Bruno Apolloni
Maria Marinaro
Giuseppe Nicosia
Roberto Tagliaferri (Eds.)

LNCS 3931

Neural Nets

16th Italian Workshop on Neural Nets, WIRN 2005
and International Workshop on Natural
and Artificial Immune Systems, NAIS 2005
Vietri sul Mare, Italy, June 2005
Revised Selected Papers

Springer

Lecture Notes in Computer Science 3931

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

Editorial Board

Bruno Apolloni Maria Marinaro
Giuseppe Nicosia Roberto Tagliaferri (Eds.)

Neural Nets

16th Italian Workshop on Neural Nets, WIRN 2005
and International Workshop on Natural
and Artificial Immune Systems, NAIS 2005
Vietri sul Mare, Italy, June 8-11, 2005
Revised Selected Papers

Springer

Volume Editors

Bruno Apolloni
Università degli Studi di Milano, Dipartimento di Scienze dell'Informazione
Via Comelico 39/41, 20135 Milano, Italy
E-mail: apolloni@dsi.unimi.it

Maria Marinaro
International Institute for Advanced Scientific Studies
Via G. Pellegrino 19, 84019 Vietri sul Mare (Salerno), Italy
E-mail: iiass.direzione@tin.it

Giuseppe Nicosia
Università di Catania, Dipartimento di Matematica e Informatica
Viale A. Doria, 6, 95125 Catania, Italy
E-mail: nicosia@dmi.unict.it

Roberto Tagliaferri
Università di Salerno, Dipartimento di Matematica e Informatica
via Ponte don Mellilo, 84048 Fisciano (Salerno), Italy
E-mail: rtagliaferri@unisa.it

Library of Congress Control Number: 2006922801

CR Subject Classification (1998): F.1.1, I.5, I.2.6, J.3, I.4, G.3, H.2.8

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-540-33183-2 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-33183-4 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11731177 06/3142 5 4 3 2 1 0

Preface

This volume reports the proceedings of the 16th Italian Workshop on Neural Nets WIRN 2005 and the satellite International Workshop on Natural and Artificial Immune Systems NAIS 2005. The workshops, held in Vietri sul Mare (SA) from 8 to 11 June, were jointly organized by the International Institute for Advanced Scientific Studies "Eduardo R. Caianiello" (IIASS) and the Società Italiana Reti Neuroniche (SIREN). The new format of the SIREN annual meeting—WIRN plus satellite conference—stems from the aim of sharing neural network methodologies with close research communities. The goal this year was to recognize and exploit a commonality of subsymbolic methods used both in neural networks and immune system research to study complex systems with a distinguishing common feature. All these systems are indeed constituted by a mass of elementary organisms/computational units generating ensemble functionalities, such as optimization and learning, denoting traits of intelligent thought. Moreover, like neural networks, the area of *natural and artificial immune systems* is also developing in response to needs for a mix of inter- and multidisciplinary expertise, spanning immunology, biology, mathematics, computer science, and medicine, among others, having rebounding of models and results from biological to computational realms and *vice versa* as a great strength point.

The volume contains invited review papers and selected original contributions presented in oral or poster sessions by both Italian and foreign researchers. The contributions have been assembled, for reading convenience, into eight sections. The first section reports the lecture given by the winner of the Premio Caianiello award. Sections II to V contain wide-spectrum papers on neural networks, split in four parts; models, architectures and algorithms, signal processing, and pattern recognition, respectively. The sixth section focuses on methodologies and applications of fuzzy and neurofuzzy computing promoted by a joint Spanish–Italian working group. The last two sections gather papers on immune systems divided into natural systems and artificial systems.

The editors would like to thank the invited speakers and all the contributors whose highly qualified papers helped the success of the workshops. Finally, special thanks go to the referees for their accurate work.

June 2005

Bruno Apolloni
Maria Marinaro
Giuseppe Nicosia
Roberto Tagliaferri

Organization

Organizing Scientific Commitee

B. Apolloni (Univ. Milano), A. Bertoni (Univ. Milano), N.A. Borghese (Univ. Milano), D.D.Caviglia (Univ. Genova), P. Campadelli (Univ. Milano), F. Celada (Univ. Genova), A. Chella (Univ. Palermo), A. Clivio (Univ. Milano), A. Colla (ELSAG Genova), V. Cutello (Univ. Catania), D. Dasgupta (Univ. Memphis, USA), F. Esponda (Univ. New Mexico, USA), A. Esposito (Univ. Napoli II), F. M. Frattale Mascioli (Univ. Roma "La Sapienza"), C. Furlanello (ITC-IRST Trento), S. Giove (Univ. Venezia), M. Gori (Univ. Siena), D. Lee (IBM-KAIST Bio-Computing Research Center, Korea), H. Linger (Monash Univ., Australia), M. Marinaro (Univ. Salerno), F. Masulli (Univ. Pisa), C. Morabito (Univ. Reggio Calabria), P. Morasso (Univ. Genova), P. Mussio (Univ. Milano), G. Nicosia (Univ. Catania), G. Orlandi (Univ. Roma "La Sapienza"), A. Pagnoni (Univ. Milano), T. Parisini (Univ. Trieste), E. Pasero (Politecnico Torino), A. Petrosino (Univ. Napoli "Parthenope"), V. Piuri (Univ. Milano), R. Serra (Univ. di Modena e Reggio Emilia), F. Sorbello (Univ. Palermo), A. Sperduti (Univ. Padova), R. Tagliaferri (Univ. Salerno), J. Timmis (Univ. Kent, UK)

Referees

Anguita D.
Bassis S.
Bertoni A.
Borghese A.
Campadelli P.
Capasso V.
Ceccarelli M.
Ciaramella A.
Clivio A.
Colla A.M.
Di Claudio E.
Eleuteri A.
Esponda F.
Esposito A.
Fiori S.
Frattale Mascioli F.M.
Furlanello C.
Giove S.
Lee D.
Malchiodi D.
Marinaro M.
Martinelli G.
Martone R.
Masulli F.
Morabito C.
Muselli M.
Nicosia, G.
Palmieri F.
Parisi R.
Parisini T.
Pasero E.
Pessa E.
Piuri V.
Raiconi G.
Serra R.
Sessa S.
Sperduti A.
Staiano A.
Tagliaferri R.
Timmis J.
Uncini A.
Zanaboni A.M.
Valentini G.

Sponsoring Institutions

International Institute for Advanced Scientific Studies (IIASS) "E.R. Caianiello"
Dip. di Fisica "E.R. Caianiello", University of Salerno

Dip. di Matematica ed Informatica, University of Salerno
Dip. di Scienze dell'Informazione, University of Milano
Dip. di Informatica e comunicazione, University of Milano
Dip. di Scienze Precliniche Lita Vialba, University of Milano
Dip. di Matematica e Informatica, University of Catania
Centro di Ricerca Ippari, University of Catania
Società Italiana Reti Neuroniche (SIREN)
IEEE Neural Network Society, Italian Chapter
INNS, SIG Italy
Artist, Network for Artificial Immune Systems
Istituto Italiano per gli Studi Filosofici, Napoli
Italian Chapter of IEEE Computational Intelligence Society
Public Health Italian Ministry
Provincia di Salerno

Acknowledgments

Special thanks go to Simone Bassis for his Web management. Furthermore thanks go to Tina Nappi and Ornella De Pasquale for their secretarial work and to Michele Donnarumma for his technical work.

Table of Contents

I Eduardo R. Caianiello Lecture

Kernel Methods for Clustering
Francesco Camastra 1

II Models

A New Neural Network Model for Contextual Processing of Graphs
Alessio Micheli, Antonio S. Sestito 10

Approximation Properties of Positive Boolean Functions
Marco Muselli 18

Switching Neural Networks: A New Connectionist Model for Classification
Marco Muselli 23

III Architectures and Algorithms

Ensembles Based on Random Projections to Improve the Accuracy of Clustering Algorithms
Alberto Bertoni, Giorgio Valentini 31

Recursive Neural Networks and Graphs: Dealing with Cycles
M. Bianchini, M. Gori, L. Sarti, F. Scarselli 38

A System for Transmitting a Coherent Burst of Activity Through a Network of Spiking Neurons
J. Bose, S.B. Furber, J.L. Shapiro 44

NEC: A Hierarchical Agglomerative Clustering Based on Fisher and Negentropy Information
Angelo Ciaramella, Giuseppe Longo, Antonino Staiano, Roberto Tagliaferri 49

A Recurrent ICA Approach to a Novel BSS Convolutive Nonlinear Problem
Daniele Vigliano, Raffaele Parisi, Aurelio Uncini 57

IV Signal Processing

Hourly Forecasting of SO_2 Pollutant Concentration Using an Elman Neural Network
U. Brunelli, V. Piazza, L. Pignato, F. Sorbello, S. Vitabile 65

Nonlinear Exploratory Data Analysis Applied to Seismic Signals
Antonietta M. Esposito, Silvia Scarpetta, Flora Giudicepietro, Stefano Masiello, Luca Pugliese, Anna Esposito 70

Artifact Cancellation from Electrocardiogram by Mixed Wavelet-ICA Filter
Fabio La Foresta, Nadia Mammone, Francesco Carlo Morabito 78

Intelligent Predictive Control of Micro Heat Exchanger
Mehdi Galily, Farzad Habibipour Roudsari, Masoum Fardis, Ali Yazdian 83

Transient Power System Stabilizer Design Using Simple Neuron Structure
Farzad Habibipour Roudsari, Behzad Jalaie, Mehdi Galily 90

Short Term Local Meteorological Forecasting Using Type-2 Fuzzy Systems
Arianna Mencattini, Marcello Salmeri, Stefano Bertazzoni, Roberto Lojacono, Eros Pasero, Walter Moniaci 95

Learning and Data Driver Methods for Short Term Meteo Forecast
Eros Pasero, Walter Moniaci 105

Automatic Dictionary Creation by Sub-symbolic Encoding of Words
Filippo Vella, Giovanni Pilato, Ignazio Motisi, Salvatore Gaglio 113

V Pattern Recognition

An Automatic Feature Based Face Authentication System
Stefano Arca, Paola Campadelli, Elena Casiraghi, Raffaella Lanzarotti 120

On the Preprocessing of Mass Spectrometry Proteomics Data
M. Cannataro, P.H. Guzzi, T. Mazza, G. Tradigo, P. Veltri 127

Clustering Causal Relationships in Genes Expression Data
Sergio Pozzi, Italo Zoppis, Giancarlo Mauri 132

Application of EαNets to Feature Recognition of Articulation Manner in Knowledge-Based Automatic Speech Recognition
Sabato M. Siniscalchi, Jinyu Li, Giovanni Pilato, Giorgio Vassallo, Mark A. Clements, Antonio Gentile, Filippo Sorbello 140

VI Fuzzy and Neurofuzzy Computing

Granular Regression
B. Apolloni, D. Iannizzi, D. Malchiodi, W. Pedrycz 147

Fuzzy Logic Activities at the Microelectronics Institute of Seville
Angel Barriga, Santiago Sánchez-Solano, Iluminada Baturone, Francisco Moreno-Velo, Piedad Brox, Federico Montesino, Nashaat M. Hussein, María Brox, Andrés Gersnoviez 157

Generalized Fuzzy Similarity Indexes
Narcís Clara 163

Environmental Time Series Prediction by Improved Classical Feed-Forward Neural Networks
Maurizio Campolo, Narcís Clara, Carlo Francesco Morabito 171

New Similarity Rules for Mining Data
Vito Di Gesù, Jerome H. Friedman 179

Image File Compression Using Region Growing and Interpolation
Antonio Di Nola, Nicla Paladino, Barnabás Bede 188

Fuzzy Connectivity and Its Application to Image Segmentation
Gabriele Martino, Alfredo Petrosino 197

Soft Rank Clustering
Stefano Rovetta, Francesco Masulli, Maurizio Filippone 207

VII Natural Immune Systems

An Evolution Hypothesis of Bacterial Populations
Bruno Apolloni, Alberto Clivio, Simone Bassis, Sabrina Gaito, Dario Malchiodi 214

Modelling the Immune System with Situated Agents
Stefania Bandini, Franco Celada, Sara Manzoni, Roberto Puzone, Giuseppe Vizzari 231

Current Paradigms in Immunology
Eugenio Cesana, Silvia Beltrami, Antonia Emanuela Laface, Astrid Urthaler, Alessandra Folci, Alberto Clivio 244

Supporting Collaborative Knowledge Work: A Methodology for Developing ICT Tools for Biomedical Research
Henry Linger 261

Profiling Network Attacks Via AIS
Anastasia Pagnoni, Andrea Visconti 272

Fuzzy Continuous Petri Net-Based Approach for Modeling Immune Systems
Inho Park, Dokyun Na, Doheon Lee, Kwang H. Lee 278

A General Learning Rule for Network Modeling of Neuroimmune Interactome
D. Remondini, P. Tieri, S. Valensin, E. Verondini, C. Franceschi, F. Bersani, G.C. Castellani 286

VIII Artificial Immune Systems

On Diversity and Artificial Immune Systems: Incorporating a Diversity Operator into aiNet
Paul S. Andrews, Jon Timmis 293

Lipschitzian Pattern Search and Immunological Algorithm with Quasi-Newton Method for the Protein Folding Problem: An Innovative Multistage Approach
A.M. Anile, V. Cutello, G. Narzisi, G. Nicosia, S. Spinella 307

A Clonal Selection Algorithm for Coloring, Hitting Set and Satisfiability Problems
Vincenzo Cutello, Giuseppe Nicosia 324

Artificial Immune-Based Optimization Technique for Solving Economic Dispatch in Power System
Titik Khawa Abdul Rahman, Saiful Izwan Suliman, Ismail Musirin 338

Securing IPv6-Based Mobile Ad Hoc Networks Through an Artificial Immune System
Julian L. Rrushi 346

Challenges for Artificial Immune Systems
Jon Timmis 355

Author Index 369

Kernel Methods for Clustering

Francesco Camastra

Department of Applied Science University of Naples "Parthenope",
via A. De Gasperi 5, 80133 Napoli, Italy
francesco.camastra@uniparthenope.it

Abstract. Kernel Methods are algorithms that implicitly perform, by replacing the inner product with an appropriate Mercer Kernel, a nonlinear mapping of the input data to a high dimensional Feature Space. In this paper, we describe a Kernel Method for clustering. The algorithm compares better with popular clustering algorithms, namely *K-Means*, *Neural Gas*, *Self Organizing Maps*, on a synthetic dataset and three UCI real data benchmarks, *IRIS data*, *Wisconsin breast cancer database*, *Spam database*.

1 Introduction

Kernel Methods [5] [22] are algorithms that implicitly perform, by replacing the inner product with an appropriate Mercer Kernel [2], a nonlinear mapping of the input data to a high dimensional Feature Space. Powerful supervised Kernel Methods have been developed to solve classification and regression problems. Recently some kernel methods for clustering have been proposed. They can be grouped in three big families. The first family is formed by algorithms [25] based on *Kernelising* the Metric, i.e. the metric is computed by means of a Mercer Kernel in a Feature Space. The second family [8][19] is formed by Kernel Methods that implements K-Means in the Feature Space. The third group is formed by Kernel Methods based on support vector data description. In this paper, for sake of brevity, we review an algorithm [4], a Kernel Method for Clustering, that belongs to the third family. The algorithm maps data from the Input Space to a high dimensional Feature Space using a Mercer Kernel. Then it considers K centers and computes for each center the smallest ball that encloses the data that are closest. The algorithm uses a K-Means-like strategy, i.e. it moves repeatedly the centers, computing for each center the smallest ball, until no center changes anymore. Unlike other popular clustering algorithms [10] that yield piecewise linear borders among data, the algorithm allows to get nonlinear separation surfaces in the data. This is the main quality of the algorithm. The plan of the paper is as follows. In Section 2 we review K-Means that inspired the algorithm; in Section 3 we describe One Class SVM, the basic step of the algorithm; in Section 4 we present the Kernel Method for Clustering; in Section 5 some experimental results are reported; finally some conclusions are drawn in Section 6.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 1–9, 2006.

2 K-Means

Let $D = (x_1, x_2, \ldots, x_m)$ be a data set with vectors $x_i \in \mathbb{R}^N$. We call *codebook* the set $W = (w_1, w_2, \ldots, w_{K-1}, w_K)$ where each element $w_c \in \mathbb{R}^N$ and $K \ll m$. The *Voronoi Region* (R_c) of the codevector w_c is the set of all vectors in $\mathbb{R}^N$ for which w_c is the *nearest codevector* $R_c = \{x_i \in \mathbb{R}^N \mid c = arg\min_j \ \|x_i - w_j\|\}$. The *Voronoi Set* ($V_c$) of the codevector w_c is the set of all vectors in D for which w_c is the *nearest vector* that is $V_c = \{x_i \in D \mid c = arg\min_j \ \|x_i - w_j\|\}$. The partition of $\mathbb{R}^N$ formed by all Voronoi regions is called *Voronoi Tessellation*. The most popular clustering technique is K-Means (or *Batch K-Means*) [13]. K-Means works by repeatedly moving all codevectors to the arithmetic mean of their Voronoi sets. The theoretical foundation of the algorithm is that it can be proven [9] that a *necessary* condition for a codebook W to minimize the quantization error is that each codevector w_c fulfills the *centroid condition*. In the case of finite data set D and the Euclidean distance, the centroid condition reduces to $w_c = \frac{1}{|V_c|} \sum_{x_i \in V_c} x_i$ where V_c is the Voronoi set of the codevector w_c. K-Means consists of the following steps:

1. Initialize the codebook W to contain K codevectors w_i ($W = \{w_1, w_2, \ldots, w_K\}$) with vectors chosen *randomly* from the training set D
2. Compute for each codevector $w_i \in W$ its Voronoi Set V_i
3. Move each codevector w_i to the mean of its Voronoi Set,

$$w_i = \frac{1}{|V_i|} \sum_{x_i \in V_i} x_i \tag{1}$$

4. Go to step 2 if any codevector w_i has been changed in step 3 otherwise return the codebook.

The second and the third step form a so-called *Lloyd iteration*. It is guaranteed that after a Lloyd iteration the quantization error does not increase. Besides, K-Means is an *Expectation-Maximization* (*EM*) [3] [6] algorithm. The second and the third step are respectively the Expectation and the Maximization stages. Since each EM algorithm is convergent [24], the convergence of the K-Means algorithm is guaranteed. K-Means is a *batch* algorithm, that is all inputs are evaluated first before any adaptations are done in the codebook, unlike other algorithms called (*on-line*) that make a codebook update after the evaluation of each input. The main K-Means drawback is the lacking in robustness with respect to the outliers. If we consider (1), we see that the presence of the outliers can affect notably the codevector computation.

3 One Class SVM

Firstly we recall the definition of *positive definite kernel* (or *Mercer kernel*) [2].

Definition 1. *Let X be a nonempty set. A function $G : X \times X \to I\!R$ is called a* positive definite kernel *(or* Mercer kernel*) if and only if is* symmetric *(i.e $G(x,y) = G(y,x)\ \forall x, y \in X$) and $\sum_{j,k=1}^{n} c_j c_k G(x_j, x_k) \geq 0$ for all $n \geq 2$, $x_1, \ldots, x_n \subseteq X$ and $c_1, \ldots, c_n \subseteq I\!R$. Each Mercer kernel G can be represented as: $G(x,y) = \langle \Phi(x), \Phi(y) \rangle$ where $\langle \cdot, \cdot \rangle$ is the inner product and $\Phi : X \to \mathcal{F}$, $\mathcal{F}$ is called* Feature Space*.*

An example of Mercer kernel is the *Gaussian* $g(x,y) = e^{-\frac{x-y}{\sigma^2}^2}$ where $\sigma \in \mathbb{R}$. *One Class SVM (1SVM)* [1] [20] [21] is a unsupervised Kernel Method based on support vector description of a data set. In Feature Space, One Class SVM computes the smallest sphere that encloses the image of the input data. Let $D = (x_i \in \mathbb{R}^N,\ \ i = 1, 2, \ldots, m) \subseteq \mathcal{X}$, with $\mathcal{X} \subseteq \mathbb{R}^N$. We project the data x_i into some Feature Space $\mathcal{F}$ using a nonlinear transformation $\Phi : \mathcal{X} \to \mathcal{F}$. Then we look for the smallest sphere of radius R, in the Feature Space, that encloses the data projections $\Phi(x_i)$. This is described by the constraints:

$$\|\Phi(x_j) - a\|^2 \leq R^2 \quad j = 1 \ldots m$$

where $\|\cdot\|$ is the Euclidean norm and a is the center of the sphere. The constraints can be relaxed by using *slack* variables ξ_j:

$$\|\Phi(x_j) - a\|^2 \leq R^2 + \xi_j \tag{2}$$

with $\xi_j \geq 0$. We solve the problem of finding the smallest sphere introducing the *Lagrangian* $\mathcal{L}$:

$$\mathcal{L} = R^2 - \sum_{j}^{m} (R^2 + \xi_j - \|\Phi(x_j) - a\|^2)\beta_j - \sum_{j}^{m} \xi_j \mu_j + C \sum_{j}^{m} \xi_j \tag{3}$$

where $\beta_j \geq 0$ and $\mu_j \geq 0$ are Lagrange multipliers, C is a constant and $C \sum_{j}^{m} \xi_j$ is a penalty term. If we consider the following stationary conditions $\frac{\partial \mathcal{L}}{\partial R} = \frac{\partial \mathcal{L}}{\partial a} = \frac{\partial \mathcal{L}}{\partial \xi_j} = 0$, we get

$$\sum_{j}^{m} \beta_j = 1 \tag{4}$$

$$a = \sum_{j}^{m} \beta_j \Phi(x_j) \tag{5}$$

$$\beta_j = C - \mu_j \tag{6}$$

The Karush-Kuhn-Tucker conditions yield

$$\xi_j \mu_j = 0 \tag{7}$$

$$(R^2 + \xi_j - \|\Phi(x_j) - a\|^2)\beta_j = 0 \tag{8}$$

It follows from (8) that the image of a point x_j with $\xi_j > 0$ and $\beta_j > 0$ lies outside the sphere. The equation (7) states that such a point has $\mu_j = 0$, hence we conclude from the equation (6) that $\beta_j = C$. This will be called a *bounded support vector* (*BSV*). A point x_j with $\xi_j = 0$ is mapped to the inside or to the surface of the Feature Space sphere. If $0 < \beta_j < C$ then (8) implies that $\Phi(x_j)$ lies on the surface of the Feature Space sphere. Such a point will be referred to as a *support vector* (*SV*). Support Vectors lie on cluster boundaries, BSVs lie outside the boundaries and all other points lie inside them. The constraint (4) implies that, when $C \geq 1$, no BSVs exist. Using these relations we may eliminate the variables R, a and μ_j, turning the Lagrangian into the *Wolfe dual form* [22] $\mathcal{W}$ that is a function of the variables β_j:

$$\mathcal{W} = \sum_{j}^{m} \Phi(x_j)^2 \beta_j - \sum_{i}^{m}\sum_{j}^{m} \beta_i \beta_j \Phi(x_i) \cdot \Phi(x_j) \tag{9}$$

Since the variables μ_j do not appear in the Lagrangian they may be replaced with the constraints: $0 \leq \beta_j \leq C \quad j = 1, \ldots, m$. We compute the inner products $\Phi(x_i) \cdot \Phi(x_j)$ by an appropriate Mercer kernel $G(x_i, x_j)$ (*kernel trick*). Since polynomial kernels [21] do not permit getting contour representations of a cluster, the usual choice is to use, as Mercer Kernel, the Gaussian one. We have adopted this choice in the experimentations described in the Section 5. The Lagrangian $\mathcal{W}$, using the kernel trick, becomes

$$\mathcal{W} = \sum_{j}^{m} G(x_j, x_j)\beta_j - \sum_{i}^{m}\sum_{j}^{m} \beta_i \beta_j G(x_i, x_j) \tag{10}$$

At each point x the distance $R(x)$ of its image in Feature Space from the center a of the sphere is:

$$R^2(x) = \|\Phi(x) - a\|^2 = G(x, x) - 2\sum_{j}^{m} \beta_j G(x_j, x) + \sum_{i}^{m}\sum_{j}^{m} \beta_i \beta_j G(x_i, x_j) \tag{11}$$

The radius of the sphere is given by the distance between a support vector and the center of the sphere.

4 The Kernel Method for Clustering

Given a data set D, we map our data in some Feature Space $\mathcal{F}$, by means a nonlinear map Φ. Unlike One Class SVM, we consider K centers in Feature Space ($a_i \in \mathcal{F} \quad i = 1, \ldots, K$). We call the set $A = (a_1, \ldots, a_K)$ *Feature Space Codebook* since in our representation the centers in the Feature Space play the same role of the codevectors in the Input Space. In analogy with the codevectors in the Input Space, we define for each center a_c its *Voronoi Region* and *Voronoi*

Set in Feature Space. The *Voronoi Region in Feature Space* (FR_c) of the center a_c is the set of all vectors in F for which a_c is the closest vector $FR_c = \{\xi \in \mathcal{F} \mid c = arg\min_j \|\xi - a_j\|\}$. The *Voronoi Set in Feature Space* (FV_c) of the center a_c is the set of all vectors x_i in D such that a_c is the *closest vector* for their images $\Phi(x_i)$ in the Feature Space

$$FV_c = \{x_i \in D \mid c = arg\min_j \|\Phi(x_i) - a_j\|\} \tag{12}$$

These definitions, as in the case of the codevectors in the Input Space, induce a *Voronoi Tessellation of the Feature Space*. Now we describe the *Kernel Method for Clustering* (KMC). The algorithm uses a K-Means-like strategy, i.e. moves repeatedly all centers a_c in the Feature Space, computing One Class SVM on their FV_c, until no center changes anymore. To make robust the algorithm with respect to the outliers One Class SVM is computed on $FV_c(\rho)$ of each center a_c. $FV_c(\rho)$ is defined as

$$FV_c(\rho) = \{x_i \in FV_c \quad and \quad \|\Phi(x_i) - a_c\| < \rho\} \tag{13}$$

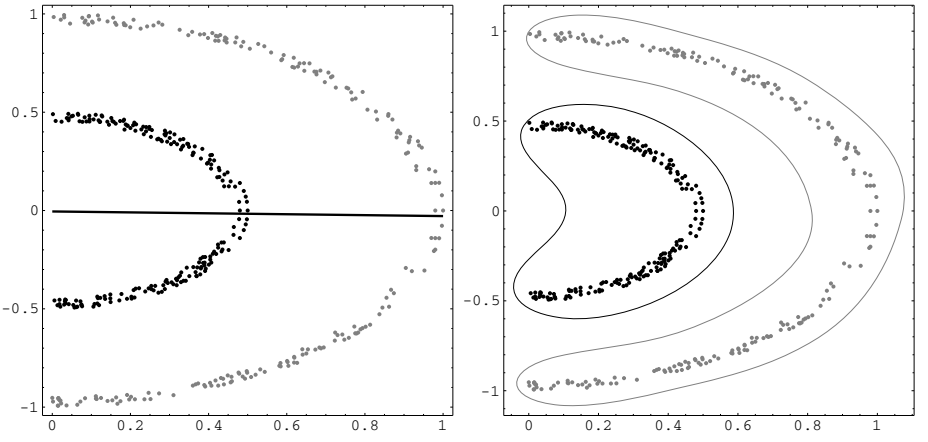
$FV_c(\rho)$ is the Voronoi set in the Feature Space of the center a_c without outliers, that is the images of data points whose distance from the center is larger than ρ. The parameter ρ can be set up using model selection techniques [3]. The algorithm has the following steps:

1. Project the data Set D into a Feature Space $\mathcal{F}$, by means a nonlinear mapping Φ. Initialize the centers $a_c \quad c = 1, \ldots, K \quad a_c \in \mathcal{F}$
2. Compute for each center a_c $FV_c(\rho)$
3. Apply One Class SVM to each $FV_c(\rho)$ and assign to a_c the center yielded, i.e. $a_c = 1SVM(FV_c(\rho))$
4. Go to step 2 until any a_c changes
5. Return the Feature Space codebook.

The second step is the expectation stage of an EM algorithm. With regard to the third step, when the constant C is taken not lower than 1, one class SVM computes the smallest ball that encloses all data. Intuitively under this condition the third step is the maximization stage of an EM algorithm and the algorithm convergence is guaranteed, since each EM algorithm is convergent. The Kernel Method, with $C \geq 1$ and a ρ fixed during the different iterations, has always converged in all our experiments. Finally the Feature Space codebook, in the step one, can be initialized considering a small part (RD) of the data set D, dividing it in K disjoint subsets $RD_i \quad i = 1, \ldots, K$ and assigning $a_i = 1SVM(RD_i)$ $i = 1, \ldots, K$.

5 Experimental Results

KMC has been tried on a synthetic data set (*Delta Set*) and on three UCI data sets, that is the *IRIS* Data [7], the *Wisconsin's breast cancer* database [23] and



(a) K-Means (b) KMC

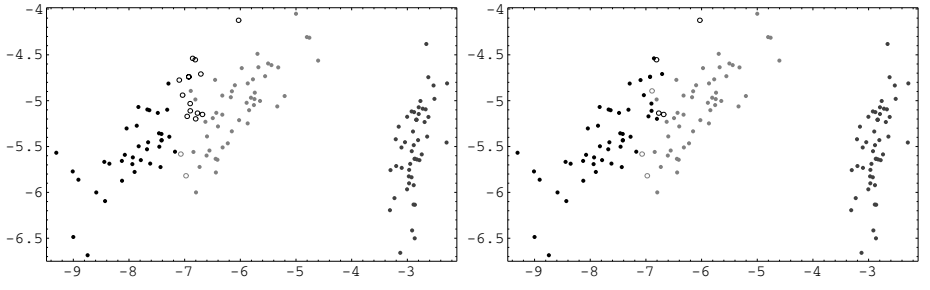
Fig. 1. (a) K-Means on Delta Set. The solid line indicates the separation line determined by K-Means. (b) KMC on Delta Set. The variance σ of the gaussian kernel is 0.4. The region delimited by the black line identifies the input data whose images in the Feature Space have distance from the center a_1 less than 0.75. The region delimited by the gray line identifies the input data whose images in the Feature Space have distance from the center a_2 less than 0.84.

the *Spam* dataset. Delta Set[1] is a bidimensional set formed by 424 points of two classes nonlinearly separated. Therefore the two classes cannot be separated by clustering algorithms that use only two codevectors in the Input Space, since two codevectors permit only linear separation of the data. To confirm that, we tried K-Means, using two codevectors, on Delta Set. As shown in the figure 1, K-Means cannot separate the clusters. K-Means shares this limitation with other not-kernel-based clustering algorithms, e.g. SOM [11] [12] and Neural Gas [15] [16]. Then we tried KMC on Delta Set using only two centers. As shown in figure 1, the algorithm can separate the two clusters, unlike other clustering algorithms. Besides, it is important to remark that the counterimages $\Phi^{-1}(a_1)$ and $\Phi^{-1}(a_2)$, in the Input Space, of the centers a_1 and a_2 do not exist. Iris Data[2] is the most famous real data benchmark in Machine Learning.

Iris Data, proposed by Fisher [7] in 1936, is formed by 150 points, that belong to three different classes. One class is linearly separable from the other two, but the other two are not linearly separable from each other. Since the IRIS Data dimension is 4, Iris Data is usually represented, projecting the data along their two major principal components. We tried K-Means, Neural Gas, SOM and KMC on IRIS data using three centers, one center for each class. Besides,

[1] Delta dataset can be downloaded from the following ftp address: ftp.disi.unige.it/person/CamastraF/delta.dat

[2] Iris Data can be downloaded from the following ftp address: ftp.ics.uci.edu/pub/machine-learning-databases/iris



(a) K-Means (b) KMC

Fig. 2. K-Means (a) and KMC (b) on Iris Data Set. The variance σ of the gaussian kernel is 1.1. Each data class is represented with a different gray level. Filled disks indicate the data correctly classified. Circles represent the data missclassified.

Table 1. SOM, K-Means, Neural Gas, Ng-Jordan algorithm and KMC average performances, in terms of correctly classified points, on Iris, Wisconsin and Spam database. The variances σ of the gaussian Kernel are: 1.1 (Iris Data), 0.9 (Wisconsin Data), 2.0 (Spam Database).

Algorithm	Iris Data	Wisconsin Database	Spam Database
SOM	121.5 ± 1.5 (81.0%)	660.5 ± 0.5 (96.7%)	1210 ± 30 (78.9%)
K-Means	133.5 ± 0.5 (89.0%)	656.5 ± 0.5 (96.1%)	1083 ± 153 (70.6%)
Neural Gas	137.5 ± 1.5 (91.7%)	656.5 ± 0.5 (96.1%)	1050 ± 120 (68.4%)
Ng-Jordan Algorithm	126.5 ± 7.5 (84.3%)	652 ± 2 (95.5%)	929 ± 0 (60.6%)
KMC	142 ± 1 (94.7%)	662.5 ± 0.5 (97.0%)	1247 ± 3 (81.3%)

we also tried a spectral clustering algorithm [17], *Ng-Jordan algorithm* [18]. The results using K-Means and KMC are shown in the figure 2. The table 1 shows the average performances of K-Means, Neural Gas, SOM, Ng-Jordan algorithm and KMC on IRIS on 20 runs, obtained changing algorithm initializations and parameters (e.g. gaussian kernel variance σ and ρ value). As shown in the table, KMC performances are better than other clustering algorithms. Wisconsin's breast cancer database[3] has been proposed by the physician Wolberg [14] [23] in 1990. The breast cancer database collects 699 cases for such diagnostic samples. We have removed 16 database samples with missing values, therefore the database considered in the experiments has 683 patterns. The patterns belong to two different classes, benign and malignant tumor. Each pattern (patient) is represented by a nine-dimensional feature vector. Spam database[4] collects 1534

[3] Wisconsin's breast cancer database can be downloaded from the following ftp address: ftp.ics.uci.edu/pub/machine-learning-databases/breast-cancer-wisconsin

[4] Spam database can be downloaded from the following ftp address: ftp.ics.uci.edu/pub/machine-learning-databases/spam

samples. The pattern belong to two different classes, spam and not-spam. Each pattern is represented by a 57-dimensional feature vector. We tried K-Means, Neural Gas, SOM, Ng-Jordan algorithm and KMC on Wisconsin and Spam databases using two centers, one center for each class. The table 1 shows the average performances of the algorithms on Wisconsin and Spam databases on 20 runs, obtained changing algorithm initializations and parameters. As shown in the table, KMC performances are better than other clustering algorithms.

6 Conclusion

In this paper we have described a Kernel Method for Clustering. The Kernel Method is a batch clustering algorithm, therefore its performance is not affected by the pattern ordering in the training set, unlike on-line clustering algorithms. The main quality of the algorithm consists, unlike most clustering algorithms published in the literature, in producing nonlinear separation surfaces among data. The Kernel Method compares better with popular clustering algorithms, K-Means, Neural Gas and Self Organizing Maps, on a synthetic dataset and three UCI benchmarks, *IRIS data*, *Wisconsin breast cancer database* and *Spam database*. These results encourage the use of the Kernel Method for the solution of computer vision problems, for instance the segmentation of color images.

Acknowledgments

The author dedicates the work and Eduardo R. Caianiello prize 2005 to his mother, Antonia Nicoletta Corbascio, in the most difficult moment of her life.

References

1. A. Ben-Hur, D. Horn, H.T. Siegelmann, and V. Vapnik, "Support Vector Clustering", *Journal of Machine Learning Research*, vol. 2, pp. 125-137, 2001.
2. C. Berg, J.P.R. Christensen, and P. Ressel, *Harmonic analysis on semigroups*, Springer-Verlag, New York, 1984.
3. C. Bishop, *Neural Networks for Pattern Recognition*, Cambridge University Press, Cambridge (UK), 1995.
4. F. Camastra and A. Verri, "A novel Kernel Method for Clustering", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 801-805, 2005.
5. N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*, Cambridge University Press, Cambridge (UK), 2000.
6. A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum Likelihood from Incomplete Data via the EM algorithm", *Journal Royal Statistical Society*, vol. 39, no. 1, pp. 1-38, 1977.
7. R.A. Fisher, "The Use of Multiple Measurements in Taxonomic Problems", *Annals of Eugenics*, vol. 7, pp. 179-188, 1936.
8. M. Girolami, "Mercer kernel based clustering in feature space", *IEEE Transactions on Neural Networks*, vol. 13, no. 3, pp. 780-784, 2002.

9. R.M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Press, Dordrecht, 1992.
10. A.K. Jain, M.N. Murty, and P.J. Flynn, "Data Clustering: A review", *ACM Comput. Surveys*, vol. 31, no. 3, pp. 264-323, 1999.
11. T. Kohonen, "Self-Organized formation of topologically correct feature maps", *Biological Cybernetics*, vol. 43, no. 1, pp. 59-69, 1982.
12. T. Kohonen, *Self-Organizing Map*, Springer-Verlag, New York, 1997.
13. S.P. Lloyd, "An algorithm for vector quantizer design", *IEEE Transaction on Communications*, vol. 28, no. 1, pp. 84-95, 1982.
14. O.L. Mangasarian and W.H. Wolberg, "Cancer diagnosis via linear programming", *SIAM News*, vol. 23, no. 5, pp. 1-18, 1990.
15. T.E. Martinetz and K.J. Schulten, "A neural gas network learns topologies", In *Artificial Neural Networks*, pp. 397-402, 1991.
16. T.E. Martinetz and K.J. Schulten, "Neural-gas network for vector quantization and its application to time-series prediction", *IEEE Transaction on Neural Networks*, vol. 4, no. 4, pp. 558-569, 1993.
17. M. Meila, "Comparing Clusterings", UW Technical Report 418, 2003.
18. A.Y. Ng, M.I. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an algorithm", In *Advances in Neural Information Processing Systems 14*, pp. 849-856, 2001.
19. B. Schölkopf, A.J. Smola, and K.-R Müller, "Nonlinear component analysis as a kernel eigenvalue problem", Technical Report No. 44, Max Planck Institut für biologische Kybernetik, Tübingen, 1996.
20. B. Schölkopf, R.C. Williamson, A.J. Smola, J. Shawe-Taylor, and J. Platt, "Support vector method for novelty detection", In *Advances in Neural Information Processing Systems 12*, pp. 526-532, 1999.
21. D.M.J. Tax and R.P.W. Duin, "Support Vector domain description", *Pattern Recognition Letters*, vol. 20, no. 11-13, pp. 1191-1199, 1999.
22. V. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.
23. W.H. Wolberg and O.L. Mangasarian, "Multisurface method of pattern separation for medical diagnosis applied to breast citology", *Proceedings of the National Academy of Sciences, USA*, vol. 87, pp. 9193-9196, 1990.
24. C.F.J. Wu, "On the convergence properties of the em algorithm", *The Annals of Statistics*, vol. 11, no. 1, pp. 95-103, 1983.
25. K. Yu, L. Ji and X. Zhang, "Kernel Nearest-Neighbor Algorithm", *Neural Processing Letters*, vol. 15,no. 2, pp. 147-156, 2002.

A New Neural Network Model for Contextual Processing of Graphs

Alessio Micheli and Antonio S. Sestito

Dipartimento di Informatica, Università di Pisa,
Largo B. Pontecorvo 3, 56127 Pisa, Italy
micheli@di.unipi.it

Abstract. We propose a novel simple approach to deal with fairly general graph structures by neural networks. Using a constructive approach, the model **Neural Network for Graphs** (**NN4G**) exploits the contextual information stored in the hidden units progressively added to the network, without introducing cycles in the definition of the state variables. In contrast to previous neural networks for structures, NN4G is not recursive but uses standard neurons (with no feedbacks) that traverse each graph without hierarchical assumptions on its topology, allowing the extension of structured domain to cyclic directed/undirected graphs. Initial experimental results, obtained on the prediction of the boiling point of alkanes and on the classification of artificial cyclic structures, show the effectiveness of this new approach.

1 Introduction

The processing of structured domains is becoming increasingly important in Machine Learning area. In fact, varying-size structures, e.g. sequences, trees and graphs, can be a useful tool to represent data for many real-world problems occurring for example in Natural Language processing, document and structured text analysis, Cheminformatics and Bioinformatics.

In the neural computing area, advanced models such as Recurrent Neural Networks for sequence domains and Recursive Neural Networks (RNN) for structured domains have been introduced [4, 9]. RNN are powerful models suitable to deal with recursive (hierarchical) data structures such as rooted trees or directed acyclic graphs (DAG) with supersource. The RNN are based on a causality assumption [4] to realize an encoding matching the input structures, i.e. transductions on the structured domain. In a DAG a causal transduction assumes that the computation for a vertex depends only on the current vertex and vertexes descending from it. Accordingly, the model traverses the directed structures from the leaves to the supersource respecting topological ordering. However, causality can be a unsuitable constraint for tasks requiring contextual information, e.g. considering both possible directions in sequences or the neighborhood of each vertex in a structure. Recently, a recursive contextual model has been introduced to partially relax the causality assumption for directed acyclic positional graphs (DPAG)(see [7] and the references therein for previous approaches).

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 10–17, 2006.

Processing cyclic graphs is another relevant issue. For RNN, extensions to cyclic graphs are studied both by introducing cycles in the state transition system or by rewriting the original graph into hierarchical structures, e.g. [9, 1]. However, more efficient solutions, leading to models able to deal with general graphs, including either directed/undirected, cyclic/acyclic, ordered/unordered graphs deserve further research. Finally, the design of fast and effective training algorithms for RNN, due to its complex dynamical properties, still constitutes an ongoing issue of research (e.g. [5]).

In this paper, we propose a simple approach based on a local contextual processing of each vertex of a graph. In the proposed approach we relax the full power of a recursive approach, in favour of a simpler architecture and learning design, moving toward the possibility to treat a more general class of graphs. This shows that, at least in principle, it is possible to process general graphs without the need of a full-power dynamical system.

2 Neural Network for Graphs (NN4G)

In this paper we assume a fairly general class of input structured patterns as labeled graphs. Given a set of labeled graphs $\mathcal{G}$ and a graph $g \in \mathcal{G}$, we denote the set of the vertexes of g as $Vert(g)$ and the set of its edges as $Edg(g)$, where an edge is a couple of vertexes. Given a vertex $v \in Vert(g)$, we define $edg(v)$ the set of the edges incident on v; the graph can be directed or undirected if the edge set consists of ordered or unordered pairs of vertexes, respectively. In an ordered graph, for each vertex v a total order on the edges incident on v is defined. In a undirected graph, we call the set of adjacent vertexes, or *neighbors*, of the vertex v as

$$\mathcal{N}(v) = \{u \in Vert(g) | (u, v) \in edg(v)\} \ .$$

Each vertex has a label representing a multidimensional numerical attribute $\mathbf{l}(v)$, and $l_i(v)$ is the i-th of L components. A subclass of structures is the class of *free trees*, which are connected acyclic undirected graphs. *Rooted trees* are free trees with a distinctive (supersource) vertex called the *root*.

In the Neural Network for Graphs (NN4G) a state variable $x(v)$ is associated to each vertex v. A constructive network is developed through learning, according, for instance, to a Cascade Correlation approach [3], which progressively adds new hidden units to a set of already trained (frozen) units. Each hidden unit i computes a state variable $x_i(v)$ defined as

$$\begin{cases} x_1(v) = f\left(\sum_{j=0}^{L} \bar{w}_{1j} l_j(v)\right) \\ x_i(v) = f\left(\sum_{j=0}^{L} \bar{w}_{ij} l_j(v) + \sum_{j=1}^{i-1} \hat{w}_{ij} \sum_{t \in \mathcal{N}(v)} x_j(t)\right) \quad i=2,...,N \end{cases} \tag{1}$$

where f is a linear or a sigmoid function of its argument, $\bar{w}_{ij}$ is the weight of the j-th entry of the label to the i-th hidden unit, $\hat{w}_{ij}$ is the weight associated to the

connection between the current unit i and the j-th preceding frozen hidden unit, which brings state values of all the adjacent vertexes to v. The bias is included in the label. Note that, in the second line of Eq. 1, the first sum corresponds to the "present" information, i.e. the label attached to v, and the second sum corresponds to the "contextual" information taken from each preceding hidden unit, i.e. the state values computed for the neighbors of v. This computation is *not recursive* because $x_i(v)$ does not depend on itself but it depends on the output $x_j(v)$, $j < i$ and $i > 1$, of the other hidden units which are already frozen. Moreover, no cyclic dependencies are introduced in the definition of the variables $x_i(v)$ (state transitions). Finally, each computation is local (vertex and its neighbors) and there is no constraint to follow any topological order on the vertexes to compute the state variables.

The system specified in Eq. 1 can produce an output value for each vertex of the given graphs, implementing an IO-isomorphic transduction [4]. Alternatively, a scalar value for a whole graph can be emitted, i.e. a target value is associated to each input graph. In this case, an operator $X_i(g)$ is applied for a unit i, according to the following definitions (where $|\mathcal{S}|$ is the cardinality of the set $\mathcal{S}$):

$$X_i(g) = \frac{\sum_{v \in Vert(g)} x_i(v)}{k} \quad \text{where} \quad k = \begin{cases} 1 \\ |Vert(g)| \\ max_{\mathcal{G}}\left(|Vert(h)|\right) \quad h \in \mathcal{G} \end{cases} \tag{2}$$

i.e. various computation schemes can result defining $X_i(g)$ as a sum ($k = 1$) or an average ($k = |Vert(g)|$) or using a different average, e.g. the case three of k in 2, which defines a sum normalized in the interval $[-1, 1]$ for sigmoidal activation functions.

The output of the NN4G network is given by the output units through a linear or a sigmoid function of the *net*, which is computed distinguishing two cases

$$net_o(p) = \begin{cases} \sum_{j=0}^{N} w_{oj} X_j(g) & \text{if } p = g \in \mathcal{G} \\ \sum_{j=0}^{N} w_{oj} x_j(v) & \text{if } p = v \in Vert(g) \end{cases} \tag{3}$$

Figure 1 summarizes the pseudo-algorithm used to compute the output of the NN4G (with one output unit) when each pattern is a graph $g \in \mathcal{G}$. Note that since no topological order is required for the vertexes, the state $x_i(v)$ can be computed in parallel for each vertex of g.

We can summarize the characteristics of the NN4G model deriving from the definition of the model given in Eq. 1 (basically with respect to RNN) as:

- NN4G is not a recursive model: standard neuron (with no feedback connections) are used as computational units;
- NN4G does not assume causality over directed structures, so it is possible to use it whenever a general dependence of the output on the context is assumed; in particular, no assumption on the topological order is needed;

1. *For* $i = 1$ *to* N
2. $\quad \forall g \in \mathcal{G}$
3. $\quad\quad \forall v \in Vert(g)$
4. $\quad\quad\quad x_i(v) =$ Eq. 1
5. $\quad\quad$ *Compute* $X_i(g)$
6. $\forall g \in \mathcal{G}$
7. $\quad y = f\left(\sum_{j=0}^{N} w_j X_j(g)\right)$

Fig. 1. Algorithm used to compute the output of the NN4G model

- in the current instance, a full stationarity is assumed, i.e. the same model (each unit i) is used for each vertex in the graphs and a single parameter $\hat{w}$ is associated to each incident edge; this weight sharing allows both to reduce the effective complexity of the model and to treat adjacent vertexes with no assumption on their ordering;
- as a result of the previous properties, NN4G can deal with either cyclic/acyclic, ordered/unordered, directed[1]/undirected, labeled graphs.

Figure 2 shows both architectural details of NN4G and an example illustrating the concept of the context for the vertex v (in black) and its evolution in the network construction: the hidden unit for each layer takes a local neighborhood of each vertex as input and progressively, by composition of context developed in the previous steps, it extends the context of influence to other vertexes through state information stored in the frozen hidden units. The only information available to unit 1 is the label of each vertex (not shown in Fig. 2). Unit 2 gets the label information and the information on the context stored in the preceding unit, i.e. the state value of unit 1 for each adjacent vertex to v (central ellipse in layer 1). By this construction the state of the unit 3 for vertex v can be computed on the basis of the context extended to all the vertexes of the input structure (see all the ellipsoids in the first layer). The growth of the context is simmetric in each direction starting from each vertex of g (see Fig. 2).

The neural realization uses an automatic construction of the network. As described in [3], the Cascade Correlation algorithm adds hidden units until a desired error is reached through a learning performed by interleaving the minimization of the total error function by a simple gradient descent training of the output layer, and the maximization of the correlation of the new inserted hidden unit i with the residual error of the output unit o, $E_o(g)$, using a gradient ascent training on

$$S_i = \sum_{o=1}^{|outunits|} \left| \sum_{g \in \mathcal{G}} \left(E_o(g) - \bar{E}_o\right) \left(X_i(g) - \bar{X}_i\right) \right| \tag{4}$$

[1] Note that Eq. 1 can be specialized distinguishing the vertexes of the neighborhood on the basis of entering and leaving edges.

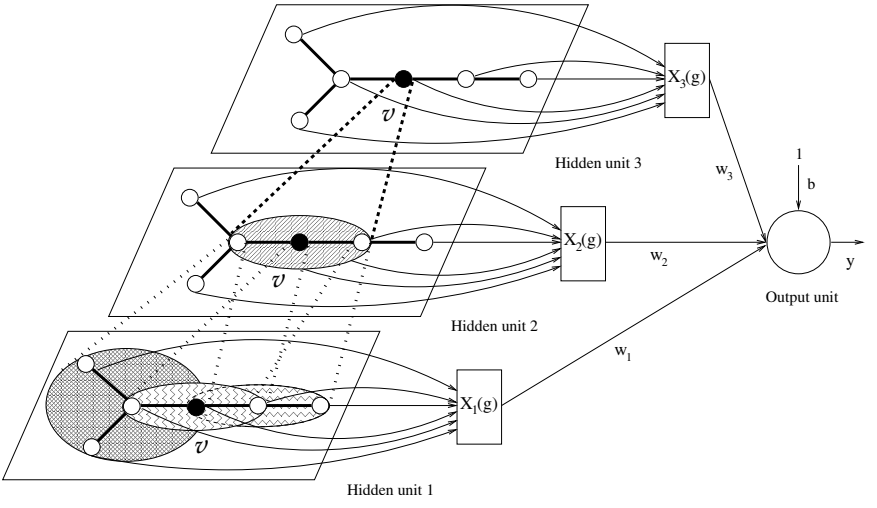


Fig. 2. Evolution of the context for the vertex v; the context grows as the number of hidden frozen units grows; the output of each hidden unit for each vertex is fed through $X(g)$ to the output unit. The label of each vertex are not shown.

where $\bar{E}_o$ is the mean residual error of the output unit o, and $\bar{X}_i$ is the mean value of $X_i(g)$ computed over all graphs. For istance, in Eq. 4 the patterns are graphs $g \in \mathcal{G}$; in case the pattern is each vertex $v \in g$, $X_i(g)$ becomes $x_i(v)$ and we compute the error on each vertex. The weight update for the hidden units is computed by the standard approach of the gradient ascent. The weight variation for the input from the state variables is computed as

$$\Delta\hat{w}_{ik} = \eta\frac{\partial S_i}{\partial\hat{w}_{ik}} = \eta \sum_{o=1}^{|outunits|} \sigma_o \left(\sum_{g\in\mathcal{G}} \left(E_o(g) - \bar{E}_o\right) \sum_{v\in Vert(g)} \frac{f\prime}{k} \sum_{t\in\mathcal{N}(v)} x_j(t) \right) \quad (5)$$

where σ_o is the sign of the correlation and $f\prime$ is the first derivative of function f. The derivation with respect to $\bar{w}$ is straightforward just considering $l_j(v)$ as input signal. A batch algorithm is used for training.

The weight update for the output unit is made by the standard approach of the gradient descent. In our experiments we have also applied the direct computation of the weights by pseudo inverse using Singular Value Decomposition (SVD) instead of normal equation [8].

3 Experimental Results

3.1 QSPR Analysis of Alkanes

We report a first assessment of our method applied to a structured domain involving chemical compounds for Quantitative Structure-Property Relationship (QSPR) analysis. This regression task has been selected in order to empirically

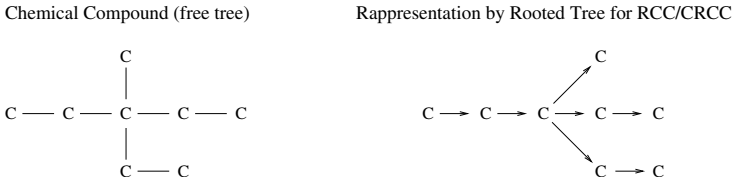


Fig. 3. Example of different hydrogens suppressed representations for an alkane (3-ethyl-3methyl-pentane)

study the new NN4G model on a real-world application where both RNN and "kernel for trees" methods have already compared favorably with respect to state-of-the-art standard approaches used in the QSPR field (see [2, 7, 6] and the references therein). In particular, we compare with respect to CRCC, an extension of recursive models (Recursive Cascade Correlation), to deal with contextual information for DPAG, which improves the results of causal recursive models [7]. The most relevant aspect is that through NN4G is possible to treat alkanes in their natural form of (undirected) free trees instead of imposing a root and a direction on the structure as needed for the RNN models (Fig. 3). Hence, we avoid the use of conventional representation rules aimed at defining rooted ordered trees for the alkanes [2], and we relax the causality assumption, enhancing the advantages deriving by contextual models. The data set used here, which is taken from [2], is based on all the 150 alkanes with up to 10 carbon atoms (C_nH_{2n+2}), represented by hydrogens suppressed graphs (Fig. 3). There is a total of 1331 vertexes in the data set. The target is the boiling point expressed in Celsius degrees (^{0}C) into the range [-164,174].

In the original RCC/CRCC experiments [2, 7], learning was stopped when the maximum absolute error for a single compound was below 8^0C (TR Max AE), or when a maximum number of hidden units (150) was reached. A 10-fold cross-validation on the dataset has been performed. For each fold 5 different learning trials were performed.

In Table 1 the average, along with the best and the worst results, and the variance, over a 10-fold cross validation, of the mean absolute error obtained for training and test set (TR and TS Mean AE), the maximum absolute error on the test set (TS Max AE), and the mean number of hidden units inserted in the model are reported. TR Max AE indicates the error tolerance used to stop the training, while (SVD) indicates the use of pseudo inverse algorithm on the output unit. In particular, it is possible to observe an improvement in average of the test mean error by the new model using the same linear output unit used for CRCC model, with a slight reduction of the number of hidden units. Further improvements in the efficiency are achieved using a direct computation of the output unit parameters by a pseudo inverse algorithm (SVD). The value of k was chosen following the third definition in Eq. 2, however no major differences were observed using $k = 1$. Since no overtraining was observed, and due to the low number of units obtained with SVD training, we explored the possibility to fix a maximum absolute error tolerance to 5^0C , showing the possibility to predict

Table 1. CRCC vs NN4G on 10-fold CV

	CRCC		NN4G					
	Average (Min/Max)	Var.	Average (Min/Max)	Var.	Average (Min/Max)	Var.	Average (Min/Max)	Var.
TR Max AE	8		8		8 (SVD)		5 (SVD)	
TR Mean AE	1.68 (1.51/1.87)	0.056	1.95 (1.80/2.14)	0.045	2.05 (1.89/2.14)	0.036	1.33 (1.27/1.55)	0.008
TS Max AE	8.29	16.28	8.53	23.46	6.86	2.40	5.03	2.54
TS Mean AE	2.56 (1.70/3.97)	0.639	**2.35** (1.83/3.81)	0.234	**2.34** (1.79/3.32)	0.098	**1.74** (1.14/2.70)	0.054
#Units	137.8		114.5		13		23.1	

the boiling point of alkanes with an average mean error under 2^0C. The number of parameters used by the NN4G model are around 6850 for TR Max AE 8, 118 for TR Max AE 8 (SVD) and 325 for TR Max AE 5 (SVD) experiments. These numbers are very small if compared with that obtained for the experiments with CRCC model, i.e. around 38500.

3.2 Classification of Cyclic/Acyclic Structures

Here we consider a classification problem involving undirected cyclic and acyclic graphs. This artificial task allows testing the capability of NN4G to learn a relevant topological feature, i.e the occurrence of cycles in the input graphs, that cannot be directly treated by RNN. The data set used is made up of 150 cyclic graphs and 150 acyclic ones (Fig. 4), with 3 up to 10 vertexes and a total of 2670 vertexes. The prediction of the type of the structure yields to a classification task with target 1 if the graph is acyclic and 0 if not.

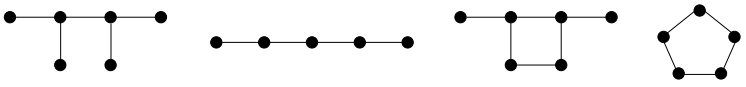


Fig. 4. Two acyclic structures and their corresponding cyclic ones

Learning was stopped when the maximum absolute error for each graph was below 0.4. For this task NN4G achieves 100% classification accuracy both for training and test set over all the folds of 10-fold cross-validation with 5 trials performed for each fold. The networks solve the task adding just 2 hidden units. In fact, the second unit is able to distinguish the ratio between the number of edges and vertexes in the graph, which is a sufficient feature to discriminate the input graphs on the basis of the occurrence of cycles in its topology.

4 Conclusions

We presented a not-recursive model based on the contextual processing of graph domains using a neural constructive approach, which we called NN4G. The basic

idea was to compositionally exploit local contextual information already stored in frozen hidden units. In such a way, the size of the context window can grow and we do not need to fix it prior to learning.

Although the main characteristics of NN4G are the simplicity of the learning algorithm (with respect to recursive approaches), and the generality of the class of structures that can be treated, the results on QSPR of alkanes showed also an improvement of the state-of-the-art performance for the task.

The cyclic/acyclic experiments allowed us to test the capability of the new model to recognize the occurrence of cycles, which is a relevant topological feature for graphs. The NN4G efficiently achieved an optimal performance, showing the potentiality of the model to treat expressive representation of the data.

Of course, various issues concerning the general capability of the new model certainly need to be investigated. However, we believe that already in the present form the NN4G model, especially because of its efficiency and simplicity, is a step toward easier and more widespread application of neural network models to structure domain learning.

Acknowledgements

The authors would like to thank Diego Sona for useful discussions.

References

1. M. Bianchini, M. Gori, and F. Scarselli. Recursive processing of cyclic graphs. In *Proc. of WCCI-IJCNN'2002*, volume 1, pages 154–159, 2002.
2. A.M. Bianucci, A. Micheli, A. Sperduti, and A. Starita. Application of cascade correlation networks for structures to chemistry. *Appl. Intell.*, 12:117–146, 2000.
3. S. E. Fahlman and C. Lebiere. The cascade-correlation learning architecture. Technical Report CMU-CS-90-100, Carnegie Mellon, August 1990.
4. P. Frasconi, M. Gori, A. Küchler, and A. Sperduti. From sequences to data structures: Theory and applications. In Kolen J.F. and Kremer S.C., editors, *A Field Guide to Dynamical Recurrent Networks*, chapter 19. IEEE Press, Inc., 2001.
5. B. Hammer and J. J. Steil. Tutorial: Perspectives on learning with rnns. In *Proc. of ESANN 2002*, pages 357–368. D-side, 2002.
6. A. Micheli, F. Portera, and A. Sperduti. A preliminary empirical comparison of recursive neural networks and tree kernel methods on regression tasks for tree structured domains. *Neurocomputing*, 64:73–92, March 2005.
7. A. Micheli, D. Sona, and A. Sperduti. Contextual processing of structured data by recursive cascade correlation. *IEEE Trans. Neural Networks*, 15(6):1396–1410, November 2004.
8. W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C: the Art of Scientific Computing, 2nd Ed.* Cambridge University Press, 1992.
9. A. Sperduti and A. Starita. Supervised neural networks for the classification of structures. *IEEE Trans. Neural Networks*, 8(3):714–735, May 1997.

Approximation Properties of Positive Boolean Functions

Marco Muselli

Istituto di Elettronica e di Ingegneria dell'Informazione e delle Telecomunicazioni,
Consiglio Nazionale delle Ricerche,
via De Marini, 6 - 16149 Genova, Italy
marco.muselli@ieiit.cnr.it

Abstract. The universal approximation property is an important characteristic of models employed in the solution of machine learning problems. The possibility of approximating within a desired precision any Borel measurable function guarantees the generality of the considered approach.

The properties of the class of positive Boolean functions, realizable by digital circuits containing only AND and OR ports, is examined by considering a proper coding for ordered and nominal variables, which is able to preserve ordering and distance. In particular, it is shown that positive Boolean functions are universal approximators and can therefore be used in the solution of classification and regression problems.

1 Introduction

An important topic in computational learning theory concerns the characterization of the class Γ of functions (models) to be adopted when searching for the solution of a classification or of a regression problem. It must be sufficiently rich to allow the treatment of a wide variety of real-world problems, it must ensure the natural handling of different kinds of input variables, it must permit the application of an efficient learning technique.

For example, if Γ is the class of multilayer perceptrons with one hidden layer and sigmoidal activation functions, well established theoretical results guarantee that Γ possesses the *universal approximation property* [1], i.e. under mild conditions any Borel measurable real function can be approximated arbitrarily well by a multilayer perceptron if a sufficient number of neurons is included in the hidden layer. The same property characterizes other widely used connectionist models, such as radial basis function networks [2] or support vector machines [3].

Also the adoption of methods based on the synthesis of Boolean functions is motivated from the well known fact that they are universal approximators, a basic property for the functioning of computing systems. If the use of the complement operator NOT is not allowed, we can only realize the subset of *positive Boolean functions*, which does not include several simple binary mappings, such as the parity function. This limitation has prevented their use in the solution of

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 18–22, 2006.

classification and regression problems, together with the lack of an efficient learning technique for the selection of a positive Boolean function that generalizes the information contained in the training set.

This paper has the aim of refuting this opinion, proving that the class $\mathcal{L}$ of positive Boolean functions possesses the universal approximation property. The adoption of $\mathcal{L}$ in the solution of classification problems gives rise to a new connectionist model, called *Switching Neural Network* (*SNN*) [6], which can be completely described through a set of intelligible rules in the **if-then** form. SNNs are trained by a specific procedure for positive Boolean function reconstruction, called *Shadow Clustering* [7].

2 Definitions and Notations

Consider the Boolean lattice $(\{0,1\}^n, \vee, \wedge, 0, 1)$, where $\vee$ and $\wedge$ are the logical sum (AND) and the logical product (OR), respectively. The addition of the complement operation NOT makes $\{0,1\}^n$ a Boolean algebra, widely used in digital circuit design and in many other scientific fields.

It is known that any Boolean function $f : \{0,1\}^n \to \{0,1\}$ can be realized through an expression in this algebra. On the contrary, the lack of the complement operation in the Boolean lattice $\{0,1\}^n$ allows to generate only the subset of *positive Boolean functions* f, for which $f(x) \leq f(y)$ if $x \leq y$, being $\leq$ the standard ordering defined on lattices.

To prove that positive Boolean functions are universal approximators, i.e. they can approximate arbitrarily well any measurable function $g : \mathbb{R}^d \to \mathbb{R}$, let us define in a formal way the concept of approximating the elements of a set X through the elements of another set Y. To this aim suppose a metric ρ exists on X.

Definition 1. *A subset Z of a metric space (X, ρ) is* ρ-dense *in X if and only if for every $\varepsilon > 0$ and for every $x \in X$ there is a $z \in Z$ such that $\rho(z, x) < \varepsilon$.*
A set Y approximates arbitrarily well *a metric space (X, ρ) if and only if a mapping $\eta : Y \to X$ can be found such that its range $\eta(Y)$ is ρ-dense in X.*

Definition 1 requires the choice of a proper mapping η to establish that a set Y approximates arbitrarily well a metric space (X, ρ). When X is a class of functions $g : A_X \to B_X$ and Y contains mappings $f : A_Y \to B_Y$ as elements, a possible way of constructing η consists in writing it as the composition $\eta(f) = \psi \circ f \circ \varphi$ of f with two mappings $\varphi : A_X \to A_Y$ and $\psi : B_Y \to B_X$. With this choice $\eta(f)$ gives a function from A_X to B_X for any $f \in Y$.

Note that if there is a mapping η from Y onto X, the set Y approximates arbitrarily well the space X, whatever is the metric ρ on it.

A classical result in approximation theory ensures that the set of discrete functions $h : I_m^d \to I_m$ approximates arbitrarily well the class of Borel measurable functions defined on $\mathbb{R}^d$. This allows to establish the universal approximation property of Boolean functions $f : \{0,1\}^b \to \{0,1\}^k$, by simply analyzing the possibility of finding proper mappings $\varphi : I_m^d \to \{0,1\}^b$ and $\psi : \{0,1\}^k \to I_m$

which makes $\eta(f) = \psi \circ f \circ \varphi$ a mapping from the class of Boolean functions onto the set of discrete functions.

The transitive property in turn allows to establish that

Theorem 1. *The class of Boolean functions $f : \{0,1\}^b \to \{0,1\}^k$ is able to approximate arbitrarily well the set of Borel measurable functions.*

It is important to observe that on any finite chain, such as I_m, a natural metric d_c can be introduced, called *counter metric*, defined as $d_c(a,b) = |a-b|$, being $|a-b|$ the length of the subchain connecting a and b. Another simple metric, which can be defined on any set, is the *flat metric* d_f given by $d_f(a,b) = 0$ if $a = b$ and 1 otherwise.

Note that the counter metric represents a usual definition for measuring distances between the values of a (discrete) ordered variable, whereas the flat metric is a natural definition for nominal variables. These metrics can easily be generalized to I_m^d by considering the extension $d_I(x,z) = \sum_{i=1}^{d} d_i(x_i, z_i)$, where $d_i(x_i, z_i) = d_c(x_i, z_i)$ or $d_i(x_i, z_i) = d_f(x_i, z_i)$, depending on whether the counter or the flat metric is adopted on I_m.

3 Approximation Property of Positive Boolean Functions

The following theorem offers a possible way to construct the mapping η to be adopted for establishing the approximation capability of the set $\mathcal{L}_n$ of positive Boolean functions.

Theorem 2. *Let A be an antichain of the poset $(\{0,1\}^n, \leq)$, i.e. for any $x, y \in A$ with $x \neq y$ neither $x < y$ nor $y > x$ holds. Then for every $f : \{0,1\}^n \to \{0,1\}$ there is a positive Boolean function $g : \{0,1\}^n \to \{0,1\}$ such that $f(x) = g(x)$ for all $x \in A$.*

Proof. Take any $f : A \to \{0,1\}$ and consider the Boolean function g defined in the following way:

$$g(x) = \begin{cases} f(x) & \text{if } x \in A \\ 1 & \text{if } x > a \text{ for some } a \in A \\ 0 & \text{otherwise} \end{cases}$$

Then, g is a positive Boolean function, since $g(x) \leq g(y)$ whenever $x \leq y$. □

Consider the mapping $P : \{0,1\}^n \to 2^{I_n}$ defined as $P(a) = \{i \in I_n : a_i = 1\}$ for any $a \in \{0,1\}^n$; it produces the subset of indices of the components a_i assuming value 1. It can be easily seen that the mapping P is an isomorphism between the posets $(\{0,1\}^n, \leq)$ and $(2^{I_n}, \subseteq)$. The inverse of P will be denoted with p; for any subset $J \subset I_n$ it gives the element $p(J) \in \{0,1\}^n$ whose i-th component $p_i(J)$ has value 1 if and only if $i \in J$.

Now, consider the subsets $Q_n^l = \{a \in \{0,1\}^n : |P(a)| = l\}$ containing the strings of n bits with l 1s. It can be easily verified that every Q_n^l is an antichain

with $C_{n,l} = \binom{n}{l}$ elements. Thus, for Th. 2 the approximation capability of the class $\mathcal{L}_n$ can be established by considering proper mappings ζ having as codomain Q_n^l for the construction of the mapping φ to be employed in η.

Denote with $\mathcal{Q}_n^l$ the collection of the subsets of I_n with cardinality l:

$$\mathcal{Q}_n^l = \{\{j_1, j_2, \ldots, j_l\} \,:\, 1 \leq j_1 < j_2 < \cdots < j_l \leq n,\ j_i \in I_n, \text{ for } i = 1, \ldots, l\}$$

When the lexicographic ordering is used to compare the relative position of its elements (ordered in an increasing way), $\mathcal{Q}_n^l$ becomes a chain. Consequently, the counter metric d_c can be defined on $\mathcal{Q}_n^l$, being $d_c(B, C)$ the length of the subchain connecting B and C for every $B, C \in \mathcal{Q}_n^l$.

According to this definition, if $B = \{1, 2, \ldots, h-1, j_h - 1, j_{h+1}, \ldots, j_l\}$ and $C = \{1, 2, \ldots, h-1, j_h, j_{h+1}, \ldots, j_l\}$, with $h < j_h < j_{h+1}$ we have

$$d_c(B, C) = \binom{j_h - 2}{h - 1} \tag{1}$$

since the length of the subchain connecting B and C is given by the number of different ways of choosing the $h-1$ indices $j_1, j_2, \ldots, j_{h-1}$ in the subset $\{1, 2, \ldots, j_h - 2\}$ having cardinality $j_h - 2$. By iterating the application of (1), if $A = \{1, 2, \ldots, h-1, h, j_{h+1}, \ldots, j_l\}$ we obtain that

$$d_c(A, C) = \sum_{i=h}^{j_h - 1} \binom{i-1}{h-1} = \sum_{i=0}^{j_h - h - 1} \binom{i+h-1}{h-1} = \binom{j_h - 1}{h}$$

having used the identity

$$\sum_{\nu=0}^{r} \binom{\nu + k - 1}{k - 1} = \binom{r+k}{k}$$

which holds for all the positive integers r, k. Note that $d_c(A, C)$ is the length of the subchain covered when moving from h to j_h the h-th element.

The least element of $\mathcal{Q}_n^l$ is $\underline{A} = \{1, 2, \ldots, l\}$ and the distance from the set $\underline{A}$ to a generic set $B = \{j_1, j_2, \ldots, j_l\}$, with $j_1 < j_2 < \cdots < j_l$, can be obtained by summing up the length of the subchains covered when moving the h-th element from h to j_h, for $h = 1, \ldots, l$. This amounts to

$$d_c(\underline{A}, B) = \sum_{h=1}^{l} \binom{j_h - 1}{h}$$

Now, if $m = C_{n,l}$ define the mapping $\omega : \mathcal{Q}_n^l \to I_m$ as $\omega(B) = 1 + d_c(\underline{A}, B)$; the following result is readily proved.

Theorem 3. *If $m = C_{n,l}$ the mapping ω is a bijection and an isometry. Consequently, ω is an isomorphism and $\mathcal{Q}_n^l$ is isomorphic to I_m.*

Since the mapping ω is 1-1, its inverse $\omega^{-1} : I_m \rightarrow \mathcal{Q}_n^l$ is uniquely determined when $m \leq C_{n,l}$. According to Th. 3, ω^{-1} is 1-1 and an isometry; if in addition $m = C_{n,l}$, then ω^{-1} turns out to be a bijection.

Consider the *mirror mapping* $\mu : \{0,1\}^n \rightarrow \{0,1\}^n$, which reverses the bits of a binary string x. The composition of μ, p and ω^{-1} generates a function $\gamma : I_m \rightarrow Q_n^l$ that maps the first m integers into the binary strings of the antichain Q_n^l. Since μ, p and ω^{-1} are all 1-1, also is the composition $\gamma = \mu \circ p \circ \omega^{-1}$.

It can be shown that the mapping γ, called henceforth *lattice coding*, is an isometry between I_m and Q_n^l if the counter metric is employed both on I_m and on Q_n^l. In addition, γ is order preserving if the lexicographic ordering is employed on Q_n^l. However, it can be easily seen that γ is still an isometry when the flat metric is adopted both on I_m and on Q_n^l. Finally, γ is a bijection if $m = C_{n,l}$. Note that the *only one coding*, widely used to code nominal variables, is equivalent to a lattice coding with $n = m$ and $l = 1$. Thus, it is always a bijection, an isometry and an order preserving mapping.

Now, the approximation property of positive Boolean functions can be derived by properly defining the mapping $\varphi : I_m^d \rightarrow Q_n^l$. If the elements a of Q_n^l are subdivided into d substrings $a^{(i)}$, the mapping φ can be obtained by concatenating the d binary strings produced by as many functions $\gamma : I_m \rightarrow Q_{n_i}^{l_i}$.

