero-Gil, Pino 346, 355
Cabello, Enrique 525
Cabrera-Gámez, Jorge 590
Campos, Manuel 241
Canton, Manuel 373
Caputo, Ludovico 426
Carrillo, Vicente 150
Castrillón-Santana, Modesto 398
Češka, Milan 275

Chávez-González, Antonia M. 247
Cirre, Cristina M. 614
Claudio, Dalcidio Moraes 9
Cobas, Juan David González 337
Collado, J.M. 537
Combarro, E.F. 179
Conde, Cristina 525
Crescenzo, Antonio Di 436
Crespo, José L. 101
Cruz, Fermín 150
Cull, Paul 457
Curto, Belén 596

Díaz, I. 179
Díaz, Víctor J. 150
Déniz-Suárez, Oscar 398
Delgado, Ana E. 531
Domínguez, Eva 91
Domínguez-Brito, Antonio C. 590

Earle, Clara Benac 140
Edtmayr, Johann 602
Egiazarian, Karen 291
Enríquez, Fernando 150
Escalera, A. de la 537
Esteban-Sánchez, M. Belén 467

Fúster-Sabater, Amparo 355
Falcón, Néstor 379
Fernández, Miguel A. 531
Fernández-Caballero, Antonio 531
Fernández-Pérez, José L. 590
Ferrer, Miguel A. 379
Fredlund, Lars-Åke 140
Freire, Enrique 69
Freire, José L. 69, 111
Fuentes-Pavón, Rafael 467

Galan, Manuel 404
García, Ricardo 33, 501, 519
García-Cabrera, Lina 223
García-Garrido, Miguel Ángel 543
Giorno, Virginia 416, 447
Gollmann, Dieter 365
González, Carla 608

González, Carlos 33, 501, 519
Gotchev, Atanas 291
Graña, Jorge 131
Grossmann, Wilfried 63
Guerra-Artal, Cayetano 398
Guil, Francisco 189, 217, 269
Guindos, Francisco 373
Gulías, Víctor M. 111, 121

Hamann, Thomas K. 56
Hernández-Goya, Candelaria 346
Hernández-Sosa, Daniel 590
Hilario, C. 537
Hoefer, Gerhard 281
Hoff, Marnes Augusto 15
Horák, Bobumil 565
Hurtado, M. Visitacíon 223

Ildefonso, Tiago 81
Isern-González, José 398
Izquierdo, Raul 235

Jang, Min-Soo 410
Jochinger, Dominik 311
Jorge, J. Santiago 111
Juarez, José M. 241

Küng, Josef 211
Křena, Bohuslav 275
Karpovsky, Mark G. 301
Kim, Sang-Jun 410
Kim, Yong-Guk 410
Klempous, Ryszard 614, 621
Kopacek, Peter 549, 584

López, María T. 531
López-Valles, José M. 531
Leal, Liara Aparecida dos Santos 9
Lee, Jeong-Eom 410
Lee, Soek-Joo 410
Licznerski, Benedykt W. 27
Lope, Javier de 513
Lopes, Gabriel Pereira 81

Machado, Júlio Pereira 50
Maciejewski, Henryk 614, 621
Maravall, Darío 513
Mariño, Castor 386
Marín, D. 205
Marín, Roque 189, 241, 269
Martín, Raquel Rivero 199
Martín-Gorostiza, Ernesto 543
Martín-Mateos, Francisco J. 571
Martin, José Antonio 33
Martinucci, Barbara 436
Mata, Javier de la 229
Mauerkirchner, Manfred 281
Mazurowski, Maciej 27
Medina, Nuria 223
Menasalvas, E. 205
Menezes, Paulo Blauth 9, 15, 44, 50
Miró-Julià, Margaret 21
Mira, J. 1, 531
Mitterauer, Bernhard 577
Molina, Alberto 373
Molinero, Miguel A. 91
Mones, R. 179
Montañés, E. 179
Montes, Raquel 525
Mora, Eduardo 101, 205
Morales, Antonio 241
Moreno, Vidal 596
Moreno-Díaz, Arminda 483
Moreno-Díaz, Roberto 483
Moreno-Díaz, Roberto Jr. 492
Moschner, Markus 63
Munteanu, Cristian 404

Naranjo, José E. 519
Nikodem, Jan 614, 621
Nikodem, Maciej 327, 614, 621
Nobile, Amelia G. 416, 447
Nuevo, Jesús 507

Obitko, Marek 565
Ocaña, Manuel 507
Olivas, Jose A. 229
Orellana, Francisco J. 217
Otero, Juan 131

Pérez, J.A. Moreno 199
Palma, José 241
Palomo, Manuel 571
París, Javier 121
Park, Gwi-Tae 410
Patete, Anna 608
Pedro, Teresa de 33, 501, 519
Penas, Marta 386
Penedo, Manuel G. 386
Pfalzgraf, Jochen 577

Pichler, Franz 27, 311
Piedra, Jose Antonio 373
Pirozzi, Enrica 426, 436, 447
Pitzer, Erik 473
Pogossova, Ekaterina 291

Quesada-Arencibia, A. 492
Quintana-Domínguez, Francisca 263

Ranilla, J. 179
Resconi, Germano 235
Reusch, Bernd 559
Ribadas, Francisco J. 169
Ricciardi, Luigi M. 416, 426
Rivas, Jose M. 33
Rodríguez, María L. 223
Rodríguez-Aragón, Licesio J. 525
Rodríguez-Millán, Jesús 608
Rodríguez-Rodríguez, Abraham 263
Rodríguez-Rodríguez, J.C. 492
Roggia, Karina Girardi 15
Rosa, Agostinho 404
Royo, Enrique Rubio 404
Rubio, Julio 159
Rudas, Imre 614
Ruiz, Javier Alonso 501
Ruiz-Alzola, Juan 392

Sánchez, Juan J. 111
Salort, José 241
Santana-Montesdeoca, José M. 467
Scharinger, Josef 317
Schierer, Edmund 584
Schwaninger, Markus 56
Segovia, J. 205
Serrano, Ángel 525
Serrano-Guerrero, Jesús 229
Smid, Jan 565
Snášel, Václav 565
Sotelo, Miguel Ángel 507, 543
Stanković, Radomir S. 301
Szczurek, Andrzej 27
Szecówka, Przemyslaw M. 27

Túnez, Samuel 217
Therón, Roberto 596
Toraño, Javier 33
Torres, M. García 199
Toscani, Laira Vieira 9
Travieso, Carlos M. 379
Troyano, José A. 150

Vázquez, N. 179
Valenzuela, Loreto 614, 621
Vega, J.M. Moreno 199
Vieira, Rodrigo Born 44
Vilares, Jesús 169
Vilares, Manuel 131
Vojnar, Tomáš 275

Wagner, Roland 211
Weiss, Norman 559
Wuerzl, Markus 584

Zorrilla, Marta E. 101, 205