By Th. 2 every function $f : Q_n^l \rightarrow \{0,1\}$ can be extended to a positive Boolean function; thus, we can conclude that the mapping η defined on $\mathcal{L}_n$ as $\eta = \psi \circ f \circ \varphi$, where ψ is the inverse of γ restricted to the subset $\gamma(I_m)$, is onto the set of discrete functions. We have then established the following general result:

Theorem 4. *The class of positive Boolean functions is able to approximate arbitrarily well the set of Borel measurable functions.*

References

1. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Networks **2** (1989) 359–366
2. Park, J., Sandberg, I.W.: Universal approximation using radial-basis-function networks. Neural Computation **3** (1991) 246–257
3. Hammer, B., Gersmann, K.: A note on the universal approximation capability of support vector machines. Neural Processing Letters **17** (2003) 43–53
4. Boros, E., Hammer, P.L., Ibaraki, T., Kogan, A., Mayoraz, E., Muchnik, I.: An implementation of Logical Analysis of Data. IEEE Transactions on Knowledge and Data Engineering **12** (2000) 292–306
5. Muselli, M., Liberati, D.: Binary rule generation via Hamming Clustering. IEEE Transactions on Knowledge and Data Engineering **14** (2002) 1258–1268
6. M. Muselli: Switching Neural Networks: A New Connectionist Model for Classification. In WIRN/NAIS 2005, vol. 3931 of Lecture Notes in Computer Science (2006) Eds. B. Apolloni et al., Berlin: Springer-Verlag, 23-30.
7. Muselli, M., A. Quarati: Reconstructing positive Boolean functions with Shadow Clustering. In Proceedings of the 17th European Conference on Circuit Theory and Design (ECCTD 2005), (Cork, Ireland, August 2005).

Switching Neural Networks: A New Connectionist Model for Classification

Marco Muselli

Istituto di Elettronica e di Ingegneria dell'Informazione e delle Telecomunicazioni,
Consiglio Nazionale delle Ricerche,
via De Marini, 6 - 16149 Genova, Italy
marco.muselli@ieiit.cnr.it

Abstract. A new connectionist model, called *Switching Neural Network* (*SNN*), for the solution of classification problems is presented. SNN includes a first layer containing a particular kind of A/D converters, called *latticizers*, that suitably transform input vectors into binary strings. Then, the subsequent two layers of an SNN realize a positive Boolean function that solves in a lattice domain the original classification problem.

Every function realized by an SNN can be written in terms of intelligible rules. Training can be performed by adopting a proper method for positive Boolean function reconstruction, called *Shadow Clustering* (*SC*). Simulation results obtained on the StatLog benchmark show the good quality of the SNNs trained with SC.

1 Introduction

Any classification problem can be viewed as an optimization problem, where a proper functional, i.e. the probability of error, has to be minimized by choosing the best classifier g inside a given collection Γ. In this sense any technique for the solution of classification problems must provide for two different actions: the class Γ of decision functions must be suitably determined (*model selection*) and the best classifier in Γ must be searched for (*training phase*).

General theoretical results advise us against choosing a too large set Γ; in fact, with this choice it is likely to incur in the problem of *overfitting*: the training phase generates a decision function $g \in \Gamma$ that performs well on the examples of the training set, but scores a high number of misclassifications on the remaining points of the input domain. On the other hand, if a too small set Γ is considered, it is impossible to obtain a sufficiently low number of errors in the training set.

A possible approach consists in choosing initially a large Γ, leaving to the learning algorithm the task of retrieving a classifier g which is enough simple, according to some index of complexity, and behaves well on the training set. This is the approach followed by support vector machines [1], where a regularization constant controls the trade-off between the complexity of the resulting decision function and the number of errors scored by it on the available training set.

In a similar way, we will introduce in the following sections a new connectionist model, called *Switching Neural Network (SNN)*, which is sufficiently rich to approximate within an arbitrary precision any measurable function. As described

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 23–30, 2006.

later, this connectionist model presents some interesting properties, among which the possibility of allowing a precise description of classifiers $g \in \Gamma$ in terms of intelligible rules.

A proper learning algorithm, called *Shadow Clustering* (*SC*), can be adopted to search for the simplest SNN that ensures a sufficiently low number of errors on the training set. Preliminary results show that SNNs trained by SC can achieve generalization errors comparable with those of best machine learning techniques. Due to space limitation, details of SC are not presented here, but can be found elsewhere [2].

2 Model Selection

Consider a general binary classification problem, where d-dimensional patterns $\boldsymbol{x} \in X \subset \mathbb{R}^d$ are to be assigned to one of two possible classes, labeled by the values of a Boolean output $y \in \{0, 1\}$. According to possible situations in real world problems, the type of the components x_i, $i = 1, \ldots, d$, can be either *continuous ordered*, when x_i belongs to a subset of $\mathbb{R}$, or *discrete ordered*, when x_i assumes values inside a finite ordered set, or *nominal*, when x_i can assume values inside a finite set, where no ordering is defined.

Denote with I_m the set $\{1, 2, \ldots, m\}$ of the first m positive integers. Without loss of generality, I_m (for a proper value of m) can be regarded as the domain of any discrete ordered or nominal variable. In the first case, I_m is viewed as a chain, whereas in the latter no ordering is assumed to be defined on it.

Now, consider the Boolean lattice $\{0, 1\}^n$, equipped with the well known binary operations '+' (*logical sum* or OR) and '$\cdot$' (*logical product* or AND). According to the standard partial ordering on $\{0, 1\}^n$, a Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ is called *positive* if $\boldsymbol{u} \leq \boldsymbol{v}$ implies $f(\boldsymbol{u}) \leq f(\boldsymbol{v})$ for every $\boldsymbol{u}, \boldsymbol{v} \in \{0, 1\}^n$.

A recent theoretical result [3] asserts that positive Boolean functions are universal approximators, i.e. they can approximate arbitrarily well every measurable function $g : \mathbb{R}^d \rightarrow \{0, 1\}$. Denote with Q_n^l the subset of $\{0, 1\}^n$ containing the strings of n bits having exactly l values 1 inside them. A possible procedure for finding the positive Boolean function f that approximates a given classifier g is based on the following three steps:

1. (*Discretization*). For every ordered input x_i, determine a finite partition $\mathcal{B}_i$ of its domain X_i such that a function $\hat{g}$ can be found, which approximates g on X within the desired precision and assumes a constant value on every set $B \in \mathcal{B}$, where $\mathcal{B} = \{\prod_{i=1}^{d} B_i \, : \, B_i \in \mathcal{B}_i, \, i = 1, \ldots, d\}$.
2. (*Latticization*). By employing a proper function $\boldsymbol{\varphi}$, map the points of the domain X into the strings of Q_n^l, so that $\boldsymbol{\varphi}(\boldsymbol{x}) \neq \boldsymbol{\varphi}(\boldsymbol{x}')$ only if $\boldsymbol{x} \in B$ and $\boldsymbol{x}' \in B'$, being B and B' two different sets in $\mathcal{B}$.
3. (*Positive Boolean function synthesis*). Select a positive Boolean function f such that $f(\boldsymbol{\varphi}(\boldsymbol{x})) = \hat{g}(\boldsymbol{x})$ for every $\boldsymbol{x} \in X$.

2.1 Discretization

Since the exact behavior of the decision function g is not known, the approximating classifier $\hat{g}$ and the partition $\mathcal{B}$ have to be inferred from the examples

$(\boldsymbol{x}_j, y_j)$ of the training set S. Every set $B_i \in \mathcal{B}_i$ must be enough large to include the component x_{ji} of some point $\boldsymbol{x}_j$ in S. Nevertheless, the resulting partition $\mathcal{B}$ must be enough fine to capture the actual complexity of the function g.

Several different discretization methods for binary classification problems have been proposed in the literature [4, 5, 6]. Usually, for each ordered input x_i a set of $m_i - 1$ consecutive values $r_{i1} < r_{i2} < \cdots < r_{i,m_i-1}$ is generated and the partition $\mathcal{B}_i$ is formed by the m_i sets $X_i \cap R_{ik}$, where $R_{i1} = (-\infty, r_{i1}]$, $R_{i2} = (r_{i1}, r_{i2}]$, ..., $R_{i,m_i-1} = (r_{i,m_i-2}, r_{i,m_i-1}]$, $R_{im_i} = (r_{i,m_i-1}, +\infty)$. Excellent results have been obtained with the algorithm Chi2 [5], which employ the χ^2 statistic to decide the position of the points r_{ik}, $k = 1, \ldots, m_i - 1$, and with the technique EntMDL [4], which adopts entropy estimates to achieve the same goal. An alternative promising approach is offered by the method used in the LAD system [6]: in this case an integer programming problem is solved to obtain optimal values for the cutoffs r_{ik}.

By applying a procedure of this kind, the discretization task defines for each ordered input x_i a mapping $\psi_i : X_i \to I_{m_i}$, where $\psi_i(z) = k$ if and only if $z \in R_{ik}$. If we assume that ψ_i is the identity function with $m_i = |X_i|$ when x_i is a nominal variable, the approximating function $\hat{g}$ is uniquely determined by a discrete function $h : I \to \{0, 1\}$, defined as $h(\boldsymbol{\psi}(\boldsymbol{x})) = \hat{g}(\boldsymbol{x})$, where $I = \prod_{i=1}^{d} I_{m_i}$ and $\boldsymbol{\psi}(\boldsymbol{x})$ is the mapping from X to I, whose ith component is given by $\psi_i(x_i)$.

2.2 Latticization

The function $\boldsymbol{\psi}$ provides a mapping from the domain X onto the set $I = \prod_{i=1}^{d} I_{m_i}$, such that $\boldsymbol{\psi}(\boldsymbol{x}) = \boldsymbol{\psi}(\boldsymbol{x}')$ if $\boldsymbol{x}$ and $\boldsymbol{x}'$ belong to the same set $B \in \mathcal{B}$, whereas $\boldsymbol{\psi}(\boldsymbol{x}) \neq \boldsymbol{\psi}(\boldsymbol{x}')$ only if $\boldsymbol{x} \in B$ and $\boldsymbol{x}' \in B'$, being B and B' two different sets in $\mathcal{B}$. Consequently, the 1-1 function $\boldsymbol{\varphi}$ from X to Q_n^l, required in the latticization step, can be simply determined by defining a proper 1-1 function $\boldsymbol{\beta} : I \to Q_n^l$. In this way, $\boldsymbol{\varphi}(\boldsymbol{x}) = \boldsymbol{\beta}(\boldsymbol{\psi}(\boldsymbol{x}))$ for every $\boldsymbol{x} \in X$.

A possible way of constructing the function $\boldsymbol{\beta}$ is to define in a proper way d mappings $\boldsymbol{\beta}_i : I_{m_i} \to Q_{n_i}^{l_i}$; then, the binary string $\boldsymbol{\beta}(\boldsymbol{u})$ for an integer vector $\boldsymbol{u} \in I$ is obtained by concatenating the strings $\boldsymbol{\beta}_i(u_i)$ for $i = 1, \ldots, d$. It can be shown [3] that a good choice for $\boldsymbol{\beta}_i$ is the *inverse only one coding*, which maps an integer $u_i \in I_{m_i}$ into the binary string $\boldsymbol{z}_i \in Q_{m_i}^{m_i-1}$ having $z_{ik} = 0$ if and only if $u_i = k$. In fact, this coding is both an isometry and a full order-preserving mapping (when x_i is an ordered input). These properties characterize also the mapping $\boldsymbol{\beta}$ if the inverse only one coding is adopted for all the $\boldsymbol{\beta}_i$.

Since $\boldsymbol{\varphi}(\boldsymbol{x})$ is obtained by the concatenation of d binary strings $\boldsymbol{\varphi}_i(x_i)$, if the discretization task has produced for each ordered input x_i a set of $m_i - 1$ cutoffs r_{ik}, as described in the previous subsection, the kth bit of $\boldsymbol{\varphi}_i(x_i)$ assumes value 0 if and only if $x_i \in R_{ik}$. Note that $x_i \in R_{ik}$ if and only if x_i exceeds the cutoff $r_{i,k-1}$ (if $k > 1$) and is lower than the subsequent cutoff r_{ik} (if $k < m_i$). On the other hand, if x_i is a nominal input the kth bit of $\boldsymbol{\varphi}_i(x_i)$ assumes value 0 if and only if $x_i = k$.

Thus, the mapping $\boldsymbol{\varphi}_i$ can be implemented by a simple device that receives in input the value x_i and compares it with a sequence of integer or real numbers.

This device will be called *latticizer*; it produces m_i binary outputs, but only one of them can assume the value 0. The whole mapping $\boldsymbol{\varphi}$ is realized by a parallel of d latticizer, each of which is associated with a different input x_i.

2.3 Switching Neural Networks

After the discretization and the latticization steps we have transformed the training set $S = \{(\boldsymbol{x}_j, y_j), j = 1, \ldots, s\}$ into a collection of examples $S' = \{(\boldsymbol{z}_j, y_j), j = 1, \ldots, s\}$, where each $\boldsymbol{z}_j = \boldsymbol{\varphi}(\boldsymbol{x}_j)$ is a binary string in the Boolean lattice $\{0,1\}^m$ for a proper value of m. The original binary classification problem can then be solved by retrieving a positive Boolean function $f(\boldsymbol{z})$ such that $\hat{g}(\boldsymbol{x}) = f(\boldsymbol{\varphi}(\boldsymbol{x}))$ minimizes the probability of misclassifying a pattern $\boldsymbol{x} \in X$.

This target can be pursued by adopting a proper technique, named *Shadow Clustering* (*SC*) [2], which is able to construct a positive Boolean function f that generalizes well starting from the examples contained in S'. To our best knowledge, SC is the first method of this kind. It adopts an overall strategy similar to Hamming Clustering [7], successfully employed in the solution of binary classification problems. A detailed description of the approach followed by SC for the synthesis of positive Boolean functions is presented in [2].

If $\boldsymbol{a} \in \{0,1\}^m$ is a binary string with length m, let $P(\boldsymbol{a})$ be the subset of I_m including the indexes i for which $a_i = 1$. A positive Boolean function can always be written in the following *Positive Disjunctive Normal Form* (*PDNF*):

$$f(\boldsymbol{z}) = \bigvee_{\boldsymbol{a} \in A} \bigwedge_{j \in P(\boldsymbol{a})} z_j \tag{1}$$

where A is an *antichain* of the Boolean lattice $\{0,1\}^m$, i.e. a collection of binary strings such that if $\boldsymbol{a}, \boldsymbol{a}' \in A$ neither $\boldsymbol{a} < \boldsymbol{a}'$ nor $\boldsymbol{a}' < \boldsymbol{a}$. The symbol $\bigvee$ (resp. $\bigwedge$) in (1) denotes a logical sum (resp. product) among the terms identified by the subscript. In particular, $\bigwedge_{j \in P(\boldsymbol{a})} z_j$ is an *implicant* for the function f; however, when no confusion arises, the term implicant will also be used to denote the corresponding binary string $\boldsymbol{a} \in A$.

The execution of SC produces the antichain A to be employed in (1) for obtaining the PDNF of the desired positive Boolean function f. This expression can be readily implemented on a two-level digital circuits including only AND, OR ports. It should be observed that only the values 1 in a binary string $\boldsymbol{a} \in A$ give rise to an incoming connection in the corresponding AND port. Thus, values 0 in $\boldsymbol{a}$ behave as don't care symbols for the associated implicant.

The approximating function $\hat{g}(\boldsymbol{x})$ that solves our binary classification problem is then given by the composition $f(\boldsymbol{\varphi}(\boldsymbol{x}))$ of the positive Boolean function f with the mapping $\boldsymbol{\varphi}$ produced by discretization and latticization steps. The device implementing $\hat{g}(\boldsymbol{x})$ is shown in Fig. 1.

It can be considered as a three layer feedforward neural network, where the first layer is responsible of the realization of the binary mapping $\boldsymbol{\varphi}(\boldsymbol{x})$, while the other two realize the expression (1) for the positive Boolean function f. Every AND port in the second layer is connected only to some of the outputs

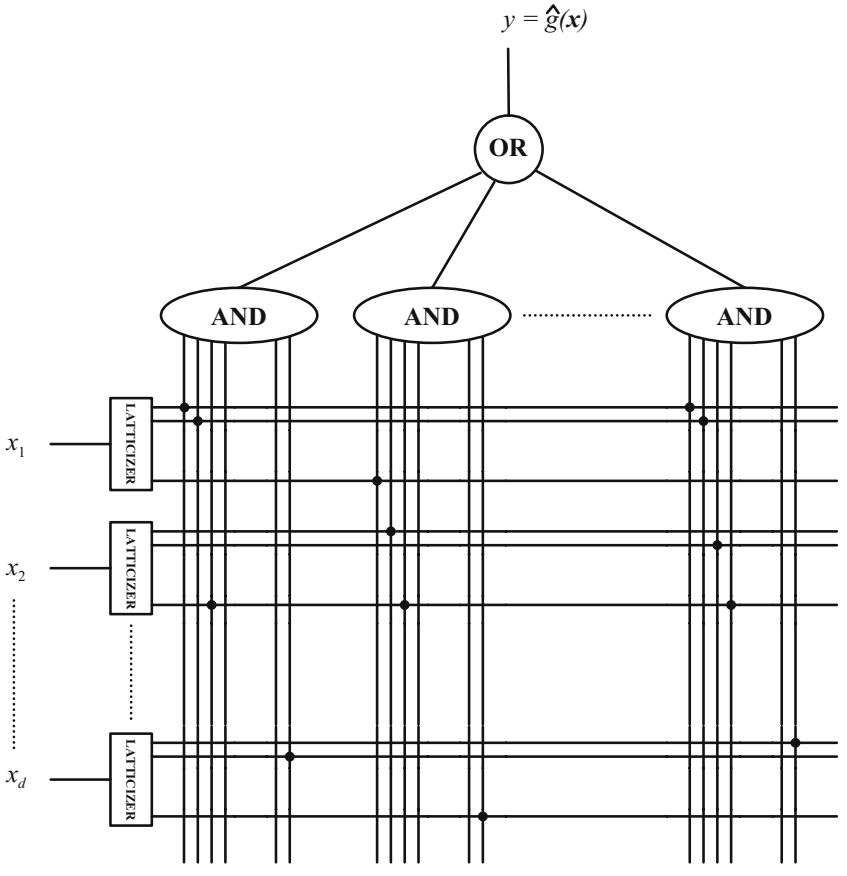


Fig. 1. Schema of a Switching Neural Network

leaving the latticizers; they correspond to values 1 in the associated implicant. These connections are represented by bold circles in Fig. 1 and can be viewed as switches that establish links between the inputs of the AND ports and the outputs of the latticizers. For this reason the connectionist model shown in Fig. 1 is called *Switching Neural Network* (*SNN*).

It is interesting to observe that, unlike standard neural networks, SNNs do not involve weights; furthermore, signals traveling on them have only one level. Thus, it should be concluded that the behavior of the function $\hat{g}(\boldsymbol{x})$ is entirely memorized in the architecture of the SNN (connections and switches). This is not a limitation, since it has been shown that SNN are universal approximators.

Every implicant $\boldsymbol{a} \in A$ generated by SC can be translated into an intelligible rule in the **if-then** form underlying the classification problem at hand. Consider the substrings $\boldsymbol{a}_i$ of $\boldsymbol{a}$ that are associated with the ith input x_i to the network. The logical product $\bigwedge_{j \in P(\boldsymbol{a})} z_j$ gives output 1 only if the binary string $\boldsymbol{z} = \boldsymbol{\varphi}(\boldsymbol{x})$ presents a value 1 in all the positions where $\boldsymbol{a}_i$ has value 1.

If x_i is an ordered variable, the execution of SC always generates binary strings $\boldsymbol{a}_i$ containing a single sequence of consecutive values 0, i.e. a run of 0. If this run begins at the $(j+1)$th position and finishes at the kth bit of $\boldsymbol{a}_i$, the logical product $\bigwedge_{j \in P(\boldsymbol{a})} z_j$ can give output 1 only if $r_{ij} < x_i \leq r_{ik}$. In the particular case where the run of 0 begins at the first position (resp. finishes at the last position), the condition becomes $x_i \leq r_{ik}$ (resp. $x_i > r_{ij}$).

As an example, suppose that an ordered variable x_i has been discretized by using the four cutoffs 0.1, 0.25, 0.3, 0.5. If the implicant $\boldsymbol{a}$ with $\boldsymbol{a}_i = 10011$ has been produced by SC, the condition $0.1 < x_i \leq 0.3$ has to be included in the **if** part of the **if-then** rule associated with $\boldsymbol{a}$.

On the other hand, if x_i is a nominal variable the portion $\boldsymbol{a}_i$ of an implicant $\boldsymbol{a}$ gives rise to the condition $x_i \in \bigcup_{k \in I_{m_i} \setminus P(\boldsymbol{a}_i)} \{k\}$. Again, if the implicant $\boldsymbol{a}$ with $\boldsymbol{a}_i = 01101$ has been produced by SC, the condition $x_i \in \{1, 4\}$ has to be included in the **if-then** rule associated with $\boldsymbol{a}$. In any case, if the binary string $\boldsymbol{a}_i$ contains only values 0, the input x_i will not be considered in the rule for $\boldsymbol{a}$.

Thus, it follows that every implicant $\boldsymbol{a}$ gives rise to an **if-then** rule, having in its **if** part a conjunction of the conditions obtained from the substrings $\boldsymbol{a}_i$ associated with the d inputs x_i. If all these conditions are verified, the output $y = \hat{g}(\boldsymbol{x})$ will be assigned the value 1. A logical OR connects all the rules obtained in this way for every implicant $\boldsymbol{a}$ in the antichain A produced by SC.

Due to this property, SC (with the addition of discretization and latticization) becomes a rule generation method, being capable of retrieving from the training set some kind of intelligible information about the physical system underlying the binary classification problem at hand.

3 Simulation Results

To obtain a preliminary evaluation of performances achieved by SNNs trained with SC, the ten classification problems included in the well-known StatLog benchmark [8] have been considered. In this way the generalization ability and the complexity of resulting SNNs can be compared with those of other machine learning methods, among which rule generation techniques based on decision trees, such as C4.5 [9]. In all these simulations the discretization method adopted in the LAD system [6] has been used to map continuous inputs into binary strings.

The tests contained in the StatLog benchmark present different characteristics that allow to evaluate the behavior of a classification algorithm under several angles. Four problems (Heart, Australian, Diabetes, and German) presents a binary output, thus permitting a direct application of the SNN approach, as described in the previous section. However, two of them (Heart and German) adopts a specific cost matrix to weight misclassified patterns.

The remaining six tests concern multiclass problems, which have been split into a sequence of binary classification problems by constructing a separate set of implicants for each output value. The class of an new pattern $\boldsymbol{x}$ has then been chosen by adopting the criteria introduced in [10], which performs a weighted sum of perfect matching rules (those verified by $\boldsymbol{x}$), having weight 1, and almost

matching rules (those verified by $\boldsymbol{x}$, except for one condition), which is assigned the weight 0.1.

The complexity of an SNN is measured through the number of AND ports in the second layer (corresponding to the number of intelligible rules) and the average number of conditions in the **if** part of a rule. Tab. 1 presents the results obtained. Accuracy and complexity of resulting SNNs are compared with those of rulesets produced by C4.5. In the same table is also reported the best generalization error reported in the StatLog report [8] for each problem, together with the rank scored by SNN when its generalization error is inserted into the list of available results.

Table 1. Generalization error and complexity of SNN, compared with C4.5 and with other methods, on the StatLog benchmark

Test	Generalization error				# Rules		# Conditions	
Problem	SNN	C4.5	Best	Rank	SNN	C4.5	SNN	C4.5
HEART	0.393	0.781	0.374	2	24.3	11.4	5.03	2.68
AUSTRALIAN	0.125	0.155	0.131	1	26.4	11.5	5.55	2.76
DIABETES	0.250	0.270	0.223	8	73.8	9.4	4.61	2.58
VEHICLE	0.278	0.266	0.150	12	91.2	26.1	5.92	4.03
GERMAN	0.716	0.985	0.535	13	95.8	21.1	8.90	2.77
SEGMENT	0.037	0.040	0.030	11	82.8	28.0	4.51	3.94
DNA	0.057	0.076	0.041	3	132.0	34.0	8.99	4.47
SATIMAGE	0.135	0.150	0.094	6	262.0	80.0	7.92	5.41
LETTER	0.115	0.132	0.064	5	1532.0	570.0	8.38	7.64
SHUTTLE	0.0001	0.001	0.0001	1	18.0	20.0	3.17	3.14

Apart from one case (Vehicle) the generalization error scored by SNN is always lower than that obtained by C4.5. On the other hand, the complexity of SNN is considerably higher (except for the Shuttle problem). As a consequence of this greater complexity, the execution time of SNN is significantly higher than that of C4.5. It ranges from 3 sec. (Australian) to a hour (Letter) for the StatLog benchmark, whereas the construction of the set of rules with C4.5 requires at most three minutes.

This behavior depends on the method employed to produce the implicants and the corresponding rules. In SNN every rule separates some patterns of a class from all the example in the training set belonging to other classes; at most a small error is accepted to avoid overfitting. On the contrary, C4.5 creates rules that are verified by a subset of patterns from different classes. Subsequent rules correct errors performed by previous ones; therefore, rules must be applied in a specific order.

It is interesting to note that in four out of the ten problems SNN achieves one of the first three ranking positions. This points out the quality of the solutions offered by SNN, even if its behavior in dealing with multiclass problems can be improved by properly adapting the SC algorithm to reconstruct in an efficient way positive Boolean functions with several outputs.

References

1. Vapnik, V.N.: Statistical Learning Theory. New York: John Wiley & Sons (1998)
2. M. Muselli, A. Quarati: Reconstructing positive Boolean functions with Shadow Clustering. In Proceedings of the 17th European Conference on Circuit Theory and Design (ECCTD 2005), (Cork, Ireland, August 2005).
3. M. Muselli: Approximation properties of positive Boolean functions. In WIRN/NAIS 2005, vol. 3931 of Lecture Notes in Computer Science (2006) Eds. B. Apolloni et al., Berlin: Springer-Verlag, 18-22.
4. Kohavi, R., Sahami, M.: Error-based and entropy-based discretization of continuous features. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (1996) 114–119
5. Liu, H., Setiono, R.: Feature selection via discretization. IEEE Transactions on Knowledge and Data Engineering **9** (1997) 642–645
6. Boros, E., Hammer, P.L., Ibaraki, T., Kogan, A., Mayoraz, E., Muchnik, I.: An Implementation of Logical Analysis of Data. IEEE Transactions on Knowledge and Data Engineering **12** (2000) 292–306
7. Muselli, M., Liberati, D.: Binary rule generation via Hamming Clustering. IEEE Transactions on Knowledge and Data Engineering **14** (2002) 1258–1268
8. Michie, D., Spiegelhalter, D., Taylor, C., eds.: Machine Learning, Neural, and Statistical Classification, London: Ellis-Horwood (1994)
9. Quinlan, J.R.: C4.5: Programs for Machine Learning. San Francisco: Morgan Kaufmann (1994)
10. Hong, S.J.: R-MINI: An Iterative Approach for Generating Minimal Rules from Examples. IEEE Transactions on Knowledge and Data Engineering **9** (1997) 709–717

Ensembles Based on Random Projections to Improve the Accuracy of Clustering Algorithms

Alberto Bertoni and Giorgio Valentini

DSI, Dipartimento di Scienze dell' Informazione,
Università degli Studi di Milano,
Via Comelico 39, 20135 Milano, Italia
{bertoni, valentini}@dsi.unimi.it

Abstract. We present an algorithmic scheme for unsupervised cluster ensembles, based on randomized projections between metric spaces, by which a substantial dimensionality reduction is obtained. Multiple clusterings are performed on random subspaces, approximately preserving the distances between the projected data, and then they are combined using a pairwise similarity matrix; in this way the accuracy of each "base" clustering is maintained, and the diversity between them is improved. The proposed approach is effective for clustering problems characterized by high dimensional data, as shown by our preliminary experimental results.

1 Introduction

Supervised multi-classifiers systems characterized the early development of ensemble methods [1, 2]. Recently this approach has been extended to unsupervised clustering problems [3, 4].

In a previous work we proposed stability measures that make use of random projections to assess cluster reliability [5], extending a previous approach [6] based on an unsupervised version of the random subspace method [7].

In this paper we adopt the same approach to develop cluster ensembles based on random projections. Unfortunately, a deterministic projection of the data into relatively low dimensional spaces may introduce relevant distortions, and, as a consequence, the clustering in the projected space may results consistently different from the clustering in the original space. For these reasons we propose to perform multiple clusterings on randomly chosen projected subspaces, approximately preserving the distances between the examples, and then combining them to generate the final "consensus" clustering.

The next section introduces basic concepts about randomized embeddings between metric spaces. Sect. 3 presents the *Randomized embedding clustering (RE-Clust)* ensemble algorithm, and Sect. 4 show the results of the application of the ensemble method to high dimensional synthetic data. The discussion of the results and the outgoing developments of the present work end the paper.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 31–37, 2006.

2 Randomized Embeddings

2.1 Randomized Embeddings with Low Distortion

Dimensionality reduction may be obtained by mapping points from a high to a low-dimensional space: $\mu : \mathbb{R}^d \rightarrow \mathbb{R}^d$, with $d' < d$, approximately preserving some characteristics, i.e. the distances between points In this way, algorithms whose results depend only on the distances $||x_i - x_j||$ could be applied to the compressed data $\mu(X)$, giving the same results, as in the original input space. In this context randomized embeddings with low distortion represent a key concept. A *randomized embedding* between $\mathbb{R}^d$ and $\mathbb{R}^d$ with distortion $1+\epsilon$, $(0 < \epsilon \leq 1/2)$ and failure probability P is a distribution probability on the linear mapping $\mu : \mathbb{R}^d \rightarrow \mathbb{R}^d$, such that, for every pair $p, q \in \mathbb{R}^d$, the following property holds with probability $\geq 1 - P$:

$$\frac{1}{1+\epsilon} \leq \frac{||\mu(p) - \mu(q)||}{||p - q||} \leq 1 + \epsilon \tag{1}$$

The main result on randomized embedding is due to Johnson and Lindenstrauss [8], who proved the following:

Johnson-Lindenstrauss (JL) lemma: Given a set S with $|S| = n$ there exists a $1 + \epsilon$-distortion embedding into $\mathbb{R}^d$ with $d' = c \log n/\epsilon^2$, where c is a suitable constant.

The embedding exhibited in [8] consists in random projections from $\mathbb{R}^d$ into $\mathbb{R}^d$, represented by matrices $d' \times d$ with random orthonormal vectors. Similar results may be obtained by using simpler embeddings [9], represented through random $d' \times d$ matrices $P = 1/\sqrt{d'}(r_{ij})$, where r_{ij} are random variables such that:

$$E[r_{ij}] = 0, \qquad Var[r_{ij}] = 1$$

For sake of simplicity, we call random projections even this kind of embeddings.

2.2 Random Projections

Examples of randomized maps, represented trough $d' \times d$ matrices P such that the columns of the "compressed" data set $D_P = PD$ have approximately the same distance are:

1. *Plus-Minus-One (PMO)*: random projections: represented by matrices $P = 1/\sqrt{d'}(r_{ij})$, where r_{ij} are uniformly chosen in $\{-1, 1\}$, such that $Prob(r_{ij} = 1) = Prob(r_{ij} = -1) = 1/2$. In this case the *JL lemma* holds with $c \simeq 4$.
2. *Random Subspace (RS)* [7]: represented by $d' \times d$ matrices $P = \sqrt{d/d'}(r_{ij})$, where r_{ij} are uniformly chosen with entries in $\{0, 1\}$, and with exactly one "1" per row and at most one "1" per column. Even if *RS* subspaces can be quickly computed, the do not satisfy the *JL lemma*.

3 Randomized Embedding Cluster Ensembles

Consider a data set $X = \{x_1, x_2, \ldots, x_n\}$, where $x_i \in \mathbb{R}^d$, $(1 \leq i \leq n)$; a subset $A \subseteq \{1, 2, \ldots, n\}$ univocally individuates a subset of examples $\{x_j | j \in A\} \subseteq X$. The data set X may be represented as a $d \times n$ matrix D, where columns correspond to the examples, and rows correspond to the "components" of the examples $x \in X$. A *k-clustering* C of X is a list $C =< A_1, A_2, \ldots, A_k >$, with $A_i \subseteq \{1, 2, \ldots, n\}$ and such that $\bigcup A_i = \{1, \ldots, n\}$. A *clustering algorithm* $\mathcal{C}$ is a procedure that, having as input a data set X and an integer k, outputs a k-clustering C of X: $\mathcal{C}(X, k) =< A_1, A_2, \ldots, A_k >$.

The main ideas behind the proposed cluster ensemble algorithm *RE-Clust* (acronym for Randomized Embedding Clustering) are based on data compression, and generation and combination of multiple "base" clusterings. Indeed at first data are randomly projected from the original to lower dimensional subspaces, using projections described in Sect 2.2 in order to approximately preserve the distances between the examples. Then multiple clusterings are performed on multiple instances of the projected data, and a similarity matrix between pairs of examples is used to combine the multiple clusterings.

The high level pseudo-code of the ensemble algorithm scheme is the following:

***RE-Clust* Algorithm:**
`Input`:

- a data set $X = \{x_1, x_2, \ldots, x_n\}$, represented by a $d \times n$ D matrix.
- an integer k (number of clusters)
- a real $\epsilon > 0$ (distortion level)
- an integer c (number of clusterings)
- two clustering algorithms $\mathcal{C}$ and $\mathcal{C}_{com}$
- a procedure that realizes a randomized map μ

`begin algorithm`
(1) $d' = 2 \cdot \left(\frac{2 \log n + \log c}{\epsilon^2}\right)$
(2) `For each` $i, j \in \{1, \ldots, n\}$ `do` $M_{ij} = 0$
(3) `Repeat for` $t = 1$ `to` c
(4) $P_t =$ `Generate_projection_matrix` (d, d')
(5) $D_t = P_t \cdot D$
(6) $< C_1^{(t)}, C_2^{(t)}, \ldots, C_k^{(t)} >= \mathcal{C}(D_t, k)$
(7) `For each` $i, j \in \{1, \ldots, n\}$
$M_{ij}^{(t)} = \frac{1}{k} \sum_{s=1}^{k} I(i \in C_s^{(t)}) \cdot I(j \in C_s^{(t)})$
`end repeat`
(8)$M = \frac{\sum_{t=1}^{c} M^{(t)}}{c}$
(9) $< A_1, A_2, \ldots, A_k >= \mathcal{C}_{com}(M, k)$
`end algorithm`.
`Output`:

- the final clustering $C =< A_1, A_2, \ldots, A_k >$

In the first step of the algorithm, given a distortion level ϵ, the dimension d' for the compressed data is computed according to the *JL lemma*.

At each iteration of the main repeat loop (step 3-7), the procedure `Generate_projection_matrix` outputs a projection matrix P_t according to the randomized embedding μ, and a projected data set $D_t = P_t \cdot D$ is generated; the corresponding clustering $< C_1^{(t)}, C_2^{(t)}, \ldots, C_k^{(t)} >$ is computed by calling $\mathcal{C}$, and a $M^{(t)}$ similarity matrix is built. The *similarity matrix* $M^{(t)}$ associated to a clustering $C =< C_1^{(t)}, C_2^{(t)}, \ldots, C_k^{(t)} >$ is a $n \times n$ matrix such that:

$$M_{ij}^{(t)} = \frac{1}{k} \sum_{s=1}^{k} I(i \in C_s^{(t)}) \cdot I(j \in C_s^{(t)}) \tag{2}$$

where I is is the characteristic function of the set C_s. After step (8), M_{ij} denotes the frequency by which the examples i and j occur in the same cluster across multiple clusterings. The final clustering is performed by applying the clustering algorithm $\mathcal{C}_{com}$ to the main similarity matrix M. Choosing different random projections we may generate different *RE-Clust* ensembles (e.g. *PMO* and *RS* cluster ensembles).

4 Experimental Results

In this section we present some preliminary experimental results with the *RE-Clust* ensemble algorithm. The Ward's hierarchical agglomerative clustering algorithm [10] has been applied as "base" clustering algorithm.

4.1 Experimental Environment

Synthetic Data Generation. We experimented with 2 different sample generators, whose samples are distributed according to different mixtures of high dimensional gaussian distributions.

Sample1 is a generator for 5000-dimensional data sets composed by 3 clusters. The elements of each cluster are distributed according to a spherical gaussian with standard deviation equal to 3. The first cluster is centered in $\mathbf{0}$, that is a 5000-dimensional vector with all zeros. The other two clusters are centered in $0.5\mathbf{e}$ and $-0.5\mathbf{e}$, where $\mathbf{e}$ is a vector with all 1.

Sample2 is a a generator for 6000-dimensional data sets composed by 5 clusters of data normally distributed. The diagonal of the covariance matrix for all the classes has its element equal to 1 (first 1000 elements) and equal to 2 (last 5000 elements). The first 1000 variables of the five clusters are respectively centered in $\mathbf{0}$, $\mathbf{e}$, $-\mathbf{e}$, $5\mathbf{e}$, $-5\mathbf{e}$. The remaining 5000 variables are centered in 0 for all clusters.

For each generator, we considered 30 different random samples each respectively composed by $60, 100$ examples (that is, 20 examples per class).

Experimental Setup. We compared classical single hierarchical clustering algorithm with our ensemble approach considering *PMO* and *RS* random projections (Sect. 2.2). We used 30 different realizations for each synthetic data set, using each

time 20 clusterings for both *PMO* and *RS* ensembles. For each *PMO* and *RS* ensemble we experimented with different distortions, corresponding to $\epsilon \in [0.06, 0.5]$.

We implemented the ensemble algorithms and the scripts used for the experiments in the *R* language (code is freely available from the authors).

4.2 Results

With *sample1* (Fig.1 (a)) for 1.10 distortion, that corresponds to projections from the original 5000 into a 3407 dimensional subspace, *RE-Clust* ensembles perform

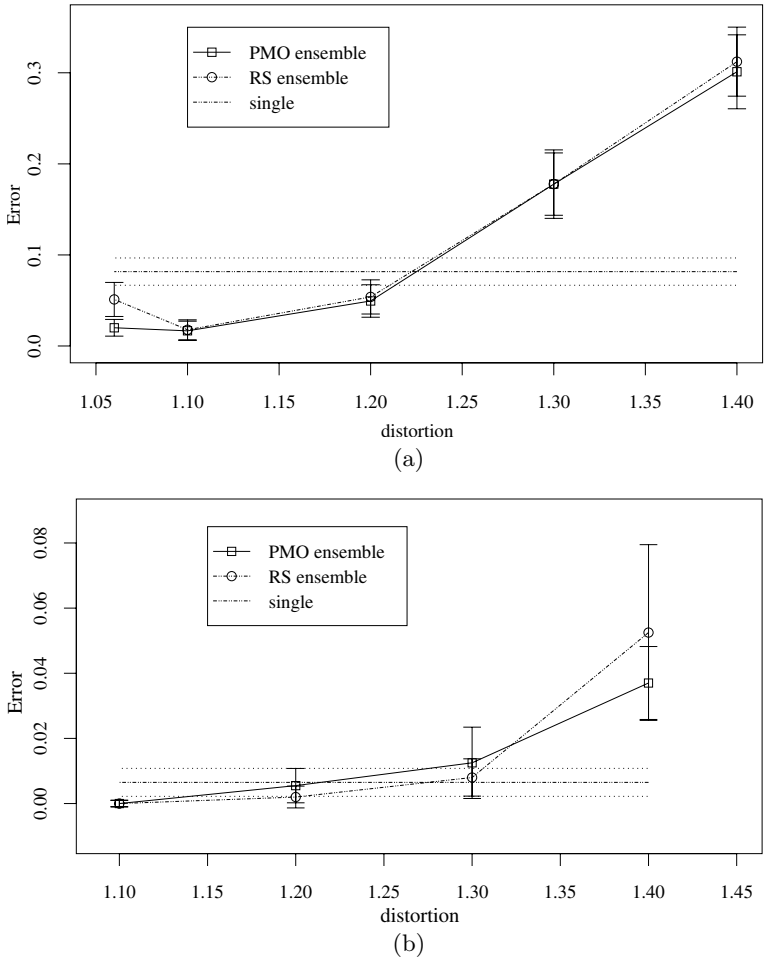


Fig. 1. Comparison of mean errors between single hierarchical clustering, PMO and RS ensembles with different $1 + \epsilon$ distortions. For ensembles, error bars for the 99% confidence interval are represented, while for single hierarchical clustering the 99% confidence interval is represented by the dotted lines above and below the horizontal dash-dotted line. (a) *Sample1* data set (b) *sample2*.

significantly better than single clustering. Indeed *PMO* ensembles achieve a 0.017 ± 0.010 mean error over 30 different realizations from *sample1*, and *RS* ensembles a 0.018 ± 0.011 mean error against a 0.082 ± 0.015 mean error for single hierarchical clustering. Also with an estimated 1.20 distortion (with a corresponding subspace dimension equal to 852) we obtain significantly better results with both *PMO* and *RS* ensembles.

With *sample2* (Fig.1 (b)) the difference is significant only for 1.10 distortion, while for larger distortions the difference is not significant and, on the contrary, with 1.4 distortion *RE-Clust* ensembles perform worse than single clustering. This may be due both to the relatively high distortion induced by the randomized embedding and to the loss of information due to the random projection to a too low dimensional space. Anyway, with all the high dimensional synthetic data sets the *RE-Clust* ensembles achieve equal or better results with respect to a "single" hierarchical clustering approach, at least when the distortions predicted by the *JL lemma* are lower than 1.30.

5 Conclusions

Experimental results with synthetic data (Sect. 4.2) show that *RE-Clust* ensembles are effective with high dimensional data, even if we need more experiments to confirm these results.

About the reasons why *RE-Clust* outperforms single clustering, we suspect that *RE-Clust* ensembles can reduce the variance component of the error, by "averaging" between different multiple clusterings, and we are planning to perform a bias-variance analysis of the algorithm to investigate this topic, using the approach proposed in [11] for supervised ensembles.

To evaluate the performance of *RE-Clust* with other "base" clustering algorithms, we are experimenting with *Partitioning Around Medoids (PAM)* and *fuzzy-c-mean* algorithms.

Acknowledgement

The present work has been developed in the context of the *CIMAINA* Center of Excellence, and it was partially funded by the italian COFIN project *Linguaggi formali ed automi: metodi, modelli ed applicazioni*.

References

[1] Dietterich, T.: Ensemble methods in machine learning. In Kittler, J., Roli, F., eds.: Multiple Classifier Systems. First International Workshop, MCS 2000, Cagliari, Italy. Volume 1857 of Lecture Notes in Computer Science., Springer-Verlag (2000) 1–15

[2] Valentini, G., Masulli, F.: Ensembles of learning machines. In: Neural Nets WIRN-02. Volume 2486 of Lecture Notes in Computer Science. Springer-Verlag (2002) 3–19

[3] Strehl, A., Ghosh, J.: Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions. Journal of Machine Learning Research **3** (2002) 583–618
[4] Hadjitodorov, S., Kuncheva, L., Todorova, L.: Moderate Diversity for Better Cluster Ensembles. Information Fusion (2005)
[5] Bertoni, A., Valentini, G.: Random projections for assessing gene expression cluster stability. In: IJCNN 2005, The IEEE-INNS International Joint Conference on Neural Networks, Montreal (2005) (in press).
[6] Smolkin, M., Gosh, D.: Cluster stability scores for microarray data in cancer studies. BMC Bioinformatics **4** (2003)
[7] Ho, T.: The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence **20** (1998) 832–844
[8] Johnson, W., Lindenstrauss, J.: Extensions of Lipshitz mapping into Hilbert space. In: Conference in modern analysis and probability. Volume 26 of Contemporary Mathematics., Amer. Math. Soc. (1984) 189–206
[9] Bingham, E., Mannila, H.: Random projection in dimensionality reduction: Applications to image and text data. In: Proc. of KDD 01, San Francisco, CA, USA, ACM (2001)
[10] Ward, J.: Hierarchcal grouping to optimize an objective function. J. Am. Stat. Assoc. **58** (1963) 236–244
[11] Valentini, G.: An experimental bias-variance analysis of SVM ensembles based on resampling techniques. IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics **35** (2005)

Recursive Neural Networks and Graphs: Dealing with Cycles

M. Bianchini, M. Gori, L. Sarti, and F. Scarselli

Dipartimento di Ingegneria dell'Informazione,
Università degli Studi di Siena,
Via Roma, 56 — 53100 Siena, Italy
{monica, marco, sarti, franco}@dii.unisi.it

Abstract. Recursive neural networks are a powerful tool for processing structured data. According to the recursive learning paradigm, the input information consists of directed positional acyclic graphs (DPAGs). In fact, recursive networks are fed following the partial order defined by the links of the graph. Unfortunately, the hypothesis of processing DPAGs is sometimes too restrictive, being the nature of some real–world problems intrinsically cyclic. In this paper, the methodology proposed in [1, 2] to process cyclic directed graphs is tested on some interesting problems in the field of structural pattern recognition. Such preliminary experimentation shows very promising results.

1 Introduction

In several applications, the information which is relevant for solving problems is encoded into the relationships between some basic entities. In [3, 4], *recursive neural networks* were introduced in order to process structured data. In fact, recursive networks are fed with directed positional acyclic graphs (DPAGs), since their learning paradigm is based on the partial order defined by the links of the graph. Unfortunately, the hypothesis of processing DPAGs is sometimes too restrictive, being the nature of some real–world problems intrinsically cyclic. For instance, Web pages can naturally be represented by graphs [5], which can be deduced directly by the HTML tags: nodes denote the logic contexts, while arcs denote the inclusion relationships between contexts, and the hyperlinks. The labels contain the words that are enclosed in the corresponding contexts. Such graphs are typically ordered, directed, and cyclic. On the other hand, both for image analysis [6] and for the prediction of the biological activity of chemical compounds [3], data can be coded as undirected graphs, whose nodes are labeled, respectively, by feature vectors and by atoms or simple molecules.

In this paper, the method proposed in [1, 2] to allow recursive processing of cyclic graphs is tested on cyclic graph recognition and subgraph detection. The paper is organized as follows. In the next section, we briefly sketch the procedure aimed at transforming cyclic graphs into *recursive–equivalent trees* [2]. Section 3 collects the experimental results, and, finally, Section 4 draws some conclusions.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 38–43, 2006.

2 Recursive Processing of Cyclic Graphs

A recursive network [3] consists of two feedforward networks: a *transition network* f and an *output network* g. In order to process a graph G, the transition network is unfolded through the structure of the input graph. The network produced by the unfolding is called *encoding network* and has the same structure of the graph, while nodes are replaced by copies of f (see Fig. 1). Then, a feedforward computation is carried out on the encoding network. In this way, at each node v of the graph, the *state* $\mathbf{X}_v$ is computed by the transition network as a function of the input label $\mathbf{U}_v$ and of the states of its children, $\mathbf{X}_{\mathrm{ch}[v]}$:

$$\mathbf{X}_v = f(\mathbf{X}_{\mathrm{ch}[v]}, \mathbf{U}_v, \theta_f), \tag{1}$$

with

$$\mathbf{X}_{\mathrm{ch}[v]} = [\mathbf{X}'_{\mathrm{ch}_1[v]}, \ldots, \mathbf{X}'_{\mathrm{ch}_o[v]}]', \quad o = \max_{v \in V}\{\mathrm{outdegree}[v]\},$$

and $\mathbf{X}_{\mathrm{ch}_i[v]}$ equal to the *frontier state* $\mathbf{X}_0$, if node v lacks of its i–th child. Following [3], we suppose to process rooted graphs and, at the root node s, an output network g computes

$$\mathbf{Y}_s = g(\mathbf{X}_s, \theta_g)\,,$$

which can be regarded as the output of the whole recursive process. Thus, the recursive network implements a function $\phi(G)$ of the input graph.

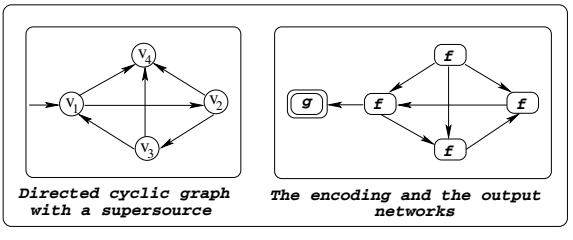


Fig. 1. The encoding and the output networks for a cyclic graph

In this learning paradigm, a state $\mathbf{X}_v$ at a node v, involved in a cycle, should be defined in terms of itself, so that the neural network should act as a dynamical system, whose stable equilibrium points are the solutions of eq. (1). In fact, for a cyclic graph, also the encoding network is cyclic (see Fig. 1).

On the cyclic encoding network, the computation is carried out by setting all the initial states $\mathbf{X}_v$ to $\mathbf{X}_0$. Then, the copies of the transition network are repeatedly activated to update the states. According to eq. (1), the transition network attached to node v produces the new state $\mathbf{X}_v$ of v. After some updates, the computation can be stopped. The whole procedure is formalized in the following algorithm.

Algorithm 1.1 : `CyclicRecursive(G)`

```
1    begin
2    for each  v ∈ V do X_v = X_0;
3    repeat
4        < Select v ∈ V >;
5        X_v = f(X_ch[v], U_v, θ_f);
6    until stop();
7    Y_s = g(X_s, θ_g);
8    end
```

The above algorithm can be easily applied also to undirected graphs, provided that $\mathbf{X}_{\mathrm{ch}[v]}$ is replaced by the states $\mathbf{X}_{\mathrm{co}[v]}$ of the nodes connected to v. Moreover, Algorithm `CyclicRecusive` is a generic framework, which describes a class of procedures. In order to implement a particular procedure, we should decide the strategy adopted to select the nodes (line 4) and the criterion used to halt the function (line 6). In fact, it can be proved that, under mild assumptions on the strategies adopted to implement the functions in lines 4 and 6, such an extended recursive model implements most of the practically useful functions on graphs.

The parameters θ_f, θ_g can be computed by BackPropagation Through Structure [7]. In order to reach this goal, a *recursive–equivalent* tree T must be generated for each cyclic graph G belonging to the training set [1]. Intuitively, a recursive–equivalent tree is a structure that contains the same information of the original structure. The following procedure describes how to produce T:

a. Insert a copy of the root s in T;
b. Visit G, starting from s; for each visited node v, insert a copy of v into T; link v to its parent preserving the information attached to each edge;
c. Continue the visit until a predefined stop criterion is satisfied, and, however, until all the arcs have been visited (at least) once.

At the end of the procedure, the target associated to s in G is attached to the root node of T. According to [1, 2], the above unfolding strategy produces a recursive–equivalent tree that holds the same information contained in G. With respect to the visit strategy (step b.) and the stop criterion (step c.), they can be defined from the functions used in lines 4 and 6 of Algorithm 1.1.

3 Experimental Results

Some experiments were carried out to evaluate the proposed method on two significant tasks: the recognition of cyclic graphs and the detection of subgraphs. The former problem consists of discovering if a graph is cyclic or not, and was chosen in order to verify whether the transformation from cyclic graphs to recursive equivalent trees preserves the information inherently contained in the graph structure. In the subgraph detection problem, a graph is explored to establish if it contains a given subgraph or not. This latter problem is often encountered in practical applications. Applications to real–world tasks can be found in [6, 8, 9, 10], where the problem of object detection in images is faced.

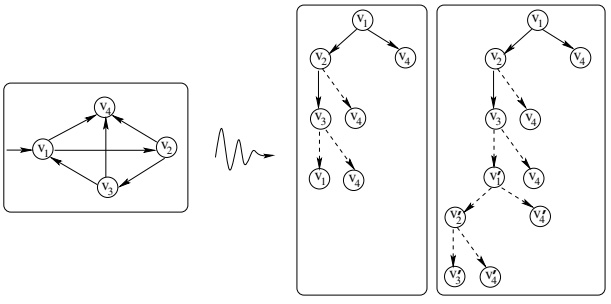


Fig. 2. The transformation from a graph to recursive–equivalent trees

The experimental data consisted of synthetic graphs generated by the following procedure:

1. For each graph, the procedure takes in input a range $[n_1, n_2]$ for the number of nodes, the number of edges e, and the maximum outdegree o.
2. First, n nodes are generated, where n is a random number in $[n_1, n_2]$.
3. Then, two random integers v_1, v_2 are generated, and the edge (v_1, v_2) is inserted into the graph, provided that the edge is not a self–connection ($v_1 = v_2$) and node v_1 has not reached the maximum outdegree o. This step is repeated until the graph contains e edges.
4. If a root node does not exist, it is added using the algorithm in [4].

For each experiment, a learning set, a validation set and a test set were generated using the above procedure. Cyclic graphs belonging to the three sets were transformed into trees using the procedure described in Section 2. With respect to the stop criterion chosen, initially all the edges were visited once, subsequently the algorithm continues to unfold the graph until a stochastic variable x becomes *true*. The variable x was *true* with probability 0.4. This halt criterion was chosen for practical reasons, since it guarantees both the recursive–equivalence of the generated trees and a small number of nodes in the trees.

Recognition of Cyclic Graphs. The problem of recognizing cycles is related to a property of the whole graph, while the recursive model is based on a local computation. Therefore, this problem is particularly interesting, since it is a difficult test for our approach. We have payed a particular attention to the selection of the parameters for generating graphs. A correct experimentation required that recursive–equivalent trees corresponding to cyclic structures and acyclic graphs cannot be discriminated only by the number of nodes. Moreover, the sets of cyclic and acyclic graphs should be balanced (with similar cardinality). In order to reach those goals, we selected the range of the number of nodes as $[5, 10]$, the number of edges as 15, and the maximum outdegree of each node as 3. Moreover, each node was labeled with a vector of five randomly generated binary values. The label was checked to verify its uniqueness. The training and the test sets collect 500 graphs, while the validation set consists of 125 graphs.

The transition function f and the output function g of the recursive network were implemented by MLPs. The number of units used for representing the states and the number of hidden neurons were chosen empirically. The collected results are described by two indexes, the accuracy and the rejection rate. The accuracy stands for the percentage of patterns correctly classified, while the rejection rate describes the percentage of patterns on which the network output belongs to $[0.4, 0.6]$, i.e. the network is unable to take a decision. In view of the difficulty of the problem, the achieved results (accuracy 84.5%, rejection rate 6.5%) are promising.

Subgraph Detection. Subgraph detection is a problem that is often encountered in Pattern Recognition. Usually, the graph represents an object and the subgraph represents a pattern that must be searched inside the object. The data set used for those experiments consisted of graphs with 5, 6, 7, 15, and 30 nodes and with maximum outdegree equal to 3. Each label contains a random integer in $[0, 50]$ (shared labels were avoided). The number of nodes of the subgraphs to be detected ranges between 3 and 5, moreover subgraphs can be cyclic or acyclic. Twenty–eight experiments were run using graphs and subgraphs of different dimensions. More precisely, two experiments were carried out for each pair (g, s), $s < g$, where $g \in \{5, 6, 7, 15, 30\}$ was the graph dimension and $s \in \{3, 4, 5\}$ was the subgraph dimension: an experiment was performed on an acyclic subgraph, whereas the other on a cyclic one. For each run, a random small subgraph was generated. In order to produce positive and negative examples, the subgraph was inserted into a random position in a half of the graphs of the data set. For each experiment, the training set contained 1000 graphs, the validation set 250 graphs, and the test set 1000 graphs. Each cyclic graph was preprocessed in order to generate the corresponding recursive–equivalent tree. The recursive network architecture was determined using a trial–and–error procedure. The achieved average results guarantee an accuracy of 75.94%, and a rejection rate of 7.58%. More details are described in Table 1 and Table 2. It is worth noting that the accuracy is larger for cyclic graphs than for acyclic ones (the third and the fourth columns of Table 1). A possible explanation of the above behavior is that the preprocessing often maps cyclic subgraphs to larger subtrees w.r.t. acyclic graphs. As the subgraph becomes larger, also the probability to discover its distinctive features increases. Moreover, the accuracy decreases when the size of the graphs increases (see Table 2). Finally, further twenty–eight experiments were carried out in order to measure the robustness of the method. Training, validation and test sets were modified by adding to the labels a random noise with a uniform distribution in $[-1, 1]$. The last two columns of Table 1 show the results.

Table 1. The effect of subgraph dimension, the presence of cycles in subgraph, and label noise on the performance

	Subgraphs with 3 nodes	Subgraphs with 4 nodes	Subgraphs with 5 nodes	Cyclic subgraphs	Acyclic subgraphs	Graphs with noise	Graphs without noise
Accuracy	73.51%	76.48%	83.67%	77.14%	74.49%	74.77%	76.87%
Rejection rate	7.15%	8.06%	6.8%	6.3%	9.2%	8.7%	6.78%

Table 2. The effect of graph dimension on the performance

Graph Size	5	6	7	15	30
Accuracy	77.05%	76.68%	75.7%	74.06%	72.42%
Uncertainty	9.05%	8.68%	7.7%	6.06%	7.36

4 Conclusions

In this paper, we have experimentally evaluated a new methodology recently proposed to process cyclic graphs using recursive neural networks. A preprocessing phase is used in order to injectively map graphs into trees, so that theoretical results about recursive processing of trees can be directly extended to generic graphs. The experimental results are promising and confirm the previous theoretical results.

References

1. Bianchini, M., Gori, M., Sarti, L., Scarselli, F.: Recursive processing of cyclic graphs. IEEE Transactions on Neural Networks with vol. 17, no. 1 (2006) 10–18.
2. Bianchini, M., Gori, M., Scarselli, F.: Recursive processing of cyclic graphs. IEEE International Joint Conference on Neural Networks (2002) 154–159
3. Frasconi, P., Gori, M., Sperduti, A.: A general framework for adaptive processing of data structures. IEEE Transactions on Neural Networks **9** (1998) 768–786
4. Sperduti, A., Starita, A.: Supervised neural networks for the classification of structures. IEEE Transactions on Neural Networks **8** (1997) 429–459
5. Gori, M., Maggini, M., Martinelli, E., Scarselli, F.: Learning user profiles in NAUTILUS. In: International Conference on Adaptive Hypermedia and Adaptive Web–based Systems, Trento (Italy) (2000)
6. Bianchini, M., Maggini, M., Sarti, L., Scarselli, F.: Recursive neural networks learn to localize faces. Pattern Recognition Letters with no. 26 (2005) 1885–1895.
7. Küchler, A., Goller, C.: Inductive learning in symbolic domains using structure–driven recurrent neural networks. In Görz, G., Hölldobler, S., eds.: Advances in Artificial Intelligence. Springer, Berlin (1996) 183–197
8. Bianchini, M., Mazzoni, P., Sarti, L., Scarselli, F.: Face spotting in color images using recursive neural networks. In Gori, M., Marinai, S., eds.: IAPR–TC3 International Workshop on Artificial Neural Networks in Pattern Recognition, Florence (Italy) (2003) 76–81
9. Bianchini, M., Gori, M., Mazzoni, P., Sarti, L., Scarselli, F.: Face localization with recursive neural networks. In Marinaro, M., Tagliaferri, R., eds.: Neural Nets — WIRN '03. Springer, Vietri (Salerno, Italy) (2003)
10. Bianchini, M., Maggini, M., Sarti, L., Scarselli, F.: Recursive neural networks for object localization. In: Proceedings of IJCNN 2004. Volume 3., Budapest (Hungary), IEEE (2004) 1911–1915

A System for Transmitting a Coherent Burst of Activity Through a Network of Spiking Neurons

J. Bose, S.B. Furber, and J.L. Shapiro

School of Computer Science, University of Manchester,
Manchester M13 9PL, UK
bosej@cs.manchester.ac.uk,
{steve.furber, jonathan.shapiro}@manchester.ac.uk

Abstract. In this paper we examine issues involving the transmission of information by spike trains through networks made of real time asynchronous spiking neurons. For our convenience we use a spiking model that is has an intrinsic delay between an input and output spike. We look at issues involving transmission of a desired average level of stable spiking activity over many layers, and show how feed-back reset inhibition can achieve this aim. We then deal with the coherence of spike trains and show that it is possible for a burst of spikes emitted by a layer to not diverge when passing through different layers of neurons. We present the results of simulations done on a multi layered feed-forward system to illustrate our method.

1 Introduction

Spiking neural models have been a source of interest [1] due to their biological plausibility and computational power. We are interested in engineering high level systems such as associative memories out of low level components such as spiking neurons, and in this paper we deal with some of the modelling issues in any such undertaking. Spiking neurons transmit information in the form of electrical pulses called spikes, whose firing times carry information. High level systems transmit information as symbols, which can be translated to a series or burst of spikes in the equivalent low level model. In a spiking neuron model of a high level system, in order to preserve the integrity of the transmitted information it is important for these spike bursts forming symbols to be stable (not die out or explode as it is transmitted through layers of neurons) and coherent (different bursts of spikes should not interfere with each other: there should be an appreciable time lag between them so that the symbols can be distinguished). In this paper we seek to tackle these two issues. There has been previous work done in these issues: studies have been made of the dynamics of activity in synfire chains [2, 3], consisting of neurons linked in a feed-forward chains propagating spiking activity.

In this paper, we have used simulations of spiking systems to illustrate our solutions, mainly because we encountered the mentioned problems during our efforts to model a real memory through spiking neurons. For our simulation

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 44–48, 2006.

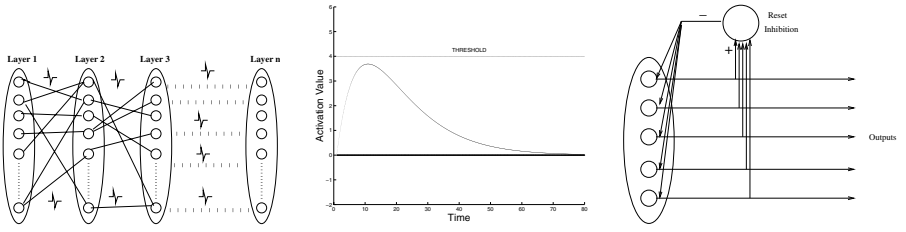


Fig. 1. (a) Architecture of the simulated network. The neurons in each layer are connected to those in the next layer with partial connectivity. The first layer fires a burst of spikes, which is fed to the second layer, whose outputs are fed to the third layer and so on. The temporal widths of the output spike bursts are measured. (b) Plot of a typical RDLIF neural activation with time, when the neuron receives a single input spike at time t=0. (c) Use of feed-back reset inhibition.

purposes we have taken a feed-forward system of layers of neurons emitting bursts of spikes (a synfire chain), as illustrated in figure 1(a). Such a system is simple and sufficient for manifesting the problems we described.

1.1 Model of Spiking Neuron

We use a rate driven leaky integrate and fire (RDLIF) model of spiking activity. As the name indicates, it is similar to the standard leaky integrate and fire (LIF) model [1], the only difference being that incoming spikes increase the driving force or the first derivative of the activation, rather than the activation itself. In this model, the behaviour of a neuron can be described by two variables: the activation a, the quantity which induces the neuron to emit a spike if it exceeds a threshold, and the activation driving force r, which controls the rate of increase of the activation. Both activation and driving force decay with time, the rate of decay being governed by their respective time constants τ_a and τ_r.

The driving force, or rate r_i of the i^{th} neuron increases with incoming spikes and decays with time t, till it reaches a resting value r_0

$$\dot{r_i} = \sum_j w_{ij} x_j - (r_i - r_0)/\tau_r \tag{1}$$

Here $x_j = \sum_n \delta(t - t_n)$ is the sum total of impulse functions of the input spikes emitted from the j^{th} input neuron and w_{ij} is the connection strength.

This driving force r_i drives the activation a_i, which itself decays with time to a resting value a_0.

$$\dot{a_i} = r_i - (a_i - a_0)/\tau_a \tag{2}$$

If the activation of the neuron exceeds its local threshold, it fires a spike and immediately its activation is reset to a refractory level, and driving force to 0 to prevent the activation from increasing.

Figure 1(b) shows the shape of the activation curve of an RDLIF neuron, following a single input spike at time 0. We see that the activation at first

increases due to the increased driving force caused by the incoming spike, but after a time the decay becomes dominant. For an RDLIF neuron to fire, it should get sufficient number of input spikes within a specified time, to enable it to reach the threshold before it 'dies out' because of the decay of the activation and activation rate. There is an inherent time lag between the input spike and the maximum activation reached by the neuron. If the system contains a feed-back loop, such a time lag is necessary, or else at least one input neuron would have to fire a spike at the same time as an output neuron fires a spike, and there would be no temporal separation between the input and output bursts. The standard LIF model cannot achieve this property. This motivated our choice of neural model. Although we have performed simulations on feed-forward networks only, in principle they should work equally well if feed-back loops are present.

2 Simulation of a Multi-layered System

We simulated a feed-forward network of partially connected layers of RDLIF neurons. We fed the first layer a uniformly distributed random set of spikes, and fed them into the second layer, the second layer spikes into the third layer and so on. Delays in the system are solely due to the second order dynamics inherent in the equations in the previous section. We then measured and plotted the temporal separation of the spike burst when passing through different layers.

2.1 Implementation Method

In this section we describe our simulation method for modelling an infinite number of different feed-forward layers of neurons. Our simulation program has a loop, each of whose iterations represents a propagation from layer i to layer (i+1). There is a different weight matrix in each iteration, representing different layers with the same average connectivity. The first layer is given a random set of spike firing times. We have an inner loop to count the time in time-steps in each such iteration, and in each time-step we check if any input or output neuron has fired. Each input spike increases the gain of the connected neuron proportional to the connection, as per our RDLIF model. The firings in the i^{th} layer cause spikes to fire in the $(i+1)^{th}$ layer. We found in our simulations that there was a time above which the gain and activation of all the neurons in a layer would decay and there would not be sufficient stimulus for any neuron to fire. We wait for this amount of time, found through simulation, which is sufficient for all the neurons in a layer to fire output spikes. We argue that this method (using time-steps and waiting for a specified time in each layer before moving on to the next layer) can be considered similar to an event-driven system.

The process of propagating the spike firings of one layer on to the next is repeated for the next iteration after copying the output vector of spike timings to the input layer. In each iteration the input is simply a vector of firing times and we get an output vector of firing times. We measure the temporal dispersion by taking the difference of first and last neuron firing times in that burst.

2.2 Sustaining Stable Activity in a Population of Neurons

Like we mentioned earlier, stability of the spiking activity over many layers is an important issue in modelling systems of spiking neurons. We found that for a given network, there was no threshold such that the system could sustain a stable average level of spiking activity over infinitely many layers. If the threshold is too low, the spiking activity dies out within a few layers. If it is too high, the activity increases with each layer, saturating to an unacceptably high level. The behaviour of the network abruptly switched from dying out of the spiking activity to saturation, as the threshold was increased. One of the ways of getting over this problem is through the use of feed-back reset inhibition. This can enable us to have a system in which a stable activity of firings could be sustained over a number of layers.

2.3 Effect of Feed-Back Reset Inhibition

Feed-back reset inhibition can be implemented by a neuron that is fed the output spikes from a layer, and fires an inhibitory spike once it gets a desired number of input spikes, say N (see figure 1(c)). This strong inhibitory spike resets all of the neurons in the layer, suppressing output activity of the layer. An RDLIF neuron with a threshold equal to the number of allowed spikes is equivalent to such a counter, provided the activation rate time constant τ_r is small and activation time constant τ_a large with respect to the input dispersion. Such systems can be used to implement N-of-M codes (when the inhibitory spike fires after exactly N neurons have fired out of a total of M), such as those used by Furber [4]. In our simulation, we implemented an 11-of-256 code, which could be sustained over infinitely many layers in this way.

2.4 Simulation Results

The parameters in our system are the individual neuron parameters (time constants, threshold) and system parameters (input time spreads and connectivity). Our model had 200 layers with 10% connectivity from layer to layer. The weights

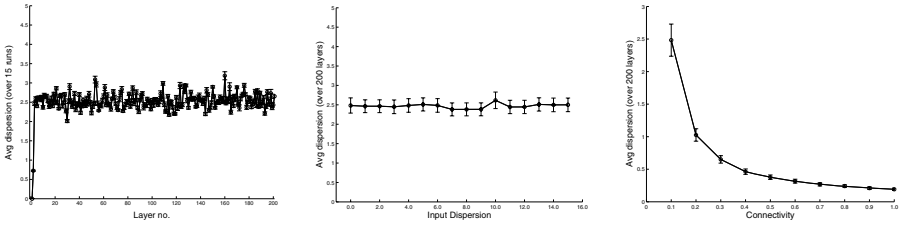


Fig. 2. (a) Plot of the average output dispersion, with initial input dispersion 0, of a burst of spikes passing through the 200-layer network. (b) Plot of the average output dispersion with varying input dispersions. (c) Plot of the variation of average output dispersion with network connectivity. As the connectivity increases, the dispersion decreases.

are real values between 0 and 0.1, chosen from an uniform distribution. The neurons have thresholds of 50 and reset values of -1, time-steps are 10 msec wide, and both time constants (τ_a and τ_r) are 1 sec each.

Figure 2(a) shows the variation of the average output dispersion (over 15 runs) with input dispersion when the initial temporal dispersion was 0. We find that the dispersion quickly tends to settle down into a range, from its input value of 0, and does not disperse much. We repeated this experiment with different values of initial input dispersion and found that we got the same behaviour with different input parameters. Thus, it is quite a stable and robust system.

We then varied other input parameters. Figure 2(b) shows the variation of the average output dispersion with the initial dispersion we gave to the first layer. We see that there is no appreciable change in the average dispersion, regardless of the input dispersion value. Figure 2(c) shows the variation with network connectivity. When the connectivity is low, the neurons have difficulty reaching the threshold and so the average spread is higher, and vice versa.

There are two important system-level time constants in our model, one for the temporal separation within a burst (our waiting time in each iteration from layer i to i+1), the other for separation between bursts (which can model axonal delays). Since we have shown that it is possible to design a system which can propagate a spike burst of desired spiking activity which can maintain coherence when passing through different layers, we can ensure that the inter-burst separation is sufficient (by inserting delays) so that successive waves of spikes do not impinge on each other. On the basis of this, we argue that it is possible to model a reliable system transmitting useful information using spiking neurons.

3 Conclusions and Future Work

In this paper we have studied issues involved in modelling systems of spiking neurons, and have shown that it is possible to propagate a coherent burst of spiking activity of a desired level over many layers. Work needs to be done in implementing real-time systems built with spiking neurons.

References

1. Wolfgang Maass, Christopher M. Bishop. Pulsed Neural Networks. MIT Press, 1999.
2. M. Abeles. Corticonics: Neural circuits of the cerebral cortex. Cambridge University Press, 1991.
3. David C. Sterratt. Spikes, synchrony, sequences and Schistocerca's sense of smell. PhD Thesis, University of Edinburgh, 2002.
4. S. B. Furber, J. M. Cumpstey, W. J. Bainbridge and S. Temple. Sparse distributed memory using N-of-M codes. Neural Networks, 2005, 10.

NEC: A Hierarchical Agglomerative Clustering Based on Fisher and Negentropy Information

Angelo Ciaramella[1], Giuseppe Longo[2], Antonino Staiano[1], and Roberto Tagliaferri[1]

[1] Department of Mathematics and Computer Science, University of Salerno,
via Ponte Don Melillo, I-84084, Fisciano, Salerno
{ciaram, astaiano, robtag}@unisa.it
[2] Dipartimento di Scienze Fisiche, University of Naples,
Polo delle Scienze e della Tecnologia, via Cintia 6 80136 Napoli, Italy
longo@unina.it

Abstract. In this paper a hierarchical agglomerative clustering is introduced. A hierarchy of two unsupervised clustering algorithms is considered. The first algorithm is based on a competitive Neural Network or on a Probabilistic Principal Surfaces approach and the second one on an agglomerative clustering based on both Fisher and Negentropy information. Different definitions of Negentropy information are used and some tests on complex synthetic data are presented.

1 Introduction

Several research areas such as data mining, pattern recognition and statistical data analysis make use of clustering to achieve some tasks which are a significant part of the process. The main objective of a good clustering algorithm is to separate classes (or clusters) arbitrarily distributed in data space, in an unsupervised way. The most simple way of clustering or classification is the use of vector quantization techniques [4] (i.e. k-means, fuzzy C-Means, SOM, etc..).

In this paper we propose a hierarchical agglomerative clustering composed by a hierarchy of two unsupervised clustering algorithms: the first algorithm is based on a competitive Neural Network or on a Probabilistic Principal Surfaces approach and the second one on an agglomerative clustering based on both Fisher and Negentropy information (NEC approach in the following).

We mark that the use of a non-euclidian metric to perform clustering was also explored in [8] and [7]. In these papers the authors use a non-parametric approach to estimate the density distribution of the data (i.e. Parzen Windows). In [8] the authors developed a clustering algorithm based on the Kullback-Leibler divergence to obtain a dissimilarity measure. We observe that this could introduce several parameters in the approach that are not very simple to manage. In [7] the authors developed a clustering algorithm based on the Kauchy-Schwarz inequality. Moreover, we note that to build a distribution in a high dimensional space is computationally expensive.

In the following, in Section 1 we introduce the clustering algorithms that accomplish the first step of the NEC approach. In Section 2 we focus our attention on the approach that accomplish the second phase in the NEC approach. In Section 3 and Section 4 we show some experimental results and conclusions, respectively.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 49–56, 2006.

2 Clustering Algorithms

We remark that to accomplish the agglomerative clustering the NEC approach needs two separated steps. In the first step a number of fixed clusters from our distribution are extracted and in the second step the clusters that have similar information are agglomerated.

The first approach that we consider is based on a competitive Neural Network (NN) [4]. The competitive NN is a Self-Organizing Map (SOM) that achieves vector quantization in a given data set.

More precisely we use a *winner takes all* approach in which during the competitive training only one neuron is updated for each input pattern.

The second approach that we consider is the Probabilistic Principal Surfaces (PPS) method that is a latent variable model which has been shown to be very effective for data mining purposes [2, 9]. PPS defines a non-linear, parametric mapping $\mathbf{y}(\mathbf{x}; \mathbf{W})$ from a Q-dimensional latent space ($\mathbf{x} \in R^Q$) to a D-dimensional data space ($\mathbf{t} \in R^D$), where normally $Q < D$. The mapping $\mathbf{y}(\mathbf{x}; \mathbf{W})$ (defined continuous and differentiable) maps every point in the latent space to a point into the data space. Since the latent space is Q-dimensional, these points will be confined to a Q-dimensional manifold non-linearly embedded into the D-dimensional data space. Substantially, the PPS approach builds a constrained mixture of Gaussians and the EM algorithm can be used to estimate the parameters of the model.

If $Q = 3$ is chosen, a spherical manifold [2] can be constructed using a PPS with nodes arranged regularly on the surface of a sphere in R^3 latent space, with the latent basis functions evenly distributed on the sphere at a lower density. After a PPS model is fitted to the data, several visualization possibilities are available for analyzing the data points [9].

3 Fisher's Linear Discriminant and Negentropy

We use both Fisher's and Negentropy information to agglomerate the clusters found in the first phase.

On one hand we note that the Fisher's linear discriminant is a classification method that projects high-dimensional data onto a line and performs classification in this one-dimensional space [1]. The projection maximizes the distance between the means of the two classes while minimizing the variance within each class. We have to note that the Fisher discriminant aims to achieve an optimal linear dimensionality reduction. It is therefore not strictly a discriminant itself, but it can easily be used to construct a discriminant.

The Fisher criterion for two classes is given by

$$J_F(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{T}} \mathbf{S}_{\mathrm{B}} \mathbf{w}}{\mathbf{w}^{\mathrm{T}} \mathbf{S}_{\mathrm{W}} \mathbf{w}} \tag{1}$$

where $\mathbf{S}_{\mathrm{B}}$ is the between-class covariance matrix and $\mathbf{S}_{\mathrm{W}}$ is the total within-class covariance matrix.

From equation 1 differentiating with respect to $\mathbf{w}$, we find that $J_F(\mathbf{w})$ is maximized when

$$\mathbf{w} \propto \mathbf{S}_{\mathrm{W}}^{-1}(\mathbf{m}_2 - \mathbf{m}_1). \tag{2}$$

On the other hand, the definition of Negentropy J_N is given by

$$J_N(\mathbf{x}) = H(\mathbf{x}_{\mathrm{Gauss}}) - H(\mathbf{x}), \tag{3}$$

where $\mathbf{x}_{\mathrm{Gauss}}$ is a Gaussian random vector of the same covariance matrix as $\mathbf{x}$ and $H(.)$ is the differential entropy. Negentropy can also be interpreted as a measure of non-Gaussianity and because Negentropy is invariant for invertible linear transformations it is obvious that finding an invertible transformation that minimizes the mutual information is roughly equivalent to finding directions in which the Negentropy is maximized [5].

The classic method of approximate Negentropy is using higher-cumulants, through the polynomial density expansions [5].

However, such cumulant-based methods sometimes provide a rather poor approximation of the entropy. There are two main reasons for this. First, finite samples estimators of higher-order cumulants are highly sensitive to outliers. Second, even if the cumulants were estimated perfectly, they mainly measure the tails of the distribution, and are largely unaffected by structure near the center of the distribution [5, 6, 3].

Now we introduce a first-order approximation of the density of maximum entropy for a continuous 1-D random variable, given a number of simple constraints. The estimated density expansion is somewhat similar to the classical polynomial density expansions by Gram-Charlier and Edgeworth. The approximation of the differential entropy is both more exact and more robust against outliers than the classical approximation based on the polynomial density expansions, without being computationally more expensive [5, 6, 3].

A special approximation is obtained if one uses two functions G^1 and G^2, which are chosen so that G^1 is *odd* and G^2 is *even*. Such a system of two functions can measure the two most important features of non-Gaussian 1-D distributions. The odd function measures the asymmetry, and the even function measures the dimension of bimodality vs. peck at zero, closely related to sub- vs. supergaussianity. Classically, these features have been measured by skewness and kurtosis, which correspond to $G^1(x) = x^3$ and $G^2(x) = x^4$.

Then the Negentropy approximation of equation 3

$$J_N(\mathbf{x}) \propto k_1 E\{G^1(\mathbf{x})\}^2 + k_2(E\{G^2(\mathbf{x})\} - E\{G^2(\upsilon)\})^2 \tag{4}$$

where υ is a Gaussian variable of zero mean and unit variance (i.e. standardized), the variable $\mathbf{x}$ is assumed to have also zero mean and unit variance and k_1 and k_2 are positive constants. We note that choosing the functions G^i that do not grow too fast, one obtains more robust estimators. See [5, 6, 3] for more details on this kind of functions. However, practical examples are the following. First, for measuring bimodality/sparsity, one might use the log-density of the double exponential (or Laplace) distribution: $G^{2a} = |\mathbf{x}|$. For computational reasons, a smoother version of G^{2a} might also be used. Another choice would be the Gaussian function, which my be considered as the

log-density of a distribution with infinitely heavy tails: $G^{2b} = \exp(-\mathbf{x}^2/2)$. For measuring asymmetry, one might use the following function: $G^1 = \mathbf{x}\exp(-\mathbf{x}^2/2)$ which corresponds to the second term in the projection pursuit index proposed in [3].

Using the above examples one obtains two practical examples:

$$J_N(\mathbf{x}) \propto k_1 E\{\mathbf{x}\exp(-\mathbf{x}^2/2)\}^2 + k_2^a(E\{|\mathbf{x}|\} - \sqrt{2/\pi})^2 \tag{5}$$

$$J_N(\mathbf{x}) \propto k_1 E\{\mathbf{x}\exp(-\mathbf{x}^2/2)\}^2 + k_2^b(E\{\exp(-\mathbf{x}^2/2)\} - \sqrt{1/2})^2 \tag{6}$$

where $k_1 = 36/(8\sqrt{3} - 9)$, $k_2^a = 1/(2 - 6/\pi)$ and $k_2^b = 24/(16\sqrt{3} - 27)$ and we remember that $H(\upsilon) = \frac{1}{2}(1 + \log(2\pi))$.

Thus we obtain approximations of Negentropy that give a very good compromise between the properties of the two classic non-Gaussianity measures given by kurtosis and skewness. They are conceptually simple, fast to compute, yet have appealing statistical properties. especially robustness [5]. We have to note that several methods to accomplish Independent Component Analysis are based on entropy information [5].

4 NEC Approach

We remark that our aim is to agglomerate by an unsupervised method the clusters (regions) that are found by a clustering approach. The information, that we call $J_{\mathbf{NEC}}$, used to merge two clusters is based both on the Fisher's discriminant and on the Negentropy:

$$J_{\mathbf{NEC}}(\mathbf{X}) = \alpha_F J_F(\mathbf{w}) + \alpha_N J_N(\mathbf{X}) \tag{7}$$

where α_F and α_N are two defined constants and $\mathbf{w}$ is the Fisher's direction.

The only *a priori* information is a dissimilarity threshold (dt in the following). We suppose to have c multi-dimensional regions X_i with $i = 1, \ldots, c$ that have been defined by the clustering approach. The NEC algorithm is described in Algorithm 1.

Algorithm 1. NEC Algorithm

```
Begin initialize dt, ĉ = c, D_i ← X_i, i = 1, ..., c
for i = 1 to ĉ-1 do
   for j = 1 to ĉ do
      calculate the Fisher's direction between D_i and D_j and project the data on it
      calculate the J_NEC information
      if J_NEC < dt then
         merge the clusters D_i and D_j and recursively all the clusters previously selected
      end if
   end for
end for
return ĉ new clusters
End
```

5 Experimental Results

In this section we show some results obtained applying the NEC approach on some synthetic data sets. We have to note that to calculate the Negentropy information we use

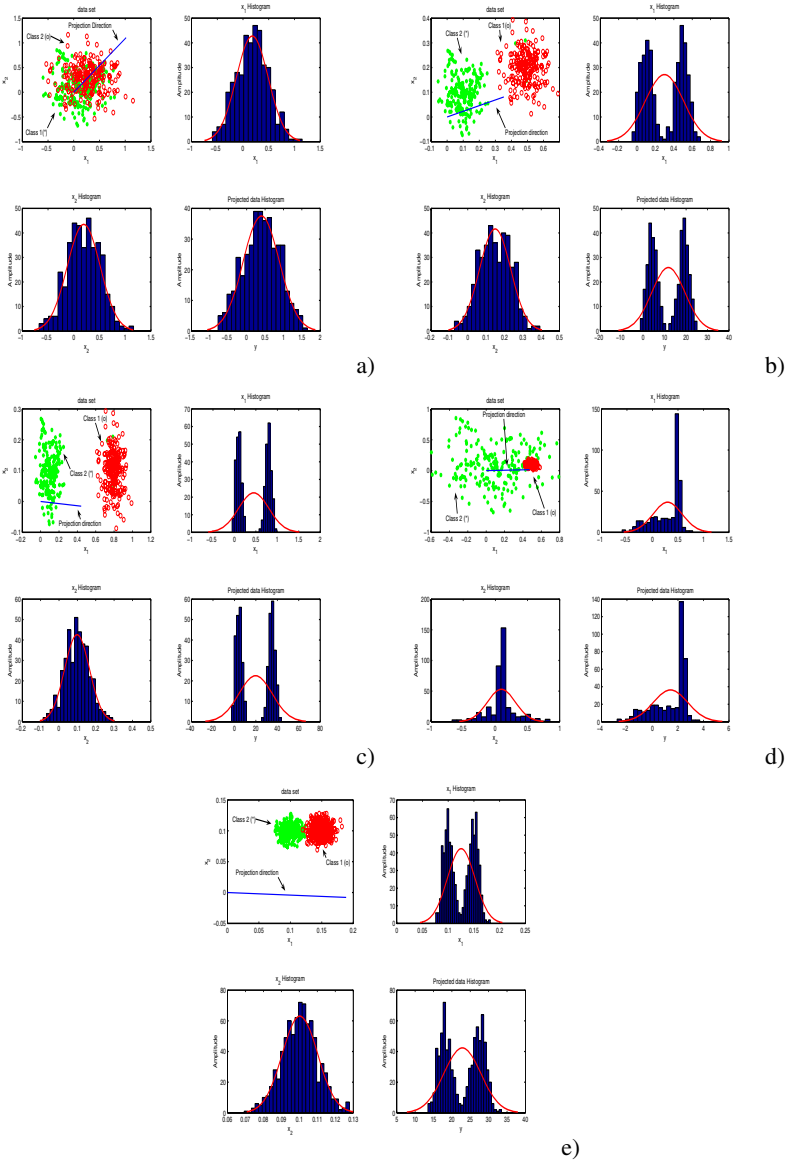


Fig. 1. $J_{\mathbf{NEC}}$ information example (from a) to e)): DATA set, histogram of the first component x_1, histogram of the second component x_2 and the histogram of the data set projected on the Fisher's direction for different data sets

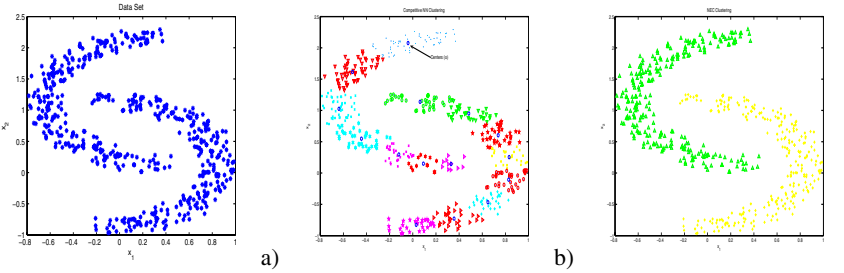


Fig. 2. NEC experimental results: a) 2 dimensional data set; b) Comparative NN clustering; c) NEC clustering

equation 7 but we stress that similar results are obtained using also the other Negentropy formulations.

At this point we remark that substantially the Fisher-Negentropy measure calculates how far the two clusters could be modeled by one single Gaussian or, in other words, if the two regions could be considered to be aligned or are part of a greater data set. This is clearer from the figure 1. In figure 1a we show four plots: a 2-dimensional data set ($\mathbf{x} = [x_1, x_2]$) composed by two close classes, the histogram of the first component x_1, the histogram of the second component x_2 and the histogram of the data set projected on the Fisher's direction. We have to note that in this case we pose $\alpha_F = .01$ and $\alpha_N = 1$. For this data set we have that $J_{\mathbf{NEC}} = 0.0079$ in which $\alpha_F J_{\mathbf{F}} = 0.0040$ and $\alpha_N J_{\mathbf{N}} = 9.73\text{x}10^{-4} + 0.0029$. In figure 1b we have a different data set where the classes are well separated and in this case we obtain that $J_{\mathbf{NEC}} = 0.4064$ in which $\alpha_F J_{\mathbf{F}} = 0.1455$ and $\alpha_N J_{\mathbf{N}} = 1.39\text{x}10^{-4} + 0.2669$. In figure 1c the two classes

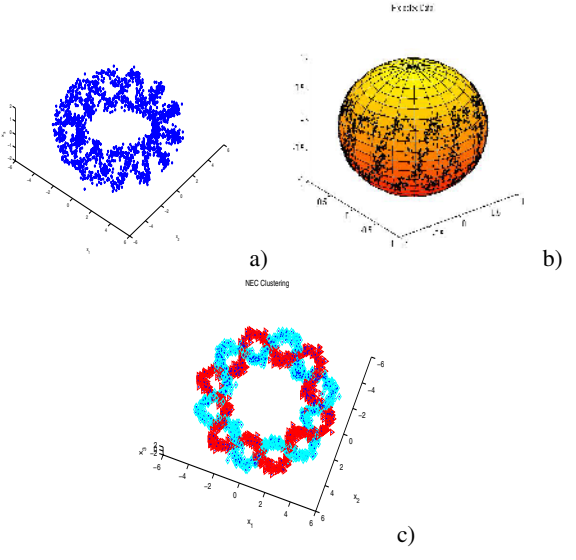


Fig. 3. NEC experimental results: a) 3-dimensional data set; b) PPS clustering; c) NEC clustering

are well separated but they have the same direction on the y axes. In this case we have $J_{\mathbf{NEC}} = 0.6794$ in which $\alpha_F J_{\mathbf{F}} = 0.3027$ and $\alpha_N J_{\mathbf{N}} = 3.49\text{x}10^{-4} + 0.3763$. Instead, in figure 1d the two classes are overlapped but could be considered separately. In this case we have $J_{\mathbf{NEC}} = 0.26603$ where $\alpha_F J_{\mathbf{F}} = 0.0188$ and $\alpha_N J_{\mathbf{N}} = 0.2196 + 0.0275$. We stress that in this case the $J_{\mathbf{NEC}}$ information is high since the asymmetry information is high. On the other hand, the $J_{\mathbf{F}}$ and bimodality/sparsity are lower than the other cases previous considered. In the last figure 1e, we plot two classes that are well defined but with a little overlapping. In this case we obtain that $J_{\mathbf{NEC}} = 0.2861$ where $\alpha_F J_{\mathbf{F}} = 0.0910$ and $\alpha_N J_{\mathbf{N}} = 3.64xx10^{-4} + 0.194876$. We note that here the bimodality/sparsity information is higher than the other information. We can conclude from this experiments that the $J_{\mathbf{NEC}}$ contains all the information that we need to agglomerate regions.

Now we apply our approach on two different data sets that are proposed in [8]. We mark that the showed examples are obtained by using $\alpha_F = 0.1$, $\alpha_N = 10$ and the threshold $dt = 3$ in all the cases. We have also to stress that in all the cases that we present and in all the experiments that we made the choice of the threshold is a very simple task since the difference between the $J_{\mathbf{NEC}}$ information that the regions could be merged and the $J_{\mathbf{NEC}}$ information that they could not be merged is high.

The first data set that we consider is composed by two 2-dimensional classes with a complex distribution (see figure 2). In figure 2a we plot the data classes. In figure 2b we show the clusters obtained using the Competitive NN and in figure 2c after the NEC approach.

The second data set is composed by two 3-dimensional classes (two springs interlaced). In figure 3a and in figure 3b we plot the data set and the clustering obtained using the PPS approach, respectively. In figure 3c we plot the resulting classes obtained applying the NEC approach.

6 Conclusions

In this paper we presented a hierarchical agglomerative clustering, i.e. a hierarchy of two unsupervised clustering algorithms, called by us NEC. The first algorithm is based on a competitive Neural Network or on a Probabilistic Principal Surfaces approach and the second one on an agglomerative clustering based on both Fisher and Negentropy information. We also presented different definitions of Negentropy information. Some of the several tests that we made have been introduced to clarify the power of the dissimilarity information and the behavior of the NEC algorithm on complex synthetic data. We mark that using different clustering approaches (i.e. K-Means, Fuzzy C-Means, etc..) we have to define the number of clusters and in particular for the cases we showed using two clusters we do not obtain the separation (see also [8]). We however note that since the NEC clustering does not use non-parametric estimation of the distribution it is very fast and very stable. In fact, in many cases the non-parametric estimation could give worse results depending on the values assigned to the parameters (parameters of the Parzen Window in [8]). In the next future the authors will focus their attention on the application of the algorithm to real data and on the study of the structure of the data by using the dissimilarity threshold.

Acknowledgment

The authors thanks the authors in [8] for their courtesy and to have shared their benchmark data.

References

1. C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University PRESS, 1995
2. K. Chang, J. Ghosh, A unified Model for Probabilistic Principal Surfaces, IEEE Transactions on pattern Analysis nd Machine Intelligence, 23, n.1
3. D. Cook, A. Buja, J. Cabrera, Projection Pursuit Indexes Based on Orthonormal Function Expansions, Journal of Computational and Graphical Statistics, 2(3): 225-250, 1993
4. R. O. Duda, P. E. Hart, D. G. Stork, Pattern Classification, John Wiley & Sons Inc., Second Edition, 2001
5. A. Hyvärinen, J. Karhunen, E. Oja, Independent Component Analysis", John Wiley & Sons, 2001
6. A. Hyvärinen, New Approximations of Differential Entropy for Independent Component Analysis and Projection Pursuit, Advences in Neural Information Processing Systems, vol. 10, pp. 273-279, MIT PRESS, 1998
7. R. Jenssen, T. Eltoft, J. C. Principe, Information Theoretic Clustering: A Unifying Review of Three Recent Algorithms, Proceedings of the 6th Nordic Signal Processing Symposium - NORSIG 2004, June 9-11, 2004 Espoo, Finland
8. A. de M. Martins, A.D.D. Neto, J.D. de Melo, J.A.F. Costa, Clustering Using Neural Networks and Kullback-leibler Divergency, International Joint Conference on Neural Networks - IJCNN - 2004, pp. 01-06
9. A. Staiano, L. De Vinco, A. Ciaramella, G. Raiconi, R. Tagliaferri, G. Longo, G. Miele, R. Amato, C. Del Mondo, C. Donalek, G. Mangano, D. Di Bernardo, Probabilistic Principal Surfaces for Yeast Gene Microarray Data-Mining,ICDM'04 - Fourth IEEE International Conference on Data Mining, Brighton (UK), pp. 202-209, 2004

A Recurrent ICA Approach to a Novel BSS Convolutive Nonlinear Problem

Daniele Vigliano, Raffaele Parisi, and Aurelio Uncini

Dipartimento INFOCOM, Università di Roma "La Sapienza" – Italy Via Eudossiana, 18, 00184 Roma – Italy
daniele.vigliano@poste.it, parisi@infocom.uniroma1.it, aurel@ieee.org

Abstract. This paper introduces a Recurrent Flexible ICA approach to a novel blind sources separation problem in convolutive nonlinear environment. The proposed algorithm performs the separation after the convolutive mixing of post nonlinear convolutive mixtures. The recurrent neural network produces the separation by minimizing the output mutual information. Experimental results are described to show the effectiveness of the proposed technique.

Keywords: Blind Source Separation, Flexible ICA, Spline Adaptive function, Mutual Information, Recurrent networks, IIR filters.

1 Introduction

The first studies about Independent Component Analysis aimed at resolving the famous cocktail party problem first in instantaneous, then in reverberant environments. A critical issue is that linear mixing models are too unrealistic and "poor" in a lot of real situations; recently it starts to grow the interest in non linear convolutive separation.

Important theoretical results in nonlinear instantaneous ICA framework are in [Hyvarinen et al., 1999]. Several papers explore Post Nonlinear mixing problem (PNL) in instantaneous [Taleb, 2002] and in convolutive environment [Milani et al., 2002][Zade et al., 2002] but only few of them (see [Taleb et al., 1999][Hyvarinen et al., 1999]) deal with the issue of existence and uniqueness of the solution. Recent advances in BSS of nonlinear mixing models have been reviewed in [Jutten et al., 2003]. If there are particular request of performance, separation quality or strictness of the mixing environment, became important the pdf matching of the signals. The so called Flexible ICA algorithm performs the adaptive estimation of parameters bound to the pdf of signals, this lead to a better pdf matching; it improves the quality of separation and allows a faster learning.

Actually recent studies try to improve the severity of mixing models moving from single-block nonlinear structures (convolutive or at least instantaneous) to multi-block structures. In [Solazzi et al., 2004] sources are recovered from a PNL mixing followed by an instantaneous mixing (so called PNL-L mixture); in [Vigliano et al., 2005] the mixing environment is composed by a PNL mixing block followed by a convolutive one (PNL-C mixture) moreover the issue of existence and uniqueness of

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 57 – 64, 2006.

the solution has been explored. Other improvements of the severity of mixing environment have been presented in [Vigliano et al., 2004] which explores how behave a FIR network in separating sources from the convolutive mixing of a convolutive post-nonlinear mixture (CPNL-C mixture).

Recent works start performing the separation using multilayer neural networks (see [Woo et al., 2004] for details).

This paper explores the performance of a full recurrent network in separating sources from the CPNL-C mixing environment (already introduced in [Vigliano et al., WIRN2004]); CPNL-C at this moment is the most general convolutive nonlinear environment in literature.

2 Separation in Nonlinear Convolutive Environment

This section introduces BSS problem in nonlinear environment. Be $\mathbf{x}[n]$ the N vector of mixed accessible signals and $\mathbf{s}[n]$ the vector of hidden independent sources The expression of the hidden mixing model in closed form is: $\mathbf{x}[n] = \mathcal{F}\{\mathbf{s}[n],...,\mathbf{s}[n-L]\}$, in which $\mathcal{F}\{\cdot\}$ is a convolutive nonlinear distorting function. The solution of the BSS problem can be expressed as: $\mathbf{y}[n] = \mathcal{H}\{\mathbf{s}[n]\} = \mathcal{G} \circ \mathcal{F}\{\mathbf{s}[n]\}$. In instantaneous environment ICA recovers the original sources up to some trivial acceptable non-uniqueness: outputs can be scaled and delayed version of flipped inputs.

In the more general convolutive nonlinear case, the issue of separating mixtures with the only constraint of output independence and no other a priori assumption is affected by a strong non uniqueness [Jutten et al., 2003], [Vigliano et al., 2005]. Given independent inputs, several well known examples show the existence of maps that can produce independent outputs even with non diagonal Jacobian matrix. Independence constraint alone is not strong enough to recover original sources from generic nonlinear mixing environments [Taleb, 2002].

The most important issue for generic nonlinear problems is to ensure the presence of conditions granting, at least theoretically, the possibility to achieve the desired solution. In [Hyvarinen et al., 1999] authors proposed a constructive way (a Gram-Schmidt like method) to obtain solutions of the separation problem in an instantaneous nonlinear mixing environment; in order to grant the uniqueness of the solutions some constraints have been applied. In [Vigliano et al., 2005] authors introduced a theoretical proof of existence and uniqueness of the solution in a convolutive nonlinear environment stricter than the Post Nonlinear one: the PNL-C; in that paper a general idea has been applied: adding "soft" constraints (a priori assumptions) to the problem can produce the uniqueness of the solution. Following such consideration, in this paper too, the a priori knowledge about the mixing model is exploited to design the recovery network: the so called "mirror" demixing model is used. Convolutive mixing environments add to the solution one more strong non-uniqueness: the filtering indeterminacies. Convolutive mixtures are separable but applying channel-by-channel filters to the independent recovered signals, outputs are still independent.

This indeterminacy may be unacceptable since it can strongly distort the sources. In any case after separation it is possible to equalize the outputs in order to obtain better results. According to these reasons filtering indeterminacy will no more considered in the rest of this paper.

3 The Mixing-Demixing Architecture

This section explores the recovery of separated sources from nonlinear convolutive mixing; the a priori knowledge about the mixing model has been used to design the recovering network. The mixing environment is represented in figure 1.

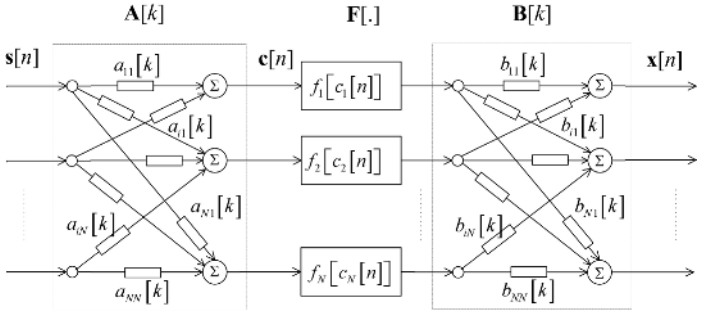


Fig. 1. The block diagram of the convolutive nonlinear mixing model: the CPNL-C model

In which $\mathbf{A}[k]$ and $\mathbf{B}[k]$ are $N \times N$ FIR matrices with respectively L_a and L_b filter taps and $\mathbf{F}\left[\mathbf{c}[n]\right] = \left[f_1\left[c_1[n]\right], \quad f_N\left[c_N[n]\right]\right]^T$ is the $N \times 1$ vector of nonlinear distorting functions. The closed form for mixing model is:

$$\mathbf{x}[n] = \mathcal{F}[\mathbf{s}] = \mathbf{B}[n] * \mathbf{F}\left[\mathbf{A}[n] * \mathbf{s}[n]\right] \tag{1}$$

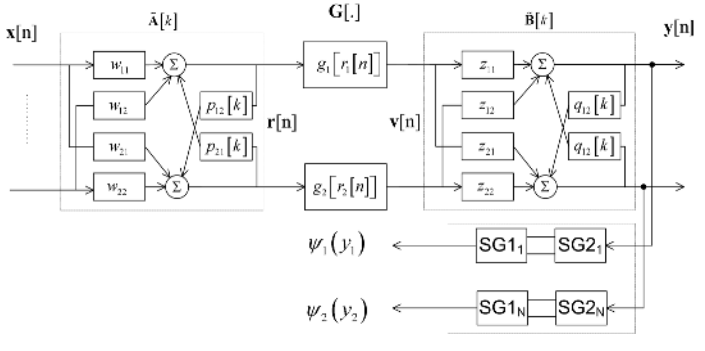


Fig. 2. Recurrent network used for the nonlinear blind deconvolution and separation

it is the so called CPNL-C mixing environment: convolutive mixing of a convolutive post-nonlinear mixture. Most of mixing environment already used in literature can be rewritten as particular case of the CPNL-C model. In order to grant the uniqueness of the solution, the recovering structure mirrors the mixing model but here convolutive blocks are realized by IIR architectures. The application of MIMO recurrent networks to the cocktail party problem in convolutive environment has been already exposed in several papers. The architecture presented in one of them [Choi et al., 1997] has been here adapted to this context and exploited to design $\tilde{\mathbf{A}}$, $\tilde{\mathbf{B}}$. Figure 2 shows the recovering structure, with N=2 for sake of simplicity.

The expression of the i-th output channel is:

$$\begin{aligned} y_j[n] &= \sum_{h=1}^{N} z_{jh} v_h[n] + \sum_{\substack{h=1 \\ h\neq j}}^{N} \sum_{k=1}^{L_q} q_{jh}[k] v_h[n-k] \\ v_h[n] &= g_h\left[r_h[n]\right] \\ r_h[n] &= \sum_{h=1}^{N} w_{jh} x_h[n] + \sum_{\substack{h=1 \\ h\neq j}}^{N} \sum_{k=1}^{L_p} p_{jh}[k] x_h[n-k] \end{aligned} \tag{2}$$

In which $\mathbf{G}[.]$ is the $N\times 1$ vector of nonlinear compensating functions, one for each channel, $\mathbf{W}$ and $\mathbf{Z}$ are $N\times N$ matrices, $\mathbf{Q}[k]$ and $\mathbf{P}[k]$ are FIR filter with L_p and L_q filter taps; the networks used in this paper have $q_{ii}[k]\equiv 0,\ p_{jj}[k]\equiv 0\ \ \forall i,j,k$.

Introducing the knowledge about the particular kind of mixing model is the key to avoid the strict non uniqueness of the solution; such assumption limits the weakness of the output independence condition reducing the cardinality of all possible independent output solutions; with this constraint the problem of recovery the original sources is not ill posed any more.

4 Blind Demixing Algorithm and Network Model

This section explores the blind demixing algorithm, the adaptive recurrent network and the network that performs the nonlinear function estimation. The blind algorithm performs an adaptive learning of the network parameters $\mathbf{\Phi}$ on the base of the output independence estimation.

The learning is realized minimizing the Mutual Information $I\{\mathbf{\Phi},\mathbf{y}\}$ between outputs, with a steepest descent algorithm: $\mathbf{\Phi}(k+1)=\mathbf{\Phi}(k)-\eta_{\mathbf{\Phi}}\left[\partial I\{\mathbf{\Phi},\mathbf{y}\}/\partial\mathbf{\Phi}\right]$. The choice of a gradient based minimization procedure lead to terms like:

$$\frac{\partial}{\partial\mathbf{\Phi}}\log\left[p_{y_i}(y_i)\right]=\frac{\partial p_{y_i}(y_i)/\partial y_i}{p_{y_i}(y_i)}\frac{\partial y_i}{\partial\mathbf{\Phi}}=\psi_i(y_i)\frac{\partial y_i}{\partial\mathbf{\Phi}} \tag{3}$$

in which $\psi_i(y_i)$ are the so called Score Functions (SF). In this paper, Spline Neurons are used to perform the on-line estimation of both Score Functions and

nonlinear compensating functions (for a detail about Spline Neurons see [Solazzi et al., 2004][Uncini et al., 1998]). The most attractive property of Spline Neurons, as function estimators, is the local learning: for each learning step only the four control points nearest to the training input sample are considered; no matter how many control points the Spline curve has.

For direct estimation of SF has been performed a MSE approach (see [Taleb et al., 1999] for details) but learning rules result still blind:

$$\frac{\partial \mathcal{E}}{\partial \mathbf{Q}_i^{\psi_j}} = \left[\tfrac{1}{4} \mathbf{T}_u \mathbf{M} \mathbf{T}_u \mathbf{M} \mathbf{Q}_i^{\psi_j} + \tfrac{1}{\Delta} \dot{\mathbf{T}}_u \mathbf{M} \right] \tag{4}$$

in which $\mathbf{M}$ is a matrix of coefficients, $\mathbf{T}$ is the vector local abscissa and Δ is the distance between the abscissas of adjacent control points.

The network used to perform the separation is a cascade of blocks well described in literature and previously used to resolve more simple problems. Deriving the cost function $I\{\mathbf{\Phi}, \mathbf{y}\}$ with respect to the learning parameter $\mathbf{\Phi}$ results:

$$\begin{aligned} &\frac{\partial I\{\mathbf{\Phi}, \mathbf{y}[n]\}}{\partial \mathbf{\Phi}} \simeq \frac{\partial \Im\{\mathbf{\Phi}, \mathbf{y}[n]\}}{\partial \mathbf{\Phi}} = \\ &= -\frac{\partial}{\partial \mathbf{\Phi}} \sum_{h=0}^{M} \left[\begin{array}{l} \log|\det \mathbf{Z}| + \log|\det \mathbf{W}| + \sum_{i=1}^{N} \log\left(g_i'\left[v_i[n-h] \right] \right) \\ + \sum_{i=1}^{N} \log p_{y_i}\left(y_i[n-h] \right) \end{array} \right] \end{aligned} \tag{5}$$

In (5) the expected value has been replaced by the instantaneous value. The learning rules for each parameter of the set $\mathbf{\Phi} = \left\{ z_{ij}, w_{ts}, p_{ml}[k], q_{nr}[h], \mathbf{Q}^g, \mathbf{Q}^{\Psi} \right\}$ are:

$$\partial \Im / \partial \mathbf{Z} = -\mathbf{Z}^{-T} - \mathbf{\Psi}_y^{\,T} \mathbf{v}[n] \tag{6}$$

$$\partial \Im / \partial \mathbf{Q}_i^{g_j} = -\left[\dot{\mathbf{T}}_u \mathbf{M} / \dot{\mathbf{T}}_u \mathbf{M} \mathbf{Q}_i^{g_j} + \tfrac{1}{2} \mathbf{\Psi}_y(\mathbf{Z})_j \mathbf{T}_u \mathbf{M} \right] \tag{7}$$

$$\begin{aligned} \partial \Im / \partial \mathbf{W} = &-\mathbf{W}^{-T} - \left[\ddot{g}_1(r_1) / \dot{g}_1(r_1) \cdots \quad \ddot{g}_N(r_N) / \dot{g}_N(r_N) \right]^T \mathbf{x}[n] + \\ &-diag\left[\mathbf{Z}^{\mathbf{T}} \mathbf{\Psi}^{\mathbf{y}} \right] \left[\dot{g}_1(r_1) \cdots \dot{g}_N(r_N) \right]^T \mathbf{x}[n] \end{aligned} \tag{8}$$

$$\partial \Im / \partial q_{ij, i \neq j}[k] = -\Psi_i^{\,y} y_j[n-k] \tag{9}$$

$$\begin{aligned} \partial \Im / \partial p_{ij}[k] = &-\left[\ddot{g}[n]_1(r_1) / \dot{g}_1(r_1) \cdots \quad \ddot{g}_N(r_N) / \dot{g}_N(r_N) \right]^T \mathbf{r}[n-k] + \\ &-diag\left[\mathbf{Z}^{\mathbf{T}} \mathbf{\Psi}^{\mathbf{y}} \right] \dot{\mathbf{v}}^T[n] \mathbf{r}[n-k] \end{aligned} \tag{10}$$

in which $\mathbf{M}$ and $\mathbf{T}$ have the same sense as in (4) and the operator *diag*[$\mathbf{r}$] transforms the vector $\mathbf{r}$ in a diagonal squared matrix.

Recurrent networks have been used because they allow more compressed representation of models. If compared with a FIR architecture, the IIR network here proposed requires a reduced number of weights and then allows faster and more accurate learning.

5 Experimental Results

This section collects separation results of two mixtures obtained applying the same mixing environment to different sets of independent sources. Although the algorithm should be able to perform the separation of *N*-channel mixtures, for proper visualization of results each set of sources is composed only by a couple of signals. The first set is composed by white signals (a gaussian noise and a uniform distributed signal), the second one by correlated signals (a male and a female voice speaking respectively "*Le donne i cavalier l'arme*" and "*Riperdo una seconda volta quegli esigui beni*").

Figure 3 a-c) show the joint pdf of mixed signals and figure; figure 3 b-d) show the joint pdf of separated signals resulting after training, one note the typical plot of the joint pdf of separated signals. The demixing network has Spline Neurons (**g** for distortion compensation and **Ψ** for Score Function estimation) with 53 control points and filters **P**[*k*] and **Q**[*k*] with 5 taps.

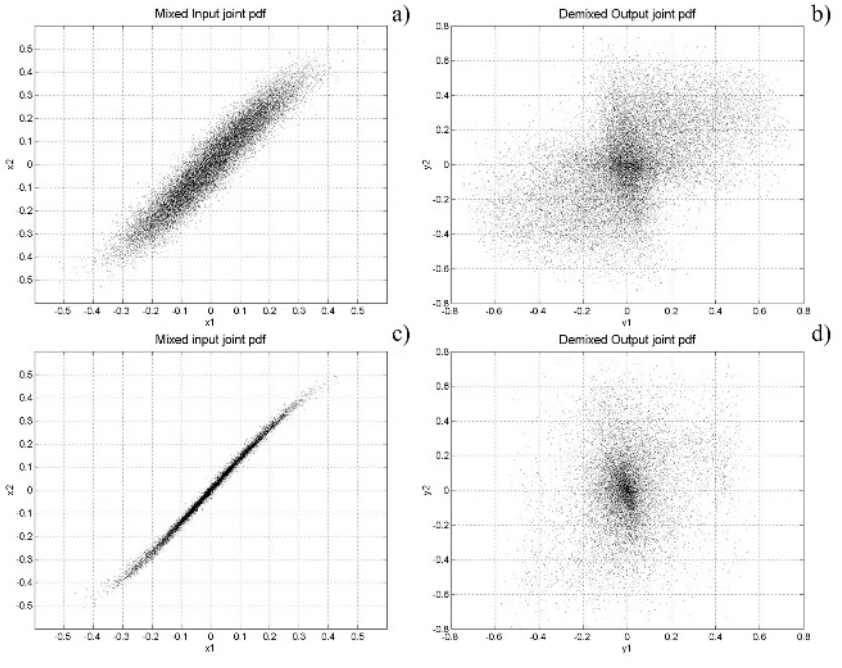


Fig. 3. a-b) white signals: joint pdf of input mixture and joint pdf independent output; voices c-d) joint pdf of input mixture and joint pdf of independent output

The applied nonlinear distortions for both input signals are: $\mathbf{F}\left[f_1(p_1), f_2(p_2)\right] = \left[p_1 + 0.5p_1^3, 0.5p_2 + 0.8 * \tanh(5p_2)\right]$ (note that input signals are normalized in order to spread the nonlinear range of these functions). Considering the notation of figure 1, the FIR matrices of the mixing environment are: $\mathbf{A} = \begin{bmatrix} 0.8 + 0.4z^{-1} + 0.2z^{-2} & 0.6 + 0.3z^{-1} + 0.1z^{-2} \\ 0.5 + 0.3z^{-1} - 0.1z^{-2} & 0.8 - 0.4z^{-1} + 0.2z^{-2} \end{bmatrix}$, and $\mathbf{B} = \begin{bmatrix} 0.8 - 0.3z^{-1} + 0.06z^{-2} & 0.3 + 0.2z^{-1} - 0.06z^{-2} \\ -0.3 + 0.3z^{-1} + 0.11z^{-2} & 0.7 + 0.3z^{-1} - 0.1z^{-2} \end{bmatrix}$.

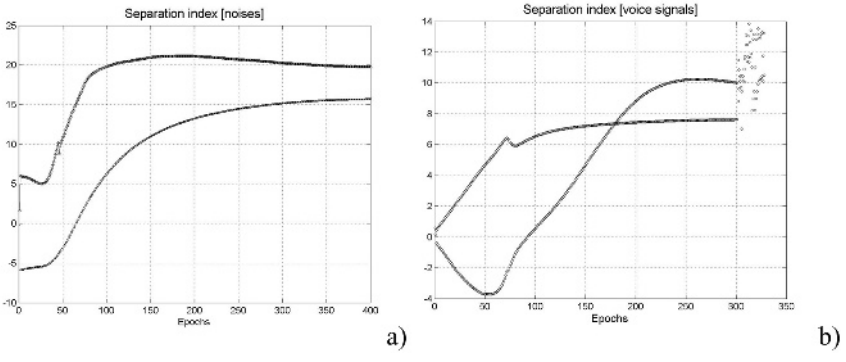


Fig. 4. Separation index during the training; a) separation of white signals, b) separation of vocal signals

The so called "Separation Index" S_j (dB) introduced in [Shobben et al., 1999] gives a measure of how much channel *j*-th is separated from the others; here the Separation Index is evaluated for each channel.

$$S_j = 10\log\left[E\left\{\left(y_{\sigma(j),j}\right)^2\right\} \Big/ E\left\{\sum_{k\neq j}\left(y_{\sigma(j),k}\right)^2\right\}\right] \tag{11}$$

In (11), $y_{i,j}$ is the *i*-th output signal when only the *j*-th input signal is present while $\sigma(j)$ is the output channel corresponding to the *j*-input. The trend of this index (Figure 4, a-b) confirms the growing of separation during the training for both tests. For each of two tests, Fig. 3 and Fig. 4 together show how the algorithm is successful in performing the separation of the output signals. In [Vigliano et al., 2004] a similar mixing environment was approached with a FIR-based architecture; the separation performances obtained with the recurrent algorithm here exploited result improved moreover while the FIR matrix required 15-tap filters, the recurrent structure required 5-tap filters. This leads to a significant gain in terms of computational effort. Even if the separation well behaves in demixing voices, it reaches the best results with white signals; the reason of this behaviour lie in the construction of cost function (5).

6 Conclusion

This paper explores a novel recurrent ICA approach to the BSS problem in the CPNL-C mixing environment. The use of mirror model confirms the existence and the uniqueness of the solution to the CPNL-C separation problem. Results assure good separation and good nonlinear compensation. The use of the recurrent architecture allows a more compact representation of the recovering model and grants a more accurate demixing.

The recovering network performs the on line estimation of both nonlinear compensating functions and Score Functions by Spline Neurons leading to a better matching and producing a more accurate learning.

References

Jutten, C., Karhunen, J., (2003), "Advances in Nonlinear Blind Sources Separation", 4th International Symposium on ICA and BSS (ICA2003), April 2003, Nara, Japan.

Taleb, A., (2002), "A Generic Framework for Blind Sources Separation in Structured Nonlinear Models", In IEEE Trans. on signal processing, vol. 50. no 8 August 2002.

Taleb, A., Jutten, C., (1999), "Sources Separation in post nonlinear mixtures", In IEEE Trans. on signal processing, vol. 47. no 10 August 1999.

Hyvarinen, A., Pajunen, P., (1999), "Nonlinear Independent Component Analysis: Existence and Uniqueness Results", Neural Networks 12(2): 429-439, 1999.

Solazzi, M., Uncini, A., (2004), "Spline Neural Networks for Blind Separation of Post-Nonlinear-Linear Mixtures", In IEEE Trans. on Circuits and Systems I Fundamental Theory and Applications, Vol. 51 , No. 4, pp 817 – 829, April 2004.

Uncini, A., Vecci, L., Piazza, F., (1998), "Learning and approximation capabilities of adaptive Spline activation function neural network", In NN, Vol. 11, no. 2, pag. 259-270 March 1998.

Milani, F., Solazzi, M., Uncini, A., (2002), "Blind Source Separation of convolutive nonlinear mixtures by flexible spline nonlinear functions", Proc. of IEEE ICASSP'02, Orlando, USA, May, 2002.

Zade, M. B., Jutten, C., Najeby, K., (2001), "Blind Separating, Convolutive Post nonlinear Mixture", ICA 2001 In Proc. of the 3rd Workshop on Independent Component Analysis and Signal Separation (ICA2001), San Diego (California, USA), 2001, pp. 138–143.

Shobben, D., Torkkola, K., Smaragdis, P., (1999), "Evaluation of blind signal separation methods", In Proc. of ICA and BSS, Aussois, France, January 11-15, 1999.

Vigliano D., Parisi R. and Uncini A., (2004), "A flexible approach to a novel BSS convolutive nonlinear problem: preliminary result", Proc. of "Italian Workshop on Neural Networks (WIRN04)", Perugia, Springer-Verlag Ed., Sept 2004.

Vigliano, D., Parisi R., Uncini, A., (2005), An Information Theoretic Approach to a Novel Nonlinear Independent Component Analysis Paradigm, In Press On Elsevier Signal Processing Special Issue on Information Theoretic (2005).

Choi, S., Cichocki A., (1997), "Adaptive Blind Separation of speech signals: Cocktail party problem", ICSP97, Seoul, Korea, 26-28, pp. 617-622, August 1997.

Woo W.L., Khor L.C (2004), "Blind restoration of nonlinearly mixed signals using multilayer polynomial neural network", Vision, Image and Signal Processing, IEE Proceedings- ,Volume: 151 , Issue: 1 , 5 Feb. 2004.

Hourly Forecasting of SO_2 Pollutant Concentration Using an Elman Neural Network

U. Brunelli[1], V. Piazza[1], L. Pignato[1], F. Sorbello[2], and S. Vitabile[3]

[1] Dipartimento di Ricerche Energetiche ed Ambientali, Università di Palermo, Viale delle Scienze, Edificio 9, 90128, Palermo, Italy
[2] Dipartimento Ingegneria Informatica, Università di Palermo, Viale delle Scienze, Edificio 6, 90128, Palermo, Italy
[3] Istituto di CAlcolo e Reti ad alte prestazioni, Italian National Research Council, Viale delle Scienze, Edificio 11, 90128, Palermo, Italy

Abstract. In this paper the first results produced by an Elman neural network for hourly SO_2 ground concentration forecasting are presented. Time series has been recorded between 1998 and 2001 and are referred to a monitoring station of SO_2 in the industrial site of Priolo, Syracuse, Italy. Data has been kindly provided by CIPA (Consorzio Industriale per la Protezione dell'Ambiente, Siracusa, Italia). Time series parameters are the horizontal and vertical wind velocity, the wind direction, the stability classes of Thomas, the base level of the layer of the atmospheric stability, the gradient of the potential temperature and the difference of the potential temperature of reference.

1 Introduction

Air pollution is a meteorological phenomenon men created for maximizing the profit without attention to the health of nature. The increased sensibility of the populations to environmental problems has obliged the state administrations to emit laws that govern as the pollutant emission of the industries as the maximum values of pollutant's concentrations over the ground.

Obviously the control of exhaust gases emission is primary in order to keep air pollution below the specified limits of the laws in force. For an adequate health warning system accurate forecasts of pollutant concentrations function of time and of location are necessary. Models for forecasting at short lead time (1 - 2 hours) are adequate to plan, in sites with a high concentrations of industries, a health warning system in reducing pollutants concentration using cleaner fuels and arriving to shut-down the production of some industry. Such a models can be divided in deterministic, probabilistic or stochastic and employing neural networks. The impossibility to use, in areas with high industrial concentration and tormented orography, deterministic models is due to the uncertainty over the emission rate of pollutants and the intrinsic structure of mathematical representation (advection-diffusion equation).

The probabilistic approach allows studying the future behavior of a time dependent phenomenon, in which the influence of considered factors is not possible to get in deterministic mode. The statistical methodology is concerned with models in which the observations are assumed to vary independently. The stochastic methods, used for analysis of data time dependent series, are well described by Box and Jenkins [1] and

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 65 – 69, 2006.

were used by many researchers [2], [3] forecasting future values of a time series from current and past values. This methods, certainly better than deterministic approach, are not adequate in environmental field to predict the values of concentration when occur a great increment due to thermal inversions. In this case, in which the methods above delineated are not plentiful to give good results, the use of a neural network can be un-hoped solutions. Neural networks [4] are able to treat incomplete and uncertain information, as human brain, and, after the training phase, they are able to find the expected solutions for a datum classification task. Conventional ANNs are widely used for static problems where outputs depend only on the current inputs. Memory of conventional ANNs, called long-term memory, is represented by network connection weights (or any other trainable parameters) that are updated after each training epoch. Once ANNs have been successfully trained, the long-term memory remains fixed during the operation of the network. In forecasting tasks, neural networks have to deal with input and output patterns that vary across time. Using the taxonomy proposed in [6], the Space-Temporal Connectionist Networks (STCNs) also include a short-term memory that allows these networks to deal better with time series. Conventional connectionist networks compute the activation values of all nodes at time *t* based only on the input at time *t*. By contrast, in STCNs the activations of some nodes at time *t* are computed based on activations at time *t-1*, or earlier. These activations serve as a short-term memory.

In this paper a forecaster based on Elman Neural networks trained with RPROP algorithm for hourly SO_2 concentration prediction is proposed. The forecasting is performed one hour ahead. Horizontal and vertical wind velocity, the wind direction, the stability classes of Thomas, the base level of the layer of the atmospheric stability, the gradient of the potential temperature and the difference of the potential temperature of reference composed the time series. The time series was recorded from Apr 1st, 1998 to Dec 31st 2001 and are referred to a monitoring station of SO_2 with a lead-time of an hour. These data are kindly provided by CIPA (Consorzio Industriale per la Protezione dell'Ambiente, Siracusa, Italia).

2 Data Gathering and Pre-processing

The region of Melilli's coastal strip is occupied by the industrial settlements extending from South to North. The region is limited by land not only on the Eastern side, but also in the Northern side (Capo Izzo with the small town of Augusta). More important, there are relatively high elevations at W-SW (Climiti Mountains, about 400-500 m high) witch probably breezes to generate vertical patterns. As it will be seen, they influence the classical partition of the wind directions near the ground, causing an apparent reduction of the frequency of the breezes coming from the sea when compared to those coming from the land. In the natural cavity surrounding the bay there are scattered villages, roads and single houses at different heights above the sea. The pollution sources on the coast can be roughly divided in area and point sources. The first ones cover the whole costal strip with high degree of continuity, while the second ones consist mainly in stacks of relevant heights (frequently comprised between 100 m and 200 m). This is usually considered a good design, because the maximum ground concentration on a flat plane is inversely proportional to the square of the source height. The CIPA (Consorzio Industriale per la Protezione

dell'Ambiente, Siracusa, Italia) is located on a slightly elevated spot (30m above sea) about 1 Km inland. The facilities of interest for the paper are:

- A RASS (Radio Acoustic Sounding System)
- A pole of 10m supporting an anemometer located at 10m and two thermometers at 2 and 10m. Both thermometers are shielded to avoid ground radiation and provide an accurate estimation of temperature gradient at ground level.
- A SODAR (Sound Detection And Ranging)
- 11 fixed stations for measuring at ground level concentrations of pollutants such as NHC, CH_4, O_3, H_2S, NO, NO_2.

Traditional models cannot be used for SO_2 forecasting due wind characteristics in the region of interest. During the year 2000 has been collected the data shown in Table 1. They include wind intensity and direction at 10m height, temperature and humidity at 2 m. To speed up the evaluation all data are grouped in 8 classes whose upper and lower limits are collected in Table 1.

Table 1. Class partition used in the data evaluation

Class	Wind speed (m/s)	Wind direction	Temperature (°C)	Humidity (%)
1	<0,5	N-NE	<2.50	<5
2	0,50-1,49	NE-E	2,50-7,49	5-14,9
3	1,50-2,49	E-SE	7,50-12,49	15-14,9
4	2,50-3,49	SE-S	12,50-17,49	25-34,9
5	3,5-4,49	S-SW	17,50-22,49	35-44,9
6	4,50-5,49	SW-W	22,50-27,49	45-54,9
7	5,50-6,49	W-NW	27,50-32,49	55-64,9
8	>6,5	NW-N	>35,5	>65

One of the main components in the success of some neural network solution is the data pre-processing. Each neural network's input pattern is composed by seven (hourly) values: the horizontal and vertical wind velocity, the wind direction, the stability classes of Thomas, the base level of the layer of the atmospheric stability, the gradient of the potential temperature and the difference of the potential temperature of reference. In the experimental trials were used the above data at the instants (t - i), with i = 2, 1, 0. The prediction was performed one hour ahead. Each value was normalized in the range [-1, 1] using the following linear transformation:

$$X' = \frac{X - V_m}{V_{max} - V_{min}} \tag{1}$$

where X' is the new normalized value, X is the old value, V_{max} is the maximum of the considered data set, V_{min} is the minimum of the considered data set and V_m is the average value of the considered data set.

3 Experimental Results

The primary aim of developing an ANN is to generalize the features of the processed time series. A popular technique to achieve generalization, avoiding over fitting, is the early stopping method presented in [7]. According to this method, the generated data set was divided into three subsets: a training set, a validation set, and a test set. After the training set, the validation set was used to tune the parameters of the network (architecture, epochs, etc). The whole training phase was stopped when the lowest error on the validation set was reached. In our experimental trials, the training set was composed by about 18,000 values, the validation set by 400 values and the test set by 168 values. Several Elman network topologies were implemented and tested changing the number of layers, the number of the hidden and context units. Neural model simulations were performed using the Stuttgart Neural Network Simulator [8]. The adopted training algorithm was the Resilient-BackPROPagation (RPROP) algorithm. The parameters of the learning algorithm (see eq. 1 and eq. 2) were: η^+=1.2, η^-=0.5, Δ_0=0.1 (fixed starting value for Δ_{ji}), Δ_{max}=50 (the upper limit for Δ_{ji}). The connection weights were initialised to zero-mean random values with adequate upper and lower bounds of (-1, 1). Training epochs were set to 500 and networks performance were evaluated through the Mean Absolute Error (MAE):

$$MAE = \frac{\sum_{i=1}^{N} |O_i - P_i|}{N} \qquad r = 1 - \frac{\sum (O_i - P_i)^2}{\sum (P_i - P_m)^2} \qquad (2)$$

where O_i is the observed value at time i, P_i is the predicted value at time i, P_m is the medium value of the observed values, and N is the total number of observations.

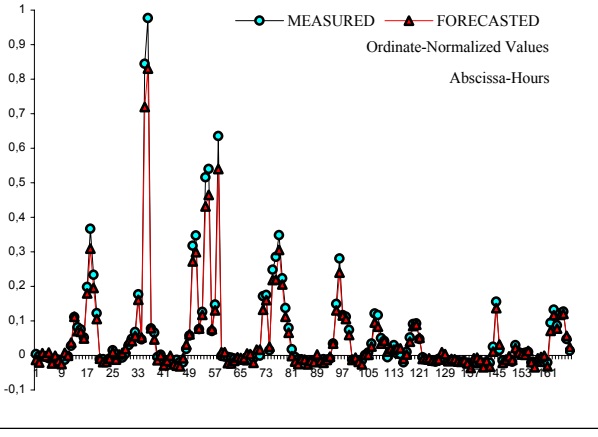


Fig. 1. Comparison between the measured and forecasted SO_2 pollutant values

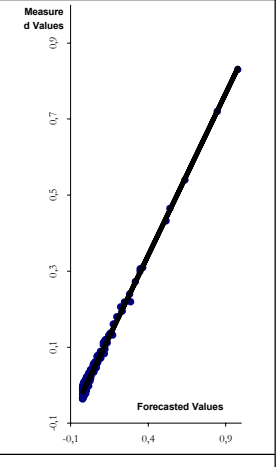


Fig. 2. The dispersion diagram

The Elman Neural Network architecture with the topology 21-30-1 (30 hidden units and 30 hidden context units, 1 output unit and 1 output context unit) with the Resilient-BackPROPagation training algorithm has shown the best forecasting accuracy with MAE=0,014424 and r = 0.999. The sigmoid function was used as activation function for the hidden layer, whilst the *RM* function was used as activation function for the output layer. The *RM* function expression is:

$$a_j(t) = \begin{cases} 0.85 * a_j(t-1) + 0.15 * net_j(t) * (1 - a_j(t-1)) & \text{if } net_j(t) > 0 \\ 0.85 * a_j(t-1) + 0.15 * net_j(t) * (1 + a_j(t-1)) & \text{if } net_j(t) \le 0 \\ \text{where } net_j(t) = \sum_i w_{ij} o_i \end{cases} \quad (3)$$

Elman NNs are capacitated to internally encode temporal contexts from their feed-back connections. They evolve as a sequential system and, consequently, can describe a dynamical system evolution in a more efficient way. In Figures 1, 2 the measured and forecasted SO_2 pollutant values are reported.

4 Conclusions

In this paper, a forecaster based on an Elman Neural Network trained with Resilient BackPROPagation algorithm has been presented. The prediction task was related to the industrial site of Priolo, Syracuse, Italy for hourly ground concentration of SO_2 pollutant prediction. The goodness of the model is showed in the figures 1 and 2 and confirmed from values of MAE and Correlation Linear Coefficient.

References

1. Box G E. P. and Jenkins G. M. Time Series Analysis: Forecasting and Control. Holden-Day. S. Francisco. ISBN 0-8162-1104-3
2. McCollister G.M. and Wilson K. R. Linear stochastic models for forecasting daily maxima and hourly concentrations of air pollutants. Atmospheric Environment Vol. 9 page416-423 1974
3. Finzi G., Zannetti P., Fronza G. and Rinaldi S. Real time prediction of SO_2 concentration in the Venetian lagoon area Atmospheric Environment Vol. 13 page1249-1255 1978
4. Elman, J. L. (1990), "Finding structure in time". In Cognitive Science, n.14 pp. 179-211.
5. Alireza Khotanzad, Malcon H. Davis, Alireza Khotanzad, Malcom H. Davis, Alireza Abaye (1996) "An Artificial Neural Network Hourly Temperature Forecaster With Applications in Load Forecasting" IEEE Transaction on Power System, Vol. 11 N 2 May 1996
6. S.C. Kremer, Space-temporal connectionist networks: A taxonomy and review," Neural Computation, vol. 13, no. 2, pp. 249-306, 2001.
7. W.S. Sarle, Stopped training and other remedies for over fitting, Proc. of the 27th Symp. On the Interface of Computing Science and Statistic, 352-360, (1995).
8. SNNS - Stuttgart Neural Network Simulator, url: http://www-ra.informatik.uni-tuebingen.de/SNNS/.

Nonlinear Exploratory Data Analysis Applied to Seismic Signals

Antonietta M. Esposito[1,2], Silvia Scarpetta[1], Flora Giudicepietro[2], Stefano Masiello[1], Luca Pugliese[4], and Anna Esposito[3]

[1] Dipartimento di Fisica, Università di Salerno, INFN, and INFM Salerno
[2] Osservatorio Vesuviano, INGV, Napoli, Italy
aesposito@ov.ingv.it
[3] Seconda Università di Napoli, and INFM Salerno, Italy
[4] IIASS, via Pellegrino 19, Vietri sul Mare (SA), Italy

Abstract. This paper compares three unsupervised projection methods: Principal Component Analysis (PCA), which is linear, Self-Organizing Map (SOM) and Curvilinear Component Analysis (CCA), which are both nonlinear. Performance comparison of the three methods is made on a set of seismic data recorded on Stromboli that includes three classes of signals: explosion-quakes, landslides, and microtremors. The unsupervised analysis of the signals is able to discover the nature of the seismic events. Our analysis shows that the SOM algorithm discriminates better than CCA and PCA on the data under examination.

1 Introduction

Dimension reduction techniques are widely used for the analysis and visualization of complex datasets. These techniques may be distinguished into two classes. In the former there are linear methods like Principal Component Analysis (PCA) [3] or the classical Multidimensional Scaling (MDS) [3]. In the latter there are nonlinear methods like Self-Organizing Map (SOM) [5] or nonlinear variants of MDS, like the recently proposed Curvilinear Component Analysis (CCA) [1].
PCA is able to perform eigenvalue decomposition on the data, detecting linear dependencies between vectors of features that constitute the data set of interest. However, the detection of linear dependencies is a limitation when projection methods which are able to capture the higher order structure of non-Gaussian distributed data are required. Among these, Self-Organizing Maps (SOM) is one of the most powerful since can transform input data of arbitrary dimension into a low dimensional map, by performing a nonlinear topology preserving mapping. However, SOM transforms unknown input distribution onto a fixed topological structure and this represents a prior constraint. When no matching takes place between the predetermined structure and the intrinsic structure of the input data, this technique leads to sub-optimal mappings. In this case, it is suggested to use the CCA that has proved to be successful for several applications [2] since it permits to obtain a more consistent representation of the input data. However, in this article we have shown that, for our class separation task, SOM performs better than CCA, and that CCA does not outperform PCA. This analysis is further supported by the results reported in [9]. In the following, we first illustrate the data set and describe the preprocessing performed on the data. We

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 70 – 77, 2006.

describe the three projection methods in section 4 and the comparison among these techniques and their results in the last section.

2 Data Description

The volcanic island Stromboli has a permanent eruptive activity, monitored by a network of 13 digital stations. Seismic signals are characterized by microtremors and explosion-quakes. However, in Dec 2002 there was also a large landslide that generated a small tsunami. This event created the necessity to automatically discriminate among the several seismic activities, that inspired this research. Our data set consists of three classes of signals acquired with a sampling rate of .02s, composed of 430 explosion-quakes, 267 landslides, and 462 microtremors. Each recording is 24s long (more details on the data acquisition are available at www.ov.ingv.it/stromboli/).

3 Extraction of Seismic Features

An important stage in the data analysis task is the feature extraction. This step is performed in order to extract unfailing information from signals, eliminating as much as possible redundancy, in order to obtain a compact and significant data representation.

There are several signal analysis techniques: the selection of the appropriate algorithm depends on the particular application and computational complexity. Generally, the identification of the features that are "significant" for the specific application is a critical point.

The preprocessing method used in this work is the *Linear Predictive Coding* (LPC) [7]. The basic idea of the LPC is to model each event s_n as a linear combination of a certain number p of its past values as:

$$\overline{s_n} = \sum_{k=1}^{p} c_k s_{n-k} + G \tag{1}$$

where c_k are the *prediction coefficients, G* is the *gain* and p represents the *model order*. The c_k estimation is obtained by an optimization procedure which tries to minimize the error function between the real signal at time t and its LPC estimate. The choice of the order p is problem dependent and is generally made estimating the LPC residual error over the dataset at the hand. Applying the LPC coding to each signal and computing the residual error averaged over all the dataset, it was possible to estimate the model order p that best fits our data. Figure 1 shows both this error and Akaike's Final Prediction Error (FPE) computed as a function of the order p. From the figure it is possible to observe that the value for p=6 used in this work seems to be a good trade-off between the compactness of the data representation and the error made using such a representation. Moreover, this choice for the p value is also justified by the fact that in a previous work [8], where a supervised learning algorithm was used, we found that p=6 was the order value that gives the best performance on the same seismic data set. Thus, we extract 6 coefficients plus the gain from each one of the 8 Hanning windows in which we divided the signal, each window overlapping with the previous one by 2.56 seconds.

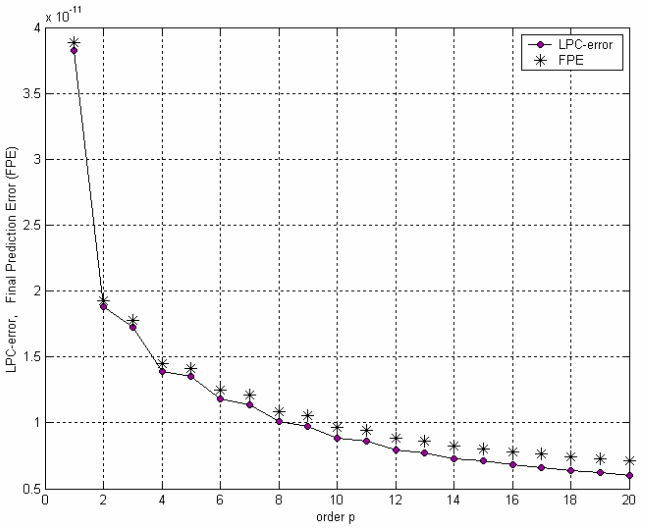


Fig. 1. LPC residual error (filled circles) and Akaike's Final Prediction error (filled stars) computed as a function of the model order p and averaged over all the events in the dataset

The length of the analysis window was fixed to 5.12 seconds under suggestion of the seismologists since they experimentally found that this time interval permits to take into account all the frequencies of interest for the signals at the hand. Moreover these specific seismic signals are assumed by the experts to be stationary in the above specified time interval.

LPC efficiently encodes signal frequency features. In order to preserve also signal time-domain information we encoded in the data representation also a waveform information obtained as the properly normalized difference between the maximum and the minimum amplitude of the signal in a 1s long window. The use of both spectral and temporal features is justified by the fact that seismologists exploit this information in the visual inspection of the seismic signals. After the preprocessing each recorded signal was encoded with 56 LPC coefficients plus 23 time features giving for each signal a feature vector of 79 components.

4 PCA, SOM and CCA Algorithms

We have applied the PCA, the CCA, and the SOM algorithms to our feature vectors in order to identify a possible structure in the data under examination. The PCA algorithm finds the axes of maximum variance of the input data and represents them by a linear projection onto a subspace of reduced dimension p [3]. The CCA [1] instead, performs a nonlinear dimensionality reduction in two steps: (1) a vector quantization (VQ) of the input data into K quantized n-dimensional prototypes and (2) a nonlinear projection of these quantized vectors onto a p-dimensional output space. After learning the quantized prototypes, the prototype pairs (xi,yi) are used to interpolate the continuous mapping between the n-dimensional input space X and the p-dimensional output space Y. The nonlinear mapping is done minimizing the cost function

described in the equation (2) where $X_{ij}=d(xi,xj)$ and $Y_{ij}=d(yi,yj)$ are the Euclidean distances between quantized vectors and output vectors, respectively, and $F(Y_{ij},\lambda)=exp(-Y_{ij}/\lambda)$ is a weighting function that favors the preservation of the data topology.

$$E = 1/2 \sum_i \sum_j (X_{ij} - Y_{ij})^2 F(Y_{ij}, \lambda) \tag{2}$$

A drawback of this procedure is that the values of the neighborhood parameter λ and of the learning rate η have to be properly chosen by the user. While the identification of η requires no particular attention, on the contrary, the neighborhood parameter λ is critical. As noted also in [6] the CCA performance critically depends on the choice of λ (and its decreasing time-speed): if λ decreases too slowly, the nonlinear dependencies are not well unfolded, whereas, a fast decrease compromises the CCA convergence. A help in the choice of the CCA parameters comes from the use of the *dydx* plot. The *dydx* plot shows the joint distribution of the distances $dx=Xij$ and $dy=Yij$ in the input and output space, respectively. In this representation, a perfect match ($Xij \approx Yij$) clusters the points around the identity function. A locally good mapping is shown by a distribution close to the identity function near the origin (local projection), while unfolding is revealed by bent and spread with $dy>dx$ in average. Even though the *dydx* plot can be of some help to the user, this dependency on critical parameter raises several difficulties.

Like the CCA, the Kohonen Self-Organizing Map (SOM) performs non-linear mapping of an *n*-dimensional input space onto a two-dimensional regular grid of processing units known as "neurons". A prototype vector is associated with each neuron (node). The fitting of the prototype of each node is carried out by a sequential regression process that minimizes the differences between each input vector and the corresponding winning node prototype. Namely, at each time step $t = 1,2,...$ a sample $x(t)$ is extracted and the winner index c (best match) is identified by the condition:

$$\forall i, \left\| \mathbf{x}(t) - \mathbf{m}_c(t) \right\| \le \left\| \mathbf{x}(t) - \mathbf{m}_i(t) \right\| \tag{3}$$

where $x(t)$ is the feature vector of the signal extracted at step t, and $m_i(t)$ are the prototypes of node i. After that, all prototypes are updated as:

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{c(x),i}(\mathbf{x}(t) - \mathbf{m}_i(t)) \tag{4}$$

where $h_{c(x),i}$ is the "neighborhood function", a decreasing function of the distance between the *i-th* and *c-th* nodes on the map grid. The neighborhood kernel can be written in terms of the Gaussian function $h_{c(x),i}=\eta_t \exp(-d_{c,i}^2/2\sigma_t^2)$, where η_t is a scalar valued learning rate, σ_t the neighborhood radius at step t, $d_{c,i}$ is the distance between map units c and i on the map grid. Both η_t and σ_t are sometimes monotonically decreasing functions and their exact forms are not critical [4]. The SOM algorithm accomplishes two important steps: (a) a clustering of the input data into nodes, and (b) a spatial ordering of the map in the sense that the prototypes are ordered on the grid so that similar inputs fall into topologically close nodes. Such an ordered map of the data items facilitates the understanding of the structures in the data set. The clusters structure in the data set can be made more visible by displaying the distances between prototype vectors of neighboring nodes. The Euclidean distances between

prototype vectors of neighboring nodes can be visualized on the map using gray levels. In such a way, the SOM map gives a good representation of the clustering structure, by graphically depicting both the density of the data in the prototype plane and the Euclidean distances between the prototypes.

5 Results

The application of the three methods above described on the seismic data set allows a bi-dimensional representation of them. Applying the PCA technique on our data set we obtain the projection shown in Fig. 2.

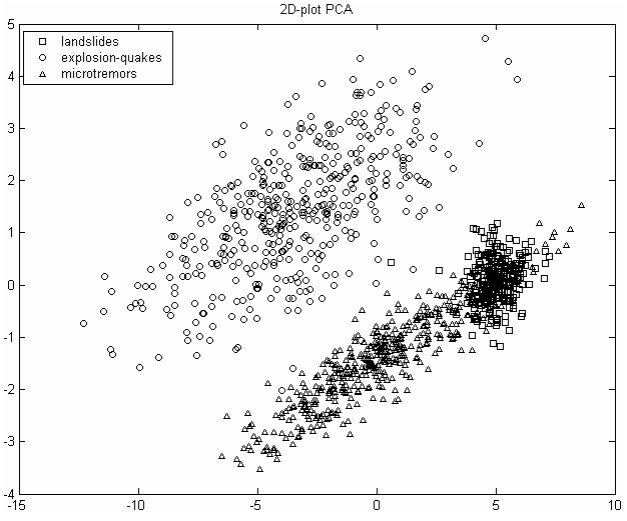


Fig. 2. Bi-dimensional projection of the data vectors using the PCA. The plot is made according to the known classification: the empty *triangles* indicate microtremors, the empty *circles* indicate explosion-quakes and the empty *squares* represent landslides.

The plot has been made according to the signal classification performed by the experts: the empty *triangles* indicate microtremors, the empty *circles* are the explosion-quakes and the empty *squares* represent landslides. As we can see from Fig. 2, the PCA visualization permits to discriminate the explosion-quakes from the other two classes of signals, which instead appear not well separated.

While for the PCA, once chosen the output space dimension p, no parameter must be adjusted by the user, the CCA performance strongly depends on the adjustment of the two parameters η and λ. For the adjustment of λ values we exploited the *dydx* plot. After several trial and error tests, a value of λ was found that gives the plot reported in Fig. 3 showing the *dydx* plot obtained setting λ equal to 48.4. As it has been observed in [1] in this plot *"correct matches give points close to the line corresponding to the identity function, whereas unfolding is revealed by bent and spread dydx characteristics*". Therefore the dispersion of data in the *dydx* representation indicates

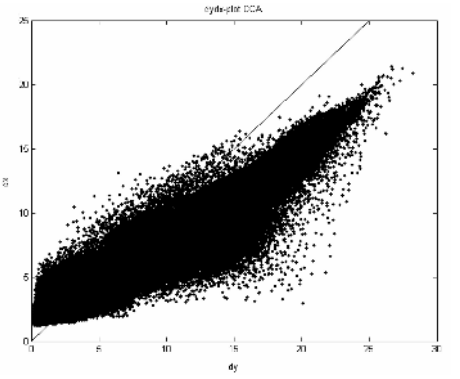


Fig. 3. The *dydx* plot obtained for this λ value equal to 48.4. The dispersion of data in the *dydx* representation indicates that the three classes are not well separated.

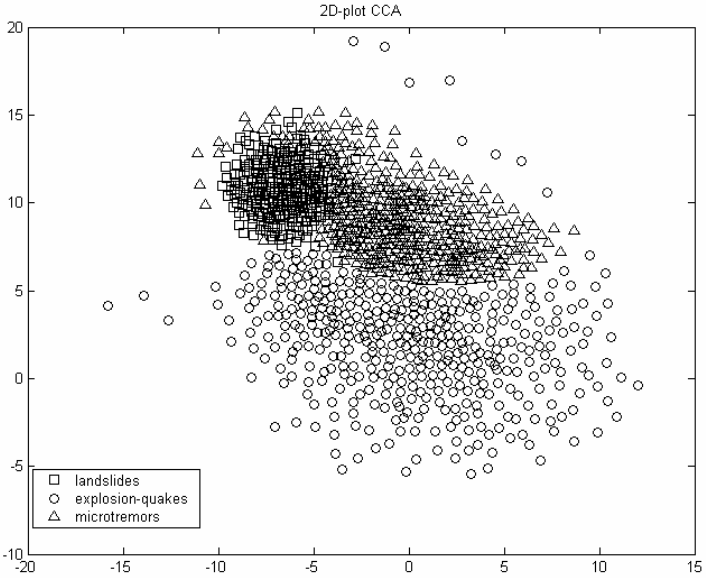


Fig. 4. Projection of the data set using the CCA algorithm. The bi-dimensional visualization is made according to the known classification: the *triangles* indicate microtremors, the *circles* indicate explosion-quakes and the *squares* indicate landslides.

that the three classes are not well separated. This could be better observed in Fig. 4 that shows the poor discrimination obtained with the CCA algorithm for the three classes under examination (visualized according the known classification).

The results in Fig. 4, obtained after several attempts for determining an appropriate λ value, show that the CCA projection was able to adequately separate the explosion-quakes (empty circles) from the landslides (empty squares) and the microtremors (empty triangles) even though the microtremors and landslides still overlap.

Contrarily to the CCA, the SOM maps do not show a critical dependence from its parameters. In our experiments, the SOM learning parameters have been settled in agreement with the prescriptions reported in [4] and the resulting map is shown in Fig. 5.

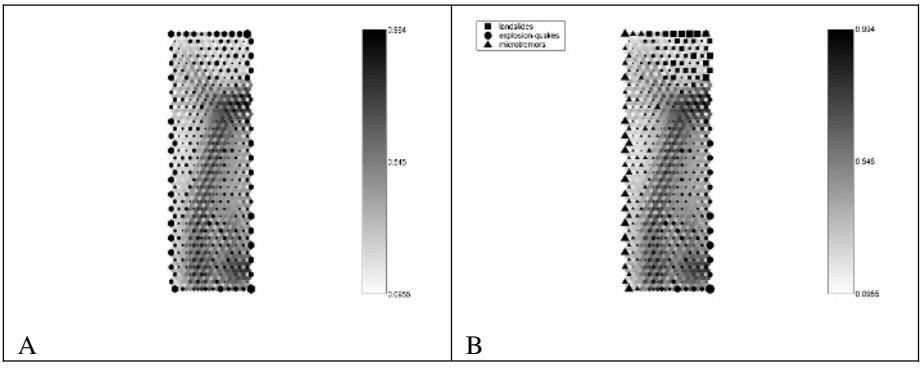


Fig. 5. A SOM Map with 36x11=396 nodes. A) The node (black hexagon) size indicates the number of samples that belong to that node. The gray level coloring indicates the distance between prototypes of neighboring nodes. B) The SOM clustering made according the known classification. The t*riangles* indicate microtremors, the *circles* are for explosion-quakes and the s*quares* for landslides.

Fig. 5A shows the topological distance among the three classes. In this figure each prototype is visualized as a black hexagon, whose size represents the number of samples associated with that node. Distances between the prototypes are visualized on the map using gray level coloring. According to this coloring, high distance values between two prototypes correspond to dark gray hexagons separating the two corresponding nodes. In order to have a qualitative measure of the matching between the cluster structure revealed by the SOM and the known structure of our dataset, each node on the map has been represented according to the classification made by the seismologists. Observing the map in Fig. 5A, it is possible to distinguish a dark gray boundary (from the upper right to the lower left) that separates two large clusters. Moreover, within the two clusters some sub-clusters are also present. In Fig. 5B the *triangles* indicate microtremors, the *circles* indicate explosion-quakes, and the *squares* indicate landslides. If different types of signals belong to the same node a combination of the above symbols is used. In particular, the cluster on the lower right side of the map, that individuates the explosion-quakes (see Fig. 5B), has a large gray zone that well separates it from the landslides and the microtremors. The microtremors are mainly located in the upper left side of the map, and the landslides in the upper right side. Therefore, the dark gray line (corresponding to large distances between nodes) clearly separates the explosion-quakes from the other two classes. The distance between the landslides and the microtremors seems to be less marked suggesting that the two classes may share similar features and therefore they appear closer to each other. However, the results reported in Fig. 5B show that the three clusters visualized by the SOM map have a clear correspondence with the three classes of seismic signals identified by the experts.

5 Conclusions

Exploratory data analysis methods can be used as tools in knowledge discovery of data structure. The aim of this work is to compare three unsupervised projection methods, the linear PCA and the nonlinear CCA and SOM, in order to identify, among them, the one that can better give an easily understandable bi-dimensional representation of three different classes of seismic signals while preserving as much as possible the essential information needed to separate the above three classes. These techniques work without assumption about the data distribution and no external information, like class labels, is provided to obtain the final output. The analysis is unsupervised, and the possible class labels have been used afterwards to aid in the interpretation of the results, without affecting the structures that have been found. The results show that the SOM performs the cluster separation task better than the PCA and the CCA obtaining minor overlaps with respect to the other two methods. The poor performance for the PCA can be due to the fact that PCA cannot take into account nonlinear structures, since it describe the data in terms of a linear subspace, whereas the principal drawback of the CCA is the strong dependence of its performance on the complex adjustment of its parameters, in particular the neighborhood parameter λ. Finally, it is important to underline that the resulting clustering can be used by the experts for an automatic labeling of the seismic events.

Acknowledgements. We would like to thank Prof. Maria Marinaro for the numerous and useful discussions that have made possible the realization of this work and for her critical reading of the manuscript. Miss Tina Marcella Nappi is acknowledged for her editorial help.

References

1. Demartines, P., Herault, J.: Curvilinear Component Analysis: A Self Organizing Neural Network for Nonlinear Mapping of Data Sets. IEEE Trans. on Neural Networks, Vol. 8 (1997) 48-154
2. Herault, J., Guerin-Dugue, A., Villemain, P.: Searching for the Embedded Manifolds in High-dimensional Data, Problems and Unsolved Questions. Proceedings of ECANN Bruge (2002)
3. Jollife. I.T.:Principal Component Analysis. Springer Verlag, New York (1986)
4. Kohonen, T., Hynninen, J., Kangas, J., Laaksonen, J.: SOM_PAK Program Package. Report A31. Helsinki University, Finland (1996)
5. Kohonen, T.: Self-Organizing Maps. Series in Information Sciences, Vol. 30. Springer, Heidelberg. Second ed (1997)
6. Lee, J.A., Lendasse, A., Donckers, N., Verleysen, M.: A Robust Nonlinear Projection Method. Proceedings of ESANN'2000 D-Facto pubbl. (2000) 13-20
7. Makhoul, J.: Linear prediction: a Tutorial Review. IEEE Vol. 63 (1975). 561-580.
8. Esposito, A. M., Giudicepietro, F., Scarpetta, S., D'Auria, L., Marinaro, M., Martini .M.: Automatic Discrimination of Landslides Seismic Signals at Stromboli Volcano using Neural Network (submitted)
9. Masiello, S., Esposito, A. M., Scarpetta, S., Giudicepietro, F., Esposito, A., Marinaro, M.: Application of Self Organized Maps and Curvilinear Components Analysis to the Discrimination of Vesuvius Seismic Signals. To appear in the Proceedings of the Workshop on Self Organizing Map (WSOM), Paris, 5-8 September (2005)

Artifact Cancellation from Electrocardiogram by Mixed Wavelet-ICA Filter

Fabio La Foresta, Nadia Mammone, and Francesco Carlo Morabito

DIMET, *Mediterranea* University of Reggio Calabria,
via Graziella Loc. Feo di Vito,
I 89100 Reggio Calabria, Italy
morabito@unirc.it, nadia.mammone@ing.unirc.it,
laforesta@ingegneria.unime.it

Abstract. In this paper a novel method, called WICA, based on the joint use of wavelet transform (WT) and independent component analysis (ICA) is discussed. The main advantage of this method is that it encompasses the characteristics of WT and ICA. In order to show the novelty of our method, we present a biomedical signal processing application in which ICA has poor performances, whereas WICA yields good results. In particular, we discuss the artifact cancellation in electrocardiographic (ECG) signals. The results show the ability of WICA to cancel some artifact from ECG when only two signals are recorded.

1 Introduction

When the biomedical signals are corrupted by some artifact, a preprocessing step is needed in order to extract some clinical information from the data. For this reason the artifact cancellation is a key topic in biomedical data processing [1]-[6]. In particular, the artifact removal is often necessary for the clinical study of the electrocardiographic (ECG) signal [1], [2], [3]. The ECG signal shows the repeating and almost periodic pattern. This characteristic of physiological signals was explored in order to synchronize the parameters of the filter with the period of the signal; Liang and Lin [6] have also discussed the effectiveness of their method based on discrete wavelet transform (DWT) in order to perform the cancellation of stimulus artifact in the serosal recordings of gastric myoelectric activity. However, those filter fail to remove the interference when it has the same frequency of the ECG signal. On the other hand, the independent component analysis (ICA) was implemented in order to remove artifacts in biomedical signals [2], [4], [5]. In particular Barros *et al.* [2] have shown that, under some conditions, ICA can separate ECG signal from the interference.

In this paper a novel method, called WICA, based on the joint use of DWT and ICA is presented. In the next sections we discuss the new method and we show the ability of WICA to cancel some artifacts from ECG when only two signals are recorded.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 78–82, 2006.

2 The WICA Approach for ECG Artifact Rejection

The WICA method was proposed for the first time for an application on electromyographic signals [7, 8]. This method merges the advantages of the DWT and the ICA; here, wavelets are not only used as a denoising or filtering tool, but the wavelet decomposition is a preprocessing step which projects each raw data into a n-dimensional orthogonal basis (consisting of the scaling function and the wavelet functions) where n-1 is the number of levels, while the redundancy is increased and the ICA performance is improved, therefore the decomposition is an integral part in the separation process. In Figure 1 the WICA method fitted to ECG artifact rejection is depicted: the first stage is a discrete wavelet decomposition of the ECG signal, the number of levels was set at 7. The wavelet decomposition-reconstruction were performed by biorthogonal wavelets [9], because the wavelet functions belonging to this family have a shape which is close to the ECG shape. Once the raw data had been so projected into the n-dimensional space, a new dataset was built with some selected wavelet components, we will discuss the choice of the number of levels and the selection of the wavelet components in the next section, since they depend on the spectral features of the signals. The new dataset was processed by ICA based on the extended-INFOMAX algorithm [10]. During this step the independent components (ICs) related to the artifacts were removed (for more details see [11]). The last stage of WICA procedure is the inverse discrete wavelet decomposition which reconstructs the cleaned ECG signals.

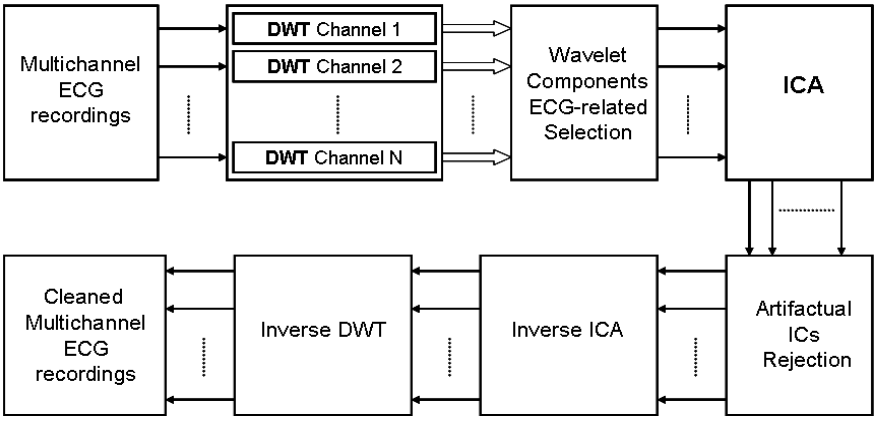


Fig. 1. Block scheme of the WICA method for ECG artifact rejection

3 Results

In order to test the performances of the WICA method for ECG artifact rejection, we decided to mix some real ECG signals with some real artifact signals and to process the mixed data.

The ECG signals and the noise signal are from an archive of recorded physiologic signals (available online at http://www.physionet.org/). The ECG data are two-channel ambulatory ECG recordings and they were digitized at 360 samples per second per channel with 11-bit resolution over a 10 mV range. The artifact signal was recorded by some electrodes placed on the limbs in positions in which the ECG signal of the subjects was not visible, it is an electromyographic (EMG) artifact and it was digitized at the same sampling rate and with the same quantization settings as the ECG. We focused on the early six seconds data segment of the three *sources* (two-channels ECG and EMG-artifact), then we linearly mixed them in order to mimic an ECG corrupted by an EMG signal during a real two-channel ambulatory ECG recording; we show the mixed signals in Figure 2. At first, we tried to extract the artifact from the corrupted dataset by ICA, but it was unable to isolate the artifact component; this was to be expected because the number of *sources* was larger than the number of channels and, therefore, we were dealing with an overcomplete ICA problem which can not be solved by INFOMAX algorithm. It is a very troublesome problem, it is even more troublesome when the number of sources is not known and in the real environment this is a very likely condition [12]. Thus, we tested the WICA procedure on the corrupted dataset. The first stage of the method is the wavelet decomposition: we performed a seven levels decomposition by biorthogonal wavelets. Note that we set the number of levels at seven because the frequency band of the signals was [0-180] Hz and we aimed to capture the low frequency waves related to the breathing artifact (which could affect the recorded ECG signals) and the low frequency spectral content of our EMG artifact. The seven levels decomposition provided an approximation, for each channel, whose spectrum was in the range [0-1.12] Hz and which captured the extremely low frequency spectral content of the EMG artifact. The approximations were so discarded in the wavelet components dataset construction. The first and the second levels details, whose frequency ranges were [90-180] Hz and [45-90] Hz respectively, were also discarded because they accounted for noise. Instead, the third level detail ([22.5-45] Hz) and the fourth level detail (11.25-22.5] Hz) were selected for the dataset, the fifth, sixth and seventh level details ([5.5-11.25] Hz, [2.25-5.5] Hz, [1.125-2.25] Hz, respectively) were merged in a single component and then it was inserted in the dataset, they were merged and inserted separately because we did not need to increase the redundancy so much. The constructed dataset was

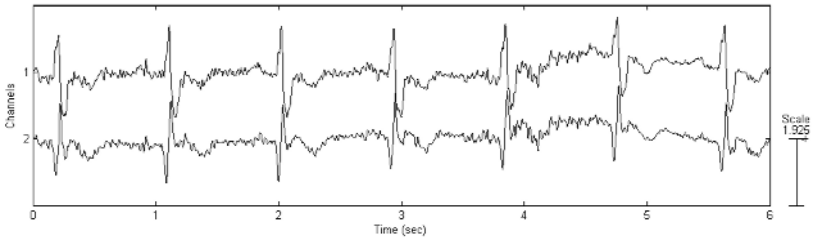


Fig. 2. The corrupted dataset yielded by the linear mixing of the ECGs and the EMG

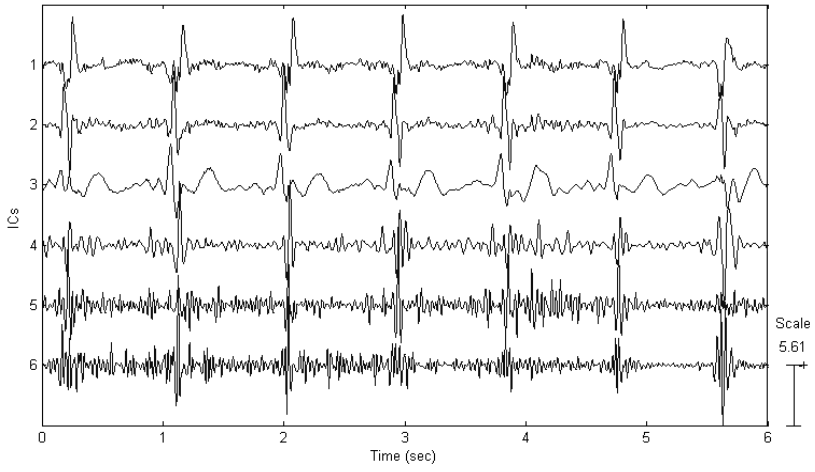


Fig. 3. The independent components from the wavelet components dataset

processed by ICA and the ICs are plotted in Figure 3, we can point out that the early three ICs account for ECG, in particular, the early two signal account for QRS waves and the third one for P and T waves [13], the last ones account for noise and therefore they were rejected. The cleaned wavelet component dataset was yielded by the inverse ICA on the remaining ICs, this dataset was processed by the inverse DWT which yielded the cleaned multichannel ECG recordings (Figure 4). The correlation coefficient between the cleaned dataset and the original signals is 0.9, they are close indeed, apart from a residual little amount of noise.

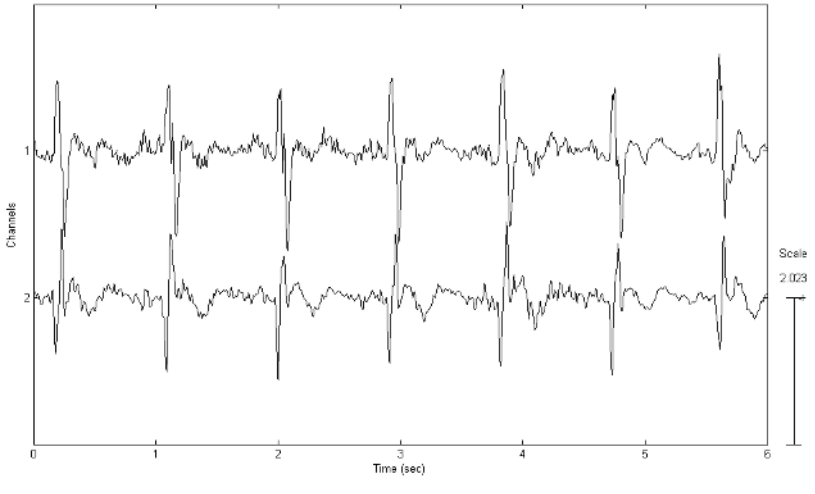


Fig. 4. The cleaned dataset yielded by the WICA method

4 Conclusions

In this paper a novel WICA method was used in order to perform the artifact cancellation in corrupted ECG signals. The WICA procedure was based on a mixed filter that encompassed the characteristics of wavelet transform and ICA. The results have shown the ability of WICA to cancel some artifacts from ECG when only two signals were recorded. The WICA approach allows extending the artifact removal in those clinical applications in which both DWT and ICA, when applied separately, have poor performances.

References

1. Romberg, D.: An ECG lead system for reducing artifacts in the temporal averaging of cardiac signals. Proceedings of Computers in Cardiology (1991) 397–400
2. Barros, A.K., Mansour, A., Ohnishi, N.: Removing artifacs from electrocardiographic signals using independent component analysis. Neurocomputing **22** (1998) 173–186
3. Tong, D.A., Bartels, K.A., Honeyager, K.S.: Adaptive reduction of motion artifact in the electrocardiogram. Proceedings of The Second Joint EMBS/BMES Conference **2** (2002) 1403–1404
4. Jung, T., Humphries, C., Lee, T.W., Makeig, S., McKeown, M.J., Iragui, V., Sejnowski, T.J.: Removing Electroencephalographic Artifacts: Comparison between ICA and PCA. Neural Network for Signal Processing **VIII** (1998) 63–72
5. Jung, T., Makeig, S., Westerfield, M., Townsend, J., Courchesne, E., Sejnowski, T.J.: Removal of eye activity artifacts from visual event-related potentials in normal and clinical subjects. Clinical Neurophysiology **111** (2000) 1745–1758
6. Liang, H., Lin, Z.: Stimulus artifact cancellation in the serosal recordings of gastric myoelectric activity using wavelet transform. IEEE Trans. Biomedical Engineering **49** (2002) 681–688
7. Azzerboni, B., Finocchio, G., Ipsale, M., La Foresta, F., Morabito F.C.: A New Approch to Detection Muscle Activation by Independent Component Analysis and Wavelet Transform. LNCS **2486** (2002) 109–116
8. Azzerboni, B., Carpentieri, M., La Foresta, F., Morabito F.C.: Neural-ICA and Wavelet Transform for Artifacts Removal in surface EMG. Proceedings of The IJCNN Conference (2004) 3223–3228
9. Cohen, A., Daubechies, I., Feauveau, J.C.: Biorthogonal basis of compactly supported wavelets. Comm. Pure Appl. Math. **45** (1992) 485–560
10. Lee, T.W., Girolami, M., Sejnowski, T.J.: Independent Component Analysis using an extended infomax algorithm for mixed sub-Gaussian and super-Gaussian sources. Neural Computation **11**(2) (1999) 606–633
11. Azzerboni, A., La Foresta, F., Mammone, N., Morabito F.C.: A new approach based on wavelet-ICA algorithms for fetal electrocardiogram extraction. Proceedings of The ESANN Conference (2005)
12. Lee, T.W., Lewicki, M.S., Girolami, M., Sejnowski, T.J.: Blind Source Separation of more sources than mixtures using overcomplete representations. IEEE Signal Processing Letters **6**(4) (1999) 87–90
13. Rosén, K.G., Amer-Wahlin, I., Luzietti, R., Norén, H.: Fetal ECG waveform analysis. Best Prectice & Research Clinical Obstetrics and Gynaecology **18**(3) (2004) 485–514

Intelligent Predictive Control of Micro Heat Exchanger

Mehdi Galily, Farzad Habibipour Roudsari, Masoum Fardis, and Ali Yazdian

Iran Telecom Research Center (ITRC), Ministry of ICT, Tehran, Iran
m.galily@gmail.com

Abstract. An intelligent predictive control to temperature control of a micro heat exchanger is addressed. First, the dynamics of the micro heat exchanger is identified using a locally linear model. Then, the predictive control strategy based on this model of the plant is applied to provide set point tracking of the output of the plant.

1 Introduction

Most of the existing predictive control algorithms use an explicit process model to predict the future behavior of a plant and because of this, the term model predictive control (MPC) is often utilized [1,2] for this control strategy. An important characteristic, which contributes to the success of the MPC technology, is that the MPC algorithms consider plant behavior over a future horizon in time. Thus, the effects of both feedforward and feedback disturbances can be anticipated and eliminated, fact, which permits the controller to drive the process output more closely to the reference trajectory.

Although industrial processes usually contain complex nonlinearities, most of the MPC algorithms are based on a linear model of the process. Linear models such as step response and impulse response models derived from the convolution integral are preferred, because they can be identified in a straightforward manner from process test data. In addition, the goal for most of the applications is to maintain the system at a desired steady state, rather than moving rapidly between different operating points, so a precisely identified linear model is sufficiently accurate in the neighborhood of a single operating point. As linear models are reliable from this point of view, they will provide most of the benefits with MPC technology. Even so, if the process is highly nonlinear and subject to large frequent disturbances, a nonlinear model will be necessary to describe the behavior of the process [3-5].

Recently, neural networks have become an attractive tool in the construction of models for complex nonlinear systems [6-8]. Most of the nonlinear predictive control algorithms imply the minimization of a cost function, by using computational methods for obtaining the optimal command to be applied to the process. The implementation of the nonlinear predictive control algorithms becomes very difficult for real-time control because the minimization algorithm must converge at least to a sub-optimal solution and the operations involved must be completed in a very short time (corresponding to the sampling period). In this paper, we will apply a predictive controller to output temperature tracking problem in a electrically heated micro heat exchanger plant [9,10]. First, the nonlinear behavior of the process is identified using a Locally Linear Model Tree (LOLIMOT) network [11,12] and then predictive

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 83 – 89, 2006.

Controller is applied to the plant. Using the proposed strategy, the tracking problem of the temperature profile will be tackled. The performance of the proposed controller is compared with that of a classic PID controller, which simulation results show better match for the proposed predictive controller.

2 Electrically Heated Micro Heat Exchanger

Electrically heated micro heat exchangers have been developed to accelerate the fluid and gas heating in a reduced space [9,10]. This system consists of a diffusion bonded metal foil stack with many grooves, the heating element are placed between the foils (Fig. 1). In a small volume, powers to 15 kW can be converted. The advantages of this heat exchanger are

- Fluids and gas heated by electrical power and not by additional flow cycle
- Efficient transformation of electrical energy in thermal energy
- Fast temperature change of the media and temperature good fit for sensitive media
- Compact construction due to micro system technology.

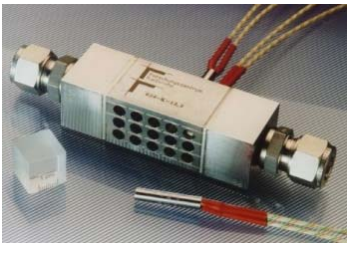


Fig. 1. Electrically heated micro heat exchanger

3 Locally Linear Model Tree Identification of Nonlinear Systems

The network structure of a local linear neuro-fuzzy model [11,12] is depicted in Fig. 2. Each neuron realizes a local linear model (LLM) and an associated validity function that determines the region of validity of the LLM that are normalized as

$$\sum_{i=1}^{M} \varphi_i(\underline{z}) = 1 \tag{1}$$

for any model input $\underline{z}$.

The output of the model is calculated as

$$\hat{y} = \sum_{i=1}^{M} (w_{i,o} + w_{i,1}x_1 + ... + w_{i,n_x}x_{n_x})\varphi_i(\underline{z}) \tag{2}$$

where the local linear models depend on $\underline{x} = [x_1,...,x_{n_x}]^T$ and the validity functions depend on $\underline{z} = [z_1,...,z_{n_z}]^T$. Thus, the network output is calculated as a weighted sum

of the outputs of the local linear models where the $\varphi_i(.)$ are interpreted as the operating point dependent weighting factors. The network interpolates between different Locally Linear Models (LLMs) with the validity functions. The weights w_{ij} are linear network parameters. The validity functions are typically chosen as normalized Gaussians. If these Gaussians are furthermore axis- orthogonal the validity functions are

$$\varphi_i(\underline{z}) = \frac{\mu_i(\underline{z})}{\sum_{j=1}^{M} \mu_j(\underline{z})} \tag{3}$$

with

$$\mu_i(\underline{z}) = \exp(-\frac{1}{2}(\frac{(z_1 - c_{i,1})^2}{\sigma_{i,1}^2} + \ldots + \frac{(z_1 - c_{i,n_z})^2}{\sigma_{i,n_z}^2})) \tag{4}$$

The centers and standard deviations are *nonlinear* network parameters. In the fuzzy system interpretation each neuron represents one rule. The validity functions represent the rule premise and the LLMs represent the rule consequents. One-dimensional Gaussian membership functions

$$\mu_{i,j}(z_j) = \exp(-\frac{1}{2}(\frac{(z_j - c_{i,j})^2}{\sigma_{i,j}^2})) \tag{5}$$

can be combined by a t-norm (conjunction) realized with the product operator to form the multidimensional membership functions in (3). One of the major strengths of local linear neuro-fuzzy models is that premises and consequents do not have to depend on identical variables, i.e. $\underline{z}$ and $\underline{x}$ can be chosen independently.

The LOLIMOT algorithm consists of an outer loop in which the rule premise structure is determined and a nested inner loop in which the rule consequent parameters are optimized by local estimation.

1. *Start with an initial model*
2. *Find worst LLM*
3. *Check all divisions*
4. *Find best division*
5. *Test for convergence*

For the termination criterion various options exist, e.g., a maximal model complexity, that is a maximal number of LLMs, statistical validation tests, or information criteria. Note that the *effective* number of parameters must be inserted in these termination criteria. Fig. 3 illustrates the operation of the LOLIMOT algorithm in the first four iterations for a two-dimensional input space and clarifies the reason for the term ”tree” in the acronym LOLIMOT. Especially two features make LOLIMOT extremely fast. First, at each iteration not all possible LLMs are considered for division. Rather, Step 2 selects only the worst LLM whose division most likely yields the highest performance gain. For example, in iteration 3 in Fig. 3

only LLM 3-2 is considered for further refinement. All other LLMs are kept fixed. Second, in Step 3c the local estimation approach allows to estimate only the parameters of those two LLMs which are newly generated by the division. For example, when in iteration 3 in Fig. 3 the LLM 3-2 is divided into LLM 4-2 and 4-3 the LLMs 3-1 and 3-3 can be directly passed to the LLMs 4-1 and 4-3 in the next iteration without any estimation.

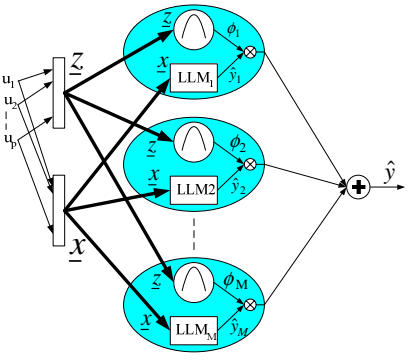


Fig. 2. Network structure of a local linear neuro-fuzzy model with M neurons for nx LLM inputs x and nz validity function inputs z

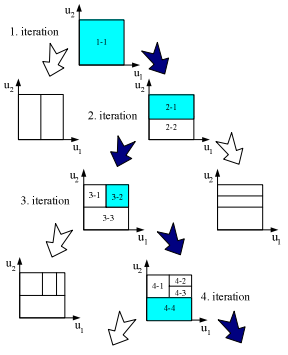


Fig. 3. Operation of the LOLIMOT structure search algorithm in the first four iterations for a two-dimensional input space ($p = 2$)

Using the above strategy, a locally linear model is adopted to the system for input data as shown in fig. 4. The identified and actual outputs can be seen in Fig. 5. As it can be seen, the error between these two values is not considerable and the identified model can match the system well. In the next section, we will use of this model in predictive control block.

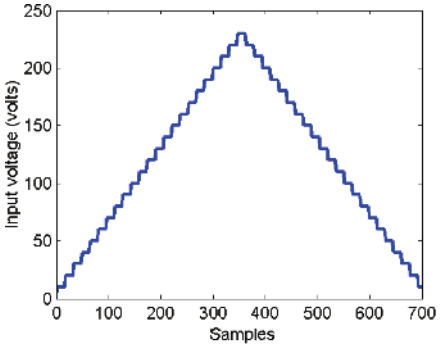


Fig. 4. Input voltage for system identification

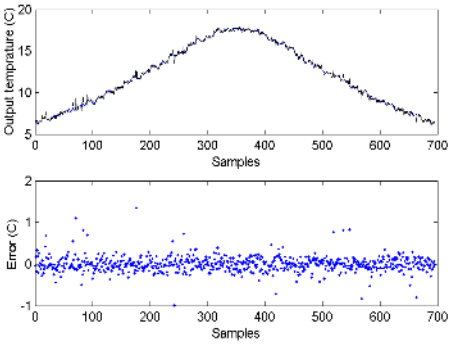


Fig. 5. The identified and actual output temperature with error between theses two values

4 Predictive Controller Design

The objective of the predictive control strategy using LOLIMOT predictors is twofold: *(i)* to estimate the *future output* of the plant and *(ii)* to minimize a *cost function* based on the error between the predicted output of the processes and the reference trajectory. The cost function, which may be different from case to case, is minimized in order to obtain the optimum control input that is applied to the nonlinear plant. In most of the predictive control algorithms a quadratic form is utilized for the cost function:

$$J = \sum_{i=N_1}^{N_2} \left[y(k+i) - r(k+i) \right]^2 + \lambda \sum_{i=1}^{N_u} \Delta u^2 (k+i-1) \tag{9}$$

with the following requirements

$$\Delta u(k+i-1) = 0 \qquad 1 \le N_u < i \le N_2 \tag{10}$$

where N_u is the control horizon, N_1 and N_2 are the minimum and maximum prediction horizons respectively, i is the order of the predictor, r is the reference trajectory, λis the weight factor, and Δ is the differentiation operator.

The command u may be subject to amplitude constraints:

$$u_{\min} \le u(k+i) \le u_{\max} \qquad i = 1,2,..., N_u \tag{11}$$

The cost function is often used with the weight factor λ=0. A very important parameter in the predictive control strategy is the control horizon N_u, which specifies the instant time, since when the output of the controller should be kept at a constant value.

The output sequence of the optimal controller is obtained over the prediction horizon by minimizing the cost function J with respect to the vector U. This can be achieved by setting

$$\frac{\partial J}{\partial U} = 0 \qquad U = \left[u(k-d),...,u(k-d+N_u-1) \right]^T \tag{12}$$

However, when proceeding further with the calculation of $\partial J/\partial U$, a major inconvenience occurs. The *analytical approach* to the optimization problem needs for the differentiation of the cost function and, finally, leads to a nonlinear algebraic equation; unfortunately this equation cannot be solved by any analytic procedure. This is why a *computational method* is preferred for the minimization of the cost function, also complying with the typical requirements of the real-time implementtations (guaranteed convergence, at least to a sub-optimal solution, within a given time interval).

The advantage of this nonlinear neural predictive controller consists in the implementation method that solves the key problems of the nonlinear MPC. The implementation is robust, easy to use and fulfills the requirements imposed for the minimization algorithm. Changes in the parameters of the neural predictive controller (such as the prediction horizons, the control horizon, as well as the necessary constraints) are straightforward operations. A simple block diagram of predictive control strategy is depicted in Fig. 6.

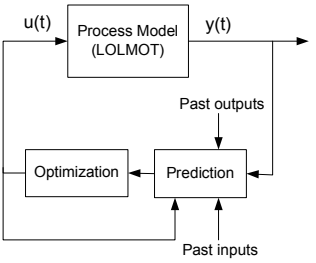


Fig. 6. The scheme of predictive control

The closed-loop system response using the predictive control algorithm based on LOLIMOT model is shown in Fig. 7. In order to investigate the performance of the predictive controller, we will provide another simulation using conventional PID controller. Using trail and error algorithm, the best values for PID controller in which the closed-loop system is stable and has almost satisfactory performance is adopted as follows

$$\text{PID Controller} \Rightarrow k_p + \frac{k_i}{s} + k_d s$$

$$k_p = 1.24 \quad k_i = 2.27 \quad k_d = 0.17$$

The closed-loop system response using PID controller with above parameters is shown in Fig. 8. Comparing Fig. 7 with Fig. 8, we can see that the performance of the system using the predictive controller is much better than that of PID one.

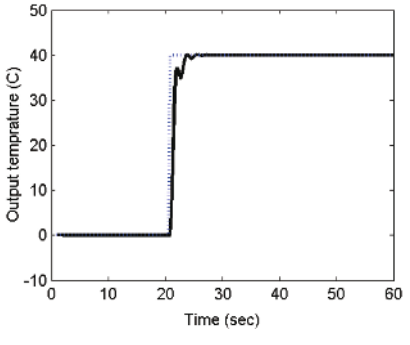


Fig. 7. Closed-loop system response using proposed predictive controller

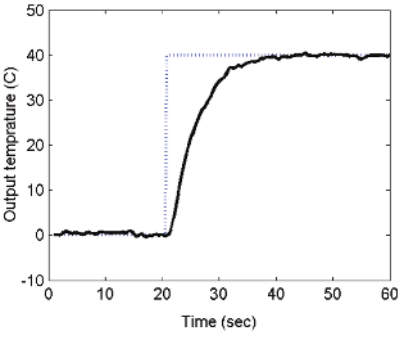


Fig. 8. Closed-loop system response using PID

5 Conclusion

In this paper, a predictive Controller (BELBIC) was applied to electrically heated micro heat exchanger, which is a highly nonlinear plant. To this end, the dynamics of the system was identified using Locally Linear Model Tree (LOLIMOT) algorithm. Then, a controller based on predictive strategy was applied to the system to tackle the

output temperature tracking problem. The closed-loop system performance using the proposed predictive controller was compared with that of PID one, which the result of predictive controller was much better than that of PID controller.

References

1. Camacho, E.F. Model predictive control, Springer Verlag, 1998.
2. Garcia, C.E., Prett, D.M., and Morari, M. Model predictive control: theory and practice- a survey, Automatica, 25(3), pp.335-348, 1989.
3. Badgwell, A.B., Qin, S.J. Review of nonlinear model predictive control applications, In Nonlinear predictive control theory and practice, Kouvaritakis, B, Cannon, M (Eds.), IEE Control Series, pp.3-32, 2001.
4. Parker, R.S., Gatzke E.P., Mahadevan, R., Meadows, E.S., and Doyle, F.J. Nonlinear model predictive control: issues and applications, In Nonlinear predictive control theory and practice, Kouvaritakis, B, Cannon, M (Eds.), IEE Control Series, pp.34-57, 2001.
5. Babuska, R., Botto, M.A., Costa, J.S.D., and Verbruggen, H.B. Neural and fuzzy modeling on nonlinear predictive control, a comparison study, Computatioinal Engineering in Systems Science, July, 1996.
6. Nelles, O. Nonlinear system identification: from classical approach to neuro-fuzzy identification, Springer Verlag, 2001.
7. Narendra, K. S., and Parthasarathy, K., Identification and control of dynamic systems using neural networks. IEEE Transactions on Neural Networks, 1, pp.4–27, 1990.
8. Jalili-Kharaajoo, M. and Araabi, B.N. Neural network control of a heat exchanger pilot plant, to appear in IU Journal of Electrical and Electronics Engineering, 2004.
9. Brander, J., Fichtner, M., Schygulla, U. and Schubert, K., Improving the efficiency of micro heat exchangers and reactors. In Irven, R. [Hrsg.] Micro reaction Technology: 4th International Conference; AIChE Spring Nat. Meeting, Atlanta, Ga., March 5-9, 2000.
10. Stief, T., Langer, O.U. and Schuber, K., Numerical investigations on optimal heat conductivity in micro heat exchangers. In Irven, R. [Hrsg.] Micro reaction Technology: 4th International Conference; AIChE Spring Meeting, Atlanta, Ga., March 5-9, 2000.
11. Nelles, O., Nonlinear system identification: From classical approaches to neural networks and fuzzy models. Springer, 2001
12. Nelles, O., Local linear model tree for on-line identification of time variant nonlinear dynamic systems. Proc. International Conference on Artificial Neural Networks (ICANN), pp.115-120, Bochum, Germany, 1996.

Transient Power System Stabilizer Design Using Simple Neuron Structure

Farzad Habibipour Roudsari, Behzad Jalaie, and Mehdi Galily

Iran Telecom Research Center, Ministry of ICT, Tehran, Iran
roudsari@itrc.ac.ir

Abstract. In this paper, a new power system stabilizer, which guarantees the voltage regulation as well, is designed using a simple neuron structure. The Neuro-controller is trained on-line based on a modified function. It consists of one neuron, one weight, hard limit activation function, and a constant input. Some simulation studies on a nonlinear third order generator model with the proposed neuro-controller are provided. The obtained results demonstrate the effectiveness of this simple neuro-controller.

1 Introduction

An important problem, which is frequently considered in power system stabilizer design, is to maintain steady acceptable voltage under normal operating and disturbed conditions, which is referred as the problem of voltage regulation [1,2]. The Artificial Neural Network (ANN) technology has matured enough to be applied successfully in many control fields [3,4]. However, its success will eventually depend on its ability to remove a major obstacle, i.e. the lake of a firm theory. As a first step towards a solution for many problems that face practical applications of neural network in control field, an s-domain model of a simple neuro-controller is developed in this paper. Training of the proposed neuro-controller is on-line by the back propagation (BP) algorithm using a modified error function. By representing the neuro-controller learning equations in the s-domain, the controller parameters can be determined analytically. Using the calculated parameters, applications of the neuro-controller as a field excitation controller for a synchronous generator in a single-machine infinite-bus power system are illustrated by simulation studies.

2 Model of Power System

In this paper, a simplified dynamic model of a power system, namely, a Single-Machine to Infinite-Bus (SMIB) power system is considered [5]. This model consists of a single synchronous generator connected through a transmission line to a very large network approximated by an infinite bus. The model is shown in Fig. 1.

- *Mechanical equations*

$$\dot{\delta}(t) = \omega(t) - \omega_o \; ; \quad \dot{\omega}(t) = -\frac{D}{2H}(\omega(t) - \omega_o) - \frac{\omega_0}{2H}(P_e(t) - P_m) \tag{1}$$

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 90 – 94, 2006.

The mechanical input power P_m is treated as a constant in the excitation controller design, i.e., it is assumed that the governor action is slow enough not to have any significant impact on the machine dynamics.

- *Generator electrical dynamics*

$$\dot{E}'_q(t) = \frac{1}{T'_{do}}(E_f(t) - E_q(t)) \tag{2}$$

More details about power system modeling and the definition of the parameters are [1,5].

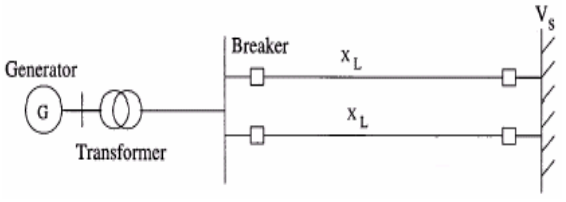


Fig. 1. A single machine infinite bus power system

3 Training of the Neuron Based Controller

In order to regulate generator terminal voltage to the prefault value, a simple neuro-VR is used [2,3]. The overall control system with the proposed neuro-controller consisting of one neuron is shown in Fig. 2. The neuro-controller uses a linear hard limit activation function and a modified error feedback function.

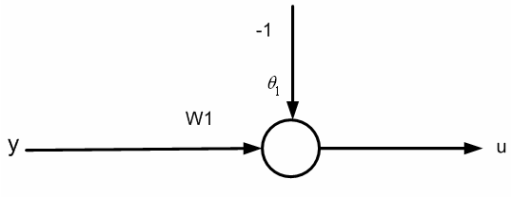


Fig. 2. single neuro configuration

The neuro-controller uses a simple procedure to update its weight on-line. There is no need for any off-line training. There is no need for parameter identification or reference model. It uses the sampled values of the system output to compute the error using the modified error function. This error is back propagated through the single neuron to update its weight. Then, the output of the neuro-controller is computed, which is equal to the neuron weight. The neuro-controller output can be derived as:

$$u(t) = W(t) \tag{3}$$

$$W(t) = W(t-1) + \eta \times WCT(t) \tag{4}$$

where *WCT* is the neuron weight correction term based on the modified error function. *W(t), u(t),* η are neuron weight, neuron output and the learning rate,

respectively. Based on equations (3) and (4), the neuro-controller model in s-domain can be obtained. In time domain, (4) can be written as:

$$W(t) - W(t - \Delta t) = \eta \times WCT(t) \tag{5}$$

Dividing (5) by Δt , we have:

$$\frac{W(t) - W(t - \Delta t)}{\Delta t} = \frac{\eta \times WCT(t)}{\Delta t} \tag{6}$$

Using the differential form, (6) can be written as:

$$\frac{dW(t)}{dt} = \eta_1 \times WCT(t) \quad ; \quad \eta_1 = \frac{\eta}{\Delta t} \tag{7}$$

Representing (7) in s-domain results in:

$$sW(t) = \eta_1 \times WCT(t) \tag{8}$$

From (3) and (8), we have:

$$U(s) = \frac{\eta_1 \times WCT(t)}{s} \tag{9}$$

The general form of the weight correction term is:

$$WCT(s) = R(s) - C(s) f(s) \tag{10}$$

where *R(s), C(s)* and *f(s)* are reference input, system output and feedback function, respectively. Complete model of the proposed neuro-controller in s-domain is shown in Fig. 3, where *G(s)* represents the controlled system.

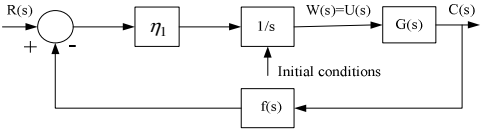


Fig. 3. Neuro-voltage controller model in s-domain

The Complete procedure of training of the network to obtain the best controller parameters has been presented in [2]. The neuro-controller as a voltage regulator (VR) and power system stabilizer with the simplified machine model is shown in Fig. 4.

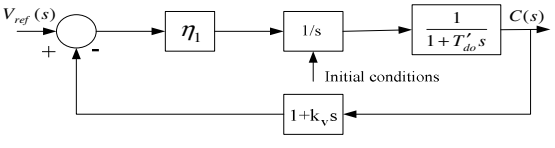


Fig. 4. Neuro-voltage controller as an AVR for simple model of synchronous generator

4 Simulation Results

The proposed neuro-AVR is used to simulate the dynamic behavior of the generating unit connected to a constant voltage bus through two parallel transmission lines [3]. The parameters of the model mentioned in section 2 and the initial conditions of the system are as the following

$$x_d = 1.863 \quad x'_d = 0.257 \quad x_T = 0.127 \quad x_L = 0.4853 \quad x_{ad} = 1.712 \quad H = 4$$
$$D = 5 \quad \omega_o = 314.159 \quad k_c = 1 \quad x_{ds} = 2.23265 \quad x'_{ds} = 0.62665 \quad x_s = 0.36965 \quad T'_{do} = 6.9$$
$$\delta_o = 34.2^o \qquad P_{mo} = .8 p.u. \qquad V_{to} = 1 p.u.$$

Without modified error function [2], the system response to a 0.05pu step change in reference voltage is shown in Fig. 5. As it is clear from this figure, this controller is not suitable to control the generator terminal voltage. By introducing the modified error function in the design of the controller, system response to a 0.05pu step increase in reference voltage at 1sec is depicted if Fig. 6. It is seen from Fig. 6 that the terminal voltage response is excellent. It also demonstrates that the parameters obtained based on a simplified linear model can be used for the non-linear model as well. The performance of the proposed NAVR is compared with the commercial AVR (the AVR obtained from MATLAB toolbox). The results are shown in Fig. 7. As it can be seen the performance of the NAVR is much better than that of AVR in response to 0.05pu step disturbance in voltage reference.

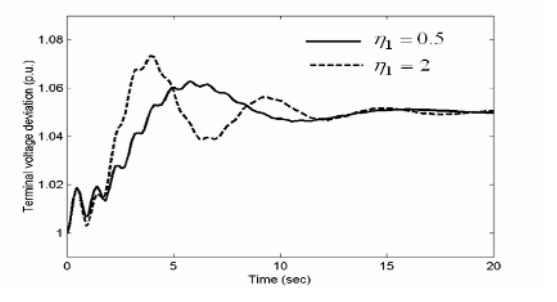


Fig. 5. Neuro -controller performance using unity feedback function for the third-order model of the synchronous generator

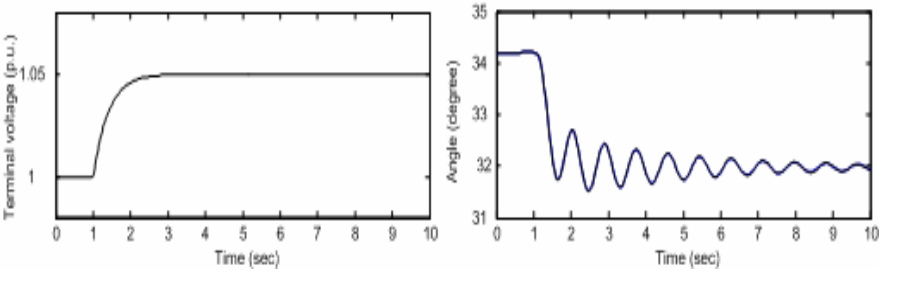


Fig. 6. Neuro-controller performance using modified error function for the third-order model of the synchronous generator

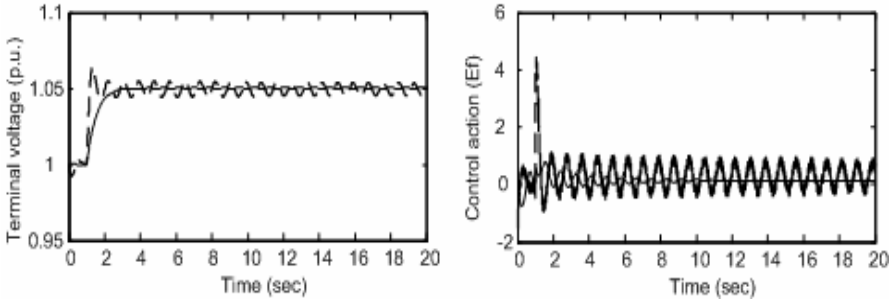


Fig. 7. System response to 0.05pu step disturbance in voltage reference using NAVR (solid line) and commercial AVR (dashed line)

5 Conclusions

In this paper, a neuro-controller with a simple structure for a synchronous generator was presented. The Neuro-controller is trained on-line based on a modified function. The neuro-controller consists of one neuron, one weight, hard limit activation function, and a constant input. Having the neuro-controller in *s*-domain, its stability analysis with a simplified generator linear model was presented. The neuro-controller parameters were obtained analytically to ensure system stability. The controller parameters, which were calculated, based on a simplified linear model, can be used for a nonlinear model. The proposed controller was used to function as an AVR for a SMIB power system. Results showed that the neuro-controller acts as an adaptive controller with better performance than the commercial AVR.

References

1. Jalili-Kharaajoo, M., Moezi-Madani, K.: Optimal Nonlinear Transient Controller for a Single Machine-Infinite Bus Power System, in Proc. IEEE International Conference on Control Applications (CCA'03), Istanbul, Turkey (2003) 306-311
2. Jalili-Kharaajoo, M., Mohammadi-Milasi, R.: Design of simple neuro-controller for global transient control and voltage regulation of power systems, International Journal of Control, Systems and Automation, 3(2) (2005) 302-307
3. Shamsollahi, P., Malik, O.P.: An Adaptive Power System Stabilizer Using On-Line Trained Neural Networks, IEEE Transactions on Energy Conversion, 12(4) (1997) 382-387
4. Salem, M., El-Zahab, A., Malik, O.P.: Generating Unit Excitation Neuro-Controller. Proceedings, in Proc. IFAC Symposium on Power Plants and Power Systems Control, Brussels, Belgium (2000) 97-102
5. Kundur, P.: Power system stability and control, In the EORI Power System Engineering Series, New York, Mc Grow Hill (1994)

Short Term Local Meteorological Forecasting Using Type-2 Fuzzy Systems

Arianna Mencattini[1], Marcello Salmeri[1], Stefano Bertazzoni[1], Roberto Lojacono[1], Eros Pasero[2], and Walter Moniaci[2]

[1] Dip. Ingegneria Elettronica, Università di Roma "Tor Vergata," via del Politecnico 1 – 00133 Roma RM, Italy
{mencattini, salmeri, bertazzoni, lojacono}@ing.uniroma2.it
http://micro.eln.uniroma2.it

[2] Dip. Elettronica, Politecnico di Torino, c.so Duca degli Abruzzi 24 – 10129 Torino TO, Italy
{eros.pasero, walter.moniaci}@polito.it
http://www.neuronica.polito.it

Abstract. Meteorological forecasting is an important issue in research. Typically, the forecasting is performed at "global level," by gathering data in a large geographical region and by studying their evolution, thus foreseeing the meteorological situation in a certain place. In this paper a "local level" approach, based on time series forecasting using Type-2 Fuzzy Systems, is proposed. In particular temperature forecasting is inspected. The Fuzzy System is trained by means of historical local time series. The algorithm uses a detrend procedure in order to extract the chaotic component to be predicted.

1 Introduction

The recent evolution of the power computation of the modern computers has allowed a great improvement in the field of meteorological forecasting. Typically, it is performed at "global level," gathering data over a large geographical region. The study of the evolution of this data permits to foresee the meteorological situation in a certain place. Because of the huge amount of data to be processed, this procedure is time consuming and needs very powerful computers. On the contrary, a complementary approach permits to operate at "local level," processing only the historical data related to that place in order to predict its future evolution. Many meteorological variables, such as the temperature, the humidity, the pressure, the wind speed can be taken into account and analyzed. The sampled values of the interesting variables are acquired in uniformly spaced intervals and used to predict one or more samples in the following intervals. This procedure, known as "Time Series Prediction," is composed of two phases. In the first one the prediction system is trained using known time series. In the second one the following value is predicted starting from a certain number of previously acquired samples. Many approaches for the prediction systems design have been proposed in literature. Some of them exploit analytical properties of the series

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 95–104, 2006.

and compute the following values starting from a proper expansion of the signal. Others consist of a neural network trained on previous historical values [1].

Our approach uses a particular Fuzzy Logic System (FLS), known as Type-2 FLS, suitable to take into account non-stationary noise both in the training and operating phases.

The paper is organized as follows. In Section 2 Type-2 FLS are pointed out in order to show how they can be used in forecasting. In Section 3 the time series prediction issue is explored and in Section 4 a particular set of meteorological data, acquired during a year, are analyzed in order to identify its components in time and frequency domains. In Section 5 the architecture of the whole system is described and some preliminary simulation results are shown. Finally, in Section 6 some aspects of the proposed approach will be highlighted.

2 Type-2 Fuzzy Logic Systems

A classical FLS, also denoted as Type-1 FLS, can be represented as in Fig. 1(a). As shown, rules play a central role in the FLS framework. They can be provided by experts or can be extracted from numerical data. The IF-part of a rule is an antecedent while the THEN-part of a rule is its consequent. Fuzzy sets are associated with terms that appear both in the antecedent and in the consequent, while Membership Functions (MFs) are used to describe these fuzzy sets. Recall that a Type-1 fuzzy set A, can be represented, in terms of a single variable $x \in X$, as

$$A = \{(x, \mu_A(x)) | \forall x \in X\},$$

where Type-1 MF, $\mu_A(x)$, belongs to the range $[0, 1]$, for all $x \in X$ and it is a 2D function, that depends on both x and A.

However, in many applications there are at least four sources of uncertainties that Type-1 FLS cannot handle [2].

1. The meanings of the worlds that are used in the antecedents and consequents of rules can be uncertain.
2. Consequents may have a histogram of values associated with them.

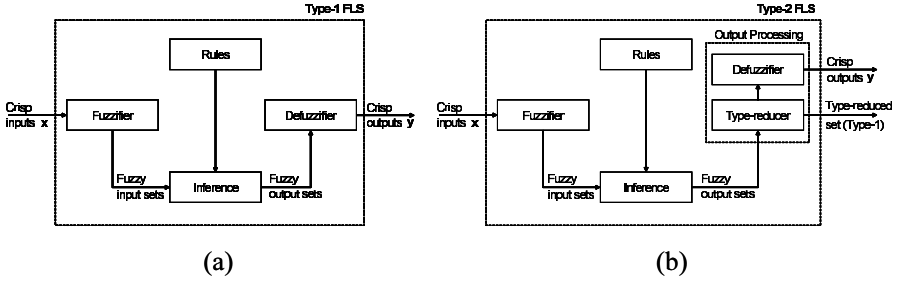


Fig. 1. Type-1 and Type-2 FLS scheme

3. Measurements that activate a Type-1 FLS may be noisy and therefore uncertain.
4. The data that are used to tune the parameters of a Type-1 FLS may also be noisy.

On the contrary, Type-2 FLS, first introduced by Zadeh in 1975 [3], succeed in modeling such uncertainties because their MFs are themselves fuzzy. In order to introduce the structure of Type-2 MFs, imagine blurring the Type-1 membership function depicted in Fig. 2 (a) by shifting the points on the triangle either to the left or to the right and not necessarily by the same amounts, as in Fig. 2 (b). Then at a specific value of x, called x', there no longer is a single value for the

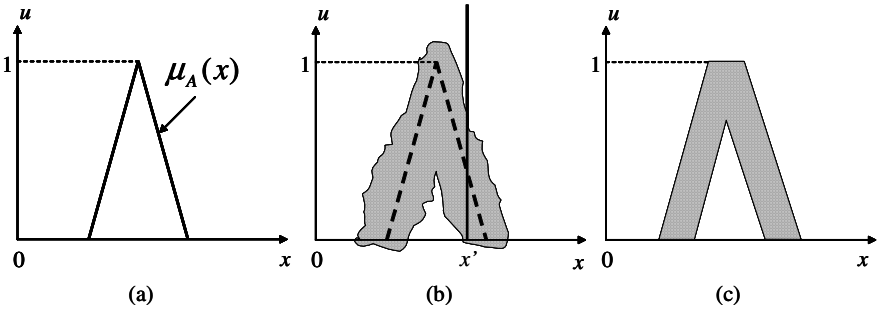


Fig. 2. (a) Type-1 MF, (b) Blurred Type-1 MF, (c) FOU

membership function, whereas the membership function takes on values wherever the vertical line intersects the blur. Those values need not all be weighted the same, and we can assign an amplitude distribution to all of those points. So, we can create a three dimensional membership function, a Type-2 MF, that characterizes a Type-2 fuzzy set. Thus, we can define a Type-2 fuzzy set $\tilde{A}$

$$\tilde{A} = \{(x,u), \mu_{\tilde{A}}(x,u) | \forall x \in X, \forall u \in J_x \subseteq [0,1]\},$$

where $\mu_{\tilde{A}}(x,u) \in [0,1]$. From this definition it follows that, fixed $x = x'$, we have a set of possible values, even with different weights, that we call *secondary MF*, $\mu_{\tilde{A}}(x',u)$, while the domain of this secondary membership function is called *primary membership* of x, J_x in the expression above. Consequently we have primary and secondary degree of membership to a fuzzy set. Figure 2 (c) represents the set of all possible primary MFs embedded in a Type-2 fuzzy set, also denoted as *Footprint of Uncertainty (FOU)*. This term seems very useful because it provides a convenient way to describe the entire support of the secondary grades and in many applications allows to correctly choose appropriate MFs by first thinking about their appropriate FOUs. The FOU can be also described through the concepts of *lower* and *upper* MFs [4] that represent respectively two Type-1 MFs that are bounds for the FOU of a Type-2 fuzzy set $\tilde{A}$. An example is provided in Fig. 3, where one can notice the three-dimensional nature of such a

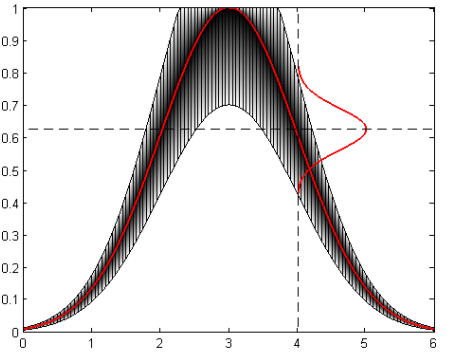


Fig. 3. Footprint of uncertainty

MF, pointed out by the gaussian secondary membership degree associated with the value $x = 4$.

Many choices are possibile for the secondary MFs, Gaussian, triangular, trapezoidal, etc. In particular when the secondary membership degree of x is rectangularly shaped, we say that the secondary MFs are *interval sets*, so that Type-2 FLS is also denoted as *Interval Type-2 FLS* [5]. This subclass of Type-2 fuzzy sets makes this new set of systems more attractive owing to a substantial simplification in the characterization and in the tuning of its parameters. In fact, despite of the growing importance of this class of FLS [6], characterizing a Type-2 FLS is not as easy as characterizing a Type-1 FLS.

By computational aspects, the main difference between Type-2 FLS and Type-1 FLS is in the output processor, as shown in Fig. 1(b). While the output stage in Type-1 FLS is just a defuzzifier and produces a single value as output, the output stage of a Type-2 FLS can be further subdivided in the cascade of two blocks: the first one maps a Type-2 set into a Type-1 set (type reduction [7, 8]) and the second one is a classical defuzzifier. In order to emphasize the difference between classical Type-1 FLS and Type-2 FLS note that the output of a Type-1 FLS is just like the mean of a unknown probability density function (pdf), while a Type-2 FLS can provide information about the dispersion around this mean, just like a variance of that pdf. Despite apparent simplicity, also common operations of union, intersection and complement, needed to implement a classical Type-1 FLS, must be readapted using Zadeh's Extension Principle [3].

During recent years, many works have examined and proved interesting properties of Type-2 FLS making them an effective tool in many classical fields, such as control of mobile robots, decision making, forecasting of time series, function approximation, preprocessing of radiographic images, transport scheduling, co-channel interference elimination from nonlinear time-varying communication channels [7, 9]. In particular, forecasting is a very important issue that appears in many disciplines. For example weather forecast can save lives in the event of a catastrophic hurricane, or financial forecasts can improve the return of an investment. The problem of forecasting a time series can be formalized as follows.

Given a set of p past measurements of a process $s(k)$, namely $x(k-p+1)$, $x(k-p+2), \ldots, x(k)$, determine the estimate of a future value of s, $\hat{s}(k+l)$, where p and l are fixed positive integers.

If the measurements are noise free, then the set of past values are replaced by $s(k-p+1)$, $s(k-p+2), \ldots, s(k)$. When $l = 1$ we obtain the single stage forecaster of s, while in general we obtain the l-stage forecaster of s. Suppose now we collected N data points, $x(1)$, $x(2), \ldots, x(N)$, then we must divide them in two set: the *training* data subset $x(1)$, $x(2), \ldots, x(D)$ and the *testing* data subset $x(D+1)$, $x(D+2), \ldots, x(N)$. Because we use p data points in order to estimate the next data point, we will have $D-p$ training pairs. The training data can be used in order to establish the rules of the FLS in at least two ways.

- The data are used to initialize the center of the fuzzy sets both in the antecedents and in the consequents of the rules.
- A possibile architecture of the FLS is chosen and the training data are used to optimize its parameters.

3 Time Series Prediction

Time series represent a certain phenomenon time evolution, usually by uniformly spaced time intervals. Many physical, economic, and biological phenomena can be represented by time series. They can be classified basing on the relation between the time dependent variable and the time itself. Often, in real cases, a certain phenomenon is described by a discrete time series rather than a continuous time series because the measurements acquired to characterize it are obtained as regularly separated samples during the time.

Time series can be usually modeled as the sum of a deterministic and a chaotic component. The first one deals with those characteristics of the observed phenomenon that can be computed analytically, while the second one concerns those parts of the phenomenon that appear as "casual" events. So, such series can be described by a model as $z(t) = f(t) + a(t)$, where $z(t)$ is the value of the time series, $f(t)$ is the value of the deterministic component, and $a(t)$ is the value of the chaotic component.

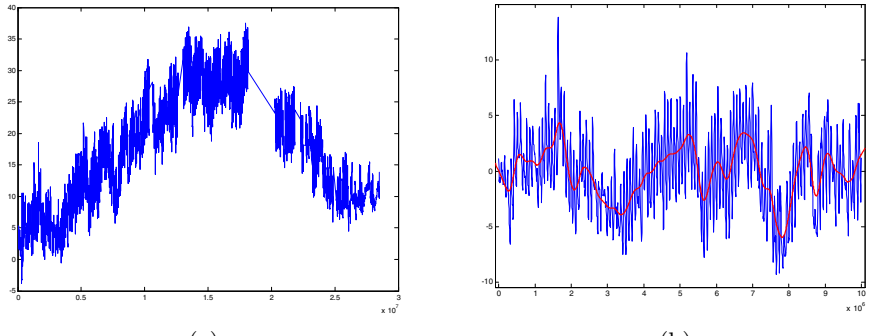
The deterministic component can be often divided in more fundamental components, namely the *trend* that represents the very long-term series behavior (often it is the mean value of the series), and long- and short-term components that represent the deterministic evolution of the time series along long-time and short-time period, respectively. All these components are reasonably affected by (even non-stationary) noise resulting from measurements, extraction and data processing.

Common approaches for modeling univariate time series are the autoregressive (AR) model, the moving average (MA) model, the Box-Jenkins [10] model, and ARIMA models. However, building good ARIMA models generally requires more experience than commonly used statistical methods such as regression. Recently many researcher have taken advantages in the use of wavelet transform in those

forecasting applications where the observed data are correlated, originated from non-stationary process [11, 12]. In this last case, variance and covariance, or equivalently the spectral structure of the process, are likely to change over time. Obviously, in order to develop a meaningful approach, one needs to control this deviation from stationary, and hence to model it. Usually, the simplest approach consists in considering it piecewise stationary, or approximately so, while others suppose a local stationary, applying classical results, locally. In recent years there has been considerable interest in the application of intelligent technologies such as neural networks and fuzzy logic [13, 14, 15]. More recently, these two computationally intelligent techniques have been considered as complementary leading to developments in fusing the two approaches [16, 17], with a number of several successful neurofuzzy systems reported in literature [16, 18, 19, 20, 9]. The application of artificial-neural-network (ANN) and fuzzy logic based decision support systems to time-series forecasting has gained attention recently [18, 19]. ANN-based forecasts give large errors when the profile of the series changes very fast and extremely slow training procedure, in order to select proper structure of the neural networks, make this method not suitable for real time applications. On the other hand, the development of a fuzzy decision system (fuzzy expert system) for forecasting requires detailed analysis of data and a fuzzy rule base has to be developed heuristically. The rules fixed in this way may not always yield the best forecast. The shortcomings of the neural-network drawback can be partly solved by encoding the human knowledge by a fuzzy expert systems, and then integrated into a fuzzy neural network, improving the learning speed of the ANN.

4 Historical Data Analysis

The first phase of the research has been the study of historical time series of meteorological measurements of temperature acquired in a certain place during about one year by the Neuronica Lab at the Politecnico of Torino (Italy) which values are depicted in Fig. 4(a). The measurements have been performed every



(a) (b)

Fig. 4. Temperature during one year (a) and chaotic component of its mean value (b)

900 s, that is a quarter of an hour. This sampling rate is justified by the slow rate of the considered phenomenon. Some intervals in which the data are missing, owing to loss of measurements, can be located in graphs.

The first remark we can highlight is the presence of an annual component. The period of this component is obviously exactly equal to one year and corresponds to the trend component of the global signal. So, this is a deterministic component and it can be deleted in order to isolate the others components of the time series. This procedure allows us to avoid biasing in the following steps. Because the very low frequency of this component, that one can suppose sinusoidal (this approximation does not affect the performance of the prediction system), it cannot be deleted through a classical digital filtering procedure. In fact, in this case the filter needs the samples of some periods, that corresponds to acquire samples for some years. The used approach was a sinusoidal regression analysis of the signal in order to detect both the amplitude and the phase of the annual component. Its frequency is in fact obviously known. Once the characteristic of the annual component has been detected, it can be extracted from the global signal. Figure 4(b) shows a portion of the time series without the annual component with the chaotic mean value highlighted. One can see how much the resulting signal range has been reduced by this procedure.

At this point, two ways can be followed in order to perform long term or short term forecasting.

The first one consists in subdividing the signal into more components (deterministic and chaotic) [21], predicting them independently and reconstructing the predicted sample at the end of the procedure. It is possible because a sharp separation of each component in frequency domain occurs.

The second one, proposed in this paper, predicts the following value of the series directly without further signal decomposition [22]. This method can be applied when the other components of the signal slowly vary in time with respect to the sample time. This approach is very suitable in case of short time prediction avoiding processing errors due to signal decomposition and reconstruction.

5 System Design and Simulation Results

Starting from the consideration highlighted in Section 4, the whole system depicted in Fig. 5 has been designed. Several historical time series values are used to detect the characteristics of the trend component of the signal by a sinusoidal regression procedure which identifies amplitude A and phase Φ of that component. The value of the estimated trend value is subtracted from the actual value of the series. The result goes in input to a Type-2 FLS tuned to predict the chaotic component of the corresponding signal through a proper training phase on historical values.

We highlight one more time that the filtering process that delete the annual component is essential because the training and working phases must have in input values in the same range. In the other hands, we cannot train the FLS on winter temperature range and hope to have correct predicted values in summer.

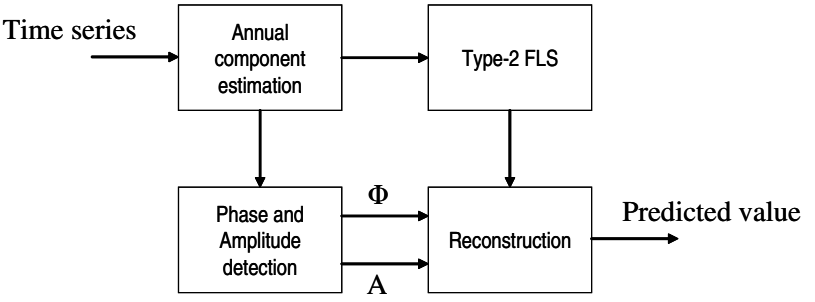


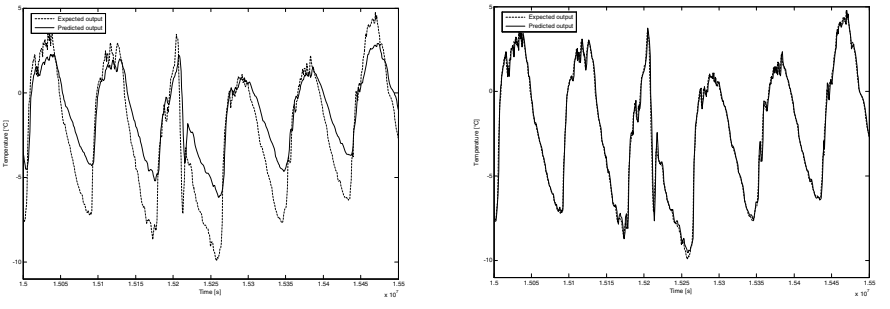
Fig. 5. System architecture

Obviously, once the next value of the series has been predicted, the amplitude value A is added in order to forecast the corrected temperature value.

The described procedure can predict only one following value starting from four past values, that is $\tilde{T}(k)$ is computed using $T(k-p)$ for $p \in [1,4]$. In the prediction of $\tilde{T}(k+1)$ two approaches could be implemented. The first one computes $\tilde{T}(k+1)$ from $T(k-1), T(k-3), T(k-5)$, and $T(k-7)$. The second one uses the estimated value $\tilde{T}(k)$ and the previous $T(k-p)$ for $p \in [1,3]$. The same procedure can be iteratively applied for the prediction of following values. Each method has different advantages. The latter has weakly higher estimation error, but the FLS maintains the same configuration for every following prediction, thus needing only one training procedure. On the contrary, the former needs as many training procedures as the number of predictions.

The described algorithm has been implemented in Matlab® using 5000 samples to train the Type-2 FLS and other 2000 samples for the test.

Some preliminary results of the proposed forecasting approach in the case of temperature prediction are shown in Fig. 6. It displays an example of the prediction, performed through a Type-1 FLS and a suitably tuned Type-2 FLS, of a portion of temperature vs. the expected behavior. We can notice that the results achieved through a Type-2 FLS are much more close to the expected ones.



(a) (b)

Fig. 6. Prediction of temperature with Type-1 FLS (a) and Type-2 FLS (b)

Note that the used measurements were affected by a very low noise, above all due to the quantization procedure. However, simulation performed on a phantom chaotic time series (Mackey-Glass) artificially corrupted by relatively high noise has shown a drastic performance increasing of Type-2 FLS with respect to many other approaches.

6 Conclusions

This paper presents a novel approach to forecast meteorological phenomena. It is based on an architecture using a Type-2 FLS in order to predict the chaotic component of the signal. Using Type-2 FLS is very important because this class of systems has shown small sensitivity to measurements uncertainty and noise. The whole system has been designed and simulated on historical meteorological data previously acquired. First results show promising performance with respect to other methodologies.

The following step in research will be the implementation of the system in HW. It can be noticed that, because the time interval between two consecutive measurements is very large (900 s), we can use very slow devices in order to carry out the forecasting. This fact will permit the production of low cost equipments.

Acknowledgment

The authors would like to thank Jerry Mendel for his support and useful advices in Type-2 FLS design.

References

1. Pasero, E., Moniaci, W.: Artificial neural networks for meteorological nowcast. In: Intern. Symp. on Comp. Intel. for Meas. Syst. and Appl. (CIMSA 2004), Boston, USA (2004)
2. Mendel, J.M., John, R.I.B.: Type-2 fuzzy sets made simple. **10** (2002) 117 – 127
3. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning - 1. Information Sciences **8** (1975) 199 – 249
4. Mendel, J.M., Liang, Q.: Pictorial comparisons of type-1 and type-2 fuzzy logic systems. In: Proc. IASTED Intern. Conf. on Intelligent Systems and Control, Santa Barbara, CA, USA (1999)
5. Liang, Q., Mendel, J.M.: Interval type-2 fuzzy logic systems: Theory and design. **8** (2000) 535 – 550
6. Mendel, J.M.: Uncertain Rule-Based Fuzzy Logic Systems. Prentice Hall, Upper Saddle River, NJ, USA (2000)
7. Karnik, N.N., Mendel, J.M.: Introduction to type-2 fuzzy logic systems. In: Proc. 1998 IEEE FUZZ Conf., Anchorage, AK, USA (1998) 915 – 920
8. Karnik, N.N., Mendel, J.M.: Type-2 fuzzy logic systems: Type-reduction. In: Proc. IEEE Conf. on Systems, Man and Cybernetics, San Diego, CA, USA (1998) 2046 – 2051

9. Mendel, J.M.: Uncertainty, fuzzy logic, and signal processing. Signal Processing Journal **80** (2000) 913 – 933
10. Box, G.E.P., Jenkins, G.M.: Time Series Analysis, Forecasting and Control. 3rd edn. Prentice Hall, Englewood Clifs, NJ (1994)
11. Fryzlewicz, P., Bellegem, S.V., Sachs, R.V.: Forecasting non-stationary time series by wavelet process modelling. Annals of the Institute of Statistical Mathematics **55** (2003) 737 – 764
12. Antoniadis, A., Sapatinas, T.: Wavelet methods for continuous-time prediction using hilbert-valued autoregressive processes. Journal of Multivariate Analysis **87** (2003) 133 – 158
13. Lapedes, A.S., Faber, R.: Non-linear signal processing using neural networks: Prediction and system modelling. Technical Report LA-UR-87-2662, Los Alamos National Laboratory, USA (1987)
14. Moody, J., Darken, C.: Fast learning in networks of locally tuned processing units. Neural Computation **1** (1989) 281 – 294
15. Wang, L.X., Mendel, J.M.: Generating fuzzy rules by learning from examples. **22** (1992) 1414 – 1427
16. Takagi, H.: Fusion technology of fuzzy theory and neural networks: Survey and future directions. In: Int. Conf. on Fuzzy Logic and Neural Networks. (1990) 13 – 26
17. Brown, M., Harris, C.: Neurofuzzy Adaptive Modelling and Control. Prentice Hall, Englewood Clifs, NJ (1994)
18. Jang, R.J.S.: Predicting chaotic time series with fuzzy IF-THEN rules. In: Int. Conf. on Fuzzy Systems. Volume 2. (1993) 1079 – 1084
19. Horikawa, S., Furuhashi, T., Uchikawa, Y.: On fuzzy modelling using fuzzy neural networks with the back propagation algorithm. **3** (1992) 801 – 806
20. Karnik, N.N., Mendel, J.M.: Introduction to Type-2 fuzzy logic systems. In: Int. Conf. on Fuzzy Systems, Anchorage, AK (1998) 915 – 920
21. Mencattini, A., Salmeri, M., Bertazzoni, S., Lojacono, R., Pasero, E., Moniaci, W.: Local meteorological forecasting by Type-2 fuzzy systems time series prediction. In: IEEE Intern. Conference on Computational Intelligence for Measurements Systems and Applications, Giardini Naxos, Italy (2005)
22. Mencattini, A., Salmeri, M., Bertazzoni, S., Lojacono, R., Pasero, E., Moniaci, W.: Short term local meteorological forecasting using Type-2 fuzzy systems. In: Italian Workshop on Neural Network, Vietri sul Mare, Italy (2005)

Learning and Data Driver Methods for Short Term Meteo Forecast

Eros Pasero and Walter Moniaci

Department of Electronic Engineering, Polytechnic of Turin,
INFN of Turin, 10129 Turin, Italy
eros.pasero@polito.it, walter.moniaci@polito.it

Abstract. In this paper an "intelligent systems" based on neural networks and a statistical non parametric method evaluates the future evolution of meteorological variables and the occurrence of particular phenomena such as rain and road ice. The meteorological variables forecast system is based on a recurrent feed forward multi layer perceptron which make the forecasts using data coming from synoptic electronic sensors. Road ice is forecasted using an analytical model.

1 Introduction

Whereas a traditional weather forecast system provides you with a powerful system able to give indications about the weather evolution in next days, a "nowcast" system is used to make short-term prediction and warning decisions. This is made possible thanks to a powerful suite of detection and prediction algorithms. The usefulness of this approach is that the knowledge of the next three hours weather conditions, for example, can be used to organize outdoor activities. The knowledge about the rain event in next two hours can be used to choose the right set of tyres in a formula one competition. The knowledge of the ice occurrence in next three hours allows the airport and motorways maintenance staff to take in advance important decisions in order to avoid accidents. The prediction of meteorological events in restricted areas within short time is therefore an interesting and useful approach to the weather forecast system. The Neuronica laboratory at the Polytechnic of Turin has a long experience in numerical series analysis and prediction using both statistical tools and artificial neural networks. Last year NEMEFO [1], an artificial neural network used to forecast ice on mountain highways was awarded as "electronic project of the year" [2]. This tool uses meteorological data sampled every hour by means of a traditional meteorological station located in a dangerous point of the highway and give indications about humidity, rain, wind and solar radiation) which concur to determine the ice formation. Two algorithms are the engine. The first one is based on the Parzen statistical method and computes the cross entropy among the different classes of parameters. The best parameters are used to train a recurrent neural network which forecasts each single meteorological datum. The predicted data are used by an analytical model which determines the danger of ice formation. Main characteristics of this approach is that it uses only local parameters coming from a meteorological station. This means that no data coming from satellites or other stations are required.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 105 – 112, 2006.

A simple personal computer is requested to use the software. In the next section the neural network – Parzen system is described; in section 3 is shown the ice analytical model and finally the results are commented in section 4.

2 Neural Network – Parzen System

Neural Network

The system profile can be modeled as the output of some dynamic system, influenced by weather variables, time and other environmental variables. A neural network with feedback can simulate a discreet time dynamically system. The general feed-forward topologies with weights sharing can represent feedback connections by unfolding in time the basic network. Of course using this technique, feedback can be followed for a finite time (also arbitrary large). In this way general feed-forward networks can simulate the dynamic systems, but only for their transient behavior. It's necessary to define a network topology (*equations and connections*). The relation between the network input and output depends on several weights that can be modified. Then we must fix a *training rule* in order to adjust the weights and, consequently, to reduce the difference between the network output and the real value (*target*). We have used a recurrent feed-forward network [3] with three layers, as shown in figure 1:

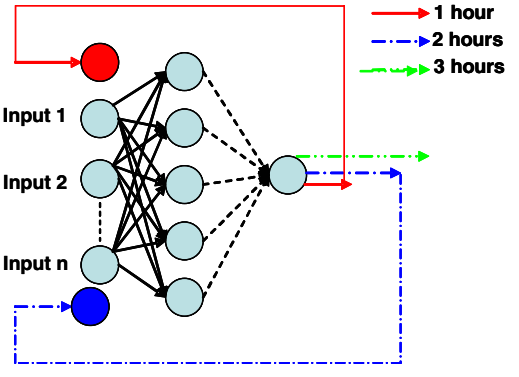


Fig. 1. Recurrent neural network

Each neuron has, as transfer function, the hyperbolic tangent function

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \tag{1}$$

We decided to use back propagation rule to train the network. In this method we choose the weights in order to minimize the square error on training set

$$E = \sum_{t=1}^{T} E(t) = \sum_{t=1}^{T} \sum_{i=1}^{n} 0.5 \cdot [\hat{y}_i(t) - y_i(t)]^2 \tag{2}$$

In this formula $\hat{y}_i(t)$ is the network output whereas $y_i(t)$is the target value. The learning rate $\eta(t)$ is the parameter that conditions the system velocity and efficiency.

It gives an idea of how much the weights change at each iteration. A simple rule to establish the learning rate value for each epoch is the following:

$$\begin{aligned} &E(t) < E(t-1) \Rightarrow \eta(t+1) = 1.01 \cdot \eta(t) \\ &E(t) > E(t-1) \Rightarrow \eta(t+1) = 0.99 \cdot \eta(t) \\ &E(t) \approx E(t-1) \Rightarrow \eta(t) \end{aligned} \tag{3}$$

E(t) and *E(t-1)* are the error function values for two consecutive training epochs.

For the optimization of the free parameters of the network (*weights*) we used the Levenberg – Marquardt [4] algorithm that is a general non linear downhill minimization algorithm for the case when derivatives of the objective function are known. It dynamically mixes Gauss – Newton and gradient descent iterations.

Parzen Method

The goal is to choose the best "predictors" for the rain forecast. We define the entropy of the parameters in the following way:

$$e(x) = -\int_{-\infty}^{+\infty} \log(p(x)) \cdot p(x) dx \tag{4}$$

In this formula *p(x)* is the Probability Density Function (PDF) of a random variable *x*. *e(x)* is an index of dispersion that lies in the range]-∞,+∞[. Let Z be the vector of all the possible predictors that can be used to foresee the variable *y* (predictand). Now let X1 and X2 be two particular subsets of predictors taken from Z. The number of elements in X1 and X2 has to be the same. In order to establish the best set of predictors between X1 and X2 we calculated the following entropy difference:

$$\begin{aligned} &d(X, y) = e(X, y) - e(x) = \\ &= -\int \log P(X, y) \cdot P(X, y) dX dy + \\ &+ \int \log P(x) \cdot P(x) dX \end{aligned} \tag{5}$$

where p(X,y) is the joint PDF of X and y and p(X) is the PDF of X (the predictors). Both PDFs are unknown and therefore it's impossible to evaluate the integrals in (4). We circumvented this problem by estimating the unknown PDFs through the Parzen method [5]. This method estimates the unknown probability density functions making a sum of Gauss kernels, each one centered on a record of the database. The formula is the following:

$$P_X^*(X; \mathbf{D}, \Lambda) = \frac{1}{n} \sum_{i=1}^{n} \prod_{j=1}^{m} \frac{1}{\sqrt{2\pi\lambda_j^2}} \cdot \exp\left[-\frac{(x_j - x_{ij})^2}{2\lambda_j^2} \right] \tag{6}$$

where,

- o **D = {X_1, ……X_n}**: it' s the data vector
- o n: database dimension (number of records)
- o xi,xij: these are the j-th component of *X* and X_i
- o Λ: standard deviation vector, Λ= (λ1,….,λm)

Each standard deviation in Λ. regulates the resolution of the estimator along the corresponding dimension. In its turn this allows us to estimate d(X1,y) and d(X2,y). The best between X1 and X2 is the one that gives rise to the smallest entropy

difference. At the end of the Parzen simulation we obtain the most important meteorological variables that the neural system has to use for the prediction of a particular meteorological parameter. For example we found that the last 4 hour humidity values were sufficient to make a good rain prediction prediction.

3 Ice Analytical Model

This model [6] calculates the amount of ice that is present on the road at the end of the forecast period. It needs receiving the forecast values of the meteorological parameters by the forecast neural system described in section II. It is made up by four modules:

- **1st module:** it [7, 8] calculates water plus ice mass balance on the road.
- **2nd module:** it [7, 8] reckons water minus ice mass balance on the road. This modulus is used only when the road temperature is 0°C. In fact only in this case it's possible the water change phase. As far as this process is concerned it's very important the energy balance on the road.
- **3rd module:** it calculates the snow heap and the snow-melting on the road.
- **4th module:** this module, according to the air – asphalt interface temperature, integrates opportunely the previous modules. On the starting integration instant we used the observed meteorological parameters whereas on the last instant we used that one forecast.

A 1st Module

The initial hypothesis is that we have two different water reserves:

- water in liquid phase
- water in solid phase (ice)

It's assumed that there is only liquid water on the road when the interface air – asphalt temperature is above 0°C. On the contrary we have only water in solid phase when the temperature is under 0°C. When the temperature is equal to 0°C we have the presence of both the water phases. This module calculates the amount of water plus ice on the road in the following way:

$$\frac{dwls}{dt} = k \cdot pt - E - h \cdot wls - 1 \tag{7}$$

where:

- k: it's a coefficient that makes the measure units homogeneous.
- pt: it's the precipitation parameter.
- E: it's the evaporation and condensation term. It depends on wind and relative humidity gradient between the air and the ground.
- h: it's the runoff term and it stands for the amount of water that is taken away by air natural convection.
- wls: it's the instantaneous value of water plus ice.

The evaporation/condensation term is computed as follows:

$$E = -C \cdot \rho \cdot (uar - usat) \tag{8}$$

where:

- *C*: water vapour transfer coefficient in air. It depends on wind velocity.
- ρ: air density that changes according to the air temperature and pressure values.
- *uar*: specific air humidity at the air temperature.
- *usat*: specific air humidity at the interface temperature and in saturated conditions.

This ordinary differential equation is integrated between the measured values of the meteorological parameters and the forecast values given by the neural forecast system.

B 2nd Module

This module calculate the amount of water minus ice that it's on the road as follows:

$$\frac{dwl\_s}{dt} = \frac{2 \cdot (Rn - gt))}{L} - h \cdot 0.003 \cdot (wl\_s - 1) \tag{9}$$

where:

- *Rn*: it's the net solar radiation that allows the water changing phase (from solid to liquid or on the contrary).
- *gt*: it's the heat transferred by conduction through the asphalt layer. It depends on the ground temperature gradient and on the asphalt thermal conductivity.
- *L*: it's the water solidification latent heat.

It's important to underline that this module is present only when the interface temperature is equal to 0°C because it represents the amount of water or ice that has changed phase.

C 3th Module

This module integrates the previous three modules according to the interface temperature values. There are 5 cases that depends on the interface temperature:

- Ts(in)≥0 and Ts(fin)>0
- Ts(in)≥0 and Ts(fin)<0
- Ts(in)≤ 0 and Ts(fin)<0
- Ts(in)≤0 and Ts(fin)>0
- Ts(in)=0 and Ts(fin)=0

Ts(in) and Ts(fin) are the interface temperatures at the initial and final integration instant. At the end of the integration we have always the amount of water and ice on the road.

4 Results

A Classical Meteorological Variables

First we tested our system using 5 years of data of an Italian agricultural meteorological site in Spilimbergo del Friuli. Now the system is running "on line"

tied to a professional meteorological station situated on the roof of the Polytechnic of Turin. Forecast are visible on the web site www.neuronica.polito.it. Figures 2 and 3 show a comparison between measured and forecast values for temperature and humidity. Values are referred to a 24 hours period.

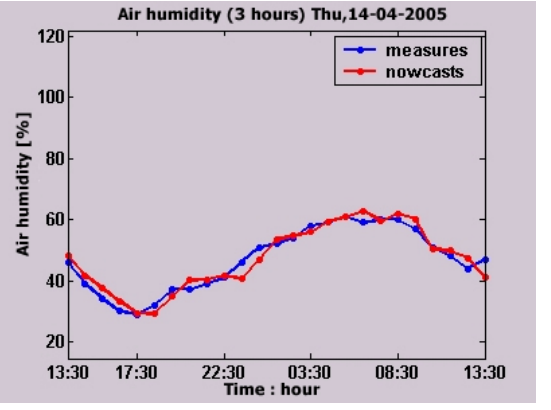


Fig. 2. Humidity forecast – Polytechnic of Turin

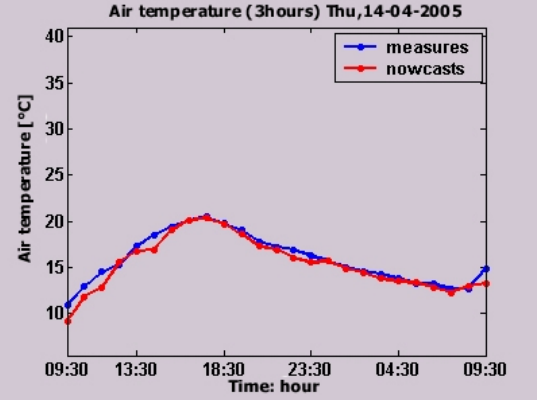


Fig. 3. Temperature forecast - Turin

As far as the rain forecast the observation period is between 15/03/2004 and 29/03/2004. In order to evaluate the quality of the predictions we calculate the mean square error for all the forecasts at Turin site. Then we calculate also the Variance Reduction Coefficient as shown in the following formula:

$$VRC = \frac{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}{\frac{N}{\sigma^2}} \tag{10}$$

where,

- x_i : it's the forecast value
- $\hat{x}_i$: it's the measured value
- N: it's the number of observations

This index is very meaningful: in fact it puts in evidence how much the variability of the process is explained by the predictor used. Lower is the VRC value higher the predictor is correlated to the forecasted variable.

Table 1. Forecast mean square error

	Rain	Air temperature	Humidity
MSE	0.37 mm^2 (1h) 0.42 mm^2 (2h) 0.49 mm^2 (3h)	1.34 $°C^2$ (3h)	15 $\%^2$ (3h)
VRC	0.39 (1h) 0.43 (2h) 0.51 (3h)	0.19 (3h)	0.1 (3h)

We can see that the results are good; in fact regards the rain forecast even in the three hour forecast the mean square error is low if we consider that the pluviometer sensibility is 0.2 mm. Also in the case of temperature and humidity forecasts, the error is comparable with the instrument accuracy (±3.5% for humidity and ±0.5°C for temperature).We can deduct that, being the VRC value less than 1 in all cases, the mean square error is less than the intrinsic variability, expressed by the variance, of the predictand variable.

B Road Ice

We tested the model on the period November (2004) – February (2005). The following graphics show the results.

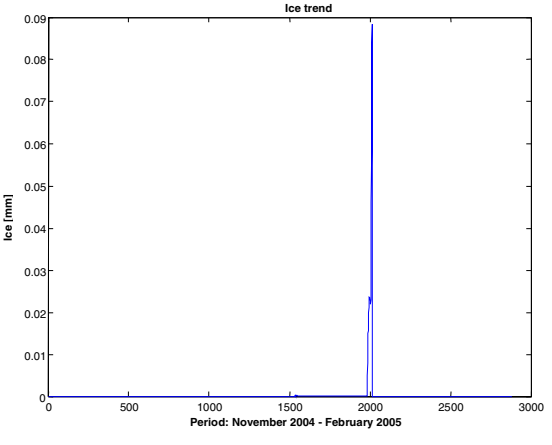


Fig. 4. Ice on the road

5 Conclusion

In this paper we have shown how a hybrid neural statistical system can be applied to a short-term weather forecasting problem. The nature of weather is dynamic, and his change is not only influenced by present weather parameters, but also by the past state variables. Only a neural model incorporates the previous state information. We can get an improvement in the efficiency of the forecast system, increasing database extension. In order to select the critical points (shaded zone) in which meteorological stations have to be positioned, we can create a thermal profile of the road layout. Furthermore in this papers it has been put in evidence how it' s important to investigate the relations among several meteorological parameters that , on sight, have no connections among them. In fact, using the Parzen method, we found that the future evolution of a particular meteorological variable can be influenced by the past evolution of other meteorological parameters. In the future, we could consider the measures of an ice sensor positioned within the asphalt to improve the quality of the weather forecast neural system. Furthermore the spatial information can be added in order to have network of forecasts and not only punctual information. It would be very interesting to apply this system to a runway.

Acknowledgment

This work is partly funded by INFN research project "HAPTIC".

References

1. E. Pasero, W. Moniaci, "Artificial Neural Networks for nowcast systems," presented at the 2004 Int. Symposium. On Computational Intelligence for Measurement Systems (CIMSA), Boston, USA, 14-16 July.
2. Design in Award 2003 for environment design, Padova, Italy, 2003.
3. P. Werbos, "Beyond regression: New Tools for Prediction and Analysis in the Behavioural Sciences," Ph.D. dissertation, Committee on Apply. Math , Harvard Univ., Cambridge, MA, November 1974.
4. D. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *SIAM J. Appl. Math.*, 1963, Vol. 11, pp. 431ñ441.
5. E. Parzen, On Estimation of a Probability Density Function and Mode, Ann. Math. Stat., Vol 33, pp 1065-1076, 1962.
6. B. H. Saas, "A Numerical Model for Prediction of Road Temperature and Ice". *J Appl. Meteor.*, Vol.36 , 801-817, 1992.
7. P. Crevier and Y. Delage, "METRo: A New Model for Road – Condition Forecasting in Canada". *J Appl. Meteor.*, Vol. 40 , 2001.
8. Y. Delage, "Parameterizing Sub-grid Scale Vertical Transport in Atmospheric Models under Statically Stable Conditions". *Bound.-Layer meteor.*, Vol. 82 , 23-48, 1997.

Automatic Dictionary Creation by Sub-symbolic Encoding of Words

Filippo Vella[1], Giovanni Pilato[2], Ignazio Motisi[1], and Salvatore Gaglio[1,2]

[1] DINFO - Dipartimento di ingegneria INFOrmatica,
University of Palermo,
Viale delle Scienze - 90128 Palermo - Italy
{vella, motisi}@csai.unipa.it, gaglio@unipa.it
[2] ICAR - Istituto di CAlcolo e Reti ad alte prestazioni,
Italian National Research Council,
Viale delle Scienze - 90128 Palermo - Italy
pilato@pa.icar.cnr.it

Abstract. This paper describes a technique for automatic creation of dictionaries using sub-symbolic representation of words in cross-language context. Semantic relationship among words of two languages is extracted from aligned bilingual text corpora. This feature is obtained applying the Latent Semantic Analysis technique to the matrices representing terms co-occurrences in aligned text fragments. The technique allows to find the "best translation" according to a properly defined geometric distance in an automatically created semantic space. Experiments show an interesting correctness of 95% obtained in the best case.

1 Introduction

Today, the large availability of multilingual repositories, as well as multingual web sites, allows to apply information retrieval algorithms not only to find information written in different languages but also to find semantic relation among terms belonging to different languages. The problem of information extraction from bilingual corpora has been addressed by many authors and the attention has been mainly aimed to the creation of trans-linguistic information retrieval systems [2][4][7][8].

In this paper a sub-symbolic approach for an automatic, data-driven, generation of dictionaries starting from bilingual parallel corpora is presented. This methodology can be useful for translating terms belonging to not common languages and/or when a specialized bilingual dictionary is required but not available. The proposed solution is divided in two phases. In the first phase a matrix of term co-occurrence is built and then processed to reduce the term relationship due to accidental co-occurrence. In the second phase, a technique commonly used to represent the semantics of words, named LSA[9], is applied to the processed matrix. The aim is to create a new sub symbolic representation of terms where only relevant connections are preserved while the particular information, relative to the finite number of examples in the corpus, is discarded.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 113–119, 2006.

The technique allows to find the "best translation" according to a properly defined geometric distance in an automatically created semantic space. Experiments show a correctness of 95% obtained in the best case.

The paper is organized as follows: Section 2 describes techniques for the generation of dictionary from bilingual corpora; in Section 3 some background knowledge about the Latent Semantic Analysis technique is recalled; in Section 4 the proposed approach for the creation of automatic dictionaries is presented. Finally in Sections 5 and 6 are given the results of the experiments and the conclusions.

2 Related Works

In [7] the application of Latent Semantic Analysis (LSA) [9] to Information Retrieval with Cross-Language documents has been presented.

A matrix has been formed considering as rows the terms of the two languages and as columns the bilingual paragraphs. This representation allows to put in relation documents with terms of the two languages and is used for the retrieval of documents presenting a query in a language and retrieving documents written in both languages. The corpus for the experiments was limited to 1500 bilingual documents and although the approach gives good results for information retrieval, not optimal words translation is produced[7].

Techniques for automatic dictionaries creation have been presented in [8] and [2]. In [8], the creation of a dictionary has been described starting from parallel documents retrieved on the web. After filtering and alignment of the documents, the dictionary has been built counting the co-occurrence of words in parallel sentences and applying the EMIM score[10].

In [2] the creation of a bilingual word-word dictionary for alignment during translation has been presented. The dictionary has been built considering the matrix of co-occurrence among words of the two languages. To extract the connections between words, the matrix has been filtered with symmetrical or asymmetrical thresholding to values taking into account a fixed threshold and the number of times the potential translation term is present in the corpus. The size of the dictionary has been determined by the value of thresholds. Higher thresholds allow to reduce spurious translations but conversely the size of the dictionary results decreased.

In [4] a geometric view of a technique based on PLSA [5] has been presented. The method is based on the matrix built with the context vectors of the couples (s_i, t_j) present in a given dictionary. The PLSA is applied to create terms clustering where synonyms appear in the same cluster and polysemous words appear in more than a cluster. As difference with the method proposed in this paper, the technique reported in [4] is based on a previous bilingual dictionary and aims to achieve a clustering for the dictionary words. Instead in the technique proposed in this paper the relation among word, the semantic clustering and the meaning are all extracted from the bilingual corpora applying a suitable technique based on Latent Semantic Analysis and an ad-hoc built geometric distance in a semantic space.

3 Latent Semantic Analysis

Latent Semantic Analysis (LSA) [9] is a paradigm to extract and represent the meaning of words by statistical computations applied to a large corpus of texts. LSA is based on the *vector space method*: given a text corpus of M words and N documents, a matrix A with rows associated to words and columns associated to documents is built. The content of the (i, j)-th cell of A is a function of the i-th word frequency in the j-th text; then the matrix A is replaced with a low-rank (k-dimension) approximation generated by the truncated singular-value decomposition (SVD) technique [9]:

$$A \approx \hat{A} = U \Sigma V^T \tag{1}$$

where U is the (Mxk) left singular matrix, Σ is (kxk) diagonal matrix with decreasing values $s_1 \geq s_2 \ldots \geq s_R > 0$ and V is the (Nxk) right singular matrix. U and V are column-orthogonal and so they both identify a basis to span the k dimensional space. Terms (represented as $u_i \Sigma$) are projected on the basis formed by the column vectors of the right singular matrix V and documents (represented as $v_j \Sigma$) are projected on the basis formed by the column of the matrix U to create their representation in the k-dimensional space.

4 The Proposed Approach

The aim of the proposed method is to generate a data driven dictionary starting from aligned bilingual corpora. The technique is based on the automatic detection of relationships between the terms of the two languages from the data contained in the corpora. The basic hypothesis is that aligned paragraphs convey the same meaning also if they are expressed with the lexicon and the syntax of their own language.

Data coming from the aligned bilingual parallel corpus are used to generate a matrix A that connects the terms of the language $\mathcal{L}_1$ with the terms of the language $\mathcal{L}_2$. Considering both the listing of the words of the language $\mathcal{L}_1$ and the language $\mathcal{L}_2$, the i-th row of A is associated to i-th word of the first language and the j-th column of A is associated to the j-th term of the second language. An aligned bilingual text fragment is a couple of aligned sentences in the two language excerpted from the corpus (i.e. each sentence is the translation of the other one present in the aligned bilingual text fragment). Let p_i be the probability of the presence of the word $w_{i_{\mathcal{L}_1}}$ in language $\mathcal{L}_1$, p_j the probability of the presence of the word $w_{j_{\mathcal{L}_2}}$ in the language $\mathcal{L}_2$ and $p_{i,j}$ the probability the word $w_{i_{\mathcal{L}_1}}$ and the word $w_{j_{\mathcal{L}_2}}$ are present in the aligned text fragments. The elements of A connect the terms of the two languages considering the probability $p_{i,j}$ minus the product of the single probabilities p_i, p_j of the single terms w_i and w_j:

$$a_{i,j} = \begin{cases} p_{i,j} - p_i \cdot p_j & \text{if } p_{i,j} - p_i \cdot p_j \geq th \\ 0 & \text{otherwise} \end{cases}$$

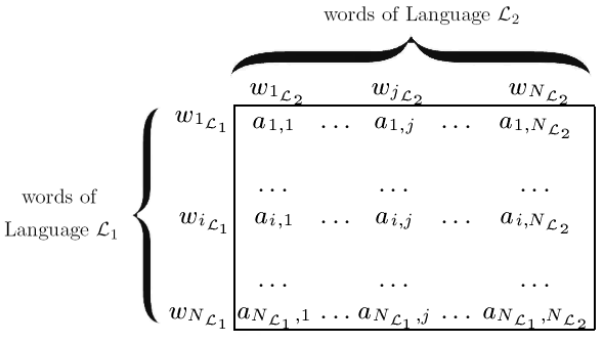


Fig. 1. Matrix of statistical connections among terms of the two languages

Values are filtered with an experimentally fixed threshold th to maintain only the strongest statistical connections.

The LSA technique is applied to the matrix A creating an approximation $\hat{A}$ that is its low-rank (k-dimension) approximation generated by the truncated singular-value decomposition (SVD) theorem (see eq. 1).

The application of LSA let emerge the semantic dependency between words. The result of this processing is the sub-symbolic representation of terms belonging to different languages in a k-dimensional vector space having the property that similar vectors are generated by their statistical presence in the same context. Thanks to the semantic properties of the vector space created with LSA, vector terms that have a little geometric distance and belong to the same language are synonyms while terms that have a little distance but belonging to different languages are one the translation of the other. The characteristic vectors of the words of the first language are the rows of

$$U_c = U\sqrt{\Sigma_k} \tag{2}$$

and in similar way are represented the words of the second language. The characteristic vectors represent the sub symbolic encoding words of the first or of the second language with a k-dimensional vector that takes into account its specific semantics.

To evaluate semantic relationship among terms in this space a metric has to be used. Considering that the components of the characteristic vectors can be interpreted as the belonging degree to sets related to semantic hidden features, it is sound to consider as distance between the sub-symbolic representation of terms a distance inspired to the *Tanimoto* measure [3] that takes into account the intersection and the union of two sets. The similarity measure, adapted to the representation of words vectors is:

$$\mathcal{D}_T \triangleq \frac{u_i \cdot v_j}{\|u_i\|_2 + \|v_j\|_2 - u_i \cdot v_j} \tag{3}$$

The more two terms are similar the higher will be the value of $\mathcal{D}_T$.

Starting from a characteristic vector for the term $w_{i_{\mathcal{L}_1}}$, the characteristic vector in language $\mathcal{L}_2$ producing the highest value of $\mathcal{D}_T$ will be the translation of $w_{i_{\mathcal{L}_1}}$. Analogously the characteristic vectors of terms in $\mathcal{L}_1$ producing higher values, will be the synonyms of $w_{i_{\mathcal{L}_1}}$.

5 Experimental Results

To test the above described approach, experiments on parallel bilingual corpora excerpted from a multi-lingual version of Bible [1] and from the proceedings of the European Parliament[6] gathering more than 750,000 text fragments, have been conducted. Paragraphs of Bible were aligned according to the verse of the text. The European Parliament parallel corpus was aligned automatically with the Church and Gale algorithm [6]. Terms in both corpora have been divided in four classes according to their frequency in the whole text to evaluate the approach with classes of words based on words frequency. Class 1 gathers terms with number of occurrences higher than 1000, Class 2 terms with occurrences between 1000 and 100, Class 3 between 100 and 10 and Class 4 less than 10.

The assessment has been done evaluating the right translation of words belonging to the four classes. For each term $w_{i_{\mathcal{L}_1}}$ the $\mathcal{D}_T$ (eq. 3) between its characteristic vector and the characteristic vectors of $w_{j_{\mathcal{L}_2}}$ is evaluated. The word with the highest $\mathcal{D}_T$ is associated to $w_{i_{\mathcal{L}_1}}$. The word is considered correctly translated if the winning word is among the real translations of the term according to a dictionary.

In table 1 are shown some translations of terms with the statistical method (without LSA) and the translation with application of LSA. It can be shown that terms retrieved with the proposed technique are correctly translated also when the statistical retrieved terms are wrong or only semantically related. The relationship found with the proposed technique are definitely more fitting to the

Table 1. Examples of translations

Translation Error without LSA	**Translation Error with LSA**
Stones	**Stones**
pietre	pietra
le	pietre
Descendants	**Descendants**
figli	discendenti
di	discendenza
Spring	**Spring**
acqua	fonte
fonte	sorgenti
Carried	**Carried**
i	portato
e	portò

Table 2. Percentage of Translation Error and Improvement

Class	Translation Error without LSA	Translation Error with LSA	Performance Improvement
Class 1	17%	5%	12%
Class 2	28%	9%	19%
Class 3	46%	37%	9%
Class 4	82%	76%	6%

meaning connection between terms than the simple semantic relationship given by occurrence in the same context.

In table 2 the results of experiments, related to the word classes, are shown. The first column is referred to terms accuracy when they are translated accordingly to the values in matrix A and considering only the term connections given by the statistical contingency, the other column shows results when translations are done applying LSA to the matrix A, the third column shows the performance improvement.

Other experiments have been conducted considering the translation accuracy for terms in relation to the $\mathcal{D}_T$. Term translations have been tested retaining only word vectors with $\mathcal{D}_T$ values bigger than an experimentally detected threshold to evaluate the sensitivity of the technique to distance thresholds. The values chosen for the thresholds were 0.3, 0.4 and 0.5. Translation of terms with $\mathcal{D}_T$ bigger than 0.3 have a high percentage of correct translations (above 90%) and when the threshold is raised to 0.5 all the translations are correct.

Experimental results highlight that the application of the proposed method gives a simple and fast method for the translation of words. The technique is particularly suitable for terms in the corpus belonging to Class 1 and 2 and gives acceptable results for words belonging to Class 3.

6 Conclusions

A sub-symbolic approach has been presented for the automatic creation of a bilingual dictionary from a corpus of documents. To the matrix of the statistical dependency between terms a LSA technique is applied allowing to create a semantic space where words with the same meaning are mapped in near positions. Experimental results show that a metric can be used to detect the correct translation of word if its number of occurrences is greater than a threshold. The data representation in vector space can be used to retrieve the translation words in the target language and to search for synonymys in the same language.

Acknowledgement

Authors would like to thank Giorgio Vassallo for helpful comments and suggestions.

References

1. The new american bible. Resources avalaible at : http://www.vatican.va/archive/bible/.
2. R.D. Brown. Automated dictionary extraction for "knowledge-free" example-based translation. In *Proc. of the 7th International Conference on Theoretical and Methodological Issues in Machine Translation*, 1997.
3. T. Tanimoto D. Rogers. A computer program for classifying plants. *Science*, 132, 1960.
4. E. Gaussier, J.-M. Renders, I. Matveeva, C. Goutte, and H. Djean. A geometric view on bilingual lexicon extraction from comparable corpora. In *ACL*, 2004.
5. T: Hofmann. Probabilistic latent semantic analysis. In *Proceedings UAI 1999*, pages 289–296, 1999.
6. P. Koehn. Europarl: A multilingual corpus for evaluation of machine translation. *Unpublished*, 2003. URL : http://people.csail.mit.edu/people/koehn/ publications/europarl/.
7. M. Littman, S. Dumais, and T. Landauer. Automatic cross-language information retrieval using latent semantic indexing. In G. Grefenstette, editor, *Cross Language Information Retrieval*. Kluwer, 1998.
8. Craig J. A. McEwan, Iadh Ounis, and Ian Ruthven. Building bilingual dictionaries from parallel web documents. In *Proc. of the 24 European Colloquium on Information Retrieval Research. LNCS*, 2002.
9. P.W. Foltz T.K.Landauer and D. Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25:259–284, 1998.
10. C.J. van Rijsbergen. *Information Retrieval*. 1999. http://www.dcs.gla.ac.uk/Keith/Preface.html.

An Automatic Feature Based Face Authentication System*,**

Stefano Arca, Paola Campadelli, Elena Casiraghi, and Raffaella Lanzarotti

Dipartimento di Scienze dell'Informazione,
Università degli Studi di Milano,
Via Comelico, 39/41 20135 Milano, Italy
{arca, campadelli, casiraghi, lanzarotti}@dsi.unimi.it

Abstract. In this paper a fully automatic face verification system is presented. A face is characterized by a vector (*jet*) of coefficients determined applying a bank of Gabor filters in correspondence to 19 facial fiducial points automatically localized. The identity claimed by a subject is accepted or rejected depending on a similarity measure computed among the jet characterizing the subject, and the ones corresponding to the subjects in the gallery. The performance of the system has been quantified according to the Lausanne evaluation protocol for authentication.

1 Introduction

Human face recognition has been largely investigated for the last two decades [13]. Within this context, we can define two specific tasks: authentication and identification [10]: the former aims to verify the identity declared by a subject on the basis of some biometric characteristics, the latter aims to recognize a person who does not declare his/her identity, but who is assumed to be one of the persons which constitute a referring gallery. Faces can be biometrically characterized in the same way for these two tasks; what differs is the comparison criterion: in the case of authentication, it is necessary to define an absolute threshold, while in the case of identification the identity of the test face is determined as the one of the gallery image which has the best match with the test itself.

In [2] we presented an identification system, which automatically localizes 19 fiducial points[1] and characterizes the face applying a bank of Gabor filters in correspondence to each fiducial point. In that case given a test image, the system computes its face characterization, and looks for in a gallery the subject who maximizes a suitably defined similarity function.

* Work partially supported by project "Acquisizione e compressione di Range Data e tecniche di modellazione 3D di volti da immagini", COFIN 2003.

** Work partially supported by the PASCAL Network of Excellence under EC grant no.506778. This pubblication only reflects the authors view.

[1] The eyebrow and chin vertices, the nose tip, the eye and lip corners and upper and lower middle points, the nose lateral extremes and the mean point between the eyes.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 120–126, 2006.

In this paper we present an authentication system in which the biometric characterization is similar to the one proposed in [2], while the identification is completely new. This authentication task corresponds to an "open-universe scenario" where persons unknown to the system may claim access. The subjects whose features are stored in the gallery are referred to as *Clients* while persons claiming false identity are called *Impostors*.

The whole system has been experimented on 744 images (of 186 subjects without glasses) taken from the XM2VTS [4]. We divided the database into two sets: the clients and the impostors. Moreover the set of the clients is divided in three subsets: the first constitutes the gallery (one image per subject), the second is used as client-evaluation set, and the third as the client-test set. The impostors set is divided in the impostor-evaluation and test sets respectively. Both the evaluation sets are used to establish the verification threshold.

The performance measures (Section 4) and the results (Section 5) are shown according to the Lausanne evaluation protocol presented in [6].

2 Localization of the Facial Features and of the Corresponding Fiducial Points

The first steps consist in detecting the face in the image and localizing the corresponding facial features (eyes, nose, mouth, and chin). In [1, 7] we proposed a scale-independent method which assumes the mouth is closed and the eyes are open and without glasses.

Given the feature sub-images, we proceed processing each of them separately, with the aim of extracting the most characteristic fiducial points. In [1] we presented a method to determine robustly and efficiently the fiducial points associated to the eyebrows, the nose and the chin; regarding the eyes and the mouth we adopted the deformable template technique which estimates the whole features contour, but which is computationally very expensive. In [2] we proposed an efficient alternative for the eyes, based on the analysis of the edges obtained by means of the first derivative of Gaussian filters, while for the mouth we considered the mouth corners used for the template initialization and we derived the upper and lower middle points as a function of them. Moreover, we proposed a module able to recognize automatically which fiducial points have been

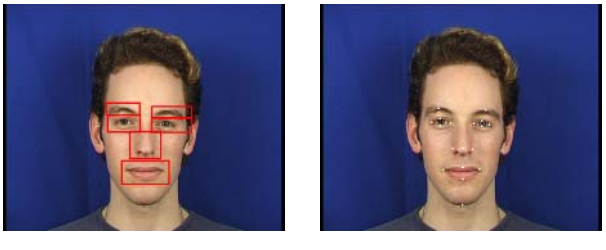


Fig. 1. Features sub-images and fiducial points

wrongly determined, and which recovers them on the basis of the positions and dimensions of the reliable features.

These modules, applied to the 744 images of the XM2VTS database, determine a good estimation of the fiducial points in the 97.5%. An example of the obtained results is shown in figure 1.

3 Face Characterization

Once the fiducial points have been extracted, we proceed characterizing each of them in terms of the surrounding gray level portion of image. Following the idea of Wiskott [12], to characterize a fiducial point, we convolve the portion of gray image around it with the following bank of *Gabor kernels*:

$$\psi_j(\boldsymbol{x}) = \frac{k_j^2}{\sigma^2} exp\left(-\frac{k_j^2 x^2}{2\sigma^2}\right)\left[exp(i\boldsymbol{k}_j\boldsymbol{x}) - exp\left(-\frac{\sigma^2}{2}\right)\right]$$

in the shape of plane waves with wave vector $\boldsymbol{k}_j$, restricted by a Gaussian envelope function. We employ a discrete set of 5 different frequencies, index $\nu = 0, ..., 4$, and 8 orientations, index $\mu = 0, ..., 7$,

$$\boldsymbol{k}_j = \begin{pmatrix} k_{jx} \\ k_{jy} \end{pmatrix} = \begin{pmatrix} k_\nu \cos\varphi_\mu \\ k_\nu \sin\varphi_\mu \end{pmatrix}, k_\nu = 2^{-\frac{\nu+2}{2}}\pi, \varphi_\mu = \mu\frac{\pi}{8}$$

with index $j = \mu + 8\nu$. The width σ/k of the Gaussian is controlled by the parameter $\sigma = 2\pi$. We observe that the kernels are DC-*free*, that is $\int \psi_j(\boldsymbol{x})d^2\boldsymbol{x} = 0$ allowing to deal with different illumination conditions.

The obtained 40 coefficients are complex numbers. A *Jet* J is obtained considering the magnitude parts only.

Applying the Gabor wavelet transform to all the facial fiducial points, we obtain the face characterization, consisting in a *jets vector* of 40×19 real coefficients.

Thus, given a test image T to be authenticated, we are interested in determining a similarity score between it and the image in the gallery corresponding to the claimed identity. To determine this score we propose a method which requires to take into account all the images as follows:

- for each fiducial point i, and for each image $k \in G$, compute the similarity between corresponding Jets:

$$S^{k,i} = S(J^{T,i}, J^{k,i}) = \frac{\sum_z J_z^{T,i} J_z^{k,i}}{\sqrt{\sum_z (J_z^{T,i})^2 \sum_z (J_z^{k,i})^2}}$$

  where $z = 0, ..., 39$.
- for each i, order the values $\{S^{k,i}\}$, and assign to each a weight $w^{k,i}$ as a function of its ordered position p.
  The weight $w^{k,i} = f(p)$ is determined as:

$$f(p) = c \cdot [\ln(x + y) - \ln(x + p)],$$

  where $y = \frac{|G|}{4}$, $x = e^{-\frac{1}{2}}$, and c is a normalization factor.

– for the gallery image A corresponding to the claimed identity, consider the set, *Best10*, of the 10 fiducial points which have got the highest weights, and determine the score:

$$\text{Score}(A) = \sum_{i \in Best10} w^{A,i} S^{A,i}.$$

If the score is greater than a threshold th the subject is authenticated otherwise he/she is rejected.

It is clear that the algorithm performance strongly depends on the value given to the threshold th. A too high value of th would indeed make difficult the access to the impostors but at the same time would reject a lot of clients; on the other hand a too low value of th would increase the detection rate of the clients, together with the number of the impostors accepted.

In order to set the value of th which makes the global error low, we studied the behavior of the authentication system on the evaluation sets varying the value of th as presented in the following section.

4 Performance Measure

To evaluate the performance of the verification system, two measures are used: the *false acceptance* (FA) and *false rejection* (FR) rates. False acceptance is the case when an impostor, claiming the identity of a client, is accepted. In contrast, false rejection is the case when a client, claiming his true identity, is rejected. The FA and FR rates are given by:

$$FA = \frac{EI}{I} \qquad FR = \frac{EC}{C} \tag{1}$$

where EI is the number of impostor accepted, I is the number of impostors trials, EC is the number of client rejected and C is the number of clients trials.

In order to set a reliable threshold, we carried out several experiments, testing the system with thresholds normalized in the range $[1 - 100]$. This step results in the ROC plotted in figure 2.

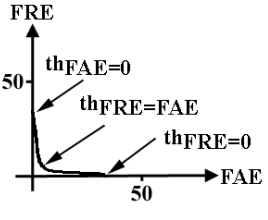


Fig. 2. ROC Curve

Starting from this curve, we focus the attention on four thresholds, which are chosen since on the evaluation data they allow to obtain desired values of the false acceptance (FAE) and false rejection (FRE) values:

$$
\begin{aligned}
th_{FAE=0} &= argmin_{th}\,(FRE|FAE=0)\\
th_{FAE=FRE} &= (T|FAE=FRE)\\
th_{FRE=0} &= argmin_{th}\,(FAE|FRE=0)\\
th_{sum} &= argmin_{th}\,(FRE+FAE)
\end{aligned}
\tag{2}
$$

This will lead to obtain 8 scores on the test set:

$$
\begin{array}{ll}
FA_{FAE=0} & FR_{FAE=0}\\
FA_{FAE=FRE} & FR_{FAE=FRE}\\
FA_{FRE=0} & FR_{FRE=0}\\
FA_{sum} & FR_{sum}
\end{array}
$$

For each threshold, the *weighted error rate* (TER) can be obtained as follows:

$$
\begin{aligned}
WE_{FAE=0} &= \omega_{FA}\cdot FA_{FAE=0}+\omega_{FR}\cdot FR_{FAE=0}\\
WE_{FAE=FRE} &= \omega_{FA}\cdot FA_{FAE=FRE}+\omega_{FR}\cdot FR_{FAE=FRE}\\
WE_{FRE=0} &= \omega_{FA}\cdot FA_{FRE=0}+\omega_{FR}\cdot FR_{FRE=0}\\
WE_{sum} &= \omega_{FA}\cdot FA_{sum}+\omega_{FR}\cdot FR_{sum}
\end{aligned}
\tag{3}
$$

The weights ω_{FA} and ω_{FR} are set depending on the relative importance of the false acceptance and rejection rates. If a general face verification system is used, we can weight the error rates equally, $\omega_{FA}=0.5$ and $\omega_{FR}=0.5$.

5 Experimental Results

We report here the experiments carried out on the subset of the XM2VTS database consisting of all the 744 images of 186 subjects without glasses (4 shots per subject). The experimental setup is shown in figure 3: the images in the database are divided to form two sets (*clients* and *impostors*).

Specifically, we randomly selected as clients 100 subjects; the remaining 86 subjects form the set of the *impostors*.

The clients' gallery (100 images) is composed by selecting for each client the first shot. The clients' evaluation set is composed by the second shot of each client; the remaining 2 images for each client, form the clients' test set. The impostors' test set is composed of all the shots of 50 randomly selected impostors;

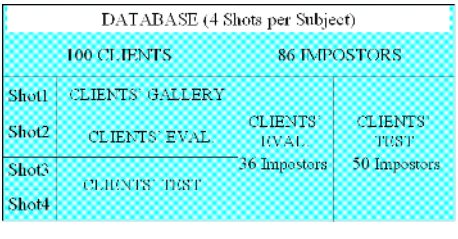


Fig. 3. Experimental setup

Table 1. Authentication results

Experimental Results						
Thresholds	**Evaluation**			**Test**		
	% FAE	**% FRE**	**% WE**	**% FA**	**% FR**	**% WE**
$th_{FAE=0}$	0	30	15	0	35.41	17.71
$th_{FAE=FRE}$	2.17	2	2.08	2.14	3.64	2.89
$th_{FRE=0}$	40.93	0	20.46	41.46	0	20.73
th_{sum}	1.78	2	1.89	1.82	3.64	2.73

while the impostors' evaluation set is composed of the remaining images of the other 36 impostors.

With this setting we obtained the results shown in table 1. For each threshold, obtained in section 4, we show the FA, FR, and the weighted error rate, WE, with respect to the evaluation and the test sets.

The best results have been obtained using the threshold t_{sum} which minimizes the weighted error WE; note that in general there is a close agreement between the results obtained on the evaluation and test sets which shows that the selected thresholds generalize well.

6 Discussion

We presented a completely automatic system for face authentication. The method is based on a module for the feature extraction and description, which is self-correcting, and determines with high reliability the correct fiducial points. Moreover it is robust to illumination and scale variations. The authentication step computes for each probe image a score on the basis of both its characteristic jets vector, and the ones in the gallery.

Since our feature detection works on images of people without glasses, we could make the experiments only on a subset of the XM2VTSDB. On this set the performance we achieve are comparable with the ones obtained in the competition reported in [9]. However our system does not require any training session and any registration (usually manually done), which are two fundamental steps of methods based on LDA [8], SVM [5, 11], Multi Layer Perceptrons and Gaussian Mixture Model [3].

Further works aim to detect additional fiducial points in order to extract further information characterizing the faces, and to deal with people wearing glasses; finally we intend to experiment and compare alternative methods to compute the scores.

References

1. S. Arca, P. Campadelli, and R. Lanzarotti. A face recognition system based on local feature analysis. *Proceedings of the International Conference on Audio- and Video-based Biometric Person Authentication, AVBPA2003 Guildford,UK published in the Lecture Notes in Computer Science*, 2688:182–189, 2003.

2. S. Arca, P. Campadelli, and R. Lanzarotti. An efficient method to detect facial fiducial points for face recognition. *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004 Cambridge,UK*, 2004.
3. F. Cardinaux, C. Sanderson, and S. Marcel. Comparison of an mlp and gmm classifiers for face verification on xm2vts. *Proceedings AVBPA 2003*, pages 911–920, 2003.
4. The XM2VTS Database. Web address:http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb/. 2001.
5. K. Jonsson, J. Kittler, Y.P. Li, and J. Matas. Support vector machines for face authentication. *Image and Vision Computing*, 20:369–375, 2002.
6. J.Kittler J.Luettin K.Messer, J.Matas and G.Maitre. Xm2vtsdb: the extended m2vts database,. *Proceedings AVBPA 99*, 1999.
7. R. Lanzarotti. *Facial feature detection and description.* PhD thesis, Università degli Studi di Milano, 2003.
8. H. Liu, C. Su, Y. Chiang, and Y. Hung. Personalized face verification system using owner-specific cluster-dependent lda-subspace. *Proceedings of International Conference on Pattern Recognition (ICPR04)*, 2004.
9. K. Messer, J. Kittler, M. Sadeghi, and al. Face verification competition on the xm2vts database. *Proceedings AVBPA 2003*, pages 964–974, 2003.
10. J. Phillips, P. Grother, R. Micheals, D.M. Blackburn, E. Tabassi, and J.M. Bone. Face recognition vendor test 2002: overview and summary. *[Online], Available: http://www.biometricsinstitute.org/bi/*, 2003.
11. F. Smeraldi and J. Bigun. Retinal vision applied to facial features detection and face authentication. *Pattern recognition letters*, 23:463–475, 2002.
12. L. Wiskott, J. Fellous, N. Kruger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. In L.C. Jain et al., editor, *Intelligent biometric techniques in fingerprints and face recognition*, pages 355–396. CRC Press, 1999.
13. W. Zhao, R. Chellappa, P.J. Phillips, and A.Rosenfeld. Face recognition: A literature survey. *ACM, Computing Surveys*, 35(4):399–458, 2003.

On the Preprocessing of Mass Spectrometry Proteomics Data

M. Cannataro[1], P. H. Guzzi[1], T. Mazza[1], G. Tradigo[2], and P. Veltri[2]

[1] Magna Græcia University of Catanzaro, Italy
[2] ICAR-CNR, Rende, Italy
{cannataro, hguzzi, t.mazza, veltri}@unicz.it,
gtradigo@si.deis.unical.it

Abstract. Mass-Spectrometry (MS) based biological analysis is a powerful approach for discovering novel biomarkers or identifying patterns and associations in biological samples. Each value of a spectrum is composed of two measurements, m/Z (mass to charge ratio) and intensity. Even if data produced by mass spectrometers contains potentially huge amount of information, data are often affected by errors and noise due to sample preparation and instrument approximation. Preprocessing consists of (possibly) eliminating noise from spectra and identifying significant values (peaks). Preprocessing techniques need to be applied before performing analysis: cleaned spectra may then be analyzed by using data mining techniques or can be compared with known spectra in databases. This paper surveys different techniques for spectra preprocessing, working either on a single spectrum, or on an entire data set. We analyze preprocessing techniques aiming to correct intensity and m/Z values in order to: (i) reduce noise, (ii) reduce amount of data, and (iii) make spectra comparable.

Keywords: Mass Spectrometry, Data preprocessing, Data cleaning.

1 Introduction

Mass Spectrometry (MS) based analysis is becoming a powerful, widely used technique in order to identify different molecular targets in different pathological conditions. MS experiments involve different and heterogeneous technological platforms thus that a clear understanding of the function and errors related to each one has to be taken into account. In particular, data produced by mass spectrometer are affected by errors and noise due to sample preparation, sample insertion into the instrument and instrument itself. Mass spectrometry-based experiments usually comprise a data generation phase, a data preprocessing phase and a data analysis phase. Analysis, whose goal is to extract information from data sets, is usually performed by using data mining algorithms, pattern extraction or peptide/protein identification. Latter phase may be performed using neural networks, also used for classification and for information extraction [5]. Data mining algorithm as well as neural networks, are always preceded by preprocessing phases. Our experience consists in considering processing phases that prepare data sample for such analysis. Indeed, each spectrum contains a very large set of measures (*m/Z, intensity*), representing the abundance (intensity) of compounds having certain mass to charge ratio (m/Z) values that have to be collected, managed and stored efficiently. In

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 127–131, 2006.

particular, MS experiments have been largely used in proteomics experiments. In this paper, after introducing Mass Spectrometry, we survey different techniques for spectra preprocessing.

2 Mass Spectrometry Data

The mass spectrometer is an instrument designed to separate gas phase ions according to their *m/Z* (mass to charge ratio) values. E.g., Matrix-Assisted Laser Desorption / Ionization - Time Of Flight Mass Spectrometry (MALDI-TOF MS) is a relatively novel technique that is used for detection and characterization of biomolecules, such as proteins, peptides, oligosaccharides and oligonucleotides, with molecular masses between 400 and 350000 Da. The Mass Spectrometry process can be decomposed in three subphases: (i) Sample Preparation (e.g., Cell Culture, Tissue, Serum in proteomics); (ii) Mass Spectrometry processing, and (iii) Data extraction. Mass Spectrometry output is represented, at a first stage, as a (large) sequence of value pairs, where each pair contains a measured *intensity*, which depends on the quantity of the detected mass and a mass to charge ratio (*m/Z*), which depends on the molecular mass detected. E.g., Figure 1 shows the spectrum of a real sample in a proteomics experiment containing *insuline* (peak m/Z=5736,85) and *mioglobine* (peak m/Z=8688,14).

A spectrum may be affected by (i) noise, (ii) peak broadening, (iii) instrument distortion and saturation, (iv) isotopes, (v) miscalibration, (vi) contaminants of different origins. Data cleaning is performed in different phases by using: (i) best-practices sample preparation; (ii) mass spectrometer software; (iii) further data pre-processing algorithms. In the rest of the paper we focus on pre-processing techniques conducted after data have been produced and eventually cleaned by the spectrometer.

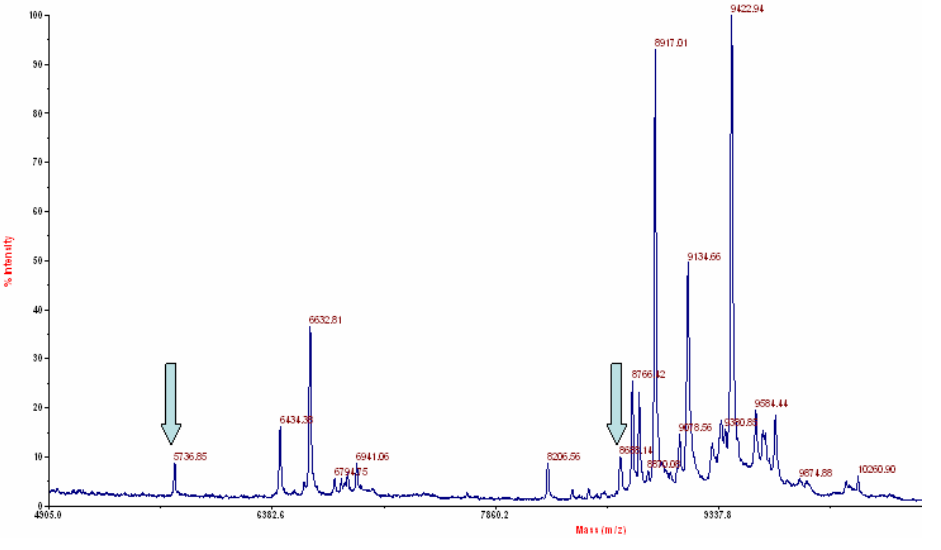


Fig. 1. Mass Spectrum of a Biological Sample

3 Preprocessing Mass Spectrometry Data

Preprocessing [6] consists of spectrum noise and contaminants *cleaning up*. Is is also used to dimensionally reduce spectra by using efficient algorithms that highlight spectra properties, without affecting biological properties. Moreover, preprocessing modifies spectra intensity and m/Z values in order to allow further manipulation and data extraction.

3.1 Noise Reduction and Normalization

Noise reduction and normalization are conducted in part by the spectrometer and in part by external preprocessing tools. In the following we describe some approaches to noise reduction and normalization.

Base line subtraction and smoothing. Such techniques are used for noise reduction. Base line subtraction flattens the base profile of a spectrum while smoothing reduces the noise level in the whole spectrum.

Base line subtraction. Each spectrum contains a base intensity level (baseline) varying through the m/z axis (see Figure 2). Such base line can be considered as a frontier between significant peaks and noise. The algorithm aims to identify such base line to flatten spectrum profile eliminating noise. It iteratively calculates the best fit straight line through a set of estimated baseline points. For each iteration the number of points across the line are counted. If the number of points above the line are less than below, they are peaks (and not noise), thus a new base line is evaluated. This process is repeated until the difference between number of points above and below the line is nearly zero. The line is then subtracted from the spectrum to get the *baseline* corrected spectrum.

Smoothing is used to increase signal to noise ratio. Each point in a spectrum is considered as an average value of its neighbors [4] [8] [6]. The most used technique consists of evaluating equidistant points and is named *moving average*. Given m the number of spectra values, an array of raw (noisy) data $[y_1, y_2, ..., y_m]$ is converted to a new array of smoothed data, where each "smoothed point" (y'_k) represents the average between $2n + 1$ consecutive points of the raw data, with n being a smoothing fixed parameter. Note that the odd number $2n + 1$ is usually named filter width.

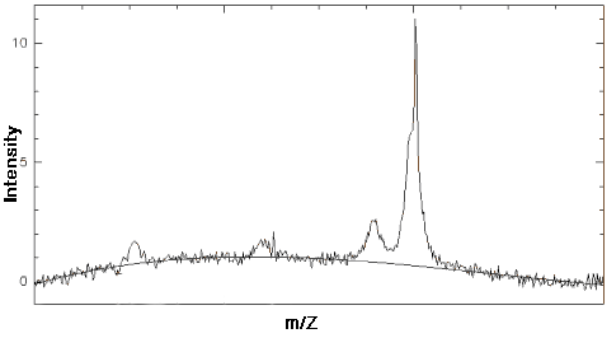


Fig. 2. Base Line Subtraction

Normalization of Intensities. Normalization consists in normalizing spectra allowing comparisons. Indeed, spectra obtained using different instruments or different calibration, need to be calibrated with respect to a common scale factor mapping each spectrum to a common set of representative values. The goal is to identify and remove sources of systematic variation between spectra due for instance to varying amounts of sample or time degradation in a sample. There are several normalization techniques, among them: the *Canonical Normalization*, the *Inverse Normalization* [3], the *Direct Normalization*, and the *Logarithmic Normalization* [9] [10].

3.2 Data Reduction

Data reduction consists of a set of techniques aiming to reduce a spectrum in a reduced number of significant values, providing a more suitable spectrum. We here report on the binning technique for space limitation.

Binning is one of the most used preprocessing technique in MS data analysis. Its aim is to perform a dimensional reduction preserving information for further processing and mining phases. Given a set of size-variable windows (called bins) defined on the m/z axis, the binning consists in associating all (m/z,intensity) values contained in each window to a representative value. We briefly sketch binning algorithm. Given a subset of N peaks in a spectrum, represented by the couples $[(I_1, m/Z_1), (I_2, m/Z_1)$ ..., $(I_N, m/Z_N)]$ contained in a window shift, binning consists in associating to them a single value $(I, m/Z)$, where *I* is function of original intensities (e.g., their sum), and *m/Z* represents a mass value (e.g. the median value). Such procedure is conducted by scanning the whole spectrum by using a sliding window [3].

3.3 Identification and Peaks Extraction

Peaks extraction consists of separating peaks corresponding to peptides, called real peaks, from peaks representing noise. Although such task can be performed by using the data-processing methods furnished by mass spectrometer software, the most efficient method is still semi-automatic, where biologist experience has to fit with automatic methods. Some methods for automatic peaks extraction have been proposed: e.g. spectral polygonalization consists of identifying relevant peaks and calculating their associated area [7]. Such method is based on a *time-series* segmentation routine that reduces data set to groups of *three strategic points* where each group defines the beginning, the medium and the ending of each located peak. Peaks with statistically insignificant height or area are then discarded [1] [7].

3.4 Peaks Alignment

To compare different spectra samples, it is necessary to align samples w.r.t. m/Z axis. Such process is known as data-calibration or *alignment* of peaks among different samples. With no alignment, peaks referring to similar biological sample (e.g. the same peptide) can have different values of m/Z across samples. To allow an easy and effective comparison of different spectra, peaks alignment methods identify a common set of peak locations (i.e. m/Z values) in a set of spectra, in such a way that all spectra have common m/Z values for the same biological entities. Once such values have been fixed, for each spectrum, intensity values are shifted with a window shift tolerance [10].

4 Conclusions and Future Work

Preprocessing algorithms described so far have been implemented and tested on different mass spectra data sets obtained in a biomedical project [2]. They are the building blocks of MS-Analyzer, a software platform for the acquisition, preprocessing, storing and analysis of mass spectrometry proteomics data.

References

1. D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its carature. *The Canadian Cartographer*, 10:112–122, 1973.
2. M. Cannataro et al. Mass Spectrometry Data Analysis for Early Detection of Inherited Breast Cancer. In *WIRN 2004, CIBB Workshop*, 2004.
3. V. Gopalakrishnan, E. William, S. Ranganathan, R. Bowser, M. E. Cudkowic, M. Novelli, W. Lattazi, A. Gambotto, and B. W. Day. Proteomic data mining challenges in identification of disease-specific biomarkers from variable resolution mass spectra. In *Proceedings of SIAM Bioinformatics Workshop 2004*, pages 1–10, Lake Buena Vista, FL, April 2004.
4. K. Herath. Effects of 'matched filter' smoothing as measured by receiver operating characteristic curve. *Phys. Med. Biol.*, 21:442–446, 1976.
5. T. Stepinski, L.Ericsson, B. Vagnhammar, and M. Gustafsson. Neural Network Based Classifier for Ultrasonic Resonance Spectra. *NDT.net*, 3(12), Dec 1998.
6. M. Wagner, D. Naik, and A. Pothen. Protocols for disease classification from mass spectrometry data. *Proteomics*, 3(9):1692–8, 2003.
7. W. Wallace, A. Kearsley, and C. Guttman. An operator-independent approach to mass spectral peak identification and integration. *Analytical Chemistry*, 76:2446–2452, 2004.
8. K. J. Worsley, S. Marrett, P. Neelin, and A. C. Evans. Searching scale space for activation in pet images. *Human Brain Mapping*, (4):74–90, 1996.
9. B. Wu, T. Abbott, D. Fishman, W. McMurray, G. Mor, K. Stone, D. Ward, K. Williams, and H. Zhao. Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. *Bioinformatics*, 1(19):1636–43, September 2003.
10. Y. Yasui, D. McLerran, BL. Adam, M. Winget, M. Thornquist, and Z. Feng. An automated peak identification/calibration procedure for high-dimensional protein measures from mass spectrometers. *Journal of Biomedicine and Biotechnology*, (4):242–248, 2003.

Clustering Causal Relationships in Genes Expression Data

Sergio Pozzi, Italo Zoppis, and Giancarlo Mauri

DISCo, Univ. Milano-Bicocca, Via Bicocca degli Arcimboldi 8,
20126 - Milano, Italy
{sergio.pozzi, zoppis}@disco.unimib.it,
giancarlo.mauri@unimib.it

Abstract. In this paper we apply a strategy to cluster gene expression data. In order to identify causal relationships among genes, we apply a pruning procedure [Chen et al., 1999] on the basis of the statistical cross-correlation function between couples of genes' time series. Finally we try to isolate genes' patterns in groups with positive causal relationships within groups and negative causal relation among groups. With this aim, we use a simple recursive clustering algorithm [Ailon et al., 2005].

Introduction

DNA microarray technology [Eisen and Brown, 1999] enables researchers to simultaneously measure the expression level for thousands of genes in different experimental conditions. To deal with the huge amounts of data obtained from these experiments, machine learning methods and in particular cluster methods are frequently used as tools of analysis (see for for example [Eisen et al., 1998]). Behind the application of a specific method, lies the ambitious realm of genetic networks construction that model how genes interact and regulate each other [Thieffry and Thomas, 1998, Liang S. and R., 1998].

Although various clustering algorithms can usefully organize sets of genes that have similar patterns, it still remains difficult to aggregate the resulting groups in structured links of activation/inhibition relationships. In this paper, we apply a strategy to group genes that seem to exhibit an activation regulatory expression and to link groups where the activity seems to be inhibitory. Our aim is both to simplify the network structure (clusters are used as vertex, in section 4) and to maximize the homogeneity of genes patterns that fall in a specific group.

The clustering problem has been modeled as a combinatorial optimization problem on graphs [Bansal, 2002]. These combinatorial approaches are NP-Hard [Bansal, 2002], but efficient polynomial time approximation schemes have been proposed [Chaitanya, 2004] [Charikar et al., 2003]. The major of them use the Williamson and Göemans technique [Göemans et al., 1995] of rounding a relaxed version of the problem. More recently, a new simple probabilistic approximation algorithm for the *Correlation Clustering Problem* has been proposed [Ailon et al., 2005]. We follow this approach in section 2.

In order to capture meaningful inferences over the course of phenotypic change we analyze time series data; specifically, we use the patterns of the budding yeast *Saccharomyces Cerevisiae* measured at 17 time step of the cell division cycle.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 132–139, 2006.

1 A Causality Model

Considering the nature of the activity of the gene processes as essentially causal, we have to admit that every gene must have one or more activators. An activator could be some sort of a biochemical signal which is responsible for the start of the transcription process of the gene. The expression level of a gene without the presence of an activator can be thought as a low level signal eventually affected by noise. As some biochemical signals could turn the gene off we have to consider also the presence of some inhibitors as well. These inhibitors could turn the gene off even if some activators counterparts are active. A Gene Regulation Network then could be viewed as a complicated network of genes and products of genes themselves that activate or inhibit other genes or products of genes. The model illustrated below [Chen et al., 1999] tries to organize, on a graph, the correlation relationships among genes coming from the preprocessing task described in section 3. Once the graph representing the supposed correlation relationships among genes is built, a pruning procedure starts. This procedure is intended to solve a combinatorial optimization problem in order to identify a subset of the genes which act as the "true" regulatory elements by deleting edges associated to spurious (not causal) relationships. The objective function embodied in the combinatorial optimization problem is naturally suggested by logical and biological considerations:

- the genes involved in a particular pathway of a cell process are of two types: activation/inhibition. These types are mutually exclusive with very few exceptions. It turns out that one of the requirements that we must expect the deleting procedure to have is to output an activation/inhibition label for the nodes of the reference graph. A direct consequence is that the labeling of nodes must be consistent with the edge labels. By deleting edge, the procedure simply has to prevent an activating/inhibiting node to be the source of a inhibiting/activating edge respectively.
- the deleting procedure has to output a graph which achieves the maximum number of nodes with both activating/inhibiting incoming edges. If a gene can be both activated and inhibited, than it can be *controlled*. What the procedure must search for is a gene network in which the number of controlled genes are maximized: the parsimonious principle.

These objective criteria give rise to a new interesting problem on graphs: *The Maximum Gene Regulation Problem*, shortly MGRN.

Given a directed graph with (A/I) labeled edges, assign each vertex either an A or I label so as to maximize the number of vertices with both A and I labeled input edges, after deleting all edges whose label differs from its parent vertex.

The problem is NP-Hard [Chen et al., 1999] so the attention is shifted toward approximation algorithms. The best algorithm known is based on the Göemans and Williamson approximation scheme and it achieves a ratio of 1/2 [Chen et al., 1999] while it is unlikely the existence of a schema since, as proved in [Pozzi et al., 2005], no polynomial time approximation algorithm with performance ratio better than $1 - \frac{1}{8(1+e^2)}$ exists unless NP=RP.

For the approximation algorithm the 1/2 performance ratio is achieved by expressing the NP-Hard problem as a integer linear programming instance. Then relaxing to the

continuous the linear programming problem, solving it and using the so obtained result to find a suitable solution for the original discrete problem. For each vertex v_j of the graph representing the instance of the MGRN problem we will associate a boolean formula C_j. We will denote as C_j^+ and C_j^- the set of subscripts of the vertices that are connected to v_j labeled as activator and inhibitor respectively. For every vertex v_j, let x_j represents a boolean variable which is true if and only if the v_j vertex is labeled as activator. Then, the boolean formula C_j, mentioned above, is: $C_j = \left(\bigvee_{i \in C_j^+} x_i \right) \wedge \left(\bigvee_{i \in C_j^-} \neg x_i \right)$.

As it can be easily verified the boolean expression C_j is satisfied if and only if the vertex v_j is controlled. Let z_j represents a boolean variable which is true if and only if the vertex v_j is controlled, then, for the MGRN problem, the following linear programming formulation can be given:

$$\max_{(\mathbf{x},\mathbf{z})} \sum_j z_j$$
$$\begin{cases} z_j \leq \sum_{i \in C_j^+} x_i & \text{for all} j \\ z_j \leq \sum_{i \in C_j^-} (1 - x_i) & \text{for all} j \\ x_i \in \{0,1\} & \text{for all} i \\ z_j \in \{0,1\} & \text{for all} j \end{cases} \tag{1}$$

As it can be seen, this formulation is similar to that of the MAXSAT problem: if we want to maximize the number of controlled vertices we have to maximize the summation $\sum_j z_j$: the objective function of (1). Let us consider a vertex v_j and the correspondent boolean variable z_j. If, for instance, we have $z_j = 1$, then the first two constraints in (1) become: $\sum_{i \in C_j^+} x_i \geq 1$ and $\sum_{i \in C_j^-} (1 - x_i) \geq 1$. In order for these two constraints to be simultaneously satisfied, at least one of the boolean variables x_i associated to the vertices in C_j^+ must be assigned to 1 and, at least one of the boolean variables associated to the vertices in C_j^- must be assigned to 0. That is, the v_j vertex results controlled. We can wonder if the required constraint of the uniqueness of the activation/inhibition behavior for a gene is correctly modeled in (1). The answer is positive since, let us suppose that an activation assignment to variable x_k is obtained ($x_k = 1$). Then let us assume there existed an inhibition edge from the correspondent vertex v_k. As it can be noticed, the value of the expression $(1 - x_k)$ in the second constraint of (1) is zero and thus the contribution of that inhibition edge can be ignored: $\sum_{i \in C_j^-} (1 - x_i) = \sum_{i \in C_j^- - \{k\}} (1 - x_i)$. A similar deduction can be done for the case $x_i = 0$. Of course, solving the integer linear programming problem (1) is NP-Hard. But if we let the variables to assume continuous values we can solve it, in polynomial time, using, for instance, the Karmakar algorithm. Finally, as shown in [Chen et al., 1999], on the basis of the relaxed solution, a discrete one can be achieved by means of a stochastic rounding procedure whose expected measure is not worse than one half of the optimal one.

2 Clustering

In the approach we use, starting from the relationships between couples of genes, which the MGRN paradigm suggests, we look for an optimal partition of genes. The optimality

Table 1. The $CC - PIVOT$ algorithm

$CC - PIVOT(G = (V, E^+ \cup E^-))$

Pick a random pivot $i \in V$
Set $C = \{i\}, V' =$

For all $j \in V, j \neq i$
If $(i, j) \in E^+$ then
Add j to C
else (If $(i, j) \in E^-$)
Add j to V'

Let G' be the subgraph induced by V'

Return clustering C,$CC - PIVOT(G')$

is with respect to a particular homogeneity measure. The problem is identified as the *Correlation Clustering Problem*. The goal is to find a partition so as to minimize the number of negative correlated elements within the same cluster and positive correlated elements crossing clusters. The *Correlation Clustering Problem* has been investigated in both the minimization version and maximization one. On complete graphs, a 4 factor approximation algorithm was known which is based on rounding a linear program formulation [Charikar et al., 2003]. A simple recursive algorithm of better approximation performance has been proposed in [Ailon et al., 2005]. In that paper the approximation factor has been improved to 3. This is the algorithm we adopt in our clustering approach. More formally, an instance of the *Correlation Clustering Problem* is specified giving an undirected graph (V, E). Between any two unordered $i, j \in V$, we either have a $(+)$ or a $(-)$ relation. We denote the set of pairs $i \neq j$ which are $(+)$-related ($(-)$-related) as E^+ (E^-) so that $E = E^+ \cup E^-$. The goal of the problem is to find a partition of V in clusters $C_1, \ldots, C_m$ minimizing the number of disagreement pairs ($(+)$ pairs in different clusters or $(-)$ pairs in the same cluster). The algorithm we use (see Table 1), was introduced in [Ailon et al., 2005] and is called the $CC - PIVOT$ algorithm. It has a very simple recursive structure. It can efficiently produce a partition of elements in linear time. It starts by randomly choosing an element i, adding it to a new cluster C. Then it includes, in the same cluster, all the other elements which are $(+)$-related with i. Eventually the procedure continues recursively with the remained elements not already clustered.

3 Data Preprocessing

The representation of a microarray is generally an $n \times m$ *expression matrix* $\mathbf{X}$ with the n rows corresponding to genes and the m columns corresponding to different conditions or different time points. This expression matrix $\mathbf{X}$ represents intensities of hybridization signals as provided by a DNA array. In our case, the element $x_{i,j}$ is the *expression*

level of gene i at time j; the entire ith row of the expression matrix is generally called *expression pattern* of gene i.

In agreement with [Chen et al., 1999], we filtered out both patterns whose expression levels were below a detection threshold (≤ 200) and patterns expressed but without significant variation over time, for instance, maximum $x_i^\star$ and average $\tilde{x}_i$ expression level of gene i satisfy $(x_i^\star - \tilde{x}_i)/\tilde{x}_i \leq 0.1$.

For all pairs of selected genes x and y, we evaluate the cross-correlation function estimation:

$$r_{xy}(k) = \sum_{t=1}^{N-k} \frac{(x_t - \tilde{x})(y_{k+t} - \tilde{y})}{\sqrt{\sum_{t=1}^{N-k}(x_t - \tilde{x})^2}\sqrt{\sum_{t=1}^{N-k}(y_{k+t} - \tilde{y})^2}} \tag{2}$$

where k is the time delay (lag) and $\tilde{x}$ and $\tilde{y}$ the respective mean values. In (2) we keep only those positive correlation at one lag head ($k = +1$). The analyzed patterns are, in fact, sampled at 10-minute intervals: by applying this filtration we do not take into account both those correlations across series which are too delayed, and those which are not significant.

Since high absolute values of cross-correlation between couple of variables can't provide proof of a causal relationship, we delete some cross-correlation relationships between couple of genes' patterns using the approach described in section 1.

4 Numerical Results

The starting point of out experimental work is a dataset collecting the expression level of 6601 genes of the *Saccharomyces cerevisiae* with 17 points time series per gene [Cho et al., 1998]. After the preprocessing task described in section 3, the genes' time series are filtered out in order to delete genes expression patterns of maximum level below a detection threshold, or genes' time series whose patterns have not a significant variation over time. Since we are interested in activation or inhibition relationships between couples of genes we have chosen to use the 1 time lag cross correlation measure (2) as our reference metrics. For our illustrative example, we have chosen a subset of 500 genes of the set of genes patterns actually obtained from the preprocessing task described in 3. Using the 500×500 cross correlation matrix we have build an instance of the MGRN problem by considering only the sign of the cross correlation for each ordered couples of genes. Then a relaxed version of the integer combinatorial optimization formulation (1) was solved. Using the rounding procedure described in [Chen et al., 1999], we have obtained a gene network. Doing so, the number of edges of the original cross-correlation graph was halved, since correlation relationships, considered as spurious, have been deleted. The activation/inhibition gene network graph so obtained has been given as input to the $CC-PIVOT$ algorithm, see Table 1. We have run the $CC-PIVOT$ algorithm with different random initial starting vertices. We have used a stop rule based on a lower bound threshold of the *Correlation Clustering* objective function. The clusters we have obtained can be seen in Figure 2 in which each of the 500 genes has been plotted using their first two principal components. A comparative clustering is shown in Figure 1 which is obtained simply by running the $CC-PIVOT$

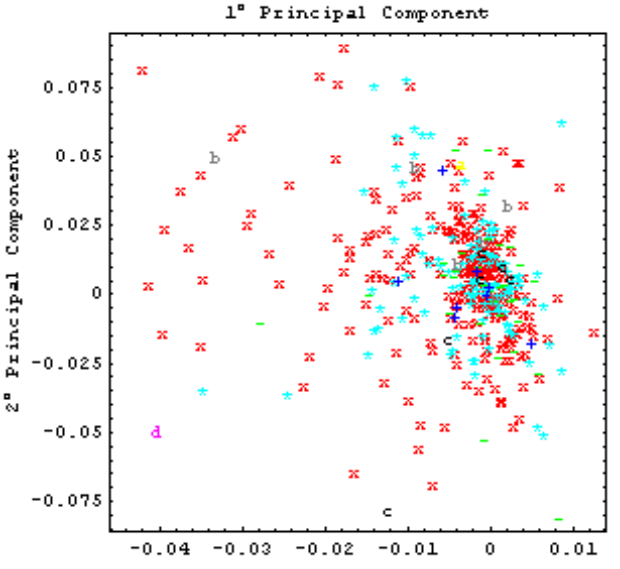


Fig. 1. The First 9 most numerous clusters for correlation relationships

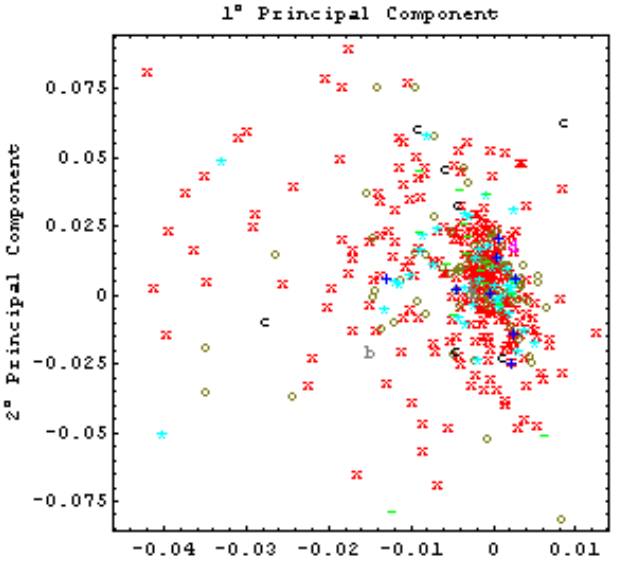


Fig. 2. The First 9 most numerous clusters for causal relationships

algorithm with the cross correlation matrix as input, without deleting the spurious correlation relationships with the procedure, whose combinatorial optimization formulation is given in (1). If we consider the clustering of the activation/inhibition relationships among genes depicted in Figure 2, we could think of each cluster as a single macro activation area. Then we could evaluate the flow of the edge weights between each macro

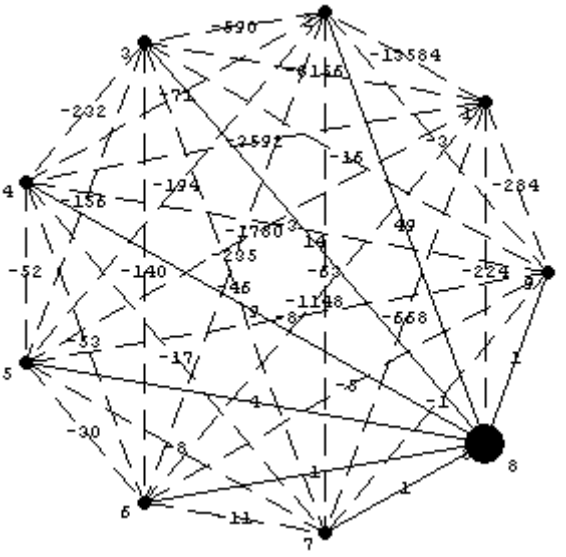


Fig. 3. Relationships among the 9 clusters of genes

area. As it is expected from the formulation of the *Correlation Clustering Problem*, the flow between each couple of clusters is likely to have negative value. This situation is depicted in Figure 3. Each node of the graph represents one of the 9 clusters of Figure 2. The edges weights are the flow of activation/inhibition relationships; the negative flows are represented in dashed lines while the positive ones in solid lines. As it can be seen, the majority of the edge weights are negative. It is interesting to note the edges involving the cluster numbered 8. The cluster number 8 has all positive weight edges but one. It seems to act as an activator for the other clusters.

5 Conclusion

We have proposed an approach for clustering causal relations among genes' patterns. The procedure consisted in solving two combinatorial optimization problems on graph, which are known to be NP-Hard, by means of a couple of approximation algorithms with moderate performance ratio. The procedure seems effective in the sense that it could deal with hundreds of genes, as in our simulations and it acts as revealing relationships among clusters of genes. As the algorithm illustrated deals with discrete information (activation/inhibition), a major improvement could be to permit the full exploitation of the continuous information coming from dataset.

References

[Ailon et al., 2005] Ailon, N., Charikar, M., and Newman, A. (2005). Aggregating Inconsistent Information: Ranking and Clustering *Proceedings of STOC2005*, to appear

[Bansal, 2002] Bansal, N., Blum, A., and Chawla, S. (2002). Correlation clustering. *Proceedings of the 43rd IEEE FOCS*, pages 238–247.

[Chaitanya, 2004] Chaitanya, S. (2004). Correlation clustering: maximizing agreements via semidefinite programming. *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms SODA 2004*, pages 526–527.

[Charikar et al., 2003] Charikar, M., Guruswami, V., and Wirth, A. (2003). Clustering with qualitative information. *To appear in Proceedings of the 44rd IEEE FOCS*.

[Chen et al., 1999] Chen, T., Filkov, V., and Skiena, S. (1999). Identifying gene regulatory networks from experimental data. *Proceedings of the third annual international conference on Computational molecular biology*, pages 94–103.

[Cho et al., 1998] Cho, R., Campbell, M., Winzeler, E., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T., Gabrielian, A., Landsman, D., Lockhart, D. and Davis, R. (1998). A genomic-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell*, 2:65–73.

[Eisen and Brown, 1999] Eisen, M. and Brown, P. (1999). Dna arrays for analysis of gene expression. *Methods in Enzymology*, 303:179–205.

[Eisen et al., 1998] Eisen, M., Spellman, P., Brown, P., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of National Academy of Sciences*, 95:14863–14868.

[Göemans et al., 1995] Goemans, M. and Williamson, D. P. (1995). Improved approximation algorithms for maximum cut and satisfiability problems. *Journal of the ACM*, 42:1115–1145.

[Liang S. and R., 1998] Liang S., F. S. and R., S. (1998). Reveal: a general reverse engineering algorithm for inference of genetic network architectures. *In Pacific Symposium Biocomputing '98*, pages 18–29.

[Pozzi et al., 2005] Pozzi, S., Della Vedova, G. and Mauri, G. (2005) An Explicit Upper Bound for the Approximation Ratio of the Maximum Gene Regulatory Network Problem *In Proceedings CMSB2004 2005, LNCS*, 3082:1–8.

[Thieffry and Thomas, 1998] Thieffry, D. and Thomas, R. (1998). Qualitative analysis of gene networks. *In Pacific Symposium Biocomputing '98*, pages 77–87.

Application of EαNets to Feature Recognition of Articulation Manner in Knowledge-Based Automatic Speech Recognition

Sabato M. Siniscalchi[1,3], Jinyu Li[1], Giovanni Pilato[2], Giorgio Vassallo[3], Mark A. Clements[1], Antonio Gentile[3], and Filippo Sorbello[3]

[1] Center for Signal and Image Processing,
School of Electrical and Computer Engineering,
Georgia Institute of Technology,
Atlanta, Georgia 30332, United States of America
{jinyuli, clements}@ece.gatech.edu
[2] Istituto di CAlcolo e Reti ad alte prestazioni,
Italian National Research Council,
Viale delle Scienze (Edif. 11), 90128 Palermo, Italy
g.pilato@icar.cnr.it
[3] Dipartimento di Ingegneria Informatica,
Universita' degli studi di Palermo,
V.le delle Scienze (Edif. 6), 90128 Palermo, Italy
siniscalchi@csai.unipa.it,
{gvassallo, gentile, sorbello}@unipa.it

Abstract. Speech recognition has become common in many application domains. Incorporating acoustic-phonetic knowledge into Automatic Speech Recognition (ASR) systems design has been proven a viable approach to rise ASR accuracy. Manner of articulation attributes such as vowel, stop, fricative, approximant, nasal, and silence are examples of such knowledge. Neural networks have already been used successfully as detectors for manner of articulation attributes starting from representations of speech signal frames. In this paper, a set of six detectors for the above mentioned attributes is designed based on the E-αNet model of neural networks. This model was chosen for its capability to learn hidden activation functions that results in better generalization properties. Experimental set-up and results are presented that show an average 3.5% improvement over a baseline neural network implementation.

1 Introduction

State-of-the-art speech recognition technology utilizes frame-based feature vectors, corresponding to about 10-20 milliseconds (ms) of speech (frame length). Within this framework, Mel-Frequency Cepstrum Coefficients (MFCCs)[4] are the most commonly employed features because of their properties to capture the main characteristics of the vocal cords and tract. Moreover, these features are usually computed by means of short-term spectral techniques, such as linear prediction (LP) analysis, or band-pass filter benches (BPFs). In addition,

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 140–146, 2006.

as the Continuous Density Hidden Markov Model (CDHMM)[1] models the sound classes, data-driven machine learning techniques allow the training of the CDHMM parameters directly from the speech data by way of dynamic programming algorithms, e.g. Baum-Welch procedure [1]. Nonetheless, even if speech researchers have learned a lot on how to build speech recognition systems, the performance of Automatic Speech Recognition (ASR) systems are comparable to Human Speech Recognition (HSR) only when working conditions match training conditions [7]. In this context, it is interesting to note that human beings integrate multiple knowledge sources in bottom-up fashion. The HSR system gathers acoustic and auditory information from the speech signal, combines them into cognitive hypotheses, and then recursively validates these hypotheses until a final decision is reached. Conversely, data-driven automatic systems, such as the Hidden Markov Model (HMM) [2] or Artificial Neural Networks (ANN) [1], address the speech recognition problem as a top-down paradigm, directly trying to convert the speech signal into words, and thus neglecting all the rich set of information that a speech signal conveys, such as gender, accent, speaking style, etc.

To overcome these limits one could attempt to incorporate the above information by collecting more data for the training data set. Nevertheless, C.-H. Lee has recently pointed out that the performance of these knowledge-ignorant modelling approaches can be improved integrating the knowledge sources available in the large body of speech science literature [11]. In the same work, he proposed a detection-based automatic speech recognition (ASR) paradigm through automatic speech attribute transcription (ASAT). Furthermore, in [6] it is showed that the idea of a direct incorporation of acoustic-phonetic knowledge, as knowledge-based features (also referred to as speech attributes in the same work) into ASR design rises the accuracy. This goal was achieved augmenting the front-end module of a conventional ASR system by means of a set of feature detectors able to capture the above-mentioned speech attributes.

The problem addressed in this paper is to build a set of detectors to recognize six attributes, namely *vowel, stop, fricative, approximant, nasal, and silence.* These attributes represent the manner of articulation, and in this paper are referred to as *manner of articulation* attributes. The choice of these attributes was dictated by not only their strong relation to human speech production [3], but also by their robustness to speech variations [6]. These six manner events are extracted directly by short time MFCCs, and represent the direct input to the six detectors.

It is well known that neural networks are widely used since they can learn a mapping from an input space to an output space realizing a compromise between recognition speed, recognition rate and hardware resources[8]. The generalization capability of neural networks is acquired during the training phase and the generalization degree achieved is strictly related to the training set characteristics. In [6], feed-forward neural networks are used to implement the six speech attribute detectors, the output of which may be interpreted as a posteriori probabilities of an attribute given the speech signal [3]. This set of detectors is used as baseline for comparisons against the model herein proposed.

Recently, a feed-forward neural architecture (EαNet) capable of learning its hidden neuron activation function has been introduced [8], and proven to perform better than traditional feed-forward neural architectures [9][10]. In this architecture, the activation functions of the hidden units are not chosen a priori, but rather they are approximated with a regression formula based on orthonormal Hermite polynomial functions. Each activation function belonging to the hidden layer, along with the neuron connection weights is then learnt during the training phase with the use of the Conjugate Gradient Descent technique with the Powells restart conditions[8].

In this paper, a set of six attribute detectors is designed based on the EαNet neural architecture. Each one of the EαNet detector classifies input speech frames into a single attribute category. The performance is evaluated on continuous phone recognition using the TIMIT database [12]. Experimental results demonstrate the effectiveness of this design for speech attribute classification, with an average 3.5% improvement with respect to the traditional ANN (maximum 8.5% improvement for plosives). The rest of the paper is organized as follows. Section 2 describes the architecture of the EαNet neural network. Section 3 describes the general framework of the *knowledge extraction* module. Section 4 presents the experimental set-up and results, with comparison to the baseline architecture. Some concluding remarks close the paper.

2 E-αNet Architecture

EαNet [8][9][10] is a feed-forward neural architecture, or multi-layer perceptron, that is able to learn the activation function of its hidden units during the training phase. Compared to a traditional feed-forward network that uses sigmoidal or sinusoidal activation functions for its hidden unit, this model is characterized by lower values of the gradient of the network output function in the surroundings of the training points. Furthermore, to avoid introduction of additional information beyond what already available in the training set (see [9][10]) in EαNet architectures the activation function is modelled through a Hermite regression formula and the optimization algorithm is based on the conjugate gradient descent with Powell restart conditions [8]. The choice of the Hermite regression algorithm is motivated by i) the smoothness of the resulting interpolation, and ii) its easy to compute first derivative [8][9].

3 Knowledge Extraction Module

The Knowledge Extraction (KE) module uses a frame-based approach to provide K manner of articulation attributes ($A_i, i \in [1, \ldots, K]$) from an input speech signal $s(t)$. In this paper the manner classes were chosen as in [6], and are listed in Table 1.

The KE module, depicted in Figure 1, is composed of two fundamentals blocks: the feature extraction module (FE), and the attribute scoring module (SC).

Table 1. Manner articulation attribute

Articulation manner	Class Elements
VOWEL	IY, IH, EH, EY, AE, AA, AW, AY, AH, AO, OY, OW, UH, UW, ER, AX, IX
FRICATIVE	JH, CH, S, SH, Z, ZH, F, TH, V, DH
STOP	B, D, G, P, T, K, DX
NASAL	M, N, NG, EN
APPROXIMANT	L, R, W, Y, HH, EL
SILENCE	SIL

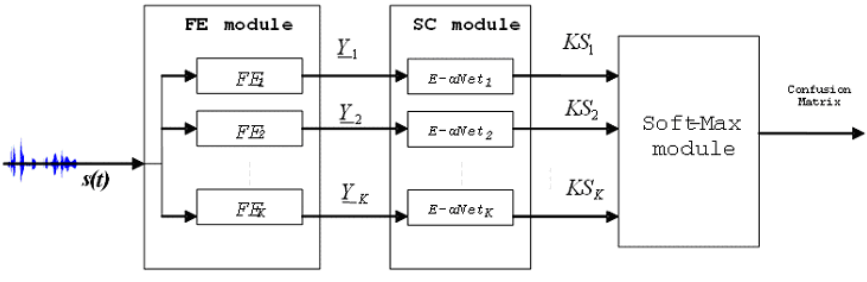


Fig. 1. Procedure of encoding the k-th word of the i-th lexical set

The FE module consists of a bank of K feature extraction blocks FE_i, where $i \in [i \dots K]$ and it maps a speech waveform into a sequence of speech parameter vectors $Y_i, i \in [i \dots K]$. Actually, each of the FE_i is fed the same speech waveform $s(t)$ and for each 10 ms-frame it computes a thirteen-MFCC feature vector X_i (12MFCCs + Energy). The frame length of 30 msec, overlapped by 20 msec.

Finally, FE_i produces, as output, a 117-feature vector Y_i combining the actual frame with the eight surrounding frames, 4 frames before and after, so that each speech parameter vector represents nine frames.

The SC module is composed of six E-αNets feed-forward neural networks, and its goal is to attach a score, referred to as *knowledge score* (KS_i), to each vector Y_i. The input of each network is a 9 frames of 12MFCCs + energy, so that the input layer is of 117 nodes. The output layer has two nodes, one for the desired class, and one for the anti-class (which are the the elements belonging to the other classes). Actually, the value obtained for the desired class for case i is defined to be the knowledge score (KS_i).

4 Experiments and Results

The evaluation of the proposed Manner of Articulation Manner Extraction module was performed on the TIMIT Acoustic-Phonetic Continuous Speech Corpus database [12], which is a well-known speech corpus in the speech recognition field.

This database is composed of a total of 6300 sentences; it has a one-channel, 16-bit linear sampling format, and it was sampled at 16000 samples/sec. The EαNet detectors were trained on 3504 randomly selected utterances, and to be consistent with [6] and [5] the four phones "cl", "vcl", "epi", and "sil" were treated as a single class, thus reducing the TIMIT phone set to a set of 45 context-independent (CI) phones.

Each of the six EαNet detectors is a three-layer network the input of which is a window of nine frames, that is, 117 parameters. The nodes of hidden layers are 100. The output layer contains two units, and a simple linear activation function is used. Finally, the soft-max module applies a soft-max function to the outputs in order to compute the overall confusion matrix.

Furthermore, an algorithm based on a mapping table was used to generate the training labels of each detectors from the phone transcription. In addition, each generated training set was normalized using the following formula:

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

where z is the new normalized value, x is the original value, μ is the training set mean value and σ is the training set standard deviation. These work-condition constrains were adopted in order to compare fairly the results presented in this paper with those shown in [6]. As previously stated, the detectors work in a frame-based paradigm, so that their performance was evaluated in term of frame error rate. Each frame was classified according to the neural network with the largest value. The global confusion matrix for the manner of articulation manner

Table 2. Phoneme accuracies (as percentages) for the manner of articulation attributes using EαNet architectures. Confusion Matrix of the manner attributes.

%	Vowel	Fricative	Stop	Nasal	Appr	Silence
Vowel	**91,00**	1,38	1,53	1,26	4,64	0,19
Fricative	3,16	**88,06**	5,53	1,02	0,89	1,24
Stop	6,32	7,41	**81,03**	1,71	1,57	1,96
Nasal	9,65	2,44	3,25	**81,45**	2,20	0,90
Approximant	30,82	2,88	3,26	2,74	**59,11**	1,19
Silence	1,10	1,09	1,88	0,61	0,58	**94,74**

Table 3. Relative improvement to the Li's Improvement of the articulation manner classification over the baseline ANN [6]

%	Vowel	Fricative	Stop	Nasal	Appr	Silence
Vowel	**2,00**	-0,12	0,03	-0,54	-0,36	-0,01
Fricative	-0,54	**2,86**	-1,27	-0,18	-0,41	-0,46
Stop	-1,28	-3,59	**8,53**	-1,19	-0,53	-1,94
Nasal	-1,55	-0,06	-1,45	**3,95**	-1,00	0,10
Approximant	-1,48	-0,02	-0,44	-0,46	**2,61**	-0,21
Silence	0,00	-0,11	-1,32	-0,09	-0,32	**1,84**

attributes is given in Table 2. The $(p, q) - th$ element of the confusion matrix measures the rate of the $p - th$ attribute being classified into the $q - th$ class.

5 Conclusions

Incorporating acoustic-phonetic knowledge into Automatic Speech Recognition designs has been proven a viable approach to rise their accuracy. Manner of articulation attributes such as vowel, stop, fricative, approximant, nasal, and silence are examples of such knowledge, and they represent speech attributes. A set of six attribute detectors was designed based on the EαNet neural architecture and their performance has been studied. The evaluation demonstrates the effectiveness of this design for speech attribute classification, with an average 3.5% improvement with respect to the use of a traditional ANN approach, showing a maximum 8.5% improvement in the case of plosives.

Acknowledgements

Authors are indebted with Prof. Chin.-H Lee, for the insightful discussions on the topics and for his help defining the general experimental framework. Part of this effort was supported under the NSF SGER grant, IIS-03-96848 and NSF ITR grant, IIS-04-27413.

References

1. L. R. Rabiner, : A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE, Vol. 77, No.2, pp. 257-286, 1989.
2. S. Haykin, : Neural Networks: a Comprehensive Foundation (2nd edition). Prentice Hall, 1998.
3. K. Kirchhoff: Combining articulatory and acoustic information for speech recognition in noisy and reverberant environments Proc. ICSLP98, Sydney, Australia, 1998.
4. S. Davis, and P. Mermelstein: Comparison of parametric representations for monosyllable word recognition in continuously spoken sentences. IEEE Trans. on Acoust., Speech and Signal Process., Vol. 28, No. 4, pp. 357-366, 1980.
5. K. F. Lee, H. W. Hon: Speaker-independent phone recognition using hidden Markov models. IEEE Trans. On Acoust., Speech and Signal Process., Vol. 37, No. 11, pp. 1641-1648, 1989.
6. J. Li, Y. Tsao and C.-H. Lee: A study on knowledge source integration for candidate rescoring in automatic speech recognition. Proceedings of the International Conference on Spoken Language Processing, Sydney, Australia, December, 1998, 891-894.
7. R. P. Lippmann: Speech recognition by machines and humans Speech Communication, Volume: 22 (1), July, 1997, pp. 1-15.
8. S.Gaglio, G. Pilato, F.Sorbello and G.Vassallo: Using the Hermite Regression Formula to Design a Neural Architecture with Automatic Learning of the 'Hidden' Activation Functions AI*IA99:Advances in Artificial Intelligence - Lecture Notes in Artificial Intelligence 1792 - Springer Verlag - pp. 226-237, 2000.

9. G.Pilato, F.Sorbello and G.Vassallo: An Innovative Way to Measure the Quality of a Neural Network without the Use of the Test Set IJACI International Journal of Advanced Computational Intelligence - Vol. 5 No 1, 2001, pp:31-36.
10. A.Cirasa, G.Pilato, F.Sorbello and G.Vassallo: An Enhanced Version of the aNet Architecture: Automatic Pruning of the Hermite Orthonormal Functions Atti del Workshop "Apprendimento e Percezione nei Sistemi Robotici" - Parma, Italy - 29-30 November 1999.
11. Lee, C.-H.: From knowledge-ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition, Proc. ICSLP, 2004.
12. J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren: DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus. U.S. Dept. of Commerce, NIST, Gaithersburg, MD, February 1993.

Granular Regression

B. Apolloni[1], D. Iannizzi[2], D. Malchiodi[1], and W. Pedrycz[3]

[1] Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano, Via Comelico 39/41, 20135 Milano, Italy
apolloni@dsi.unimi.it, malchiodi@dsi.unimi.it
[2] Dipartimento di Matematica "F. Enriques", Università degli Studi di Milano, Via Saldini 50, 20133 Milano, Italy
iannizzi@mat.unimi.it
[3] Department of Electrical and Computer Engineering, University of Alberta, ECERF, 9107 - 116 Street, Edmonton, Alberta, Canada T6G 2V4
pedrycz@ee.ualberta.ca

Abstract. We augment a linear regression procedure by a thruth-functional method in order to identify a highly informative regression line. The idea is to use statistical methods to identify a confidence region for the line and exploit the structure of the sample data falling in this region for identifying the most fitting line. The fitness function is related to the fuzziness of the sampled points as a natural extension of the statistical criterion ruling the identification of the confidence region within the Algorithmic Inference approach. We tested the method on three well known benchmarks.

1 Introductory Comments

This work concerns the use of techniques of granular computing [1] in the refinement of usual regression models [2]. The underlying concept and the design rationale can be concisely outlined in the following manner. Given experimental data, we commonly confine to the linear regression model as the first possible alternative worth exploring. Once accepted, we then focus on the refinement of the model. From the functional standpoint, there are several essential phases reflecting the rationale. First, the confidence region of the preliminary linear model (formed through the use of the confidence curves for some predefined confidence level) eliminate data points outside this region. The remaining data are subject to further usage of model building by endowing them by some properties of information granules. We consider the surroundings of those selected points as true information granules and equip them with bell-shaped membership functions like those encountered e.g. in radial basis functions (RBF) [3]. Considering the landscape constituted by a norm on the above bells, we may look for a regression line maximizing the integral of this norm along the line. In some sense this is a dual objective in respect to support vector machines [4]. With the latter we try to draw a line passing along the valleys, with the former along the crests. If we may express the above integral as a function of the drifts between sample points and interpolating line, then we may expect obtaining a

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 147–156, 2006.

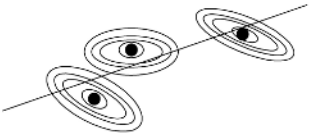


Fig. 1. Fitting the granules' information with a line

more informative regression line, just because we exploit specific information on the points represented by their attraction basins (see Fig. 1). To be more precise, starting from the usual linear model

$$y_i = a + bx_i + \epsilon_i \tag{1}$$

explaining the relation between the coordinates of the i-th sampled point, we build a confidence region at a given level δ, namely a region where regression line entirely falls with probability $1 - \delta$ according to the Algorithmic Inference approach [5]. This allows us to select a meaningful region, within which in analogy with δ-cuts in fuzzy sets [6] we will identify sample points that are mainly involved in determining a more meaningful regression line as follows. Provided that our δ-cut collects the most likely regression lines, we select the one passing as close as possible to the *relevant* sampled points, namely those included in the confidence region. We remove the points outside the confidence region, essentially considered as outliers, and focus on those inside by better analyzing their relevance through membership functions. For short: a sample point (say (x_i^*, y_i^*)) is much relevant; those (non sampled) falling in its neighborhood are less relevant, exactly according to membership functions acting as RBFs. Under lose symmetry conditions of the membership functions, denoting with ξ and ψ the coordinates of the point along a straight line r, its membership $\mu_i(\xi, \psi)$ to the RBF around the i-th sample point is a function g of the distance between (ξ, ψ) and (x_i^*, y_i^*) (see Fig. 2).

Starting from this new model we expect to identify a more suitable regression line for we better specialize the information on the single points. Namely, a simple statistical model uniformly may assume the same RBF for each point made of a Gaussian bell with mean in the sampled point and same variance on each point (this is another way of articulating the essence of regression models with Gaussian noise). We may reduce the entropy of this model once we are able to specialize the bells on each sample point. A first specialization consists in giving a Dirac function to each point outside the confidence region; hence these points do not affect our straight lines. Conversely, we connect the shape of the bells around points inside the confidence region to the mutual relations between these points as it emerges from a suitable clustering of them. Further specifications are of interest and this could be a direct result of expressing domain knowledge about specific data points.

The paper is organized as follows: Section 2 describes how the regression model is determined, while Section 3 covers some preliminary numerical experiments.

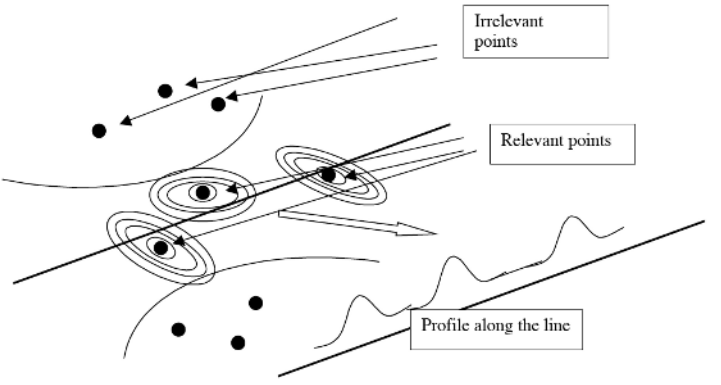
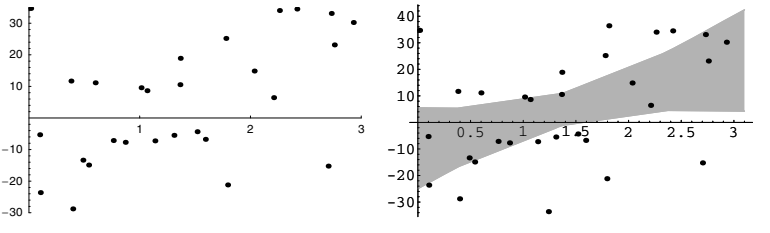


Fig. 2. A synopsis of the proposed method

Finally, we offer some conclusions and elaborate on future developments of the proposed approach.

2 The Design Method of the Model

For a given sample $\mathbf{z} = \{(x_i, y_i)\ i = 1, \ldots, n\}$ such the one shown in Fig. 3(a), the proposed method works through a sequence of steps: i) identifying the information granules, ii) endowing each granule with a relevance measure, and iii) computing a regression line on a basis of the data selected in i) and ii).



(a) (b)

Fig. 3. (a) A sample of $n = 30$ points in input to the procedure. (b) A 0.9 confidence region for the regression line (gray region) identifying $m = 12$ information granules (points).

2.1 Identifying Information Granules

The first step is devoted to the selection, among the sample points, of the information granules upon which the rest of the procedure will be based. Fixed the value of $\delta \in [0, 1]$, we identified these granules with the m sample points included in a $1-\delta$ confidence region Ω for the regression line describing the relation

among the sample points' coordinates. Hence, discarding points not belonging to Ω we obtain a *pruned version* $\mathbf{z}^* = \{(x_i^*, y_i^*) : i = 1, \ldots, m\}$ of the sample. For instance, Fig. 3(b) illustrates the 0.9-confidence Ω describing the sample in Fig. 3(a) obtained through the Algorithmic Inference regression method [7], and the corresponding pruned sample.

2.2 Assigning Relevance to the Granules

The points in $\mathbf{z}^*$ constitute the statistically drawn base of our knowledge, while the remaining ones are essentially assumed to be outliers. We also assume the former to be *information granules*, namely the centers of m fuzzy sets described by bell-shaped membership functions μ_i defined as follows:

$$\mu_i(x, y) = h_i \mathrm{e}^{-\pi h_i((x-x_i)^2+(y-y_i)^2)}. \tag{2}$$

Each of these functions resembles a Gaussian symmetric bell centered on the point $\mathbf{z}_i^* = (x_i^*, y_i^*)$, i.e. a bidimensional normal density function, whose variates' coordinates have the same variance $\sigma^2 = (2\pi h_i)^{-1}$ and covariance $\rho = 0$.

Determining the set $\{h_i, i = 1, \ldots, m\}$ is the operational way of making the model definite. This corresponds to embedding in the i-th granule some information about its *relevance* h_i. Indeed, the higher this value, the smaller the variance of the corresponding density.

In some cases this information is explicitly available, for instance in form of reliability (relevance) for each sampled point. If this is not true, a possible way for determining h_i consists in considering the topology of the pruned samples by some clustering mechanisms, say Fuzzy C-Means (FCM) [8]. Having fixed the number of clusters to be equal to c, once their centers $\{\mathbf{v}_1, \ldots \mathbf{v}_c\}$ has been identified, we compute the relevance h_i as the maximum value of its membership grades to the various clusters, that is:

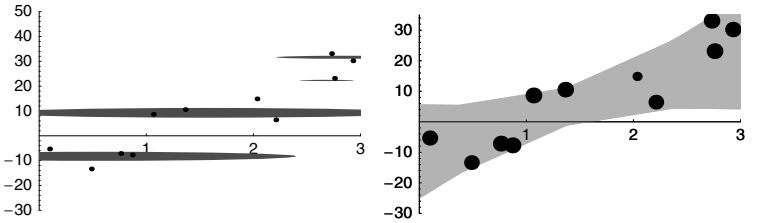
$$h_i = \max_{1 \le k \le c} \left\{ \left(\sum_{j=1}^{c} \left(\frac{||\mathbf{z}_i^* - \mathbf{v}_k||}{||\mathbf{z}_i^* - \mathbf{v}_j||} \right)^{\frac{2}{\alpha - 1}} \right)^{-1} \right\}, \tag{3}$$

where $\alpha \in \mathbb{N}$ is a fuzzification factor (> 1) whose original value has been selected when running the clustering procedure. The typical value of this factor is taken as 2.0.

For $c = 4$ and $\alpha = 2$, Figs. 4(a) and (b) respectively show the position of the clusters and the obtained granules relevance levels.

2.3 Finding the Optimal Regression Line

Among all the possible lines entirely contained in Ω, we will look now for the *optimal regression line*, i.e. the line r maximizing the sum of the integrals I_i^* of the curves obtained intersecting the membership functions $\mu_i^*(x, y)$ with the plane which contains r and in addition is orthogonal to the plane $X \times Y$ to which both r and the sample points belong. If we refer to the points of r through the



(a) (b)

Fig. 4. (a) FCM clusters (gray circles with a radius proportional to annexed points' standard deviation) against granules (black points); (b) granules (black cirlces) with embedded relevance (radius of the circle); the gray region denotes the confidence region Ω for the regression line

equation $a + bx + y = 0$, i.e. r has slope and intercept, respectively, equal to $-b$ and $-a$, the above integral will depend on the latter quantities, thus we write $I_i^*(a, b)$.

In the plane having as axes r and any line orthogonal to it, say having coordinates ξ and ψ, given the radial symmetry of the bell membership function, we may express the latter again as a bidimensional Gaussian density function

$$\mu_i^*(\xi, \psi) = h_i \mathrm{e}^{-\pi h_i((\xi - \xi_i)^2 + (\psi - \psi_i)^2)} \tag{4}$$

(where ξ_i^* and ψ_i^* are the analogous of x_i^* and y_i^* in the new space), so that the bells around our points look like in Fig. 5, where their contours at level values 0.1 and 0.005 – squeezed because of the picture aspect ratio – are reported.

Summing up, the integral $I_i^*(a, b)$ corresponding to the i-th granule is

$$I_i^*(a, b) = \int_{-\infty}^{\infty} \mu_i^*(\xi, \psi_i) \mathrm{d}\xi = \mu_i^*(\psi_i) \int_{-\infty}^{\infty} \mu_i^*(\xi | \psi_i) \mathrm{d}\xi = \mu_i^*(\psi_i), \tag{5}$$

where $\mu_i^*(\xi | \psi_i)$ has the shape and mathematical properties of a conditional density function of a random variable Ξ given the value of the companion variable

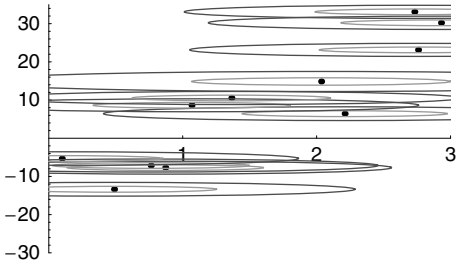


Fig. 5. Contours of the fuzzy membership to the granules (black points) at level 0.1 (light gray ellipses) and 0.005 (dark gray ellipses) suggested by FCM algorithm

$\Psi = \psi_i$ and, analogously, $\mu_i^*(\psi_i)$ is the marginal distribution of Ψ evaluated on ψ_i. Hence

$$\mu_i^*(\psi_i) = h_i^{1/2} \mathrm{e}^{-\pi h_i \psi_i^2} \tag{6}$$

and $\int_{-\infty}^{\infty} \mu_i^*(\xi|\psi_i)\mathrm{d}\xi = 1$ by definition.

Finally, as ψ_i is the distance of the point (x_i^*, y_i^*) from the r, we have

$$\psi_i = \frac{|bx_i^* + y_i^* + a|}{\sqrt{1+b^2}}, \tag{7}$$

so that the integral value is

$$I_i^*(a,b) = h_i^{1/2} \mathrm{e}^{-\pi h_i \frac{(bx_i + y_i + a)^2}{1+b^2}}. \tag{8}$$

Therefore, the optimal regression line has parameters

$$(a^*, b^*) = \arg\max_{a,b} \sum_{i=1}^{m} h_i^{1/2} \mathrm{e}^{-\pi h_i \frac{(bx_i + y_i + a)^2}{1+b^2}}. \tag{9}$$

In order to solve the related optimization problem, we can turn to an incremental algorithm, like a simple gradient descent or simulated annealing [9], exploiting the easy form of the derivatives of the integrals w.r.t. the parameters a and b of the regression line. The sole constraint we put is that the final line must not trespass the borders of the confidence region Ω. In our leading example, after some thousands iterations of the gradient descent algorithm we obtained the results shown in Fig. 6.

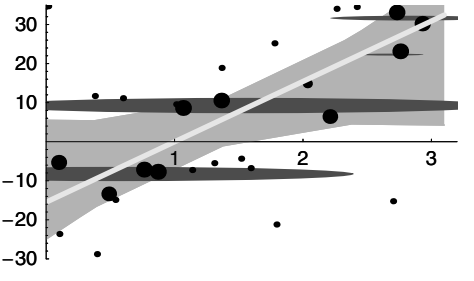


Fig. 6. Granular regression line (thick gray line), with same notations as in previous figures

3 Numerical Experiments

We tested the procedure on three benchmarks concerning respectively a calibration procedure, the observation of sunspots, and some socio-econometric indicators measured in Switzerland.

3.1 Testing Against a Well-Structured Dataset

The Norris benchmark [10], published by the US National Institute of Standards and Technology (NIST), contains 36 items concerning the calibration of ozone monitors. This is an example of highly structured data set: the corresponding scatter plot, drawn in Fig. 7(a), shows 10 granules whose centers almost lie on a straight line, having the x coordinates equally distributed. As a consequence, all points are contained in the confidence region, i.e. none of them is excluded from the information granules' set. As a further consequence, the FCM routine (run with $c = 4$ in the experiments) introduce an undue over-structure over the points; it actually partitions them in four groups of contiguous points, with the result of giving higher relevance to the points close the centers of the clusters. As a result we obtain a granular regression (GR) line that slightly shifts from least squares (LS) line because of the enhancement of the original random shifts of the sampled points, that is improperly introduced by the FCM procedure. In any case the residual errors sum to 17.86 denoting a good performance of the algorithm even in this calibrating instance.

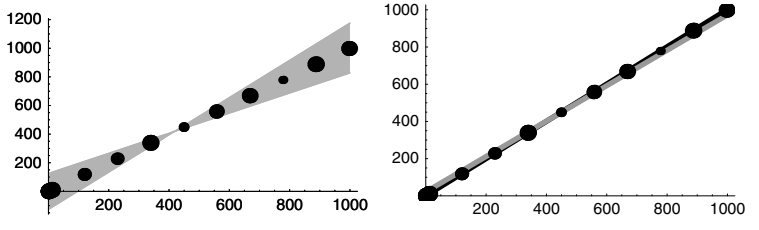
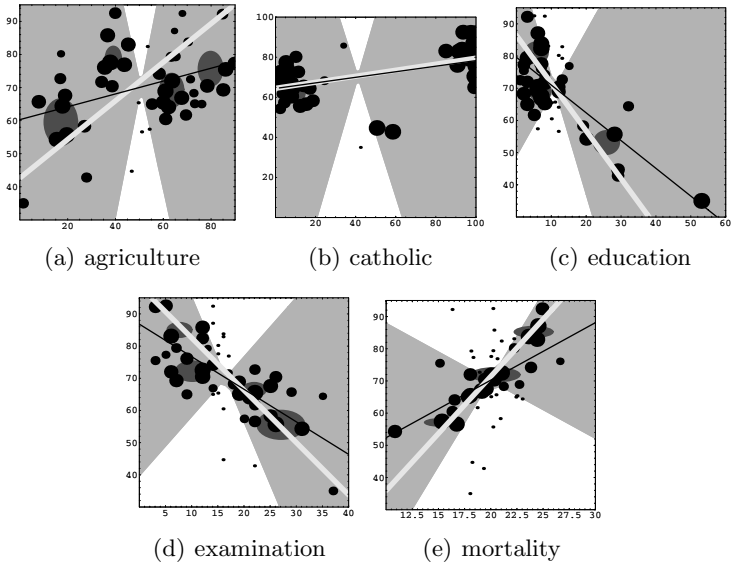


Fig. 7. Testing against the Norris dataset: (a) sample points with embedded relevance and 0.9 confidence region for the regression line (same notations of previous figures); (b) comparison between GR line (thick gray line) and LS line (black line)

3.2 Finding a Structure in Data

We have more structure inside a dataset reported in [11] that describes standardized socio-economic indicators for 47 French-speaking provinces of Switzerland at about 1968. In particular we have the scatter plots in Fig. 8 by crossing the IG fertility indicator with a second indicator that alternatively refers to the percentages of males involved in agriculture, of catholic people, of educated people, of people receiving the highest examination grade and of infant mortality in those provinces. In the pictures we see that GR line (thick dark gray line) looks for crossing crowded ensembles of points (denoting the effect of a specific phenomenon) while LS line looks for balancing the shifts. When the two targets coincide, like in Fig. 8(b), the two lines overlap. *Vice versa*, the influence of the two different targets is clearly shown in Fig. 8(d) and (e). On the contrary, in Fig. 8(a) and (c) the way through wich GR target is pursued is less clear, because it derives from a compromise between the attractions that the line bears from variously spread directions.



(a) agriculture (b) catholic (c) education

(d) examination (e) mortality

Fig. 8. Exploiting local structure in the swiss database. Same notations as in Figs. 6 and 7.

3.3 Testing Against Outliers

With the way we decided to identify structure into the data (by topologically clustering them), isolated points are considered disregardable. In addition, isolated outliers are definitely scratched by the confidence region. We get evidence in Fig. 9 of these two effects by processing a dataset describing the monthly sunspot numbers collected from 1850 to 2001 [12]. The high number of sampled

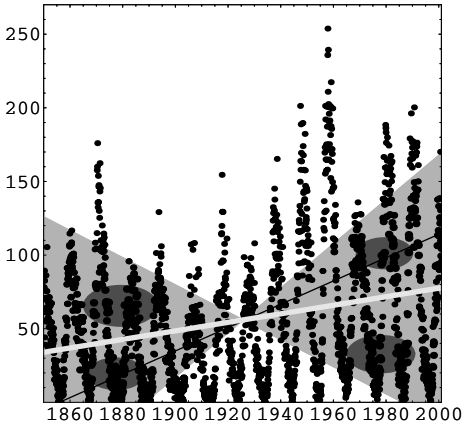


Fig. 9. Pruning outliers in the sunspots database. Same notation as in Fig. 8.

points has the side effect of inducing many local maxima in the corresponding optimization problem (besides lowering the procedure's execution time); thus we abandoned the gradient descent algorithm used in the previous experiments in favor of a simulated annealing procedure. In this way we get a GR line that is more sensitive than LS line to the concentration of points in the right bottom corner of the scatterplot.

4 Conclusions

In the perspective of probability as a way of organizing available information about a phenomenon rather than a *physical* property of the phenomenon, we consider additional information which is local, hence not gathered through a measure summing to 1 over a population. In contrast to the linear regression problem, we focus on: i) δ-cuts identified through statistical methods, and ii) a local density of clusters of points that reverberates in a membership function of population points to the information granule represented by the sample points. We set up a procedure that exploits both information kinds producing regression lines different from those obtained with simply statistical methods whenever local information are relevant. Further investigations could be pursued along the following lines:

- from one to many independent variables (vectorial x in (1)). This may involve detailed algebraic extensions maintaining the solution in our framework connected to the weighted distance of the line from selected sample points;
- from linear to more complex dependences as outlined in [1]. We plan to carry out this extension through kernel methods [13], by profiting of the rich literature developed on this matter w.r.t. support vector machines.

References

1. Pedrycz, W.: Granular computing in data mining. In Last, M., Kandel, A., eds.: Data Mining & Computational Intelligence. Springer-Verlag (2001)
2. Morrison, D.F.: Multivariate statistical methods. 2nd edn. McGraw-Hill, New York (1989)
3. Poggio, T., Girosi, F.: Networks for approximation and learning. In Lau, C., ed.: Foundations of Neural Networks. IEEE Press, Piscataway, NJ (1992) 91–106
4. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines. Cambridge University Press (2000)
5. Apolloni, B., Malchiodi, D., Gaito, S.: Algorithmic Inference in Machine Learning. International series on advanced intelligence, vol. 5. Advanced Knowledge International, Magill, Adelaide (2003)
6. Cox, E.: The fuzzy systems handbook. AP Professional, San Diego (1998)
7. Apolloni, B., Bassis, S., Gaito, S., Iannizzi, D., Malchiodi, D.: Learning continuous functions through a new linear regression method. In Apolloni, B., Marinaro, M., Tagliaferri, R., eds.: Biological and Artificial Intelligence Environments, Springer (2005) 235–243

8. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algoritms. Plenum Press, New York (1981)
9. Aarts, E., Korst, J.: Simulated annealing and Boltzmann machines: a stochastic approach to combinatorial optimization and neural computing. John Wiley, Chichester (1989)
10. National Institute of Standards and Technology: Strd dataset Norris (online, accessed April 2005) `http://www.itl.nist.gov/div898/strd/lls/data/Norris.shtml`
11. Mosteller, F., Tukey, J.W.: Data Analysis and Regression: A Second Course in Statistics. Addison-Wesley, Reading Mass. (1977)
12. Solar Influence Data Analysis Center, Royal Observatory of Belgium: Sunspot and space weather information pages (online, accessed April 2005) `http://sidc.oma.be/index.php3`
13. Schölkopf, B., Smola, A.J.: Learning with kernels : support vector machines, regularization, optimization, and beyond. MIT Press, Cambridge, Mass. (2002)

Fuzzy Logic Activities at the Microelectronics Institute of Seville

Angel Barriga, Santiago Sánchez-Solano, Iluminada Baturone, Francisco Moreno-Velo, Piedad Brox, Federico Montesino, Nashaat M. Hussein, María Brox, and Andrés Gersnoviez

Instituto de Microelectrónica de Sevilla,
Avda. Reina Mercedes s/n,Edif. CICA, 41012-Sevilla, Spain
{barriga, santiago, lumi, velo, brox, fedemp, nashaat, maria, andres}@imse.cnm.es

Abstract. In this communication we present the activities related to the development of fuzzy logic based systems at the Microelectronics Institute of Seville (Spain). These activities regard with the design of circuits and systems that operate in fuzzy logic, the development of CAD tools for fuzzy logic and the accomplishment of applications that use fuzzy logic in the resolution of certain problems.

1 Introduction

Fuzzy logic activities began at IMSE by the year 1990 within the ORBE project (ORdenador Borroso Experimental, Experimental Fuzzy Computer). In this project several Spanish groups were collaborating under the direction of Prof. Claudio Moraga (Dortmund University). Since then, a research group has been consolidated around the hardware design of fuzzy logic based systems. All this research activity has been focused on several guidelines. Basically the main area has been related to the design of digital, analog and mixed signal (analog-digital) circuits for fuzzy logic-based processing. Together with this area, the development of fuzzy logic CAD tools has also addressed our attention. The result from the latter activity has been the fuzzy logic development environment called Xfuzzy.

This paper tries to show the current activity of the fuzzy logic research group at IMSE. It is focused on three issues that are covered in the next sections: architectural design of fuzzy processors, development of CAD tools for fuzzy logic and application of fuzzy logic to different areas.

2 Fuzzy Processor Architectures

The main research activity at IMSE is related to the development of circuits and hardware systems for implementing fuzzy inference mechanisms. Nowadays, hardware development is focused on digital processing systems using different realization strategies. One of these strategies is based on a specific architecture for the fuzzy processing element [1]. An advantage of this architecture is to combine the speed increment of parallel processing with the cost reduction of sequential processing systems. This can be accomplished since the proposed architecture is based on

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 157 – 162, 2006.

singleton consequents, active rule processing, simplified defuzzification methods and pipeline techniques.

Figure 1 shows the block diagram of the proposed architecture. There are three stages (fuzzifier, inference and defuzzifier). Each stage can be implemented using different techniques. Thus, in the fuzzification stage the antecedents can be implemented by using a memory or arithmetic circuits, while in the inference stage there are different components that perform as connectives (minimum, product, and bounded-product). The rule memory can be implemented using RAM or ROM, and, finally, the defuzzification stage can implement different defuzzification schemes by using a programmable defuzzifier.

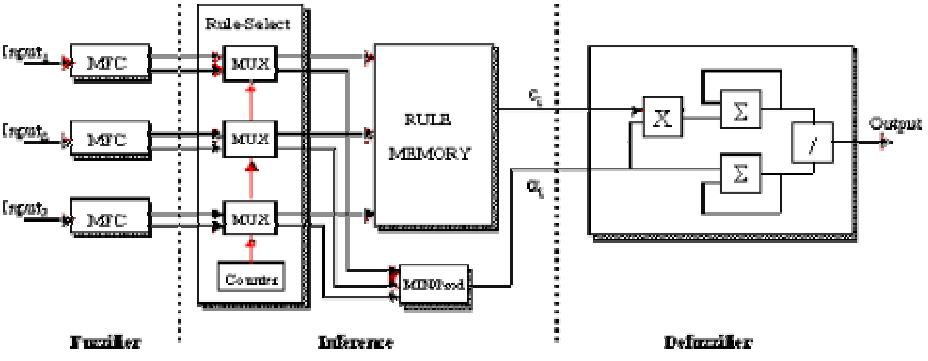


Fig. 1. Architecture of fuzzy processing element

The evolution of a fuzzy processing element up to a fuzzy processor can be accomplished by means of an embedded system based on a standard microprocessor, program and data memory and peripheral devices (interruption controller, timer, input/output interfaces, etc) [2]. This processor scheme is illustrated in figure 2. The FIM module (Fuzzy Inference Module) corresponds to the fuzzy processing element. The development of systems using this architecture is based on a hardware&software codesign methodology. Some hardware realizations have been implemented on Xilinx FPGA devices. In those implementations the general purpose processor is the 32 bit RISC architecture Xilinx processor called MicroBlaze.

Other hardware development lines focus on system design from other perspectives. One perspective consists of implementing the knowledge base as a specific combinational circuit. In this case the system structure is given by a standard digital synthesis tool. The synthesis tool generates the circuit from its behavior description using the hardware description language VHDL. Using VHDL, the fuzzy system (its membership functions, rule base, etc.) is described using the overload function property of the language [3]. The above mentioned description is synthesized by a conventional synthesis tool which produces the circuit corresponding to the inference engine. Figure 3 shows a comparison of several fuzzy system hardware implementations based on two design strategies: specific architecture and generic combinational circuit. It can be observed how the combinational logic solutions need fewer resources for precisions less than 8 bits while for precisions greater than 8 bits specific architecture has minor cost.

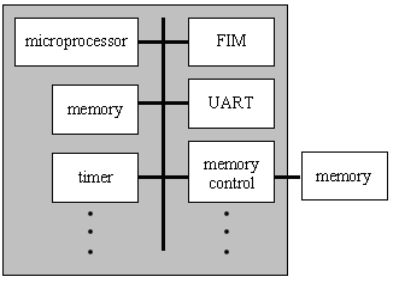


Fig. 2. Fuzzy processor architecture

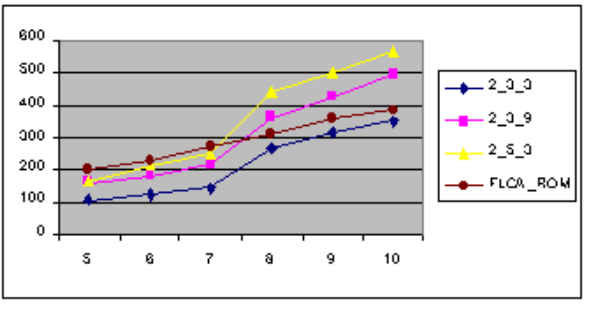


Fig. 3. Cost comparison of specific architecture implementation and combinational circuits

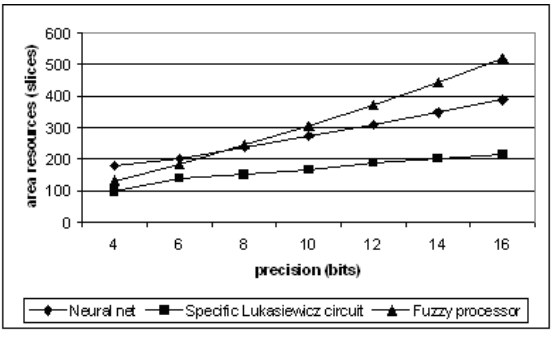


Fig. 4. Implementation comparison of a level controller

More recently there has been initiated a circuit development line based on the hardware implementation of Lukasiewicz's logic operators. Two realization strategies have been addressed: one based on neural networks and the other which uses specific combinational logic [4]. Using these operators it is possible to solve linear function interpolation problems. It has allowed to realize contributions in fields related to control and image compression. Figure 4 illustrates a comparison between different realization strategies of a level controller. The systems have been implemented on an

FPGA device. The cost in terms of FPGA slices is shown versus precision in terms of bits. It is observed that in this application the combinational logic Lukasiewicz's operators provide less cost than other approximations.

3 CAD Development: Xfuzzy

In parallel to the fuzzy logic circuits and systems design activity there is another research line related to the development of CAD tools for fuzzy logic. The result from this activity has been the fuzzy logic development environment called Xfuzzy. The Xfuzzy environment contains a set of tools that cover diverse aspects and application fields of fuzzy systems. Basically the tools within Xfuzzy facilitate the specification, adjustment, simulation and synthesis tasks of fuzzy systems.

Three versions of Xfuzzy have been developed. The current one, version 3, has been implemented completely in Java, which allows executing it independently of the platform (Windows or Unix). Xfuzzy runs under GPL license [1] [5].

The tools within Xfuzzy share a common specification language (XFL3). Figure 5 shows the structure of the environment. There are four sets of tools regarding with description, verification, tuning, and synthesis.

The description tools facilitate the graphical specification of the fuzzy system. The system is described by means of a specification using the XFL3 language. There are some graphical tools like xfedit and xfpkg that help such specification. Once a fuzzy system has been described the verification tools allow testing its functionality. This way xf2dplot and xf3dplot allow to represent graphically systems of 2 and 3 dimensions, respectively. The xfmt tool allows to modify the input variables values and to visualize the effect on the system (rule base, membership functions and output variables). Finally xfsim is a simulation tool that allows inserting the fuzzy inference engine inside a wider system containing other elements described in Java. It allows realizing behavior simulations in order to validate the specifications of the fuzzy system.

Another set of tools helps in adjusting the fuzzy system. There is a supervised learning tool that allows to fit the parameters of the fuzzy system (xfsl). There is also

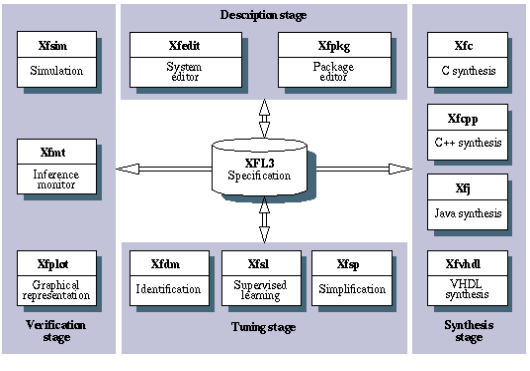


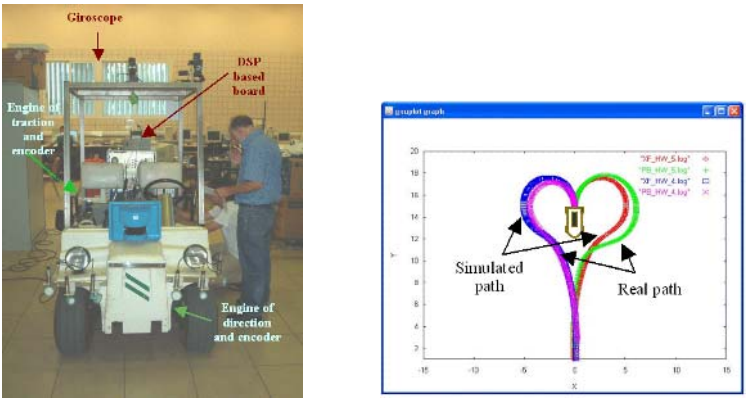
Fig. 5. Xfuzzy environment

an identification tool that permits to extract the fuzzy system specification from a data file (xfdm) as well as a knowledge base simplification tool (xfsp).

Finally there exists a set of tools that allow the implementation of the fuzzy system from two perspectives: software and hardware. The software synthesis tools xfc, xfcpp and xfj generate implementations of the fuzzy system as C, C ++ or Java programs respectively. The xfvhdl tool generates descriptions of the fuzzy system in the hardware description language VHDL. The above mentioned descriptions can be synthesized using standard synthesis tools to generate hardware implementations of fuzzy systems.

4 Application Fields

The circuit design and CAD development results can be applied to solve application problems in different fields. In particular, there are three types of applications in which the IMSE research group have proposed contributions: control of autonomous mobile vehicles, control of traffic congestion in Internet and applications to image processing.



(a) (b)

Fig. 6. a) ROMEO4 vehicle, b) Real and simulated paths

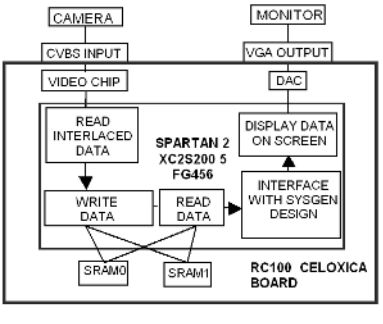


Fig. 7. Video de-interlacing system

The control of autonomous mobile vehicles has allowed to develop the hardware-software codesign methodology commented in a previous section. The prototypes were proved on the vehicle ROMEO 4R shown in figure 6a. The control problem addressed was the diagonal parking of the vehicle. For it, the knowledge base is a hierarchical structure composed by 5 rule bases. Figure 6b shows some experimental paths and their corresponding simulated paths.

Using the fuzzy processor scheme and the hardware-software codesign methodology a new application related to the control of traffic congestion in Internet is under development. In this respect, one of the working lines corresponds to the analysis of the control of congestion functions and quality of service in Internet using fuzzy logic. There are some contributions in the modeling and analysis of the network, in systems measurement and performance evaluation, as well as in the congestion control in routers and end nodes, both centralized and distributed, combined with service quality functions.

A third application line is related to image processing systems. In one side, temporal and spatial algorithms have been combined for video de-interlacing. A reconfigurable platform and three hardware architectures have been developed for the de-interlacing algorithm [6]. The designs have been realized on FPGA devices. Figure 7 shows the system scheme. In the other side, some schemes are also being developed to compress static images by applying linear functions by means of Lukasiewicz's algebra [4].

References

1. I. Baturone, A. Barriga, S. Sánchez-Solano, C. Jiménez, D.R. López, Microelectronic design of fuzzy logic-based systems, CRC Press., 2000. ISBN: 0-8493-0091-6.
2. A. Cabrera, S. Sánchez Solano, Brox, P., A. Barriga, R. Senhadji, Hardware/software codesign of reconfigurable fuzzy control systems, Applied Soft Computing, Vol. 4, Issue 3, pp. 271-285, Aug. 2004.
3. A. Barriga, S. Sánchez Solano, Brox, P., A. Cabrera, I. Baturone, Modelling and Implementation of Fuzzy Systems based on VHDL, Int. Journal of Approximate Reasoning (Accepted and pending on publication).
4. N. M. Hussein, A. Barriga, S. Sánchez-Solano, Piecewise Linear Function Interpolation Using Lukasiewicz's Operators, Int. Symposium on Innovations in Intelligent Systems and Applications (INISTA'2005), Istanbul (Turkey), June 15-18, 2005.
5. Xfuzzy home page: http://www.imse.cnm.es/Xfuzzy
6. P. Brox, S. Sánchez-Solano, I. Baturone, A. Barriga, Hardware Implementation of a De-interlacing Video Algorithm Using Programmable Logic Devices, Information Proc. and Management of Uncertainty in Knowledge-Based Systems (IPMU2004), Perugia, 2004.

Generalized Fuzzy Similarity Indexes

Narcís Clara

Universitat de Girona,
Departament d'informàtica i matemàtica aplicada
narcis.clara@udg.es

Abstract. Features Contrast Model (FCM) and Fuzzy Features Contrast Model (FFCM) are usually used to evaluate the similarity between objects. The universe of discourse is defined by two sets: A, the set of objects; and E, the set of features. A matrix is defined, which elements represent the degree in which an object verifies a feature, namely, the membership values in fuzzy terms. A classical generalization of crisp similarity indexes is well known but using some crisp properties before make the generalization, for this reason it can be said *weak* generalization. Y. A. Tolias et al, made a *strong* generalization but it does not include all the usual indexes, as simple matching or Rao's coefficients. This generalization is the goal of this paper. Some reliable conditions are proved and conditions to define a proximity relation in A are found.

1 Crisp Similarity Indexes

A very brief summary about crisp similarity indexes is presented below, focusing in those concepts that are necessary for its generalization to fuzzy sets. In the whole of the paper A and E represents two finite sets: $A = \{A_1, A_2, \ldots, A_n\}$ and $E = \{P_1, P_2, \ldots, P_m\}$.

Definition 1. *Map of FCM*
$v : A \to \wp(E)$ *defined as:* $P_k \in v(A_i) \iff A_i$ *verifies* P_k

What is equivalent to assign to each element A_i a value $\mu_{A_i}(P_k) \in \{0, 1\}$ depending on if A_i verifies or not the feature P_k synthetically

$$A_i \longrightarrow (\mu_{A_i}(P_1), \mu_{A_i}(P_2), \ldots, \mu_{A_i}(P_m)) \in \{0, 1\}^m \ . \tag{1}$$

The binary condition interpreted in the fuzzy theory is equivalent that membership function of the element A_i in the feature P_k was 1 or 0. Is immediately that $card(v(A_i)) = |v(A_i)| = \sum_{j=1}^{m} \mu_{A_i}(P_j)$ counts the number of features that verifies A_i. To evaluate the similarity we must calculate not only what verify two objects in common but what do not verify together.

Definition 2. *Similarity and dissimilarity parameters*

$$\begin{array}{ll} a = |v(A_i) \cap v(A_j)| & b = |v(A_i) \cap v(A_j)^c| \\ c = |v(A_i)^c \cap v(A_j)| & d = |v(A_i)^c \cap v(A_j)^c| \end{array} \ . \tag{2}$$

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 163–170, 2006.

Parameters a and d count -in a positive and negative sense- the common characteristics, and b and c the difference characteristics.

From similarity and dissimilarity parameters are defined many similarity indexes for crisp sets. We will focus our attention on *Jackard's coefficient*, *simple matching coefficient* and *Rao's coefficient*.

Definition 3. *Jackard's coefficent*

$$s_j(A_i, A_j) = \frac{a}{a+b+c} \ . \tag{3}$$

Definition 4. *Simple matching coefficent*

$$s_m(A_i, A_j) = \frac{a+d}{a+b+c+d} \ . \tag{4}$$

Definition 5. *Rao's coefficient*

$$s_r(A_i, A_j) = \frac{a}{a+b+c+d} \ . \tag{5}$$

2 General Definitions

As an extension of (1) FFCM represents each object A_i as a fuzzy subset $\tilde{A}_i \in \tilde{\wp}(E) = \{\tilde{A} : \ \tilde{A} \subset E\}$ of the set of features E namely

$$A_i \longrightarrow \tilde{A}_i = (\mu_{\tilde{A}_i}(P_1), \mu_{\tilde{A}_i}(P_2), \ldots, \mu_{\tilde{A}_i}(P_m)) \in [0,1]^m \ . \tag{6}$$

Definition 6. *Fuzzy similarity measure*

$$\begin{array}{l} s : \tilde{P}(E) \times \tilde{P}(E) \quad \longrightarrow \quad [0,1] \\ \textit{is a fuzzy similarity measure} \end{array} \Longleftrightarrow \begin{cases} \forall \tilde{A}, \tilde{B} \quad 0 \leq s(\tilde{A}, \tilde{B}) \leq 1 \\ s(\tilde{A}, \tilde{B}) = s(\tilde{B}, \tilde{A}) \end{cases} \ . \tag{7}$$

If, moreover, verifies the reflexive property ($\forall \tilde{A} \quad s(\tilde{A}, \tilde{A}) = 1$), it can be interpreted as a *proximity fuzzy relation* in E (see [2, 3]). These kind of fuzzy relations are very important in fuzzy clustering processes (see [9]).

Definition 7. *Metric distance* [1]

$$\begin{array}{l} d : \tilde{P}(E) \times \tilde{P}(E) \quad \rightarrow \quad [0,1] \\ \textit{is a metric distance} \end{array} \Longleftrightarrow \begin{cases} d \textit{ verifies minimality, symmetric} \\ \textit{and triangular inequality.} \end{cases} \ . \tag{8}$$

Definition 8. *Similarity measure associated to a metric distance*[2]

$$\begin{array}{l} s : \tilde{P}(E) \times \tilde{P}(E) \ \rightarrow \ [0,1] \textit{ is a similarity} \\ \textit{measure associated to a metric distance } d \end{array} \Longleftrightarrow \begin{cases} \exists f \textit{ decreasing in } [0,1] \\ s = f(d) \ (s = 1 - d) \end{cases} \ . \tag{9}$$

From this point of view, each object is associated to a n-dimensional vector. Very strong properties are fulfilled but without taking in account the fuzzy character of the data what can lead to undesirable properties. For instance, if $s = 1 - d$, two objects with the maximum uncertainty are judged as completely similar because the distance is 0 thus the similarity is 1.

[1] The same properties that in classical geometry. As are defined in $[0,1]$ must be normalized. Ex: Ecludian, L^p, Hamming, Minkowski, Mahalanobis, etc.

[2] Any similarity measure associated to a metric distance is a fuzzy similarity measure.

3 Classical Fuzzy Similarity Indexes

A very short summary of the *classical* perspective (see [5])-in which some crisp properties are applied before generalizing to fuzzy indexes- is presented to contrast with the further on proposed *generalized* indexes, in which no apriorism is made. From now on t, c and n mean a t-norm, a t-conorm and a negation respectively (see [4]).

From the crisp property $a + b + c = |v(A_i) \cup v(A_j)|$ the following definition results as a generalization of (3)

Definition 9. *Classical fuzzy Jackard's coefficient*

$$s_{cj}(\tilde{A}, \tilde{B}) = \frac{|\tilde{A} \cap \tilde{B}|}{|\tilde{A} \cup \tilde{B}|} = \frac{\sum_{k=1}^{m} t(\mu_{\tilde{A}}(P_k), \mu_{\tilde{B}}(P_k))}{\sum_{k=1}^{m} c(\mu_{\tilde{A}}(P_k), \mu_{\tilde{B}}(P_k))} \quad . \tag{10}$$

Next definition generalizes (4). It is a consequence that parameters b and c are the sum of the absolute values of the differences between the components of the respective binary vectors and, applying crisp properties $a+b+c+d = m$ because the elements that define a, b, c and d are disjointed.

Definition 10. *Classical fuzzy simple matching coefficient*

$$s_{cm}(\tilde{A}, \tilde{B}) = 1 - \frac{1}{m} \sum_{k=1}^{m} |\mu_{\tilde{A}}(P_k) - \mu_{\tilde{B}}(P_k)| \quad . \tag{11}$$

Note 1. $s_{cm}(\tilde{A}, \tilde{B}) = 1 - d_H(\tilde{A}, \tilde{B})$ where d_H is the normalized distance of Hamming.

Similarly, next definition is a fuzzy extension of (5).

Definition 11. *Classical fuzzy Rao's coefficient*

$$s_{cr}(\tilde{A}, \tilde{B}) = \frac{1}{m} \sum_{k=1}^{m} t(\mu_{\tilde{A}}(P_k), \mu_{\tilde{B}}(P_k)) \quad . \tag{12}$$

Note 2. Property $a + b + c + d = m$ is not verified if the set image of v is $\tilde{\wp}(E)$.

Proposition 1. *The following properties hold:*

1. *s_{cm} is reflexive and symmetric (fuzzy proximity relation).*
2. *$\forall$ t t-norm, s_{cr} is a fuzzy similarity measure but no reflexive.*

Proof (of proposition). The first property is trivial. The biggest t-norm of the minimum, proves the second one:

$$s_r(\tilde{A}, \tilde{A}) = \frac{1}{m} \sum_{k=1}^{m} \min(\mu_{\tilde{A}}(P_k), \mu_{\tilde{A}}(P_k)) = \frac{1}{m} \sum_{k=1}^{m} \mu_{\tilde{A}}(P_k) \leq 1 \qquad \square$$

4 Generalized Fuzzy Similarity Indexes

The general aim of this section is to generalize the whole set of crisp similarity indexes without applying any crisp property, and proving that reliable properties are fulfilled. The following definitions and propositions generalize a set of fuzzy similarity indexes -Generalized Tversky Index (see [7, 8])- which represent a fuzzified restraint set of crisp indexes. This set contains as a particular case Jackard's coefficient but Rao's coefficient or simple matching coefficient.

Definition 12. *Fuzzy similarity and dissimilarity parameters*

Equivalently as the crisp set model we introduce the similarity and dissimilarity parameters in the following form:

$$\begin{aligned} a &= |\tilde{A}_i \cap \tilde{A}_j| = \sum_{k=1}^{m} t(\mu_{\tilde{A}_i}(P_k), \mu_{\tilde{A}_j}(P_k)) \\ b &= |\tilde{A}_i \cap \tilde{A}_j{}^c| = \sum_{k=1}^{m} t(\mu_{\tilde{A}_i}(P_k), n(\mu_{\tilde{A}_j}(P_k))) \\ c &= |\tilde{A}_i{}^c \cap \tilde{A}_j| = \sum_{k=1}^{m} t(n(\mu_{\tilde{A}_i}(P_k)), \mu_{\tilde{A}_j}(P_k)) \\ d &= |\tilde{A}_i{}^c \cap \tilde{A}_j{}^c| = \sum_{k=1}^{m} t(n(\mu_{\tilde{A}_i}(P_k)), n(\mu_{\tilde{A}_j}(P_k))) \end{aligned} \quad . \tag{13}$$

Definition 13. and Definition 14. are considered suitable properties for generalized fuzzy similarity indexes.

Definition 13. *Functionality*

$$\begin{array}{ll} \exists \quad f : \tilde{P}(E) \times \tilde{P}(E) \times \tilde{P}(E) \times \tilde{P}(E) & \longrightarrow \quad \mathbf{R} \\ \forall \tilde{A}, \tilde{B}, \tilde{C}, \tilde{D} \in \tilde{P}(E) & s(\tilde{A}, \tilde{B}) = f(\tilde{A} \cap \tilde{B}, \tilde{A}^c \cap \tilde{B}^c, \tilde{A} - \tilde{B}, \tilde{B} - \tilde{A}) \end{array} \quad . \tag{14}$$

Definition 14. *Monotony*

$\forall \tilde{A}, \tilde{B}, \tilde{C}, \tilde{D} \in \tilde{P}(E)$ *must be verified:*

$$\left.\begin{array}{l} \tilde{A} \cap \ \tilde{C} \subset \tilde{A} \cap \tilde{B} \\ \tilde{A}^c \cap \ \tilde{C}^c \subset \tilde{A}^c \cap \tilde{B}^c \\ \tilde{A} - \tilde{B} \subset \tilde{A} - \tilde{C} \\ \tilde{B} - \tilde{A} \subset \tilde{C} - \tilde{A} \end{array}\right\} \quad \Longrightarrow \quad s(\tilde{A}, \tilde{B}) \geq s(\tilde{A}, \tilde{C}) \ . \tag{15}$$

The previous conditions mean that $\tilde{A}$ is more similar to $\tilde{B}$ than to $\tilde{C}$ if the shared characteristics -in a positive or negative sense- of $\tilde{A}$ and $\tilde{B}$ contain those of $\tilde{A}$ and $\tilde{C}$. Moreover, is needed that the characteristics which differentiate $\tilde{A}$ and $\tilde{B}$ were less than those of $\tilde{A}$ and $\tilde{C}$.

Due to the analogy between the crisp and fuzzy notations it could be thought that some condition was superfluous because can be deduced from the others. Next example prove the necessity of all conditions.

Example 1. Suppose that $\tilde{A}$, $\tilde{B}$ and $\tilde{C}$ are characterized by only one feature P with the following values: $\mu_{\tilde{A}}(P) = 0.7$, $\mu_{\tilde{B}}(P) = 0.8$ and $\mu_{\tilde{C}}(P) = 0.4$.

Using the usual negation $n(x) = 1 - x$ thus $\mu_{\tilde{A}^c}(P) = 1 - \mu_{\tilde{A}}(P) = 0.3$, $\mu_{\tilde{B}^c}(P) = 1 - \mu_{\tilde{B}}(P) = 0.2$ and $\mu_{\tilde{C}^c}(P) = 1 - \mu_{\tilde{C}}(P) = 0.6$. To calculate the membership function of the intersection a t-norm is needed. We have chosen the most usual: the minimum of Zadeh. We obtain

$$\min(\mu_{\tilde{A}}(P), \mu_{\tilde{C}}(P)) = 0.4 \leq 0.7 = \min(\mu_{\tilde{A}}(P), \mu_{\tilde{B}}(P))$$

$$\min(\mu_{\tilde{A}}(P), 1 - \mu_{\tilde{B}}(P)) = 0.2 \leq 0.6 = \min(\mu_{\tilde{A}}(P), 1 - \mu_{\tilde{C}}(P))$$

$$\min(\mu_{\tilde{B}}(P), 1 - \mu_{\tilde{A}}(P)) = 0.3 \leq 0.3 = \min(\mu_{\tilde{C}}(P), 1 - \mu_{\tilde{A}}(P))$$

Therefore, first, third and fourth conditions are fulfilled. On the other hand

$$\min(1 - \mu_{\tilde{A}}(P), 1 - \mu_{\tilde{C}}(P)) = 0.3 \not\leq 0.2 = \min(1 - \mu_{\tilde{A}}(P), 1 - \mu_{\tilde{B}}(P))$$

Therefore, the second condition is not verified.

Definition 15. *Generalized simple matching coefficient*

$$s_{gm}(\tilde{A}, \tilde{B}) = \frac{|\tilde{A} \cap \tilde{B}| + |\tilde{A}^c \cap \tilde{B}^c|}{|\tilde{A} \cap \tilde{B}| + |\tilde{A}^c \cap \tilde{B}^c| + |\tilde{A} - \tilde{B}| + |\tilde{B} - \tilde{A}|} \ . \tag{16}$$

Definition 16. *Generalized Rao's coefficient*

$$s_{gr}(\tilde{A}, \tilde{B}) = \frac{|\tilde{A} \cap \tilde{B}|}{|\tilde{A} \cap \tilde{B}| + |\tilde{A}^c \cap \tilde{B}^c| + |\tilde{A} - \tilde{B}| + |\tilde{B} - \tilde{A}|} \ . \tag{17}$$

Proposition 2. *s_{gm} verifies functionality and monotony.*

Proof (of proposition). The functionality is trivial from its definition.
From the hypothesis of monotony: $\forall \tilde{A}, \tilde{B}, \tilde{C}, \tilde{D} \in \tilde{P}(E)$

$$\left\{\begin{array}{c} \tilde{A} \cap \tilde{C} \subset \tilde{A} \cap \tilde{B} \\ \tilde{A}^c \cap \tilde{C}^c \subset \tilde{A}^c \cap \tilde{B}^c \\ \tilde{A} - \tilde{B} \subset \tilde{A} - \tilde{C} \\ \tilde{B} - \tilde{A} \subset \tilde{C} - \tilde{A} \end{array}\right. \Longrightarrow \left\{\begin{array}{c} |\tilde{A} \cap \tilde{C}| \leq |\tilde{A} \cap \tilde{B}| \\ |\tilde{A}^c \cap \tilde{C}^c| \leq |\tilde{A}^c \cap \tilde{B}^c| \\ |\tilde{A} - \tilde{B}| \leq |\tilde{A} - \tilde{C}| \\ |\tilde{B} - \tilde{A}| \leq |\tilde{C} - \tilde{A}| \end{array}\right. \Longrightarrow$$

$$\Longrightarrow \left\{\begin{array}{c} |\tilde{A} \cap \tilde{B}||\tilde{A} - \tilde{C}| \geq |\tilde{A} \cap \tilde{C}||\tilde{A} - \tilde{B}| \\ |\tilde{A}^c \cap \tilde{B}^c||\tilde{A} - \tilde{C}| \geq |\tilde{A} - \tilde{B}||\tilde{A}^c \cap \tilde{C}^c| \\ |\tilde{A} \cap \tilde{B}||\tilde{C} - \tilde{A}| \geq |\tilde{B} - \tilde{A}||\tilde{A} \cap \tilde{C}| \\ |\tilde{A}^c \cap \tilde{B}^c||\tilde{C} - \tilde{A}| \geq |\tilde{B} - \tilde{A}||\tilde{A}^c \cap \tilde{C}^c| \end{array}\right. \Longrightarrow$$

$$\Longrightarrow |\tilde{A} \cap \tilde{B}||\tilde{A} - \tilde{C}| + |\tilde{A}^c \cap \tilde{B}^c||\tilde{A} - \tilde{C}| + |\tilde{A} \cap \tilde{B}||\tilde{C} - \tilde{A}| + |\tilde{A}^c \cap \tilde{B}^c||\tilde{C} - \tilde{A}| \geq$$

$$\geq |\tilde{A} \cap \tilde{C}||\tilde{A} - \tilde{B}| + |\tilde{A}^c \cap \tilde{C}^c||\tilde{A} - \tilde{B}| + |\tilde{A} \cap \tilde{C}||\tilde{B} - \tilde{A}| + |\tilde{A}^c \cap \tilde{C}^c||\tilde{B} - \tilde{A}| \iff$$

$$\iff |\tilde{A} \cap \tilde{B}||\tilde{A} \cap \tilde{C}| + |\tilde{A}^c \cap \tilde{B}^c||\tilde{A} \cap \tilde{C}| + |\tilde{A} \cap \tilde{B}||\tilde{A}^c \cap \tilde{C}^c| + |\tilde{A}^c \cap \tilde{C}^c||\tilde{A}^c \cap \tilde{B}^c| +$$

$$+ |\tilde{A} \cap \tilde{B}||\tilde{A} - \tilde{C}| + |\tilde{A}^c \cap \tilde{B}^c||\tilde{A} - \tilde{C}| + |\tilde{A} \cap \tilde{B}||\tilde{C} - \tilde{A}| + |\tilde{A}^c \cap \tilde{B}^c||\tilde{C} - \tilde{A}| \geq$$

$$\geq |\tilde{A}\cap\tilde{B}||\tilde{A}\cap\tilde{C}| + |\tilde{A}^c\cap\tilde{C}^c||\tilde{A}\cap\tilde{B}| + |\tilde{A}\cap\tilde{C}||\tilde{A}^c\cap\tilde{B}^c| + |\tilde{A}^c\cap\tilde{C}^c||\tilde{A}^c\cap\tilde{B}^c| +$$
$$+|\tilde{A}\cap\tilde{C}||\tilde{A}-\tilde{B}| + |\tilde{A}^c\cap\tilde{C}^c||\tilde{A}-\tilde{B}| + |\tilde{A}\cap\tilde{C}||\tilde{B}-\tilde{A}| + |\tilde{A}^c\cap\tilde{C}^c||\tilde{B}-\tilde{A}| \iff$$

$$\iff \quad \frac{|\tilde{A}\cap\tilde{B}| + |\tilde{A}^c\cap\tilde{B}^c|}{|\tilde{A}\cap\tilde{B}| + |\tilde{A}^c\cap\tilde{B}^c| + |\tilde{A}-\tilde{B}| + |\tilde{B}-\tilde{A}|} \geq$$

$$\geq \frac{|\tilde{A}\cap\tilde{C}| + |\tilde{A}^c\cap\tilde{C}^c|}{|\tilde{A}\cap\tilde{C}| + |\tilde{A}^c\cap\tilde{C}^c| + |\tilde{A}-\tilde{C}| + |\tilde{C}-\tilde{A}|} \iff \quad s(\tilde{A},\tilde{B}) \geq s(\tilde{A},\tilde{C})$$

□

Proposition 3. *s_{gr} verifies functionality and monotony.*

Proof (of proposition). Is identically to Proposition 2. □

Proposition 4. *The following properties hold:*

1. *$s_{gr} \leq s_{gm}$.*
2. *s_{gm} and s_{gr} are fuzzy similarities measures.*
3. *s_{gm} is reflexive if the associated t-norm verifies the noncontradiction principle.*

Proof (of proposition). The first and second ones are trivial. For the third:
$s_{gm}(\tilde{A},\tilde{A}) = \dfrac{|\tilde{A}\cap\tilde{A}| + |\tilde{A}^c\cap\tilde{A}^c|}{|\tilde{A}\cap\tilde{A}| + |\tilde{A}^c\cap\tilde{A}^c| + |\tilde{A}-\tilde{A}| + |\tilde{A}-\tilde{A}|} = 1 \iff$
$\iff \quad |\tilde{A}-\tilde{A}| = |\tilde{A}\cap\tilde{A}^c| = 0$ *what means* $\forall x \in [0,1] \quad t(x,1-x) = 0$ *thus the t-norm must verify the noncontradiction principle.* □

The following two propositions show the behaviour of s_{gm} and s_{gr} in prototypical and maximum uncertainty conditions.

Proposition 5. *Prototypical conditions*
$\tilde{A} = (1,1,\ldots,1) \implies \forall \tilde{B}\ s_{gm}(\tilde{A},\tilde{B})$ *does not depend on the t-norm.*

Proof (of proposition). As $\forall t$ *t-norm,* $\forall x \in [0,1] \quad t(1,x) = t(x,1) = x$ *and* $t(0,x) = t(x,0) = 0$, *thus we conclude that* $s_{gm}(\tilde{A},\tilde{B}) = \dfrac{\sum_{k=1}^{m} \mu_{\tilde{B}}(P_k)}{m}$ □

With the same hypothesis this proposition is obviously true for Generalized Rao's coefficient.

Corollary 1. *If* $\tilde{A} = \tilde{B} = (1,1,\ldots,1) \implies s_{gm}(\tilde{A},\tilde{B}) = s_{gr}(\tilde{A},\tilde{B}) = 1$

Proof (of corollary). It is trivial. □

The following proposition solves the question proposed in Sect. 2 about the incoherent behaviour in considering absolutely similar two objects with the maximum uncertainty.

Proposition 6. *Maximum uncertainty conditions*
$\tilde{A} = (0.5, 0.5, \ldots, 0.5) \implies \forall t\ \forall \tilde{B} \quad s_{gm}(\tilde{A}, \tilde{B}) = 0.5$

Proof (of proposition). If $a = \sum_{k=1}^{m} t(0.5, \mu_{\tilde{B}}(P_k)) + t(1 - 0.5, 1 - \mu_{\tilde{B}}(P_k))$ *and*
$b = \sum_{k=1}^{m} t(0.5, \mu_{\tilde{B}}(P_k)) + t(1-0.5, 1-\mu_{\tilde{B}}(P_k)) + t(0.5, 1-\mu_{\tilde{B}}(P_k)) + t(1-0.5, \mu_{\tilde{B}}(P_k))$
It is clear that $b = 2a$ *therefore* $s_{gm}(\tilde{A}, \tilde{B}) = a/b = 0.5$ □

Unfortunately, this proposition is not true for the Rao's coefficient.

4.1 Similarity Surfaces

Let $E = \{P_1, P_2\}$ be, and $\mu_{\tilde{A}}(P_1)$ and $\mu_{\tilde{A}}(P_2)$ two reference values that define a fuzzy number $\tilde{A}$. Let $\mu_{\tilde{B}}(P_1)(x)$ and $\mu_{\tilde{B}}(P_2)(y)$ be with $(x, y) \in D_1 \times D_2$ two fuzzy numbers assigned by an expert where D_1 and D_2 are the domains of $\mu_{\tilde{B}}(P_1)$ and $\mu_{\tilde{B}}(P_2)$. In these conditions $s(\tilde{A}, \tilde{B}) = s(\tilde{A}, \tilde{B})(x, y)$, which representation[3] is a surface in $\mathbf{R^3}$.

Example 2. Suppose that $\mu_{\tilde{B}}(P_1)(x)$ and $\mu_{\tilde{B}}(P_2)(y)$ are given by two triangular fuzzy numbers $T_1 = (12, 16, 20)$ and $T_2 = (1, 3, 5)$ which representation is very well known (see [1]). Note that only points 16 and 3 have membership value equal to 1 thus the prototypical conditions are attained at the pair $(16, 3)$.

Figure 1 shows that with reference prototypical conditions the value of the similarity attaint 1 for $(\mu_{\tilde{A}}(16), \mu_{\tilde{A}}(3)) = (1, 1)$ as is proved in Corollary 1.

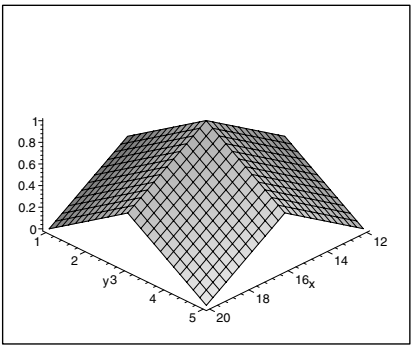


Fig. 1. s_{gm} with prototypical reference conditions

Figure 2 shows that with maximum uncertainty conditions all the similarity values are 0.5 as is proved in Proposition 6.

Figure 3 merely shows the behaviour with any other membership values. In this case $\tilde{A} = (0.85, 0.31)$ and using the t-norm of the minimum of Zadeh.

[3] Other kind of similarity indexes can be defined between fuzzy numbers (see [6]).

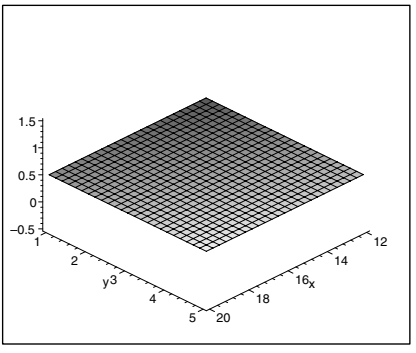


Fig. 2. s_{gm} with maximum uncertainty reference conditions

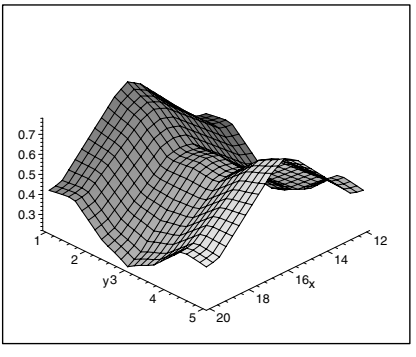


Fig. 3. s_{gm} calculated with minimum of Zadeh t-norm and $\tilde{A} = (0.85, 0.31)$

References

1. Dubois, D., Prade, H.: Fuzzy Sets and Systems, Theory and Applications. San Diego: Academic Press. 1980
2. Klir, G. J., Folger, T. A.: Fuzzy Sets, Uncerttainty, and Information. New Jersey: Prentice Hall. (1988)
3. Kundu, S.: Similarity relations, fuzzy linear orders, and fuzzy partial orders. Fuzzy Sets and Systems. **109** (2000) 419–428
4. Lowen, R.: Fuzzy Set Theory. Dordrecht: Kluwer Academic Publishers. (1996)
5. Miyamoto, S.: Fuzzy Sets on Information Retrieval and Cluster Analysis. Dordrecht: Kluwer Academic Publishers. (1990)
6. Peneva, V., Popchev, I.: Comparison of clusters from fuzzy numbers. Fuzzy Sets and Systems. **97** (1998) 75–81
7. Tolias, Y. A., Panas, S. M., Tsoukalas, L. H.: Generalized fuzzy indices for similarity matching. Fuzzy Sets and Systems. **120** (2001) 255–270
8. Tversky, A.: Features of similarity. Pshicology Review. **34** (1977) 286–377
9. Yang, M. S., Shih, H. M.: Cluster analysis based on fuzzy relations. Fuzzy Sets and Systems. **120** (2001) 197–212

Environmental Time Series Prediction by Improved Classical Feed-Forward Neural Networks

Maurizio Campolo[1], Narcís Clara[2], and Carlo Francesco Morabito[1]

[1] Università Mediterranea di Reggio Calabria,
Dipartimento di informatica, matematica, elettronica e trasporti
campolo@ing.unirc.it, morabito@unirc.it
[2] Universitat de Girona,
Departament d'informàtica i matemàtica aplicada
narcis.clara@udg.es

Abstract. The water quality at the issue of a wastewater treatment plant (WWTP) is a complex work because of its complexity and variability when conditions suddenly change. Two main techniques has been used to improve classical feed-forward neural network. First, a classical adaptative gradient learning rule has been complemented with a Kalman learning rule which is especially effective for noisy behavioral problems. Second, two independent variable selection components -based on genetic algorithms and fuzzy ranking- have been implemented to try to improve performance and generalization. The global study shows that reliable results are obtained which permit to guarantee that neural networks are a confidence tool on this subject.

1 The Set of Data

The set of data used for this study were obtained by simulation of the processes involved the most extensively system used for wastewater treatment (the activated sludge system). This system involves two main components: biochemical reactors, where biological reactions take place to remove pollutants, and settling units to clarify the treated wastewater (Fig. 1). The IAWQ's Activated Sludge Model ♯1 (see [4]) was chosen as the biological process model and the double-exponential settling velocity function (see [10]) is chosen as a fair representation of the settling process. These models are widely used and accepted by the scientific communities to model the process dynamics of WWTP. Three key operational parameters -controlled by the plant's operators- have been considered:aeration energy, pumping energy for activated sludge recycling and sludge purge.

The simulated database is formed by records chosen each 15 minutes during two weeks thus 1344 records. A period of rain makes more complex the whole set of data. Thirteen are the number of inputs; ten state variables in the affluent: t (time), readly biodegradable substrat (S_S), heterotrophic biomass (X_{BH}), Slowly biodegradable substrat (X_S), non-biodegradable particulate organic matter (X_I), nitrogen NH3+NH4 (S_{NH}), non-biodegradable soluble organic matter

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 171–178, 2006.

(S_I), soluble biodegradable organic nitrogen (S_{ND}), particulate biodegradable organic nitrogen (X_{ND}), inflow rate (Q); and three operational parameters: aeration energy (AirE), pumping energy (PumpE), and sludge purge (WAS). The choosing outputs, which describe the water quality, are the two main oxygen state variables in the effluent: biochemical organic matter (BOD) and chemical organic matter (COD) commonly defined as:

$$BOD = 0.25(S_S + X_S + 0.92(X_{BH} + X_{BA}))$$

$$COD = S_S + S_I + X_S + X_{BH} + X_{BA} + X_P + X_I$$

New variables X_{BA} and X_P represent atrophic biomass and particulate products resulting of the biomass death respectively. All the units are mg/l except for AirE (Kwh/d), PumpE (Kwh/d), WAS (Kg/d) and Q (m^3/day). The main statistical parameters are showed in Table 1. and Table 2. An schematic WWTP design (left) and complexity between input and output variables (right) is patent in Fig.1.

Table 1. Basic statistical descriptors for selected inputs variables

	S_S	X_{BH}	X_S	X_I	S_{NH}	S_I	S_{ND}	X_{ND}
Min	13.828	4.927	38.900	5.448	7.033	10	1.383	1.853
Max	120.011	42.745	293.814	109.831	50.000	30	12.001	16.071
Mean	60.533	24.446	177.965	42.049	27.945	27.76	6.053	9.190
StDev	21.178	8.573	57.481	21.610	8.632	5.51	2.118	3.223

Table 2. Basic statistical descriptors for operational parameters and outputs variables

	Q	AirE	PumpE	WAS	BOD	COD
Min	10000.0	6020.8	753.4	1930.5	2.101	33.61
Max	52126.4	8449.2	3847.1	3251.2	5.102	64.01
Mean	21329.8	7205.2	1709.3	2491.8	2.881	46.42
StDev	8997.45	708.08	701.7	304.45	0.658	3.680

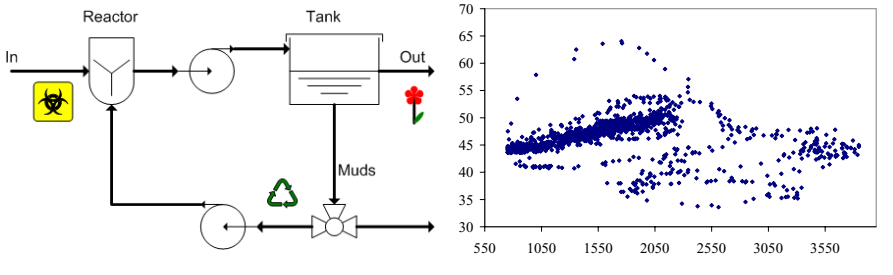


Fig. 1. Schematic WWTP design and COD versus PumpE

2 Methodology

The purpose of developing a neural model (see [2],[9]) is to produce a formula that captures essential relationships between inputs and outputs (y_i). Once developed, this formula is used to interpolate from a new set of inputs to corresponding predicted outputs ($\hat{y}_i$). In neural nets, this is called generalization.

Partition of the data set. The train set is a subset of the input data which is used to train (updated the weights of) the neural network. During training, the score on the test set determines when to stop building the network and it is also used to choose between hidden node candidates. Moreover, the test set is used as part of the model building process to prevent overfitting (the opposite of generalization). Although the explicit training of the model uses only the training set, heuristics involving the performance on the test set are used to guide choices during the construction of the model thus, it is common to use the validation data set -a subset of the input data that is different from de data used to build the network- to finally estimate model performance in a deployed environment. The last 20 records are used to prove well generalization out of the train-test-validation interval. The percentages of train, test and validation data are, respectively, the 50%, 30% and 20% of the remaining input data, all chosen randomly.

The neural network is constructed incrementally by adding hidden nodes, usually just one or two at a time. Each hidden node or pair of hidden nodes has its weights trained from several initializations. The best initialization is established in the network, and the all the weights to the output nodes of the net are retrained.

Variable selection. Effective variable selection can substantially improve model performance and generalization. The power of variable selection is the ability to find small synergistic subsets of variables which solve the problem as well as or better than the full set of measured variables.

Genetic algorithms (see [5]) are loosely based on some of the processes involved in biological evolution. Any genetic algorithm has a population of individuals which change from one generation to the next, usually by combining characteristics of two *parent* individuals to create a *child* individual. Every individual is assigned a fitness and the concept of *survival of the fittest* is implemented by selecting fitter parents more frequently than less fit parents. In our case, the individuals are sets of inputs variables. A set of inputs variables derives its fitness from how successful a model can be built based on just those variables. As the algorithm depends on the initialization values we have begin with different initializations and the variables which have consistently fail to be included in the final population are omitted.

The comparatione between the previous method and a fuzzy variable selection-based on a fuzzy ranking (FR)- makes more robust the whole variable selection process. This procedure has showed its well performance in other subjects (see [8]). Summarizing, the basic idea behind the method is to assess the flatness of

a fuzzy curve -defined by gaussian membership functions-characterizing a given input variables, since the output variable is scarcely influenced by the input variable if the related fuzzy curve is nearly flat. The most relevant variables has been taken in account to define the subset of chosen variables.

Learning rules. The adaptative learning rule uses back-propagated gradient information to guide an iterative line search algorithm (see [3]). The search direction in the weight space is modified by the previous search direction, a decay term -to avoid overfitting- is obtained by and heuristic algorithm which constructs a two dimensional grid with one axis associated with the parameter *weight decay* for the hidden layer and the other associated to the weight decay for the output layer. Finally, a random vector is added to the line search direction vector. During training, the learning rule modifies the weights in response to the training data. If left unchecked, the weights for a processing element can latch onto spurious information in the training data, such as data that does not represent a general trend in the input data. By slowly decaying the weights during the course of the training, only the general trends remain encoded in the weights.

Kalman filter learning rule considers the desired outputs to be the observations within a discrete state space transition framework. Standard non-linear Kalman filter theory is used to obtain the best estimate of the weights based on the stream of training data. Overfitting can be avoid by increasing the observation noise of the filter.

3 Results

Six methods has been implemented: adaptative gradient without variable selection (AG), adaptative gradient with genetic algorithm variable selection (AGG), adaptative gradient with fuzzy variable selection (AGF), Kalman filter with genetic algorithm variable selection (KG), Kalman filter with fuzzy variable selection (KF) and Kalman filter without variable selection (K). For each kind of methodology several analysis have been made in different conditions of noise. In Tab.3. and Tab.4. we only present the most indicative and performing. Acronyms: R=linear correlation between real and model outputs, RMS=*root mean square error*, S=score. *Mean absolute percentage errors*=$100\frac{\sum_{i=1}^{n}|y_i-\hat{y}_i|/y_i}{n}$ is calculated for the whole (E), training ($E1$), test ($E2$), validation ($E3$), out-interval (EG) and rain ($Erain$) sets of records. The real BOD versus predicted BOD(f) and generalization are plotted in Fig.2. and Fig.3. The real COD versus predicted COD(f) and generalization are plotted in Fig.4 and Fig.5. All the figures have been drawn using the results of the AG method. Similar graphics have been obtained for the rest.

Table 5. and Tab.6. show variable selection and average contribution -for some methods- for each input variable on the prediction of the outcome fuzzified from 1 (very little) to 7 (very high), respectively for BOD and COD. The acronym (r) means rain and reflects how the contribution change in the period of rain. Genetic algorithm variable selection and fuzzy variable selection are very similar for

Table 3. Main analytical results for BOD

	R	RMS	S	E	$E1$	$E2$	$E3$	EG	$Erain$
					BOD				
AG	0.9899	0.0946	0.9862	1.77%	1.72%	1.76%	1.92%	0.80%	3.83%
AGG	0.9914	0.0864	0.9934	2.25%	2.20%	2.20%	2.48%	3.11%	3.93%
AGF	0.9868	0.1133	0.9897	2.15%	2.23%	2.03%	2.06%	1.66%	4.27%
KG	0.9881	0.1027	0.9854	2.47%	2.44%	2.42%	2.65%	2.41%	2.94%
KF	0.9862	0.1152	0.9904	2.13%	2.04%	2.28%	2.23%	1.67%	4.20%

Table 4. Main analytical results for COD

	R	RMS	S	E	$E1$	$E2$	$E3$	EG	$Erain$
					COD				
AG	0.9455	1.2304	0.9551	1.77%	1.76%	1.81%	1.73%	0.59%	2.64%
AGG	0.9343	1.3327	0.9607	1.78%	1.78%	1.63%	1.97%	0.92%	2.48%
AGF	0.6370	2.8528	0.7336	3.41%	3.54%	3.16%	3.35%	0.63%	5.97%
KG	0.9278	1.3893	0.9618	1.79%	1.79%	1.60%	1.98%	1.63%	3.31%
K	0.9563	1.0919	0.9755	1.32%	1.29%	1.30%	1.43%	1.34%	2.14%

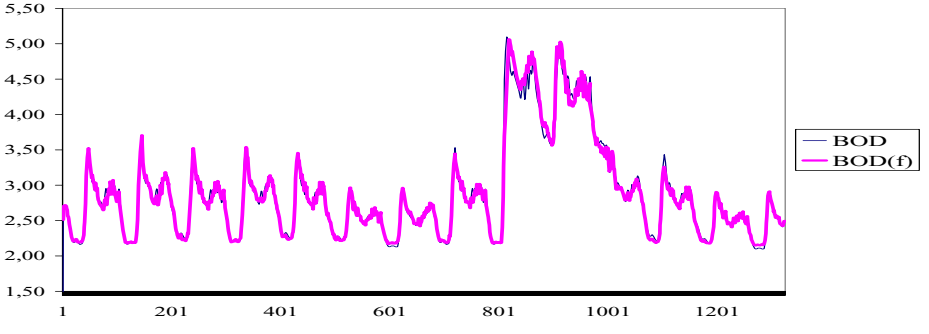


Fig. 2. BOD: real versus predicted

BOD, and absolutely agree in the most important input variables: Q and WAS. On the other hand, for COD seem very different but, in fact, the differences on the fuzzy ranking are little significative. The higher partial derivatives of output variables respect to input variables are in brackets on the same table.

Notes and Comments. Similar studies using real data were carried out by other authors (see [1]). However, the lack of reliable on-line sensors and the delay of some analytical results make difficult to build accurate predictive neural networks.

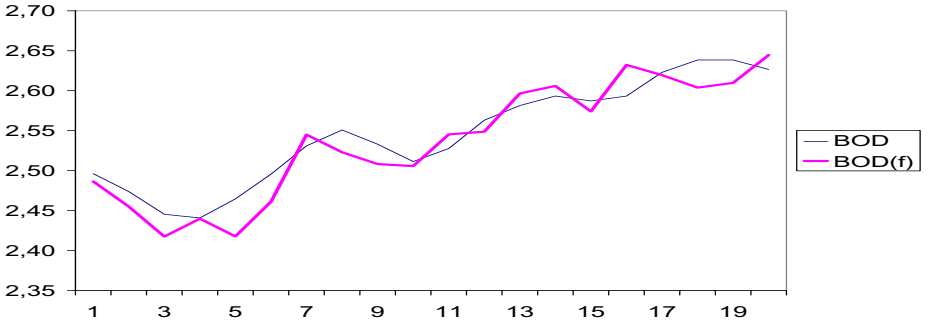


Fig. 3. BOD: 20 new data

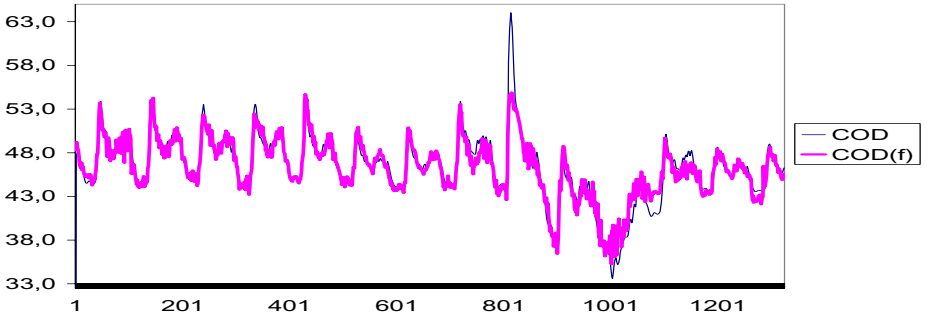


Fig. 4. COD: real versus predicted

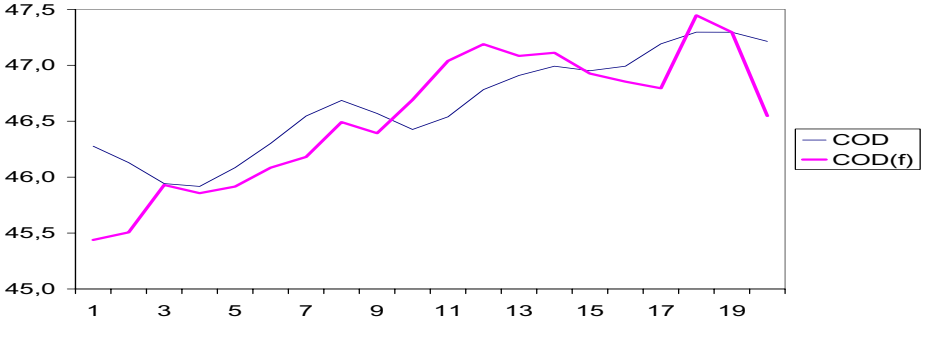


Fig. 5. COD: 20 new data

Related to that matter other approaches have been made (see [6, 7]) about the possibility of giving some qualitative indicators in order to design an early warning system alerting on the incoming of critical working phenomena such as heavy period of rain.

Table 5. Fuzzy ranking, contribution and highest sensitivity for BOD

	t	S_S	X_{BH}	X_S	X_I	S_{NH}	S_I	S_{ND}	X_{ND}	Q		AirE	PumpE	WAS	
							BOD								
FR	.2	.2	.1	.2	.2	.3	.8	.2	.1		1	.3	.8		.9
AG	4	4	4	5	4	4	5	4	4	(0.64)	5	4	4	(0.43)	5
AG(r)	4	4	4	3	6	4	3	4	4		5	4	4		7
AGF	-	-	-	-	-	-	4	-	-	(0.35)	4	-	3	(0.32)	5
AGF(r)	-	-	-	-	-	-	5	-	-		5	-	3		7
KG	-	4	-	-	-	-	-	-	-	(0.35)	5	-	-	(0.48)	6
KG(r)	-	4	-	-	-	-	-	-	-		6	-	-		7

Table 6. Fuzzy ranking, contribution and highest sensitivity for COD

	t	S_S	X_{BH}	X_S	X_I	S_{NH}	S_I	S_{ND}	X_{ND}	Q	AirE	PumpE	WAS	
							COD							
FR	.7	.9	1	1	1	.8	.9	.9	1	.7	.8	.7		.7
AG	4	4	3	6	(3.1)	5	4	(6)	4	4	3	4	4	4	4
AG(r)	4	4	3	5		5	4		4	4	4	4	4	4	4
AGF	-	4	(1.4)	4	4		4	-	4	4	4	-	-	-	-
AGF(r)	-	4	5	4		3	-		4	4	4	-	-	-	-
KG	4	-	-	-		-	-		-	4	-	4	5	4	(0.62) 4
KG(r)	4	-	-	-		-	-		-	4	-	4	5	5	4

References

1. Belanche, L.A., Valdés, J.J., Comas, J., Roda, I.R., Poch, M.: Towards a model of input-output behaviour of wastewater treatment plants using soft computing techniques. Environmental Modellin and Software. **14** (1999) 409–419
2. Bishop, Christoper M,.: Neural Networks for Pattern Recognition. Clarendon Press. (1995)
3. Fahlman, Scott E., Lebiere, C.: The Cascade-Correlation Learning Architecture. Advances in Neural Information Processing Systems 2. Morgan Kaufmann. (1988)
4. Henze, M., Grady, Jr., Gujer, W., Marais, G.v.R, and Matsuo, T.: Activated Sludge Model No. 1. IAWQ. Scientific nad Technical Report No. 1, IAWQ. London. (1987)
5. Koza, John R.: Genetic Programming. MIT Press. (1993)
6. Mappa, G., Salvi, G., Tagliaferri, R.: A Fuzzy Neural Network for the On-Line Detection of the B.O.D. Proceedings of the 7-th Italian Workshop on Neural Nets WIRN Vietri '95. M. Marinaro and R. Tagliaferri Ed.s. World Scientific Publishing. (1996) 305–310
7. Mappa, G., Tagliaferri, R., Tortora, D.: On line Monitoring based on Fuzzy Neural Techniques applied to existing hardware in Wastewater Treatment Plants. Proceedings of the International Sympsosium on Intelligent Systems, AMSE-ISIS'97, F. C. Morabitol et al. Ed.s, IOS Press. (1997) 339–342

8. Morabito, C.F., Versaci, M.: The Use of Fuzzy Curves for the Reconstruction of the Plasma Shape and the Selection of the Magnetic Sensors. Fusion Technology. **1** (1996) 937–940
9. NeuralWare.: Neural Computing: A Technology Handbook for Professional II/PLUS, and NeuralWorks Explorer. NeuralWare. (1993)
10. Takacs, I., Patry, G.G. and Nolasco, D.: A dynamic model of the clarification thickening process. Wat. Res. **25** (1991) 1263–1271

New Similarity Rules for Mining Data

Vito Di Gesù[1,2] and Jerome H. Friedman[2]

[1] Università di Palermo, DMA, Italy
[2] Stanford University, Department of Statistics , Stanford, CA , USA
digesu@math.unipa.it, jhf@stanford.edu

Abstract. Variability and noise in data-sets entries make hard the discover of important regularities among association rules in mining problems. The need exists for defining flexible and robust similarity measures between association rules. This paper introduces a new class of similarity functions, SF's, that can be used to discover properties in the feature space X and to perform their grouping with standard clustering techniques. Properties of the proposed SF's are investigated and experiments on simulated data-sets are also shown to evaluate the grouping performance.

1 Introduction

The evaluation of similarities between sets plays a considerable role in several applications. For example, the use of proximity and continuity has been introduced by the Gestalt school [1] to explain visual grouping. The association of abstract concepts has been discussed in [2]. Moreover, suitable definitions of similarity are necessary in content based pictorial retrieval problems [3]. Recently, the optimized combination of similarity functions has been considered to classify biological cells [4].

The necessity for introducing suitable SF's is a pressing problem also in genomic studies and data mining. For example, the distribution of exons within the genomic DNA has similarities to that of the dystrophin gene and it has been shown that the genomic structure of the utrophin gene is similar to the dystrophin gene [5]. Moreover, several algorithms to multiple sequence alignment are grounded on segment-based distances and similarities in genomic sequences [6, 7].

In this paper we consider the problem of finding similarities among rules defined in a given feature space, X. The definition of closeness between *rules* is an important task in order to compare them and, eventually, to perform their grouping. Informally, a rule represents a set of properties in the feature space X. Therefore, rules characterize a sub-set of X, elements of which have some common properties.

An association rule is an expression of the form $x \Rightarrow y$, where x and y are sets of items ($attribute = value$) such that no attribute appears more than once in $x \cup y$. Note that the meaning of the relation $\Rightarrow$ is: transactions that contain the items in x also tend to contain the items in y [8]. Intuitively, an association rule identifies a frequently occurring pattern of information in a database. Association

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 179–187, 2006.

rules can be generalized by replacing the sets of items with conjunctions of ($attribute = value$) equalities or validity range ($t_1 \leq attribute \leq t_2$) to include non-transactional data. Example are $(age = 40) \bigwedge (salary = 6000EURO) \Rightarrow (own_{home} = yes)$), $(25 \leq age \leq 40) \bigwedge (4500EURO \leq salary \leq 6000EURO) \Rightarrow (own_{home} = yes)$). The second example can be represented as a rectangle in the attribute space $(age, salary)$. In a bottom-up view, SF's can be than used to *discover* association rules and to *generalize* them by performing their grouping in *significant* clusters.

In Section 2 are summarized recent results in this area. Section 3 introduces the similarities proposed with their properties. Section 4 describes experiments on simulated data to assess and compare SF's. Section 5 describes how to group rules. Final remarks are given in Section 6.

2 Related Works

Discovery from rules requires deep interpretation from domain knowledge. In fact, one of the most important problems with rule induction methods is that it is very difficult for domain experts to check millions of rules generated from large data-sets. In [9], the use of multidimensional scaling has been considered to visualize rules similarity.

In [10] an Association Rule Clustering System, (ARCS), to group 2-D association rules is presented. A measure of similarity, based on bitwise operations in the attribute space, is defined by two common numeric measures assigned to each association rule: the *support*, $\frac{|xy|}{|X|}$ and the *confidence*, $\frac{|xy|}{|x|}$, note that the confidence is not symmetric. Where $|X|$ is the total number of tuples and xy indicates how often the items in x and y occur together in the same tuple.

The grouping of association rule is one of the most important procedures in data mining allowing manual inspection and support knowledge discovery after reducing significantly the number of rules. In [11] a normalized similarity to group association rules is presented. The measure is based on intersection and union operators named Market-Basket Probability distance. However, the similarity introduced is zero until rules have null intersection. An interesting feature of the proposed SF's is that they do not use any coordinate system, but only relative positions.

3 Basic Definitions

In the following, we will use the term *rule* instead of *association rule*. The feature space is a unit hyper-cube, $X \subset [0,1]^d$ ($\mathbf{x} \equiv (x_1, x_2, \ldots, x_d) \in X$ with $0 \leq x_1, x_2, \ldots, x_d \leq 1$). In this paper, rules are represented by rectangles. In the case of a categorical features, rules are Boolean conditions that state the *true* or the *false* of a given feature value. Otherwise, rules are relations defined on a numerical (integer, real) features space. In both cases rules can be represented as functions, $\mathbf{R} : X \rightarrow \{0,1\}$ that assign a truth value to each element, $\mathbf{x} \in X$, ($\mathbf{R}(\mathbf{x}) = 1$ if the rule is satisfied, 0 otherwise).

The set of all rules defined in X will be denoted by $\Re$. For sake of simplicity, we consider rules that are hyper-rectangles in X:

$$\mathbf{R} = \{\mathbf{x} \in X | a_{i_1} \leq x_i \leq a_{i_2}, \text{ for } i = 1, 2, \ldots, d\}$$

For short we will call them rectangles. These kind of rules have also been introduced in [12, 13].

Closeness between elements can be evaluated starting from SF. One of the main problems is the definition of suitable SF between rules in $\Re$. In general, a SF, σ, musts satisfy the following properties $\forall \mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3 \in \Re$:

$$\begin{array}{ll} a)\ \sigma(\mathbf{R}_1, \mathbf{R}_2) \geq 0 & (positivity) \\ b)\ \sigma(\mathbf{R}_1, \mathbf{R}_2) = \sigma(\mathbf{R}_2, \mathbf{R}_1) & (reflexivity) \end{array}$$

Moreover, the normalization condition $0 \leq \sigma \leq 1$ ($\sigma(\mathbf{R}_1, \mathbf{R}_2) = 1 \leftrightarrow \mathbf{R}_1 \equiv \mathbf{R}_2$) can be added. Usually, nothing can be said about the triangular inequality.

Note. *The transitivity property could be useful in comparisons and grouping problems. However, the transitivity implies an equivalence relation among elements of* $\Re$*, and this is not always satisfied.*

In the following we will denote by $\mathbf{R}_{12}$ the minimum rectangle that includes two rules $\mathbf{R}_1$, $\mathbf{R}_2$. $\mathbf{R}_{12}$ that is also named support (see Fig.1).

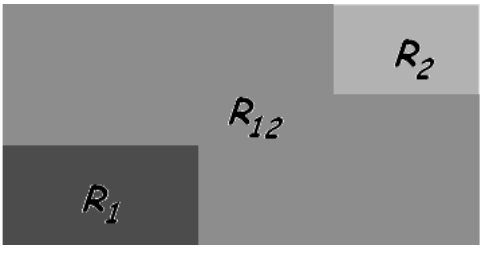


Fig. 1. Two rules and their support

In our case the feature space, X, is finite numerable, therefore the cardinality, $|\mathbf{R}|$, of a rule, $\mathbf{R}$, is the number of elements of X covered by it.

Depending on the SF considered, the cardinality of $\mathbf{R}$ will be normalized to the cardinality of the whole space X or to the cardinality of the support. It is clear that in both cases $0 \leq |\mathbf{R}| \leq 1$. Note that, the first kind of normalization does not dependent on the considered rules, while the second normalization is adaptive and will be sensitive to the positions and the cardinalities of the considered rules.

3.1 New Similarity Functions

Here, we introduce fours SF's that satisfy both positivity and reflexivity properties. The aim is to provide a measure of closeness that is insensitive to spatial features and more to their instantiations. In the following, $\mathbf{R}_{1\cap 2}$ and $\mathbf{R}_{1\cup 2}$ denote the intersection and the union of $\mathbf{R}_1$ and $\mathbf{R}_2$ respectively.

Definition 1. *Given the rules* $\mathbf{R}_1$ *and* $\mathbf{R}_2$ *the following SF's can be defined:*

$$\sigma_{min}(\mathbf{R}_1, \mathbf{R}_2) = 1 - (1 - \min(|\mathbf{R}_1|, |\mathbf{R}_2|)) \times e^{-|\mathbf{R}_1\ _2|} \quad (1)$$
$$\sigma_{ave}(\mathbf{R}_1, \mathbf{R}_2) = 1 - (1 - \tfrac{1}{2}(|\mathbf{R}_{1\cup 2}|)) \times e^{-|\mathbf{R}_1\ _2|} \quad (2)$$
$$\sigma_{union}(\mathbf{R}_1, \mathbf{R}_2) = \tfrac{1}{2}\eta_{12} \text{ where } \eta_{12} = |\mathbf{R}_{1\cup 2}| + |\mathbf{R}_{1\cap 2}| \quad (3)$$
$$\sigma_{cor}(\mathbf{R}_1, \mathbf{R}_2) = \tfrac{1+\eta_{12}}{2} \quad (4)$$
$$\text{where: } \eta_{12} = \frac{(|\mathbf{R}_1\ _2| - |\mathbf{R}_1||\mathbf{R}_2|)}{\sqrt{|\mathbf{R}_1||\mathbf{R}_2|(1-|\mathbf{R}_1|)(1-|\mathbf{R}_2|)}}$$

The term $e^{-|\mathbf{R}_1\ _2|}$ has been introduced to weight the effect due to the intersection of two rules. Both σ_{min} and σ_{ave} are sensible to the distribution model of the data covered by them. In particular, it is easy to see that their values increase when the distribution of the data covered by $\mathbf{R}_1$ and $\mathbf{R}_2$ are closer.

This property can be interesting in pattern matching applications, where it is interesting to see similarities among different patterns suited on different positions of the space. However, it must be pointed out that in these definitions the spatial closeness is taken in account.

Note that σ_{min}, σ_{ave}, and σ_{union} are normalized to the support, while σ_{cor} is normalized to the whole space X.

3.2 Properties of *SF*'s

A dissimilarity, δ, between rules can be defined from a normalized SF as it follows $\delta(\mathbf{R}_1, \mathbf{R}_2) = 1 - \sigma(\mathbf{R}_1, \mathbf{R}_2)$, with $\mathbf{R}_1, \mathbf{R}_2 \in \Re$. It is easy to show that δ is positive, symmetric and $\delta(\mathbf{R}_1, \mathbf{R}_2) = 0 \leftrightarrow \mathbf{R}_1 = \mathbf{R}_2$. Moreover:

Proposition 1. *A* δ *induced by* σ *is a distance iff* $\forall \mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3 \in \Re$*:*

$$c)\ \sigma(\mathbf{R}_1, \mathbf{R}_2) + \sigma(\mathbf{R}_2, \mathbf{R}_3) - 1 \leq \sigma(\mathbf{R}_1, \mathbf{R}_3) \text{ triangular inequality}$$

An SF that satisfies Prop.1 is said *Regular-SF*.

Proposition 2. *The SF,* σ_{union}*, satisfies the triangular inequality if* $\forall \mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3 \in \Re$, $\mathbf{R}_{1\cap 2} \equiv \mathbf{R}_{2\cap 3} \equiv \mathbf{R}_{1\cap 3} \equiv \phi$*, where* ϕ *is the empty set. In words, all rules are not intersecting.*

Proposition 3. *The SF,* σ_{union}*, satisfies the triangular inequality if* $\forall \mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3 \in \Re$, $\exists i \neq j$ and $i, j \in \{1, 2, 3\}$ *such that* $\mathbf{R}_i \equiv \mathbf{R}_j$*. In words, if at least two rules are equal.*

Fig. 2a,b show the rules configurations that satisfy Prop.s 2, 3.

Proposition 4. *The SF,* σ_{union}*, satisfies the triangular inequality if the normalization is performed respect to* $|X|$*.*

3.3 Experiments on the Triangular Inequality

To study the triangular inequality of the SF's we have performed an intensive simulation by generating 10^6 triplets of rules. The inequality has been checked

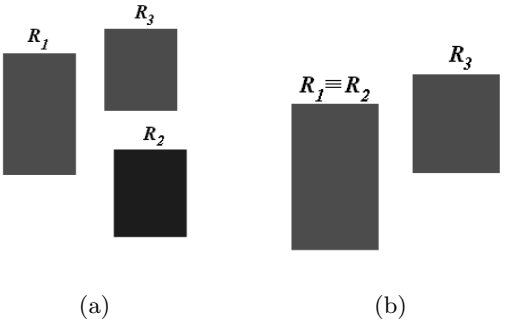


Fig. 2. Rules that satisfy the triangular inequality: (a) non intersecting rules (Prop. 2); (b) two coincident rules (Prop. 3)

in the case of *small rules* (the area of the rectangles was 0.1 the size of X) and *large rules* (the area of the rectangles was larger than 0.5 the size of X).

Experiments with small rules indicate that the triangular inequality fails 129500, 880, and 404501 times for σ_{cor}, σ_{min}, and σ_{ave} respectively; therefore these SF's do not satisfy the triangular inequality in general. Experiments with large rules indicates that that the triangular inequality fails 126850, 855, and 201410 times for σ_{cor}, σ_{min}, and σ_{ave} respectively. In both cases σ_{min} seems to be more regular than σ_{cor} and σ_{ave}. The σ_{union} the triangular inequality never failed. We claim that σ_{union} is a $Regular - SF$.

Table 1. Testing triangular inequality

Similarity	*Number of faults %*
σ_{cor}	12.9 (12.6)
σ_{union}	0
σ_{min}	0.08 (0.08)
σ_{ave}	40.4 (20.1)

Table 1 summarize the results of our experiments. The first column indicates the SF, the second column indicates the percentage of faults in the case of small (large) rules, the third column shows the upper bound of the error when the triangular inequality fails.

4 Experiments to Test New *SF*'s

Here, we consider $d = 2$, i.e. $X \subset [0,1]^2$. Experiments have been made by shifting two rectangular rules of the same size along the x-axis (see Fig. 3). The maximum of agreement is reached when the two rules are superimposed.

Elements of X are generated by a uniform distribution $\{x_i \sim [0,1]\}_1^2$, plus a number $N_c \geq 0$ of Gaussian clusters with the same variance and mean number of

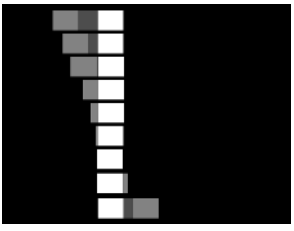


Fig. 3. Animation of the rules generation

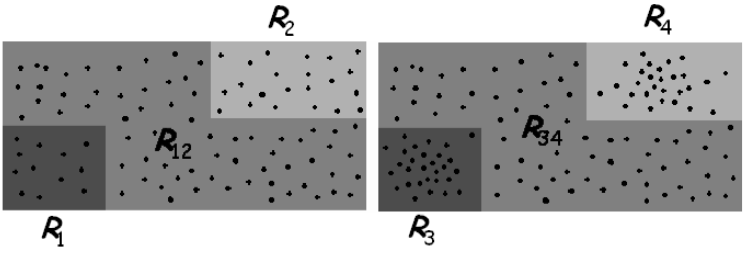


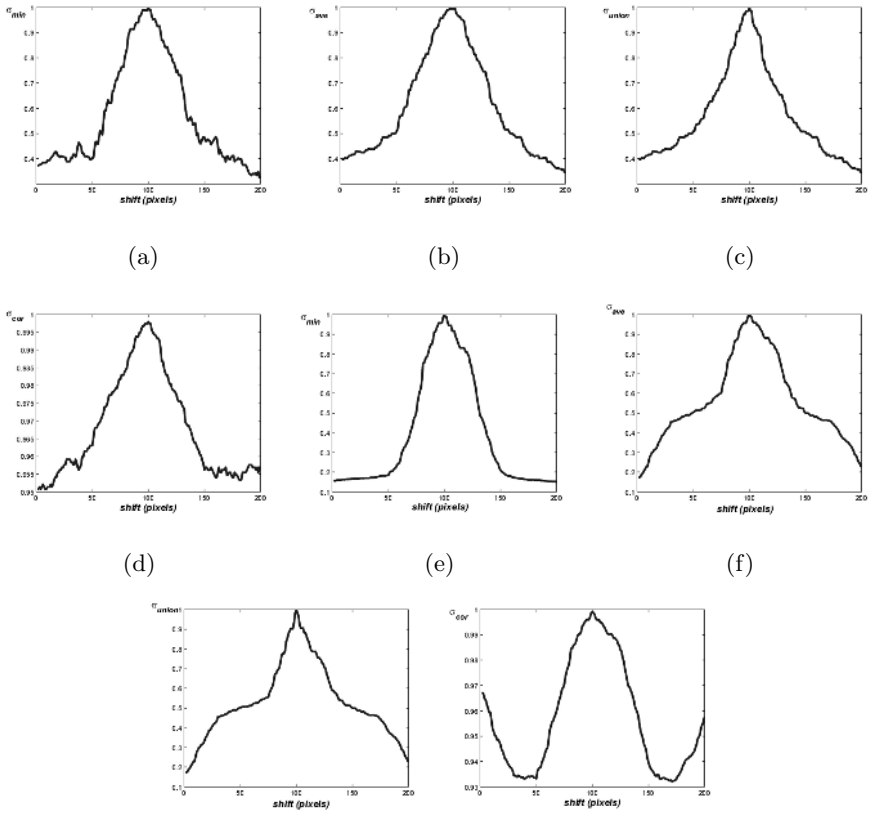
Fig. 4. Not uniform distribution of events in X may effect the estimation of the similarity

elements. The number of elements in X is 2000. In this case, the computation of the set-union, the set-intersection, and the set-difference is performed by simply counting the number of elements of X that are included the support of rules. The computation of both the union and the intersection of rules is influenced by the element distribution. For example, rules $\mathbf{R}_1$ and $\mathbf{R}_2$ in Fig. 4 should be less similar than rules $\mathbf{R}_3$ and $\mathbf{R}_4$ that cover elements with the same spatial density.

Figs 5a,b,c,d report the SF's computed by shifting two rules with the same size and support in the case of a random space without clusters.

Figs 5e,f,g,h report the results obtained in the case of a uniform space plus two clusters. Initially, the two rules are centered on the clusters, and one of them is shifted as described before. In the case of σ_{cor} there is a minimum in the middle of the two clusters and a local maximum when rules are centered on the respective clusters. Also the trend of the other SF's is influenced by the presence of structures in the X space.

Note. *The definitions of SF's are close to the concept of rule-agreement, in fact their course is continuous and their trend follows the relative spatial position of the rules. Other measure, introduced in the literature, do not satisfy fully this property in the sense that their similarity assumes positive values only if rules intersect. For example, the SF introduced in* [11]: $\sigma_{gupta}(\mathbf{R}_1, \mathbf{R}_2) = \frac{|\mathbf{R}_1 \ \ 2|}{|\mathbf{R}_1 \ \ 2|}$ *has been applied to the same data set used here. It is evident that Gupta's SF has positive values only where rules intersect. Fig. 6 shows the application of the* σ_{gupta} *when applied to the same data set used for our similarities.*

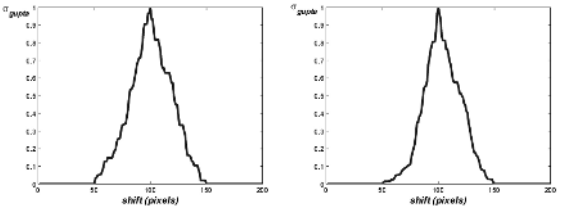


(a) (b) (c)

(d) (e) (f)

(g) (h)

Fig. 5. SF's in a uniform random space: (a) σ_{min}; (b) σ_{ave}; (c) σ_{union}, (d) σ_{cor}; SF's in a random space with two clusters: (e) σ_{min}; (f) σ_{ave}; (g) σ_{union}, (h) σ_{cor}



(a) (b)

Fig. 6. Application of σ_{gupta} to the data used to compute our SF's: (a) random space; a space with two clusters

5 Grouping Rules

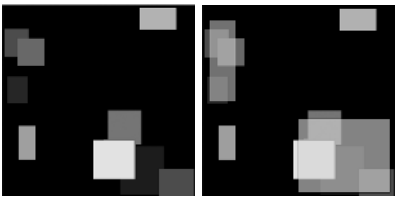
Rules clustering can be considered part of exploratory data analysis and data mining. Rules can be grouped on the basis of the above defined SF's. In order to do this, a single link algorithm has been considered; the grouping threshold was set to $\mu+\alpha\times\sigma$, where μ is the mean value and σ is the variance of the similarities, and $0 \leq \alpha \leq 1$. Rules are rectangles with both the origin (the left bottom corner) and the edges randomly generated inside X. Fig. 7a shows an example with 10 rules. X contains 2000 elements drawn from a uniform distribution $\{x_i \sim [0,1]\}_1^2$.

(a) (b) (c)

Fig. 7. Clustering rules with similarity: (a) Input rules; (b) $\sigma_{min}, \sigma_{ave}, \sigma_{union}$; (c) $\equiv \sigma_{cor}$ (uniform space)

Fig. 7b shows the clusters obtained for $\alpha = 0$ and similarity $\sigma_{min}, \sigma_{ave}$, and σ_{union}. Fig. 7c shows the clusters obtained using σ_{cor}. Also in this case σ_{cor} shows a different behavior. Grouping may suggest the generation of new rules derived from the original ones. For example, a set of rules belonging to the same cluster could be merged in a new rule that is an hyper-rectangle center of which is the center of mass, μ_R, of the rules centers and the length of edges, el, is proportional to the variance, σ_R, of μ_R: $el = \beta\sigma_R$, with $\beta > 0$.

Figs 8a,b show the input rules and their generalization, (transparent rectangles), as obtained with the proposed algorithm. In this example $\beta = 2$ and the similarity measure used to perform rules grouping is σ_{ave}.



(a) (b)

Fig. 8. An example of rules generalization: (a) Input rules; (b) Rule generalization

6 Final Remarks

This paper introduces a set of rule SF's that are size independent; all of them are positive, reflexive and symmetric. The similarity between rules has been considered in this note in the case of rectangular rules in a numeric feature space. SF's could be extended for convex sets or not simply connected sets. Moreover, SF's could be extended to not numeric feature spaces to handle linguistic entities. Another issue is the generalization starting from *basic* rules.

Acknowledgment. This work has been supported by the MIUR-PRIN project "Metodologie di data-mining per le applicazioni di e-business".

References

1. W.Khöler and H.Wallach, "Figural after-effects: an investigation of visual processes", *Proc. Amer. phil. Soc.*, Vol.88, 269-357, 1944.
2. J.B. Tenenbaum, "Rules and Similarity in Concept Learning", *Advances in Neural Information Processing Systems*, Vol.12, S. A. Solla, T. K. Leen, K.-R. Müuller (eds.), pp. 59-65, MIT Press, 2000.
3. V. Di Gesú, S. Roy, "Pictorial indexes and soft image distances", *Lecture Notes in Computer Science*, N.R.Pal and M. Sugeno Eds., pp. 200-215, 2002.
4. V. Di Gesù, G Lo Bosco, "A Genetic Integrated Fuzzy Classifier", *Pattern Recognition Letters*, Vol.26, N.4, pp.411–420, 2005.
5. M. Pearce, D.J. Blake, J.M. Tinsley, B.C. Byth, L. Campbell, A.P. Monaco, K.E. Davies, "The utrophin and dystrophin genes share similarities in genomic structure", *Human Molecular Genetics*, Vol.2, pp.1765–1772, Oxford Univ.Press, 1993.
6. B. Morgenstern, "DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment", *Bioinformatics*, Vol.15, pp. 211–218, 1999.
7. J.S.Varrè, J.P. Delahaye, and E. Rivals, "Transformation Distances: a family of dissimilarity measures based on movement of segments", *Bioinformatics*, Vol.15, pp. 194–202, 1999.
8. R. Agrawal, T. Imielinski, A. Swami, "Mining association rules between sets of items in large databases", In Proceedings of the *ACM SIGMOD International Conference on Management of Data*, Washington D.C, 1993.
9. S. Tsumoto, S. Hirano, "Visualization of Rules Similarity using Multidimensional Scaling", in Proceedings of the Third IEEE International Conference on Data Mining (ICDM03), 2003.
10. B. Lent, A. Swami, J. Widom, "Clustering Association Rules", ICDE, pp. 220-231, 1997.
11. G. Gupta, A. Strehl, J. Ghosh, "Distance based clustering of association rules", in Intelligent Engineering Systems Through Articial Neural Networks (Proceedings of ANNIE 1999), vol. 9, pp. 759–764, ASME Press, 1999.
12. J.H.Friedman and I.Fisher,"Bump Hunting in High-Dimensional Data", Stanford University, Department of Statistics,*Technical Report*, 1998.
13. J.H.Friedman and B.E.Popescu,"Importance Sampled Learning Ensamble", Stanford University, Department of Statistics,*Technical Report*, 2003.

Image File Compression Using Region Growing and Interpolation

Antonio Di Nola[1], Nicla Paladino[1], and Barnabás Bede[2]

[1] Department of Mathematics and Computer Science,
University of Salerno, Italy
dinola@unisa.it, nicla@unisa.it
[2] Department of Mechanical and System Engineering,
Budapest Tech, Népszinház u. 8 H-1081 Budapest, Hungary
bede.barna@bgk.bmf.hu

Abstract. We propose an image file compression method using Region Growing and Interpolation. Firstly, using the techniques of Region Growing, we divide the image in blocks. The compression-decompression method is based on bivariate interpolation. Error estimates in terms of the modulus of continuity of the original image are obtained. Experimental results illustrate the performances of the proposed method.

1 Introduction

Image segmentation based on Region Growing is a method used mainly for image analysis (see e.g. [4]). Image file compression techniques use also image segmentation but generally in equal blocks. Since the most of the images contain some information on a background, by using region growing this background can be separated from the important importations in the image. In this way, the compression on regions contained in the background can be made using much less data then in the rest of the image, so the compression rate can be higher, having the same quality. In this way an image should be divided in some "smooth" blocks of different sizes. This segmentation can be made using Region Growing. Region Growing is an image segmentation method where neighboring pixels are examined and added to a region if no edges are detected. We use Region Growing method as presented in section 3. Interpolation and approximation using mainly trigonometric polynomials or polynomials are the main image file compression methods. For example JPEG (before 2000) and DCT methods are using trigonometric polynomials. Also, JPEG 2000 is using wavelets, which can be also regarded as approximation operators. The compression rate in usual compression methods is related to the size of the blocks in which the image is divided. We propose in this paper a new compression-decompression algorithm using Region Growing and interpolation. The techniques of region growing are used for dividing the image in regions with similar color. Then we store the same amount of data from each region, so the compression rate is higher then in existing methods. For decompression, on each block we compute the values of the bivariate Lagrange interpolation polynomial in the missing points. After

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 188–196, 2006.

a preliminary section where we give the mathematical tools needed for the algorithm, in Section 3 we describe in detail the algorithm proposed for image file compression and we obtain error estimates. Finally we present experimental results and conclusions.

2 Preliminaries

An image plane consists of a rectangular grid, that may be represented by a matrix having each single element (pixel) corresponding to a point. Therefore, all the geometric concepts can be defined not necessarily in the same way of the classic geometry: every pixel has a neightbour composed by adjacent points. Hence the problem of defining adjacency emerges. Indeed two different types of adjacencies can be distinguished, because each pixel has 4 sides and 4 angles (see e.g. [9]). The above suggests the following definitions needed for describing the Region Growing technique.

Definition 1. *([5]) Let (i, j) be a pixel. We say that (k, l) is 4-connected with (i, j) if (k, l) is sharing one side with (i, j). If (k, l) is sharing one side or one angle with (i, j), then we say that (k, l) is 8-connected with (i, j). We denote $(i, j) \perp (k, l)$. (Figure 1 and 2)*

Definition 2. *([5]) A path from the pixel (i_0, j_0) to the pixel (i_n, j_n) is a sequence of pixels (i_0, j_0), $(i_1, j_1), ..., (i_n, j_n)$ such that the pixel (i_k, j_k) is 4(8)-connected with (i_{k+1}, j_{k+1}), $k = 0, ..., n-1$. Two pixels of a given set S are connected iff there is a path between them consisting only of elements from S.*

It is easy to see that the above relations are equivalence relations.

Definition 3. *([5]) Two pixels p and q belonging to a given set S, are* connected *if and only if there is a path from p to q composed by only elements of S.*

It's easy to check that the connectivity is an equivalence relation. Since there are two types of path, there are also two types of connectivities: 4-connectivity and 8-connectivity.

Definition 4. *([5]) Given an image I, we define* connected component *of I, any set of mutually connected pixels. Every connected component is a sequence of pixels.*

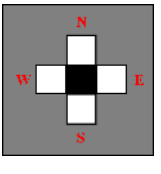


Fig. 1. 4-connected pixels

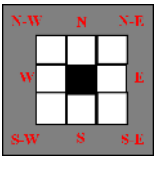


Fig. 2. 8-connected pixels

We denote by $\perp$ the 4-connectivity relation between two pixels. Let Cnm be an $n \times m$ image, and let f be a function mapping every gray-level into the set $L_n = \left\{0, \frac{1}{nm}, \frac{2}{nm}, ..., 1\right\}$. Let $\delta \in \mathbb{R}$ be called the resolution coefficient. We define a binary relation R_δ on the image by the following axioms. A1. If $xR_\delta y$ then $x \perp y$ for every x,y $\in C_{nm}$. A2. If $xR_\delta y$ then $|f(x) - f(y)| \leq \delta$. A3. There exists a unique y such that $xR_\delta y$. A4. If $xR_\delta y$, $zR_\delta t$, $x \perp z$ and $|f(x) - f(z)| \leq \delta$ then $|f(y) - f(t)| \leq \delta$ for every x,y,z $\in C_{nm}$.

Definition 5. *Let F_δ be a subset of the image such that the following axioms hold true. $\pi 1$. For any $x, y \in F_\delta$ we have $|f(x) - f(y)| \leq \delta$. $\pi 2$. For any $x \in F_\delta$ there exists $y \in F_\delta$ such that $x \perp y$. $\pi 3$. F_δ does not separate R_δ, i.e. if $x \in F_\delta$ and $xR_\delta y$ then $y \in F_\delta$. $\pi 4$. F_δ is maximal, i.e. if G_δ satisfies $\pi 1$, $\pi 2$, $\pi 3$ and $F_\delta \subset G_\delta$, then $F_\delta = G_\delta$. Then F_δ is called a connected block.*

The regions in which the image will be divided will satisfy axioms $\pi 1, \pi 2, \pi 3$ and $\pi 4$. Let us recall some definitions needed for the decompression algorithm.

Definition 6. *Let $f : [0,1]^2 \to \mathbb{R}$ be an image. The bivariate Lagrange interpolation polynomial associated to f on the knots (x_i, y_j), $i = 0, ..., m$, $j = 0, ..., n$ is*

$$L_{m,n}(f)(x,y) = \sum_{i=0}^{m} \sum_{j=0}^{n} l_{m,i}(x) l'_{n,j}(y) f(x_i, y_j),$$

where

$$l_{m,i}(x) = \frac{(x - x_0).../...(x - x_m)}{(x_i - x_0).../...(x_i - x_m)},$$

$$l_{n,j}(y) = \frac{(y - y_0).../...(y - y_m)}{(y_i - y_0).../...(y_i - y_m)},$$

and the "/" sign means that the i^{th} operand is missing.

The points (x_i, y_j) are the points which are considered for the interpolation while $f(x_i, y_j)$ is the interpolated value. These points later will be selected in the proposed algorithm and $f(x_i, y_j)$ will be the intensity of the gray-level of the image at the point having coordinates (x_i, y_j).

We will use the modulus of continuity in order to obtain error estimates for our compression-decompression method. Let us recall it's definition.

Definition 7. *Let $f : [0,1]^2 \to \mathbb{R}$ be an image. Then*

$$\omega(f, \delta_1, \delta_2) = \sup\{|f(x,y) - f(x',y')|, \text{ where } |x - x'| \leq \delta_1, |y - y'| \leq \delta_2\}.$$

is called the modulus of oscillation of the image f. If f is continuous then $\omega(f, \delta_1, \delta_2)$ is called the modulus of continuity.

Let us remark that if f is continuous, then $\omega(f, \delta_1, \delta_2) \to 0$ as $\delta_1, \delta_2 \to 0$.

We remind also some properties concerning best approximation polynomials (see [7]) Let X be a Banach space and Y a closed linear subspace of X. The error of approximation $E(f)$ of f by elements from Y is $E(f) = \inf_{P \in Y} \|f - P\|$. If this infimum is attained for some P then P is called best approximation to f. A well known result is the following.

Theorem 1. *[7, Chapter 3, Theorem 1.1] For each finite dimensional subspace X_n of X and each $f \in X$ there is a best approximation of f from X_n.*

In our case this applies as follows. Let $f : [0,1]^2 \to \mathbb{R}$, denote an image. Let $X = \{f : [0,1]^2 \to \mathbb{R} : f \text{ continuous}\}$ denote the Banach space of continuous functions with the uniform norm, i.e. $\|f\| = \sup_{x \in [0,1]^2} |f(x,y)|$. Let $X_n = \{P : [0,1]^2 \to \mathbb{R},\ P(x,y) = \sum_{i=0}^{n} \sum_{j=0}^{n} a_{ij} x^i y^j\}$. Then there is a best approximation polynomial to f, of the form $P(x,y) = \sum_{i=0}^{n} \sum_{j=0}^{n} a_{ij} x^i y^j$.

3 Description of the Algorithm

3.1 Region Growing

In this section we present an application of the formal model previously exposed.

Definition 8. *([5]) Given an image I, let G(x,y) be the function associating a gray level to every component (x,y) of I. Then we define segmentation any partition of the domain R of a discrete function f of gray levels G(x,y) in n regions R_i, based on a prefixed homogeneity criterion E, such that:*

a.) $\bigcup_{i=1}^{n} R_i = R$
b.) Every R_i is a connected component
c.) E(R_i) is satisfied in every R_i d.) E($R_i \bigcup R_j$) is not satisfied for every adjacent R_i and R_j.

Generally the homogeneity criterion E depends on specific applications, and it can be a local characteristic of the image I (gradient, texture, etc.). It is advantageously, for our aim, to choose the homogeneity criterion E such that $\mathrm{E}(\mathrm{R}_i)$ is satisfied, for every region R_i, if all the pixels of R_i have similar levels of gray (*lightness*); that is the difference of levels of gray of two any pixels in R_i has to be less or equal of a fixed threshold. The result of segmentation is a set of distinguished regions. These regions are representable by marking all the points of every region by the same label, that can be either a gray's level, a pseudo-color, or an alphanumerical symbol. To acquire the necessary data requested to describe a homogeneous region, we can make a regular growing, agglomerating every pixel, one after the other, leaving from a given point $k\,(i)$ called *seed point.* To decide if we can assimilate a new point into the region, we have to establish whether the homogeneity criterion $E\,(i)$ is satisfied. The region growing is substantially a sequential process; therefore it is opportune to place the issue of the dependency of the results from the order of exploration and, when necessary, it is opportune to take precautions to attenuate this dependency

choosing suitable seed points and homogeneity criterion (see [9]). The algorithm consists of five steps:

Step 1: The image to examine is taken in input and converted into a gray's levels image, to make the procedure computationally less expansive.

Step 2: The image C_{nm} is scaled so that the gray's degree of each pixel is always a real number between 0 and 1, via the given function

$$f : x \in C_{nm} \to f(x) \in [0, 1].$$

In this way the resolution coefficient δ is an empiricist value between 0 and 1, appropriately chosen by the customer. Taking δ enough large (near 1) makes the algorithm able to recognize less connected regions in number, but they are more extended; viceversa, δ enough small (near 0) makes so that the algorithm recognizes less extended regions, more precise and detailed, but in a greater amount. The maximum number of regions is assigned a priori and it is kept in a macro settled at 50.000; for this reason the choosing of δ must correspond with customer's effective requirement.

The region growing method allows to select (surely the selection depending on the starting points of region growing) relatively smooth regions of the image.

Step 3: The previously defined relation R_δ is taken then like homogeneity's criterion. To make effective the axiom Ax3, about uniqueness, we take the minimum in the set of turning out values among the pixels satisfactory the R_δ relation. Practically, the program starts from the pixel of coordinates (0,0), and examines every 8-connected pixel in a prefixed order. Where the relation R_δ is satisfied, the program chooses the pixel having minimal difference of the gray's levels. The cycle ends when the program mets a pixel with the gray's levels of every 8-connected pixels larger than δ. During the cycle, if the algorithm mets an isolated pixel P (a pixel is isolated when its 8-connected pixels are all labeled), then it compares the pixel P under consideration with each pixel which is 8-connected with P. If the difference between the level of gray of P and the level of gray of one of the 8-connected pixel Q is less than δ, then the program adds P the region of Q. Every time that the algorithm examines a new pixel, it verifies if the pixel was already labeled; if so, the pixel is not considered and the algorithm moves to the successive pixel, if not, this pixel is labeled with a new label and the algorithm starts to constructing a new region. The first pixel of a new region is kept as this regions's representative. At the end of the third step, the image appears subdivided in several distinguished and connected regions, that are not necessarily the real connected regions of the original image. In fact the development of the regions depends on the investigation order. Therefore a further step is needed.

Step 4: The image, output of the previous step, is seen again to recognize possible connected regions among previously constructed regions. For this purpose the program confronts the adjacent regions' representatives. If the difference of their gray's levels is less than the resolution coefficient, then the examined zones, certainly connected, are various zones of the same region. These zones must be represented by the same label. Practically the algorithm makes a merge

of adjacent regions. Otherwise every region preserves its original label. The final image is constituted by connected distinguished regions, to each of which corresponds a distinguished label.

Step 5: For a good return of the final output, the labels are ordered and then they are associated to distinguished RGB colors.

3.2 Compression

The compression algorithm has 3 main steps.

Step 1. Using Region Growing technique, the original image is divided in connected regions.

Step 2. The image is divided in rectangular blocks such that each block is contained in a connected region or has a minimal size.

Step 3. From each block we store the same number of values, selected to be equidistant and belonging to the same region. In our case we select 16 values i.e. we cover the region selected at Step 2 by a grid of 4 by 4 points, the grid having maximal length. Let us denote the coordinates of these points by (x_i, y_j). We store the intensity of the gray-level of each such point selected from the same region, i.e. if f is the gray-level intensity function of the image then we store the values $f(x_i, y_j)$. The compression is done.

Since from every (relatively smooth region) we select 4 values the error of the interpolation is expected to be small.

Since the regions obtained in Step 1 are not rectangular and since the computation of Lagrange polynomial requires rectangular domain, at the Step 2 we use the quad-tree decomposition which can be described more explicitly as follows. The initial block is the whole image.

Step 2.1. Divide the current block in 4 parts

Step 2.2. For each block b_i, $i = 1, 2, 3, 4$

Step 2.3. If the block is contained in a connected region or the size of b_i is the minimal size, then stop else go to Step 2.1.

3.3 Decompression

The values stored in the compression step are now reloaded into the decompressed image and they are considered as the knots (x_i, y_j) for the interpolation. The decompression algorithm consists of computing the Lagrange interpolation polynomial in the missing points, restoring in this way an image near to the original one.

Step 1. For each block restore the original values stored in compression in the places from where they were retrieved.

Step 2. For each missing pixel (i, j), compute the value $L_n(i, j)$.

Step 3. Go to Step 1.

3.4 Error Estimate

In what follows we obtain an error estimate if a block is contained in a connected region.

Theorem 2. *Let $f : [0,1]^2 \to \mathbb{R}$ denote the original image function in a block, and $L_n(f,x,y)$ the bivariate Lagrange polynomial (i.e. decompressed value). Then*

$$\|f - L_{n,n}\| \leq \delta \left(1 + \sup_{(x,y)\in[0,1]^2} \sum_{i=0}^{n}\sum_{j=0}^{n} |l_{n,i}(x) \cdot l'_{n,j}(y)|\right),$$

where $L_{n,n}$, $l_{n,i}$ and $l'_{n,j}$ are given by Definition 6.

In order to estimate the error in the above described method we will use some results in approximation theory. First result is the existence of the so-called best approximation polynomials associated to any continuous function. These are polynomials of a given order (say n) that are at minimal distance to a given function f (see [7]). The above cited results hold also in the bivariate case. So, f being continuous, there is a best approximation bivariate polynomial P_n of f on $[0,1]$. Then by [3], Jackson-type error estimate holds for the approximation by these best approximation polynomials, i.e.

$$E(f) = \|f - P_n\| \leq C\omega\left(f, \frac{1}{n}, \frac{1}{n}\right),$$

for an absolute constant C. By Korneichuk's Theorems the best possible constant is $C = 1$ (see [6]). We have

$$\|f - L_{n,n}\| \leq \|f - P_n\| + \|P_n - L_{n,n}\|.$$

Since the interpolation polynomials unique, P_n coincides with its own interpolating polynomial and we have

$$\|P_n - L_{n,n}\| = \|L_{n,n}(P_n) - L_{n,n}\| =$$

$$= \left\| \sum_{i=0}^{n}\sum_{j=0}^{n} l_{n,i}(x)l'_{n,j}(y)P_n(x_i,y_j) - \sum_{i=0}^{n}\sum_{j=0}^{n} l_{n,i}(x)l'_{n,j}(y)f(x_i,y_j) \right\|,$$

where $x_i = \frac{i}{n}$, $y_j = \frac{j}{n}$, $i,j = 0,1,...,n$. Then we have:

$$\begin{aligned} &|P_n(x,y) - L_n(x,y)| \leq \\ &\leq \sum_{i=0}^{n}\sum_{j=0}^{n} |l_{n,i}(x)l'_{n,j}(y)| \cdot |P_n(x_i,y_j) - f(x_i,y_j)| \leq \\ &\leq \sup_{(x,y)\in[0,1]^2} \sum_{i=0}^{n}\sum_{j=0}^{n} |l_{n,i}(x)l'_{n,j}(y)| \cdot \omega\left(f, \frac{1}{n}, \frac{1}{n}\right). \end{aligned}$$

Finally we obtain

$$\begin{aligned} \|f - L_{n,n}\| = &\left(1 + \sup_{(x,y)\in[0,1]^2} \sum_{i=0}^{n}\sum_{j=0}^{n} |l_{n,i}(x)l'_{n,j}(y)|\right) \\ &\cdot \omega\left(f, \frac{1}{n}, \frac{1}{n}\right). \end{aligned}$$

If a block is contained in a region then $\omega\left(f, \frac{1}{n}, \frac{1}{n}\right) \leq \delta$, which completes the proof.

Remark 1. Since for our purposes x, y take only the values $\frac{k}{m}, \frac{l}{m}$, for $k, l = 0, ..., m$, $m \in \mathbb{N}$, where $\frac{m+1}{n+1}$ is the compression rate in a given block, it is enough to determine

$$S_{m,n} = \sup_{(k,l)\in\{0,...,m\}^2} \sum_{i=0}^{n} \sum_{j=0}^{n} \left| l_{n,i}\left(\frac{k}{m}\right) l'_{n,j}\left(\frac{l}{m}\right) \right|.$$

By simple numerical computation we have obtained that for $n = 3$, $S_{m,3} \leq 1.5$, for $n = 4$ $S_{m,4} \leq 2$ and for $n = 5, 6$, $S_{m,5} \geq 8$ and $S_{m,6} \geq 20$ respectively. This remark leads us to use $n = 3$ (or at most $n = 4$), i.e. from each block we store 4×4 (or 5×5) values from each block. We have chosen $n = 3$. As a consequence we choose the minimal size of a block in the compression algorithm, Step 2, equal to 10×10. The degree of polynomials is set to 3 or at most 4 according to this remark. This prevents high variations of the polynomial in a region.

Remark 2. Let us remark also that the definition of the modulus of continuity insures that if the trashold δ is low, then the approximation error is also low.

As a conclusion, in blocks where some edge is detected, the compression rate is $\frac{100}{16}$. In blocks which are smooth the compression rate can be much higher. For example in blocks of size 20×20 the compression rate is $\frac{400}{16}$.

Remark 3. Since equispaced knots i.e. $x_{in} = \frac{i}{n}$, $i = 0, ..., n$ do not form a normal matrix of nodes, convergence for $n \to \infty$ cannot be obtained (see [1]).

4 Experimental Results

The algorithm was implemented in C language. Region Growing offers several advantages with respect to conventional segmentation techniques. The algorithm is very stable with respect to noise. Our region will never contain too much of the background, so long as the parameters are correctly defined. There are, however, several disadvantages of region growing. First and foremost, it is very expensive computationally. It takes both serious computing power and a decent amount of time to implement the algorithms efficiently. The method is very promising from the theoretical point of view and some results are obtained for test data. Efficient implementation and comparison to other methods is subject of further research.

References

1. E.K. Blum, Numerical Analysis and Computation Theory and Practice, Addison-Wesley Series in Mathematics, Addison-Wesley Publishing Company, XII, 1972.
2. R.O. Duda, P.E. Hart, D.G. Atock, Pattern Classification and Scene Analysis, John Wiley & Sons, New York 1973.

3. S.G. Gal, Jackson-type estimate in monotone approximation by bivariate polynomials, Journal of Concrete and Applicable Mathematics, vol. 1, no. 1(2003), p. 63-73.
4. R.M. Haralick, L.G. Shapiro, Image segmentation techniques, CVGIP, vol. 29, 1985.
5. A.K. Jain, Fundamentals of Digital Image Processing, Prentice Hall, U.S.A. 1989.
6. KorneiN.P. Korneichuk, Exact Constants in Approximation Theory, Cambridge Univ. Press, 1991.
7. G.G. Lorentz, R. A. DeVore, Constructive Approximation, Polynomials and Splines Approximation, Springer-Verlag, New York, Berlin, Heidelberg 1993.
8. T. Pavlidis, Struttural pattern recognition, Springer-Verlag, Berlin 1977.
9. P. Zamperoni, Metodi dell'elaborazione digitale di immagini, Masson, S. Donato Milanese (MI) 1990.

Fuzzy Connectivity and Its Application to Image Segmentation

Gabriele Martino and Alfredo Petrosino

Department of Applied Science, University of Naples "Parthenope", Naples, Italy
alfredo.petrosino@uniparthenope.it

Abstract. Connectivity is a concept of great relevance to image processing and analysis. It is extensively used in image filtering and segmentation, image compression and coding, motion analysis, pattern recognition, and other applications. In this paper, we review the notion of Fuzzy Connectivity and its application to Image Segmentation.

1 Introduction

The mathematical notion of *connectivity* plays an important role in image processing and analysis, and particularly in problems related to image filtering and segmentation, image compression and coding, motion analysis, pattern recognition, and other applications. In mathematics, connectivity is classically defined using either a *topological* [5], [7] or a *graph-theoretic* framework [4]. In general, topological connectivity is useful for images defined over a continuous space, whereas graph-theoretic connectivity is useful for images defined over a discrete space. Although the classical notions of connectivity have been extensively used in image processing and analysis, they are limited. One problem, associated with these notions, is that they are set-oriented and, therefore, directly applicable only to binary images.

With the advent of the theory of *fuzzy sets* [12] the classical notions of connectivity were extended to a fuzzy setting. This allowed the development of definitions of connectivity for grayscale images. A well-known example of fuzzy graph-theoretic connectivity is the fuzzy connectivity for discrete grayscale images proposed by Rosenfeld [10]. In this paper, we first review the concept of Connectivity, the concept of Graph-Theoretic Connectivity, then its extention to the fuzzy case with the Fuzzy Graph-Theoretic Connectivity. The application to Image Segmentation of some recent models based on the concept of fuzzy connectivity [3, 8] are reviewed and compared with classical clustering techinques [1].

2 Connectivity

Several different types of connectivity has been introduced in the past. We can outline them as follows:

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 197–206, 2006.

- Classical Connectivity :
  - *Topological Connectivity* over continous spaces.
  - *Graph-Theoretic Connectivity* over discrete spaces.
- Fuzzy Connectivity :
  - *Fuzzy Topological Connectivity* over continous spaces.
  - *Fuzzy Graph-Theoretic Connectivity* over discrete spaces.

Fou our purpose, since images are discret, we focused our attention on connectivity, classical and fuzzy, defined over discrete spaces, namely *Classical* and *Fuzzy Graph-Theoretic Connectivity.*

2.1 Classical Graph-Theoretic Connectivity

Connectivity in discrete spaces is best approached by using graphs. Given a graph $G = (V, L)$, a path in G between two given vertices v_1, $v_t \in V$ is a sequence $\Pi = \{v_1, v_2, ..., v_t\}$, where $v_i \in V$, such that $(v_i, v_{i+1}) \in L$, for $i = 1, 2, ..., t-1$. A graph $G = (V, L)$ is said to be connected if any two of its vertices are linked by a path in G. This can be extended to subsets of a graph by means of the concept of induced subgraph. A subset $U \subseteq V$ of vertices of a graph $G = (V, L)$ is said to be connected in G if the induced subgraph $G[U]$ is connected. Hence, U is connected in G if any two of its vertices are linked by a path with vertices in U and edges in L. The maximal connected subsets of U are called the *connected components* of U.

Two cases of interest to image processing and analysis are obtained by taking the set of vertices to be points (m, n) in a subset of the two-dimensional discrete space $\mathcal{Z}^2$. Two vertices $v = (m, n)$ and $\bar{v} = (\bar{m}, \bar{n})$ are said to be *4-adjacent* if $|m-\bar{m}|+|n-\bar{n}| = 1$, whereas v and $\bar{v}$ are said to be *8-adjacent* if $\max\{|m-\bar{m}|, |n-\bar{n}|\} = 1$. This leads to the classical notions of 4- and 8-adjacency connectivity.

2.2 Fuzzy Graph-Theoretic Connectivity

The classical notions of connectivity, examined in the last section, are all set-oriented, and therefore directly applicable only to binary images. The notions of fuzzy graphs allow extension of classical connectivity concepts to a fuzzy setting. This leads to definitions of connectivity that apply to both binary and grayscale images.

Fuzzy connectivity is defined by means of the concept of fuzzy graph. A $\mathcal{T}$-fuzzy graph is a pair $G = (V, \sigma)$, where V is the set of vertices and σ is a $\mathcal{T}$-fuzzy set on VxV, called the fuzzy edge set of G. A fuzzy graph can be thought of as a weighted graph; i.e., an ordinary graph with weights assigned to the edges.

For two vertices $v, w \in V$, the quantity $\sigma(v, w)$ indicates the *degree of adjacency*, or *strength of connection*, between v and w. Given a $\mathcal{T}$-fuzzy graph $G = (V, \bar{\sigma})$ and $U \subseteq V$, the $\mathcal{T}$-fuzzy graph $G[U] = (U, \sigma)$, where

$$\bar{\sigma}(v, w) = \begin{cases} \sigma(v, w)\ , & \text{if } v, w \in U \\ O\ , & \text{otherwise} \end{cases} \tag{1}$$

is known as the $\mathcal{T}$-fuzzy subgraph induced by U.

Given a $\mathcal{T}$-fuzzy graph $G = (V, \sigma)$, a fuzzy path in G between two given vertices v_1, $v_t \in V$ is a sequence $\Pi = \{v_1, v_2, ..., v_t\}$, where $v_i \in V$, such that $\sigma(v_i, v_{i+1}) > O$, for $i = 1, 2, ..., t-1$. Clearly, a path in an ordinary graph is a special case of a fuzzy path. The *strength* $s(\Pi)$ of a fuzzy path $\Pi = \{v_1, v_2, ..., v_t\}$ is defined as $s(\Pi) = \bigwedge_{i=1}^{t-1} \sigma(v_i, v_{i+1})$. The set of all fuzzy paths between two vertices $v, w \in V$ is denoted by Π_{vw}.

Let $G = (V, \sigma)$ be a $\mathcal{T}$-fuzzy graph. The *degree of connectivity* $c(v, w)$ between two vertices $v, w \in V$ is defined by

$$c(v, w) = \bigvee_{\Pi \in \Pi_{vw}} s(\Pi) \tag{2}$$

$$= \bigvee_{\Pi \in \Pi_{vw}} \bigwedge_{i=1}^{t-1} \sigma(v_i, v_{i+1} | v_i, v_{i+1} \in \Pi). \tag{3}$$

To simplify our discussion, we assume that the vertex set V is finite and that $\mathcal{T}$ is a chain. In this case, the strength of a path corresponds to the *weakest link* between any vertices in the path, and the degree of connectivity between two vertices corresponds to the strength of the *best path* between the vertices, since all suprema and infima involved are achieved.

A $\mathcal{T}$-fuzzy graph $G = (V, \sigma)$ is said to be fuzzy τ-connected (or, simply, τ-connected), for $\tau \in \mathcal{T} \backslash \{O\}$, if $c(v, w) \geq \tau$, for all $v, w \in V$. In other words, a $\mathcal{T}$-fuzzy graph is τ-connected if the degree of connectivity between any pair of vertices is at least τ or, equivalently, if there exists a fuzzy path of strength at least τ between any pair of vertices.

Let $G = (V, \sigma)$ be a $\mathcal{T}$-fuzzy graph. For $\tau \in \mathcal{T} \backslash \{O\}$, a subset $U \subseteq V$ of vertices of G is said to be fuzzy τ-connected (or, simply, τ-connected) if the induced $\mathcal{T}$-fuzzy subgraph $G[U]$ is τ-connected. The maximal τ-connected subsets of U are called the τ-*connected components* of U.

3 Fuzzy Connectedness

The notion of the *degree of connectedness(connectivity)* of two points was first introduced by Rosenfeld [10] in the context of studying the topology and geometry of fuzzy subsets of $\mathcal{Z}^2$. He considered contiguous paths in $\mathcal{Z}^2$ and defined the *strength of connectedness* of a contiguous path Π from v to w in a fuzzy subset $\mathcal{A}$ of $\mathcal{Z}^2$ as the smallest membership value along the path, and the degree of connectedness, denoted by $\mu_{\mathcal{R}}(v, w)$, as the strength of the strongest path between the two points. Therefore

$$\mu_{\mathcal{R}}(v, w) = \max_{\Pi \in \Pi_{vw}} [\min_{e \in \mathcal{E}(\Pi)} \mu_{\mathcal{A}}(e)], \tag{4}$$

where $\mathcal{E}(\Pi)$ is the set of all points in the path Π.

He used the notion of the degree of connectedness to define certain topological and geometrical entities which had been previously defined for hard sets of spels in $\mathcal{Z}^2$. Two points $v, w \in \mathcal{Z}^2$ are said to be connected if

$$\mu_{\mathcal{R}}(v, w) = \min(\mu_{\mathcal{A}}(c), \mu_{\mathcal{A}}(d)). \tag{5}$$

He showed that this binary (hard) relation of connectedness in a fuzzy subset is reflexive and symmetric, but not transitive, and consequently, is not an equivalence relation. Therefore, the components defined by this relation may not partition $\mathcal{Z}^2$.

Udupa and Saha [8] simultaneously introduced a different framework, bringing in a key concept of a local fuzzy relation called *affinity* on spels to capture local hanging-togetherness of spels. They showed how affinity can incorporate various image features in defining fuzzy connectedness, presented a general framework for the theory of fuzzy connectedness, and demonstrated how dynamic programming can be utilized to bring the otherwise seemingly intractable notion of fuzzy connectedness into segmentation.

Let $\mathcal{C} = (C, f)$ be any scene over (Z^n, α). In image processing, C is the image domain and f is its pixel intensity function. Udupa and Saha defined a fuzzy relation in $\mathcal{C}$, called *Fuzzy κ-Connectedness*, denoted by K. Its membership function μ_K was defined as follows:

$$\mu_K(v, w) = \max_{\Pi \in \Pi_{vw}} [\mu_{\mathcal{N}}(\Pi)]. \tag{6}$$

$\mathcal{N}$ is a *fuzzy κ-net*. Its membership function $\mu_{\mathcal{N}}(\Pi)$ is defined as follows:

$$\mu_{\mathcal{N}}(\Pi) = \min_{1 \le i < m} [\mu_\kappa(v_i, v_{i+1})]. \tag{7}$$

where $\mu_\kappa(v_i, v_{i+1})$ is the *affinity* between v_i and v_{i+1}.

A typical functional form for $\mu_\kappa(v, w)$ is:

$$\mu_\kappa(v, w) = g(\mu_\psi(v, w), \mu_\phi(v, w)). \tag{8}$$

where ψ and ϕ represent the *homogeneity-based* and the *object-feature-based* components of affinity and μ_ψ and μ_ϕ the respective membership functions. The strenght of relation ψ indicates the degree of local hanging togetherness of pixels because of their intensity similarities. The strenght of relation ϕ indicates the degree of local hanging togetherness of pixels because of the similarity of their features values to some(specified) object feature. For details and functional forms of $\mu_\psi(v, w)$ and $\mu_\phi(v, w)$ refer to [8].

In the image segmentation framework, let $O_1, O_2, ..., O_l$ be l objects in C that should be extracted from it and $S = \{o_1, o_2, ..., o_l\}$ a set of points of C such that $o_1 \in O_1, o_2 \in O_2, ..., o_l \in O_l$. The elements of S are also called *seeds*.

For any $o \in S = \{o_1, o_2, ..., o_l\}$, $b(o)$ denotes the set of reference points chosen in the object regions that represent the background regions as far as the object to which o belongs is concerned. For any points $o \in S$:

$$P_{ob(o)_k} = \{v | v \in C \text{ and } \mu_K(o, v) > \mu_K(\bar{o}, v) \text{ for all } \bar{o} \in b(o)\} \tag{9}$$

A *fuzzy-κobject $\mathcal{O}$ of $\mathcal{C}$ containing a point o relative to the background (co-objects) containing the points in b(o)* is the fuzzy subset of C defined by the following membership function. For any $v \in C$,

$$\mu_{\mathcal{O}}(v) = \begin{cases} \eta(f(v)) , & \text{if } v \in P_{ob(o)_k}, \\ O , & \text{otherwise,} \end{cases} \tag{10}$$

where η is a function that maps the image intensity values into objectness values.

4 Algorithms

Based on the theory and considerations reported in the previous sections, three algorithms have been considered for comparison.

The first algorithm adopted for comparison has been the fuzzy K-means [1].

The second algorithm is the κMRFOE [8], due to Udupa and Saho. The algorithm computes the desired fuzzy-κ objects $\mathcal{O}$, named κ-MRFOE, for multiple relative fuzzy objects extraction and is reported in Figure 1.

The third algorithm, reported in [3], due to Bloch, Martino and Petrosino, and based on the ideas reported in [2], consists in performing two passes on the image, one in the conventional sense, and one in the opposite sense. For each point c, we store the point $Q = O(c)$ from which the maximum affinity is obtained. For a point c, we don't consider all points in O_i as for exhaustive method, but only those of neighbourhood of c. Specifically, we compute the *fuzzy landscape* [2] as:

$$\mu_{f_i}(c) = \max_{d \in V(c)} t[\mu_{f_i}(O(d)), \mu_S(O(d))], \tag{11}$$

where $t[\cdot]$ denotes a t-norm function, $V(c)$ denotes the neighbourhood of c. Let d_c be the point d for which the maximum affinity value is obtained

---

▷ *Input:* $\mathcal{C} = (C, f)$, κ *a set of points* $S = \{o_1, o_2, ..., o_l\}$
▷ *Output: Fuzzy* $\kappa-$*objects* $\mathcal{O}$ *containing* $o_i \in S$ *relative to co-objects*
▷ *Auxiliary data structures: a queue* Q *containing points to processed and a* κ *scene* $f_{K_{o_i}}$ *of* o_i *in* $\mathcal{C}$ *for each* $o_i \in S$.

for $x \in S$ do
  set $f_{K_x}(v) = 0$ for all $v \in C$ except for x set $f_{K_x} = 1$
  push all points $v \in C$ such that $\mu_\kappa(x, c) > 0$ to Q
  while Q is not empty do
    remove a point v from Q;
    find $f_{max} = \max_{w \in C}[\min(f_{K_x}, \mu_\kappa(v, w))]$;
    if $f_{max} > f_{K_x}(v)$ *then*
      set $f_{K_x}(v) = f_{max}$;
      push all points e such that $\min[f_{max}, \mu_\kappa(x, e)]$ ¿ $f_{K_x}(e)$ to Q;
    end if
  end while
end for
for all $v \in C$ do
  if $f_{K_o}(v) > f_{K_{o_i}}(v)$ for all $o_i \in b(o)$
    set $\mu_\mathcal{O}(v) = \eta(f(c))$;
    else
    set $\mu_\mathcal{O}(v) = 0$
  end if
end for

Fig. 1. ALGORITHM κMRFOE

$$d_c = arg\max_{d \in V(c)} t[\mu_{f_i}(O(d)), \mu_S(O(d))] \tag{12}$$

Then, we set: $O(c) = O(d_c)$.

As final result of the second step, each point c of the image is characterized by its membership, its degree of affinity with each of the manually selected objects O_i, that is $\mu_{f_i}(c)$, $\forall c \in S$ and $\forall i = 1, 2, ...n$. Then, the fuzzy landscapes of the objects are matched with S. This is realized by means of the computation of the values of the possibility Π_i and necessity N_i. As in the computation of the fuzzy landscape, this is realized $\forall c \in S$ and $\forall i = 1, 2, ...n$. The computation of $\Pi_i(c)$ and $N_i(c)$, as regards fuzzy case, is realized as follows:

$$\Pi_i(x) = \sup_{y \in S} \mu_{f_i}(y) \quad \forall x \in S \tag{13}$$

$$N_i(x) = \inf_{y \in S} \mu_{f_i}(y) \quad \forall x \in S \tag{14}$$

The possibility and necessity can be interpreted in terms of fuzzy mathematical morphology, since the possibility is equal to the dilation of μ_S by μ_{f_i}, while the necessity is equal to the erosion [10]. The final decision is to assign a point $c \in S$ to an object for which it has the maximum degree of matching.

Figure 2 reports a sketch of the algorithm.

```
    ▷ Estimate parameters for affinity.
1   for each object i do
2       for each pixels c do
3           compute μ_{f_i}(c) according to (11);
4           for each object i do
5               for each pixels c do
6                   compute g_i(c) = (Π_i(c)+N_i(c))/2 according to (14);
7                   find p_i = argmax g_i(c);
8               end for
9           end for
10      end for
11  end for
    ▷ Output p_i
```

Fig. 2. ALGORITHM Bloch, Martino & Petrosino

5 Experimental Results

Two kinds of experiments have been performed: (a) tolerance to Gaussian noise and blur; (b) segmentation of real MRI (Magnetic Resonance Images).

In the first case an original image has been modified by means:

- superposition of 5 different levels of Gaussian noise, modelled by a Gaussian kernel with zero mean and standard deviations 10, 15, 25, 30, 35;
- blurring of the image at 5 different levels, by changing radius from 1 to 5 values by step 1.

(a) (b) (c)

Fig. 3. Segmentations obtained: (a) original image , (b) fuzzy connectivity based segmentation [3], (c) fuzzy K-means segmentation

The evaluation has been performed by means of the FOM (Figure of Merit) index defined as

$$FOM = |C_{gt}\ EOR\ C_i| \tag{15}$$

where C_{gt} is the ground-truth image, C_i the segmented image and EOR means the exclusive OR. The FOM index denotes the number of 1-valued pixels, establishing a degree of matching between the ground-truth and the segmented image.

Tables 1,2,3 show the achieved results, where it appears clear how the fuzzy connectivity based algorithms outperform the fuzzy K-means algorithm.

The second set of experiments is more complex. Based on [6], we have used 100 MRI images of the IBSR (http://www.cma.mgh.harvard.edu/ibsr/). The results obtained by this algorithm have been compared to the results of the fuzzy K-means clustering segmentation algorithm.

Table 1. FOM of the Bloch, Martino & Petrosino segmentation algorithm [3]

	Blur 1	Blur 2	Blur 3	Blur 4	Blur 5
Noise 10	0.99557	0.99507	0.99388	0.99304	0.9802
Noise 15	0.99547	0.99504	0.9938	0.99286	0.99042
Noise 25	0.99536	0.99486	0.99359	0.9922	0.99026
Noise 30	0.99552	0.99483	0.99335	0.9930	0.98956
Noise 35	0.99457	0.99368	0.99303	0.99103	0.98874

Table 2. FOM of the κMRFOE segmentation algorithm [8]

	Blur 1	Blur 2	Blur 3	Blur 4	Blur 5
Noise 10	0.99907	0.99765	0.99542	0.99316	0.99232
Noise 15	0.99849	0.99689	0.99402	0.99231	0.99216
Noise 25	0.99802	0.99626	0.99352	0.99222	0.99134
Noise 30	0.99792	0.99413	0.99309	0.99205	0.99117
Noise 35	0.99728	0.99269	0.99203	0.99025	0.98975

Table 3. FOM of the fuzzy K-means segmentation algorithm

	Blur 1	Blur 2	Blur 3	Blur 4	Blur 5
Noise 10	0.9897	0.9594	0.9287	0.9068	0.8886
Noise 15	0.989	0.9568	0.9278	0.9058	0.8876
Noise 25	0.9846	0.9562	0.928	0.9071	0.8879
Noise 30	0.9851	0.9542	0.928	0.9048	0.9052
Noise 35	0.9761	0.9531	0.9255	0.9058	0.8871

We have used the Structural Similarity Index measure, SSIM-INDEX in brief, to evaluate the degree of similarity between the original signal (groundtruth) and the distorted signal (segmented image) [11]. The SSIM-INDEX, averaged over K images, is given by:

$$\text{SSIM} - \text{INDEX} = \frac{1}{K}\sum_{k=1}^{K}\left(\frac{(2 * \overline{\mathbf{x}}_k * \overline{\mathbf{y}}_k + C_1)\,(2 * \sigma_{\mathbf{x}_k\mathbf{y}_k} + C_2)}{\left(\overline{\mathbf{x}}_k^2 + \overline{\mathbf{y}}_k^2 + C_1\right)\left(\sigma_{\mathbf{x}_k}^2 + \sigma_{\mathbf{y}_k}^2 + C_2\right)}\right), \qquad (16)$$

where $C_1 = (K_1 * L)^2, C_2 = (K_2 * L)^2, K_1 = 0.01, K_2 = 0.03$, and $L = 255$. Such measure gives values in [0,1]; the higher the value of SSIM-INDEX, the better the segmentation result.

In Tables 4, 5 and 6 the results obtained by the κMRFOE, [3] algorithm and fuzzy K-means on the 100 MRI images of the IBSR are respectively reported. Each row of the tables reports the results of 5 images; the last row of the tables reports the mean SSIM-INDEX of the 50 images. Also, the fuzzy connectivity based algorithms outperform the classical fuzzy K-means; just to give and idea of the produced results, in Fig.3 a visual comparison between the result produced

Table 4. SSIM-INDEX producted by the κMRFOE segmentation algorithm [8] over the IBSR MRI images

(a) SSIM-INDEX of the κMRFOE segmentation algorithm for the first 50 images.

Images - SSIM-INDEX					
1-5	0.9923	0.9914	0.9931	0.9853	0.9802
6-10	0.9746	0.9658	0.9467	0.9484	0.9437
11-15	0.9498	0.9442	0.9423	0.9385	0.9338
16-20	0.9375	0.9230	0.9432	0.9178	0.9302
21-25	0.9256	0.8998	0.9359	0.9237	0.8807
26-30	0.9318	0.9308	0.9241	0.9331	0.9335
31-35	0.9182	0.9043	0.9176	0.9273	0.9202
36-40	0.9282	0.9537	0.9643	0.9685	0.9704
41-45	0.9736	0.9671	0.9756	0.9740	0.9797
46-50	0.9749	0.9818	0.9858	0.9807	0.9903
Mean 0.9495					

(b) SSIM-INDEX of the κMRFOE segmentation algorithm for the last 50 images.

Images - SSIM-INDEX					
1-5	0.9717	0.9791	0.9796	0.9796	0.9562
6-10	0.9579	0.9493	0.9536	0.9436	0.9483
11-15	0.9201	0.9497	0.9329	0.9419	0.9458
16-20	0.9276	0.9344	0.9082	0.9342	0.9394
21-25	0.9377	0.9452	0.8820	0.9273	0.9109
26-30	0.9176	0.9334	0.9408	0.9207	0.9480
31-35	0.9112	0.9258	0.9417	0.9577	0.9727
36-40	0.9571	0.9654	0.9645	0.9697	0.9787
41-45	0.9747	0.9794	0.9785	0.9844	0.9766
46-50	0.9811	0.9916	0.9836	0.9936	0.9957
Mean 0.9521					

Table 5. SSIM-INDEX producted by the Bloch, Martino & Petrosino algorithm [3] over the IBSR MRI images

(a) SSIM-INDEX of the segmentation algorithm [3] for the first 50 images.

Images - SSIM-INDEX					
1-5	0.9862	0.9789	0.9702	0.9641	0.9512
6-10	0.9427	0.9548	0.9349	0.9206	0.9135
11-15	0.9202	0.9209	0.9110	0.9168	0.9116
16-20	0.9130	0.9134	0.9254	0.9285	0.9294
21-25	0.9278	0.9208	0.9333	0.9300	0.9340
26-30	0.9288	0.9174	0.9297	0.9304	0.9351
31-35	0.9424	0.9255	0.9304	0.9291	0.9305
36-40	0.9394	0.9498	0.9523	0.9532	0.9586
41-45	0.9589	0.9639	0.9734	0.9680	0.9735
46-50	0.9724	0.9747	0.9674	0.9687	0.9733
Mean 0.9420					

(b) SSIM-INDEX of the segmentation algorithm [3] for the last 50 images.

Images - SSIM-INDEX					
1-5	0.9693	0.9617	0.9671	0.9514	0.9361
6-10	0.9447	0.9328	0.9208	0.9277	0.9476
11-15	0.9213	0.9250	0.9263	0.9201	0.9353
16-20	0.8987	0.8996	0.9013	0.9094	0.9327
21-25	0.9263	0.9315	0.9214	0.9282	0.9200
26-30	0.9348	0.9325	0.9309	0.9311	0.9362
31-35	0.9416	0.9441	0.9437	0.9398	0.9516
36-40	0.9568	0.9660	0.9666	0.9705	0.9747
41-45	0.9745	0.9747	0.9683	0.9715	0.9748
46-50	0.9795	0.9789	0.9778	0.9842	0.9900
Mean 0.9450					

Table 6. SSIM-INDEX producted by the fuzzy K-means [1] over the IBSR MRI images

(a) SSIM-INDEX of the fuzzy K-means clustering segmentation algorithm for the first 50 images.

Images - SSIM-INDEX					
1-5	0.9691	0.9621	0.9616	0.9474	0.9227
6-10	0.9130	0.8992	0.9079	0.8905	0.8882
11-15	0.8806	0.8790	0.8718	0.8675	0.8522
16-20	0.8564	0.8493	0.8524	0.8541	0.8698
21-25	0.8781	0.8610	0.8676	0.8598	0.8465
26-30	0.8572	0.8481	0.8515	0.8595	0.8605
31-35	0.8606	0.8673	0.8768	0.8709	0.8787
36-40	0.8743	0.8908	0.8979	0.9082	0.9147
41-45	0.9263	0.9238	0.9318	0.9330	0.9379
46-50	0.9335	0.9425	0.9453	0.9482	0.9593
Mean 0.8942					

(b) SSIM-INDEX of the fuzzy K-means clustering segmentation algorithm for the last 50 images.

Images - SSIM-INDEX					
1-5	0.9724	0.9575	0.9310	0.9528	0.9312
6-10	0.8994	0.9127	0.9055	0.8878	0.9052
11-15	0.9121	0.8942	0.9113	0.8739	0.8907
16-20	0.8590	0.8598	0.8616	0.8761	0.8818
21-25	0.8926	0.8686	0.8863	0.8951	0.8783
26-30	0.8914	0.8990	0.8734	0.9026	0.8951
31-35	0.8852	0.8793	0.8862	0.9068	0.9124
36-40	0.9296	0.9392	0.9361	0.9455	0.9334
41-45	0.9568	0.9402	0.9322	0.9370	0.9423
46-50	0.9604	0.9748	0.9737	0.9608	0.9824
Mean 0.9135					

by a fuzzy connectivity based segmentation algorithm [8] and the fuzzy K-means clustering is reported.

As a remark, even if the results of the [3] algorithm are slightly worst, an interesting aspect is that it gets lower computational time with good results. Sometimes that is better than having quality optimal results with higher computational times.

Even if the results of the proposed algorithm are slightly worst, an interesting aspect is that it gets lower computational time, as noticed in the previous section, with good results. Sometimes that is better than having quality optimal results with higher computational times.

6 Conclusions

Connectivity is a concept of great relevance to image processing and analysis, in particular Fuzzy Connectivity is able to capture the *fuzzyness* that characerizes all images. We have analyzed Fuzzy Connectivity properties by means of the implementation and evaluation of image segmentation algorithms. They have good formal properties and obtain very good results also compared with the fuzzy K-means algorithm.

References

1. R. E. J. C. Bezdek and W. Full. *Fcm : The fuzzy c-means clustering algorithm.* Computers and Geosciences, 10:191–203, 1984.
2. I. Bloch. *Fuzzy Relative Position Between Objects in Image Processing: A Morphological Approach.* IEEE Transaction On Pattern Analisys and Machine Intelligence, vol. 21, n.7, July 1999.
3. I. Bloch, G. Martino, A. Petrosino. *A Fuzzy Mathematical Morphology Approach to Multiseed Image Segmentation.* Lecture Notes in Computer Science, vol. 3849, pp 362-368, March 2006.
4. R. Diestel. *Graph Theory.* Springer-Verlag: New York City, NY.
5. J. Dugundji. *Topology.* Allyn and Bacon: Boston, MA, 1966.
6. Jong-Min Lee, Uicheul Yoon, Sang Hee Nam, Jung-Hyun Kim, In-Young Kim, Sun I.Kim. *Evaluation of automated and semi-automated skull-stripping algorithms using similarity index and segmentation error.* Computers in Biology and Medicine vol.33, pp 495-507, 2003.
7. J.R. Munkres. *Topology: A First Course.* Prentice Hall: Englewood Cliffs, NJ, 1975.
8. J.K. Udupa, P.K Saha. *Fuzzy Connectedness and Image segmentation*, in Proc. of the IEEE, vol.91, no.10 , October 2003.
9. J.K. Udupa, P.K Saha, D. Odhner. *Scale-based fuzzy connectedness image segmentation: Theory, algorithms and applications in image segmentation.* Comput. Vision Image Understanding, vol.77, pp 145-174, 2000.
10. A. Rosenfeld. *Fuzzy digital topology.* Information and Control, Vol. 40, pp. 76/87, 1979.
11. Z. Wang, L. Lu, A.C Bovik. *Video quality assessment based on structural distortion measurement.* Signal Processing: image communication, vol.19, no.2, pp 121-132. February 2004.
12. L.A. Zadeh. *Fuzzy sets.* Information and Control, Vol. 8.

Soft Rank Clustering

Stefano Rovetta[1,2], Francesco Masulli[3,2], and Maurizio Filippone[1,2]

[1] Dipartimento di Informatica e Scienze dell'Informazione, Università di Genova,
Via Dodecaneso, 35, I -16146 Genova, Italy
[2] Istituto Nazionale per la Fisica della Materia, Unità di Genova,
Via Dodecaneso, 33, I -16146 Genova, Italy
[3] Dipartimento di Informatica, Università di Pisa,
Largo B. Pontecorvo, 3, I-56127 Pisa, Italy

Abstract. Clustering methods provide an useful tool to tackle the problem of exploring large-dimensional data. However many common approaches suffer from being applied in high-dimensional spaces. Building on a dissimilarity-based representation of data, we propose a dimensionality reduction technique which preserves the clustering structure of the data. The technique is designed for cases in which data dimensionality is large compared to the number of available observations. In these cases, we represent data in the space of soft D-ranks, by applying the concept of fuzzy ranking. A clustering procedure is then applied. Experimental results show that the method is able to retain the necessary information, while considerably reducing dimensionality.

1 Introduction

The exploration of large-dimensional data has always been an ubiquitous problem in science and information technologies. Clustering methods provide an useful tool to tackle this problem. Several clustering algorithms have been modified in the direction of incorporating fuzzy concepts (starting with the Fuzzy c-Means algorithm [1, 2]).

Many common approaches suffer from being applied in high-dimensional spaces. For instance, one of the most common methods, k-means clustering, is based on iteratively computing distances and cluster averages. Increasing the data space dimensionality may introduce a large number of suboptimal solutions (local minima), and the nearest-neighbour criterion which is the basis of the method may even become useless, in the sense that the distances of a given query point from its nearest and farthest neighbours tend to converge [3].

Clustering algorithms often seek for areas where data is especially dense. However it is often the case that the cardinality of the data sets available is not only small with respect to the size of the data space, which would lead to insufficient sampling of the space: sometimes it is even less than the number of variables. This means that the data span only a subspace within the data space. In these conditions, it is not even easy to define the concept of volumetric density, let alone estimating it.

A further problem is again related to distances in high space dimensionality. Defining clusters on the basis of distance requires that distances can be estimated. However there are results [4] stating that, when space dimensionality is high or even moderate (as low as 10-15), the distance of a point to its farthest neighbor and to its nearest neighbor tend

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 207–213, 2006.

to become equal. Therefore the evaluation of distances, and the concept of "nearest neighbor" itself, become less and less meaningful with growing dimension.

A notable complexity reduction in the presence of large-dimensional data sets is provided by representations based on mutual distances between points. If the cardinality of the data set is small compared to the input space dimensionality, then the matrix of mutual distances or other pairwise pattern evaluation methods such as kernels [5] may be used to represent data sets in a more compact way. Pękalska and Duin[6] have developed a set of methods based on representing each pattern according to a set of similarity measurements with respect to other patterns in the data set.

We adopt the same representation, whereby the data matrix is replaced by a pairwise dissimilarity matrix D. Let X be a data set of cardinality n, $X = \{x_1, x_2, \ldots, x_n\}$. We start by computing the dissimilarity matrix $d_{ik} = d(x_i, x_k) \;\; \forall i, k$ according to the dissimilarity measure $d(x, y)$ between points x and y (e.g, the Euclidean distance). The dissimilarity matrix may as well be given as input, in which case it could not even be a symmetric matrix (for instance when obtained from subjective measurements by a panel of experts, or in the behavioral sciences) and no function $d(x, y)$ may exist.

The matrix D may now be used as the representation of all points of the set X in a space with dimension n. Note that k-means type algorithms only work in metric spaces, and usually their extensions to non-metric cases are somewhat arbitrary. By representing data with this dissimilarity-based technique, we could apply this family of clustering algorithms even to non-metric data, e.g. categorical or mixed.

2 Clustering with Fuzzy Ranks

To tackle the dimensionality problem, a typical countermeasure found in traditional statistics is moving from the analysis of values (in our case, distances) to the analysis of their *ranks*. Rank is the position of a given value in the ordered list of all values. However in this work we adopt a fuzzy definition [7] of the concept of ranks.

Let ρ_{ij} be the rank of data point j with respect to data point i according to the set of dissimilarities $\{d(x_i, \cdot)\}$ when sorted in decreasing order with respect to values. This value is termed D-rank. It can be written in an algebraic fashion as:

$$\rho_{ij} = \sum_{k=1}^{n} \theta\left(d_{ij} - d_{ik}\right), \tag{1}$$

where the function $\theta(x)$ is an extended Heaviside step, taking on the values 0 for $x < 0$, 1 for $x > 0$, and 0.5 for $x = 0$, so $\rho_{ij} \in [0, \ldots, n-1] \; \forall i \in \{1, \ldots, n\}$. This extension of the Heaviside step represents the standard way to deal with ties in rank-order statistics.

It is now possible to measure the closeness of data points $x_1, \ldots, x_n$ by the concordance of their respective D-rank vectors $\rho(x_1), \ldots, \rho(x_n)$. We can therefore represent a data point x_i by the vector of its D-ranks:

$$x_i \longrightarrow \rho(x_i) = [\rho_{i1}, \ldots, \rho_{in}] \tag{2}$$

This definition has several advantages. It embeds the problem into a space of dimension n, which, by assumption, is smaller than the cardinality of the original data. Metric and non-metric cases are treated alike, since the new measure is numeric in both cases.

Using this representation of data, any metric clustering algorithm can be applied. In the experiments, we will refer to the procedure illustrated in the following section.

In a fuzzy set-theoretic perspective, it is more natural to define the relation "larger" among two numbers as a degree to which one number is larger than another. The problem of ranking fuzzy quantities has been reviewed for instance by Bortolan and Degani [8] and, more recently, by Wang and Kerre [9, 10].

For instance, suppose that we are to compare (a) $d_1 = 3$ with $d_2 = 4$, and (b) $d_1 = 3$ with $d_2 = 3.01$. Clearly in both case (a) and case (b) we can rightfully say that $d_2 > d_1$, but it is also clear that in (a) this is "more true" than in (b). Therefore, we can make the following substitution:

$$\theta\left(d_{ij} - d_{ik}\right) \longrightarrow \frac{1}{1 + e^{(d_{ij} - d_{ik})/\beta}} \tag{3}$$

where:

$$\lim_{\beta \to 0} \frac{1}{1 + e^{(d_{ij} - d_{ik})/\beta}} = \theta\left(d_{ij} - d_{ij}\right) \tag{4}$$

So the computation of fuzzy rank can be expressed as

$$\rho_{ij} = \sum_{k=1}^{n} \frac{1}{1 + e^{(d_{ij} - d_{ik})/\beta}} \tag{5}$$

The parameter β is a fuzziness parameter: for large β the ranking function is definitely fuzzy, while for $\beta = 0$ we obtain the original, crisp ranking function.

The two expressions (1) and (5) for the rank function $\rho_{\cdot\cdot}$ are compared in a simple example, illustrated in Figure 1, where the following set of values is used: $\{d, 2, 3, 5\}$. The diagram is a plot of $\rho_{d\cdot}$ (in the two expressions, crisp and fuzzy) for d in the range $[0, 7]$. Two plots are shown for the fuzzy expression, one for $\beta = 0.05$ and another for $\beta = 0.25$ (smoother).

This new definition of rank allows us to integrate into a clustering algorithm the notion that two ranks may be clearly defined (this happens when comparing very different values), and in this case the soft rank behaves similarly to the standard, crisp definition of ranks; or they may be less clearly defined (when the values to be compared are not very different), and in this case the soft rank takes into account the degree of closeness between the values. We want to exploit this added capability, and in the next section we present one possible proposal.

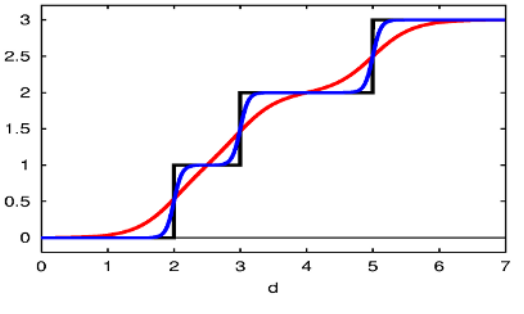


Fig. 1. Comparing crisp and fuzzy rank functions

3 Clustering Algorithms Using D-Ranks

To exploit the concept of soft-ranks, we apply a standard agglomerative hierarchical clustering algorithm to the soft D-ranks. We adopt the *agnes* procedure [11], which is available in the R language and environment [12], because it provides the agglomerative coefficient, which is defined as the average height of the mergers in a dendrogram.

We use the agglomerative coefficient to assess the value of β for which the clustering is best defined. However, with growing β the clusters in the soft D-rank space tend to collapse into one, since soft D-ranks tend all to their middle value $n/2$. Since we need to compare these values obtained on different scales, we use a different index which takes into account this problem. We define the weighted agglomerative coefficient a_w as follows:

$$a_w = a\left(\frac{max\{\rho_{ij}\} - min\{\rho_{ij}\}}{n}\right) \tag{6}$$

that equals zero for trivial clusters (one cluster for the whole data set).

4 Experiments

The method was tested on the publicly available Leukemia data by Golub et al. [13]. The Leukemia problem consists in characterizing two forms of acute leukemia, Acute Lymphoblastic Leukemia (ALL) and Acute Mieloid Leukemia (AML). The original work proposed both a supervised classification task ("class prediction") and an unsupervised characterization task ("class discovery"). Here we obviously focus on the latter, but we exploit the diagnostic information on the type of leukemia to assess the goodness of the clustering obtained.

The training data set contains 38 samples for which the expression level of 7129 genes has been measured with the DNA microarray technique (the interesting human genes are 6817, and the other are controls required by the technique). Of these samples, 27 are cases of ALL and 11 are cases of AML. Moreover, it is known that the ALL class is in reality composed of two different diseases, since they are originated from different cell lineages (either T-lineage or B-lineage). In the data set, ALL cases are the first 27 objects and AML cases are the last 11. Therefore, in the presented results, the object identifier can also indicate the class (ALL if id $\leq$ 27, AML if larger).

The test was performed according to the proposed method for a number of different fuzziness levels β. The weighted agglomeration coefficient a_w was used to assess the "best" fuzziness level, and diagrams were compared for several linkage methods. Specifically, the linkage methods used are: single (or nearest neighbor linkage); average (UPGMA); complete (or farthest neighbor linkage); weighted (WPGMA); ward (Ward's method with analysis of cluster variance).

The method which can be expected to find the most interesting clusters is Ward, since it is known to yield small and compact clusters not affected by the "chaining" effect, and this is fully confirmed by the experimental analysis.

Figure 2 shows the weighted agglomeration coefficient for the various linkage methods, with varying fuzziness level. The diagram shows that a peak is visible for a given

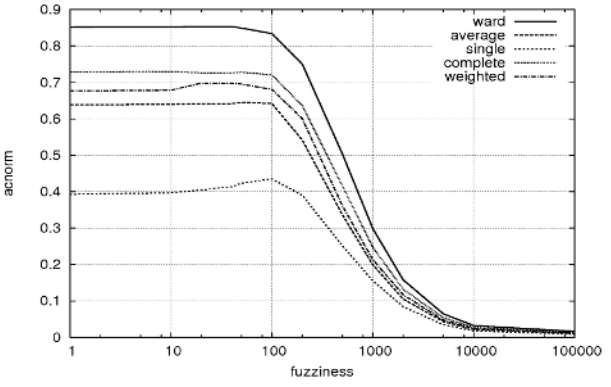


Fig. 2. Weighted agglomeration coefficient as a function of the fuzziness level β, for various linkage methods

value of β, especially for the less discriminative methods. For the method which finds the clearest clusters, the peak is hardly noticeable. Over a certain fuzziness threshold, the rank vectors tend to collapse in a single big cluster, and the agglomerative coefficient a rises; however the *weighted* agglomerative coefficient a_w plotted in the diagram correctly shows that this clustering is less and less significant.

Figure 3 shows the dendrograms obtained for some values of β with Ward's linkage method. The weighted agglomerative coefficient has a slight peak of 0.86 around $\beta = 50$, and inspection of the corresponding dendrogram reveals that indeed the clusters found are very adherent to the available domain knowledge. For instance, the first split in the dendrogram is between a cluster containing all ALL cases plus one AML case (observation 35), on the left, and a cluster containing only AML cases, on the right.

The subtypes of ALL are also fairly well evidenced, since all T-cell cases are in a major sub-cluster of the ALL cluster (on the right), with only two B-cell cases in the T-cell cluster.

For AML cases also prognostic informations are available, however the original work found no strong correlation between the molecular profile and therapy outcome (success or failure). This can also be found in the dendrogram we present here, since for cases 28-33 the outcome is failure and for cases 34-38 it is success, but no clustering is perceivable among these cases.

The analisys of clustering results thus reveals that the same conclusion which were obtained by several research step in ref. [13] are also found in a single application of our proposed method.

For comparison, we performed a cluster analysis according to three different methods: 1) agglomerative clustering on the original, 7129-dimensional data; 2) agglomerative clustering on the distance matrix; and 3) agglomerative clustering on the conventional rank matrix. In all cases, the obtained dendrograms do not show this level of match with respect to the knowledge available in the literature, thus confirming the superiority of the proposed method.

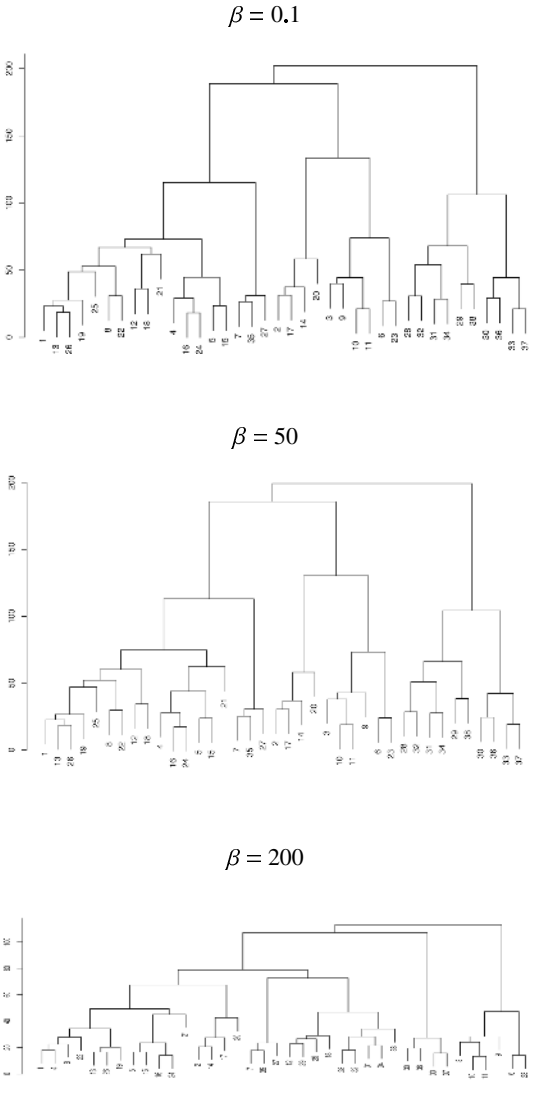


Fig. 3. Dendrograms obtained with different levels of fuzzines β with Ward's linkage method

5 Conclusions

We have presented a technique to perform clustering of high-dimensional data sets by mapping these data in a lower dimensional space, the space of fuzzy D-rank vectors. Several clustering techniques can be applied, and we used the standard *agnes* procedure to obtain an indication of the best value for the fuzziness parameter β. The analysis confirms the quality of the proposed procedure by comparison to the knowledge available in the literature, and its superiority to the other methods experimented.

Acknowledgment

Work funded by the Italian National Institute for the Physics of Matter (INFM), the Italian Ministry of Education, University and Research (2004 "Research Projects of Major National Interest", code 2004062740), and the Biopattern EU Network of Excellence.

References

1. Dunn, J.C.: A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. Journal of Cybernetics **3** (1974) 32–57
2. Bezdek, J.C.: Pattern recognition with fuzzy objective function algorithms. Plenum, New York (1981)
3. Aggarwal, C.C., Yu, P.S.: Redefining clustering for high-dimensional applications. IEEE Transactions on Knowledge and Data Engineering **14** (2002) 210–225
4. Beyer, K., Goldstein, J., Ramakrishnan, R., Shaft, U.: When is nearest neighbor meaningful? In: 7th International Conference on Database Theory Proceedings (ICDT'99), Springer-Verlag (1999) 217–235
5. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press (2004)
6. Pękalska, E., Paclík, P., Duin, R.P.W.: A generalized kernel approach to dissimilarity-based classification. Journal of Machine Learning Research **2** (2001) 175–211
7. Masulli, F., Rovetta, S.: Fuzzy variations in the training of vector quantizers. In: Proceedings of the 2003 International Workshop on Fuzzy Logics, Napoli, Italy. (2003)
8. Bortolan, G., Degani, R.: A review of some methods for ranking fuzzy sets. Fuzzy Sets and Systems **15** (1985) 1–19
9. Wang, W., Kerre, E.: Reasonable properties for the ordering of fuzzy quantities (I). Fuzzy Sets and Systems **118** (2001) 375–385
10. Wang, W., Kerre, E.: Reasonable properties for the ordering of fuzzy quantities (II). Fuzzy Sets and Systems **118** (2001) 386–405
11. Kaufman, L., Rousseeuw, P.J.: Finding Groups in Data. John Wiley & Sons, New York, USA (1990)
12. Ihaka, R., Gentleman, R.: R: A language for data analysis and graphics. Journal of Computational and Graphical Statistics **5** (1996) 299–314
13. Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C., Lander, E.: Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science **286** (1999) 531–537

An Evolution Hypothesis of Bacterial Populations

Bruno Apolloni[1], Alberto Clivio[2], Simone Bassis[3], Sabrina Gaito[1], and Dario Malchiodi[1]

[1] Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano, Via Comelico 39/41, 20135 Milano, Italy
{Apolloni, Gaito, Malchiodi}@dsi.unimi.it
[2] Dipartimento di Scienze Precliniche Lita Vialba, Università degli Studi di Milano, Via G. B. Grassi 74, 20157 Milano, Italy
Alberto.Clivio@unimi.it
[3] Dipartimento di Matematica "Federigo Enriques", Università degli Studi di Milano, Via Saldini 50, 20133 Milano, Italy
Bassis@mat.unimi.it

Abstract. We propose importing results from monotone game theory to model the evolution of a bacterial population under antibiotic attack. This allows considering the bacterium aging as a relevant phenomenon moving the evolution far away from the usual linear predator-prey paradigms. We obtain an almost nonparametric aging mechanism based on a thresholding operation, as an elementary intelligent operation, that may explain some typical patterns of the population evolution. In this paper we discuss both theoretical aspects and the results of a standardized procedure.

1 Introduction

Current evolutionary models for living microorganisms are based on a pair of randomness plus fitness criteria massively driving the population in a way that is more similar to the physical particles' laws than to higher level living organisms [1].

Said in other terms, the single particles have no age in the classical view, while the aging of a population, hence the irreversibility of their life, is decreed by external factors determining a selective death rate. The primary scheme is: bacteria randomly differentiate with time because of random changes in their genetic information; the most fitting ones with respect to an environment function survive with greater probability at each generation. Hence the rate of the fittest individuals strikes the time of the population.

Nevertheless individually aging bacteria have been observed in some experiments [2]. This moves (a part of) the population fitness into specific-goal functions of the single bacteria. As is well known, the composition of linear behaviors of single individuals may be exactly synthesized through a whole linear function at the population level. This is what exactly happens with usual predator-prey

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 214–230, 2006.

models [3, 4, 5]. Thus we expect this aging to be connected to a nonlinear behavior and focus on stepwise modifications of a state variable of the individual as a primary source of an attitude that we denote as "intelligence". Like in McCulloch and Pitts neuron [6] we site the computational ability (the intelligence indeed) of the cell into its capability of thresholding a signal coming from the outside with its inner status and using the result of this operation[1].

Actually, we consider that the common alternative way of dealing with immune system in terms of cellular automata [7] relies on exceeding logic capabilities of the involved microorganisms, as denoted by the simple fact that simulating a large scale population of them is unbearable even by modern computers. On the contrary, we look for bacteria computations that are even more elementary than McCulloch and Pitts neuron's in that:

1. we identify the status of the individual with its output ($\gamma_i = \tau_i$ with reference to Footnote 1);
2. we give a trend to the output thus synthesizing in a single operation the two levels of operations, computing and learning, describing the neuron life, where learning consists of system parameters' modifications;
3. we allow a one directional trend to the output, thus reducing the intelligence of the cell to the very primordial relation: if solicited, the cell reacts ever in the same way, but this reaction depends on the mentioned thresholding operation (which makes the true difference with non-living particles).

We have a theoretical model for understanding the potentiality of this elementary thought and a simulation tool for following the population evolution with this thinking mechanism. In the paper we review the former, that has been developed in detail in other papers, and discuss some bacterial population phenomena that will emerge from the simulations.

2 The Game and Its Biological Counterpart

We focus on a generic contest between agents having elementary game strategies in the aim of reading the evolution of a population of bacteria fighting against penicillin molecules within this paradigm. From a previous work we formulate a very essential game where nothing is known about the strategy of the players,

[1] In the McCulloch and Pitts model a neuron takes signals τ_j from other neurons, weights them with coefficients α_{ij} and compare their sum with its threshold γ_i

$$X_i(\boldsymbol{\tau}) = \sum_{j=1}^{\nu} \alpha_{ij}\tau_j + \gamma_i \tag{1}$$

hence its output is $\tau_i = h(X_i)$ with

$$h(X_i) = \begin{cases} 1 \text{ if } X_i \geq 0 \\ 0 \text{ if } X_i < 0. \end{cases} \tag{2}$$

apart from the fact that one of the players never changes strategy (hence, we speak of *game against Nature* [8]) and the other player has access to a parameter whose increment improves his ability to beat the adversary. These very poor conditions: maximum lack of knowledge plus a time direction, appear us to fit well with the roughness of the bacterial behavior, or at least with our awareness about it.

2.1 The Monotone Game

Bob and Alice (B and A for short) are playing the game described below, based on a three state monotone competition.

Game 1. *The game consists in a series of contrasts between B and A on random instances* $\boldsymbol{s}$ *drawn from a set* $\mathscr{S}$ *(huge, but finite and discrete) with the following properties:*

- *on each* $\boldsymbol{s}$ *B may i) win, ii) lose, or iii) tie with respect to A;*
- *each player owns an integer parameter, let us call them* strength γ_B *and* γ_A *respectively, which can assume a finite number of values. According to the order relation defeat* $<$ *tie* $<$ *victory, for a fixed value of* γ_A *the increment of B's strength does not diminish his contest results. Both for minimum* γ_A *and whatever* γ_B*, and for maximum* γ_B *and whatever* γ_A *B always either wins or ties;*
- *the two players have different roles:*
  - *A maintains her strength at a fixed point;*
  - *B increases his strength by one unit whenever he loses;*
- *B's goal is to find (learn) the minimal strength that will let him lose in the future with a probability below a small threshold. His strategy is to achieve this with a good confidence level. That means he will keep playing the game until he achieves this confidence.*

We solved this game elsewhere [9, 10] using two run-time statistics: the number k of defeats Bob suffered along the game history and the number $\widetilde{k}$ of defeats he should have to bear if he would have played the same contests with the current strength he has.

The main theorem is the following:

Theorem 1. *With reference to Game 1 and related statistics* $k, \widetilde{k}$ *(realizations of the random variables* $K, \widetilde{K}$*), denoting by*

- U_K *the random variable measuring the probability of defeat of a contests history having* $k, \widetilde{k}$ *as statistics of the past contests and continuing with an unlimited sequence of random drawn instances* $\boldsymbol{s}$*,*
- m *the number of played steps of our game,*
- μ_ε *the Binomial variable of parameters* m *and* ε*, and* ν_ε *the Binomial variable of parameters* k *and* ε*,*

we have

$$\mathrm{P}(\mu_\varepsilon \geq \widetilde{k}) \geq \mathrm{P}(U_k \leq \varepsilon) \tag{3}$$

$$\mathrm{P}(U_k \leq \varepsilon) \geq \mathrm{P}(\nu_\varepsilon \geq \widetilde{k} + 1) \tag{4}$$

The great value of this theorem stands in the fact that its claim expands to any game where a monotone effort–effect mechanism characterizes the players, even in the impossibility of giving it a quantitative description. A template is represented by the following:

Game 2. *Consider the atomic partition of the unitary mass of probability as in the first row of Fig. 1, where the balls, in number of ℓ, own probability $\frac{1}{\ell}$ each. Starting the game, the first $\rho(0)$ balls are white while the remaining ones are gray, and a cursor separates the two sets of balls. At each run t we draw a ball uniformly and replace it after having marked it: if the ball is white we do nothing more, if it is gray we move the cursor $\Delta\rho(t)$ balls right, changing color to the shifted balls (i.e. the new balls to the left of the cursor become white). $\Delta\rho(t)$ is not fixed a priori and can change from iteration to iteration.*

In this respect, k represents the number of cursor shifts, $\widetilde{k}$ the number of gray marked balls, and losing probability U_k the number of balls on the right of the cursor. A nice way of visualizing (3) and (4) is the seal diagram in Fig. 2 showing on the z-axis the extremes of 0.90 confidence interval for U_k (namely the 0.05 and 0.95 quantiles of this variable) as a function of k (x-axis) and $\widetilde{k}$ (y-axis). The graph shows that we may estimate the losing probability to have low values with a good accuracy when k is sensibly greater than $\widetilde{k}$, hence we may decide stopping increasing γ_B when it occurs.

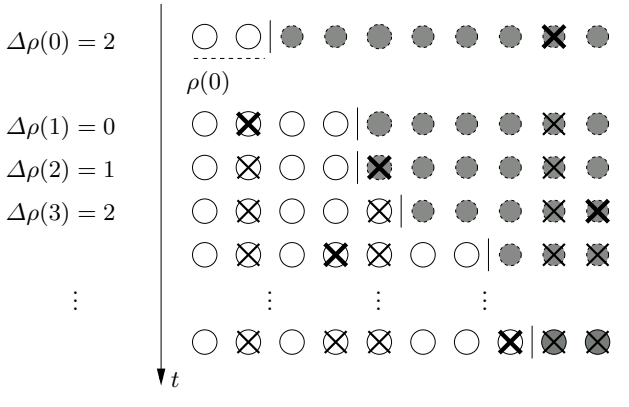


Fig. 1. An instance of Game 2 having $\ell = 10$ and $\rho(0) = 2$. The vertical axis reports the iteration number, each labeled with the corresponding $\Delta\rho(t)$. Marked balls are denoted by a cross, while bold crosses denote the balls drawn at each iteration. The cursor position is understood before the ball extraction and eventual updating of it.

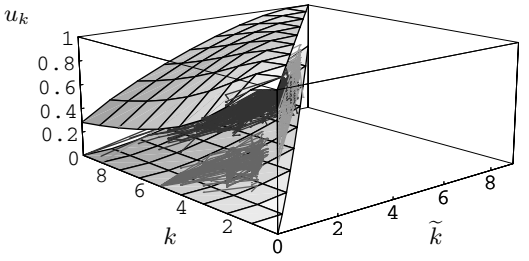


Fig. 2. 0.9 confidence intervals for losing probability u_k. Course of the lower and upper bounds (z-axis) with statistics of k (x-axis) and $\widetilde{k}$ (y-axis).

A way of enhancing $k-\widetilde{k}$ difference is the following. We play two games simultaneously: an *effective game*, where Bob uses the current strength to compute $\widetilde{k}$, and a *dummy game*, featuring a strength increment equal to zero, which lets him accumulate k. These statistics are the same as in a *virtual game* which, starting from 1, moves from this value and reaches Bob's current strength exactly when we compute the confidence interval. This game still complies with Game 2 definition and exploits the theoretical results obtained so far. Moreover it numerically denotes that the proposed strategy pushes the great part of the trajectories toward the $(k = \max, \widetilde{k} = 0)$ corner. In essence we obtain a better appreciation of the confidence intervals for the same learning process thanks to a better exploitation of the information available on them.

2.2 The Related Bacteria-Antibiotics Battle

The biological counterpart of this game may be represented by bacteria that differ in their resistance to a specific antibiotic (i.e. penicillin) as a consequence of their differential ability to produce and secrete an enzyme (penicillinase) which destroys the antibiotic provided in the culture medium. In this paper we will not consider the information processing aspects of the bacteria-antibiotic contests, such as bacterium recognition, specific mechanisms of action and so on. Rather we will deal with a predator-prey scenario constituted by a liquid medium where a bacterial colony is located in the presence of a given quantity of penicillin. In particular, we assume that a bacterium has a lever for regulating the partition of its energy between defense and reproductive activity. We may identify this lever with the strength γ_B and manage its increase with an analogous monotone rule. We essentially assume that each time B dummy loses against penicillin, so it is solicited to increase its strength, i.e. its defense attitude, unless it is refrained each time its strength is high enough in comparison with a ground state. Here ground state is represented by a mean effect of the rest of the colony B is embedded into. Namely, disregarding at the moment its physical correspondence we assume as ground state the mean strength of all colony individuals. Hence B is only pushed by its own need of surviving, being prone to increment its defense effort, i.e. penicillinase production, in diminution of the attitude to reproduce itself. The sole "altruistic" thought is to *decide* stopping increasing its strength

when it realizes to behave relatively better than the rest of the colony. The rest is pure mechanics: the concentration of penicillin decreases as much as it is destroyed by penicillinase; each B duplicates with a probability equal to the complement to 1 of the strength, etc.

Hence we model the dynamics of the population through a procedure that iteratively updates 4 quantities as follows:

1. set of living bacteria. Starting from an initial set of $n(0) = n_0$ bacteria, the set either increases or reduces depending on the fate of the single individual;
2. fate of the single bacterium. Each B either deads (Death(B)=1) or duplicates (Split(B)=1, or remains as it is ($D(\mathrm{B}) = 0$ & $S(\mathrm{B}) = 0$). Split and death are not exclusive events. Namely

$$\mathrm{P}\left(S(B) = 1\right) = 1 - \gamma_B \tag{5}$$

$$\mathrm{P}\left(D(B) = 1\right) = \begin{cases} \rho\mu_D \text{ if } S(B) = 0 \\ \rho\mu_E \text{ if } S(B) = 1 \end{cases} \tag{6}$$

   where ρ is the penicillin density in the solution, and μ_D and μ_E are mortality coefficients;
3. strength γ_B of the single bacterium. B decides incrementing γ_B of a quantity δ if the actual value of γ_B is greater than the average $\bar{\gamma}_B$ over all the colony by a factor less than a threshold τ. Namely

$$\gamma_B = \gamma_B + \delta \text{ if } \gamma_B < \tau\bar{\gamma}_B; \tag{7}$$

4. density ρ decreases of a quantity r each time an individual that is emitting penicillinase and is attached by penicillin does not die. Namely: on each time step a bacterium is invested by a penicillin molecule with probability ρ. Then we have a two level tree: at first level a bacterium either emits pencillinase (with probability γ_B) or does not emit it but duplicates (with probability $(1 - \gamma_B)$). In the first option it either dies with probability μ_E or destroyes a penicillin molecule with probability $(1 - \mu_E)$. In synthesis: $\rho = \rho - r$ if a matched emitting becterium does not die.

As a futher degree of freedom, we may consider a mixture of two variants B′ and B″ of a same bacterium. We distinguish the variants uniquely through the values of μ_i. Namely, we assume that the more resistant variant has $\mu_i' < \mu_i''$ for each $i \in \{D, E\}$.

Then, the dynamics of the populations are the same. We start from a given mixture of the two variants, and at each duplication we obtain progeny belonging to the same variant of the parent. Hence, variations of the mixture rate are uniquely due to the deaths/births history.

2.3 An Overlook on the Population Dynamics

In order to capture the gross dynamics of the population, we give the bacteria the same strength, possibly varying with time, and consider the evolution of the

expected values: $N(t)$ of the population size, $R(t)$ of the penicillin density. We derive them by solving the system of differential equations:

$$\begin{cases} N'(t) = N(t)\left((1-\gamma_B(t))(1-\mu_D R(t)) - \gamma_B(t)\mu_E R(t)\right) \\ R'(t) = -\frac{1}{L}N(t)\left(\gamma_B(t)(1-\mu_E)R(t)\right) \\ \gamma'_B(t) = \alpha(1-\gamma_B(t)) \end{cases} \tag{8}$$

with initial conditions for N, R and γ_B, α a suitable constant regulating the smoothness of the $\gamma_B(t)$ exponential trend, L the number of molecules bath where bacteria and penicillin are embedded, and disregarding some approximation problems connected with the continuity corrections starting from the differential equations actually describing our individuals and the expectations of product of variables brute force computed as the product of the single variable expectations.

Playing this simplyfied system we realize that:

a. for fixed concentration of penicillin and strength ($R' = 0, \gamma'_B = 0$) we come to a simple predator-prey process admitting equilibrium for $N(t)$ when $\gamma_B = \gamma^*_B = 1 + \frac{\rho\mu_E}{-1-\rho(\mu_D-\mu_E)}$ (or equivalently when $R = R^* = \frac{1-\gamma}{\mu_D-\gamma(\mu_D-\mu_E)}$). Obviously $N(t)$ moves to an exponential increase as soon as $\gamma_B < \gamma^*_B$ (or $R < R^*$), or decrease when $\gamma_B > \gamma^*_B$ (or $R > R^*$);
b. for decreasing concentration of penicillin and fixed strength ($R' \leq 0, \gamma'_B = 0$) we have equilibrium only for trivial conditions ($R = 0, \gamma_B = 1$). Otherwise we have either: i) an exponential decrease of $N(t)$ (in case γ_B is too high to contrast the penicillin); ii) an exponential increase, or iii) an initial decrease followed by an exponential increase (when the initial density of penicillin contrasts the reproductive ability of the bacteria, but this behavior invertes as soon as $\gamma_B(t)$ increment compensates penicillin decrease) as in Fig. 3.
c. γ_B smoothly variable from a certain value to 1 as in (8) has the benefit of greatly slackening and smoothing the dynamics of the population, thus avoiding the explosion of $N(t)$ – an undesirable condition both on the part of the bacteria and on the part of their host – on the contrary, giving the population slow decreasing trends (see Fig. 4) in a wider range of the free

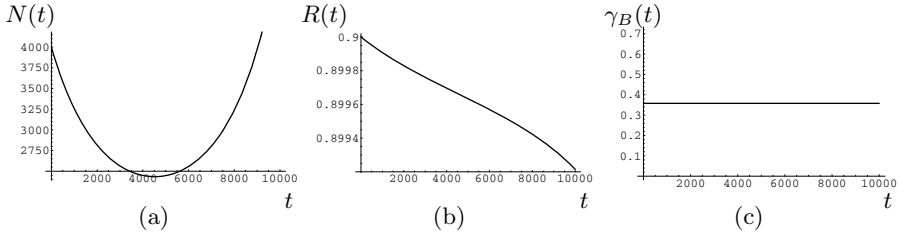


Fig. 3. Course of (a) $N(t)$ when (b) $R(t)$ is allowed to vary, $L = 10^{10}$ according to (8), and (c) $\gamma_B(t)$ is fixed to 0.36, while the mortality rates μ_E and μ_D are set to respectively 0.2 and 1

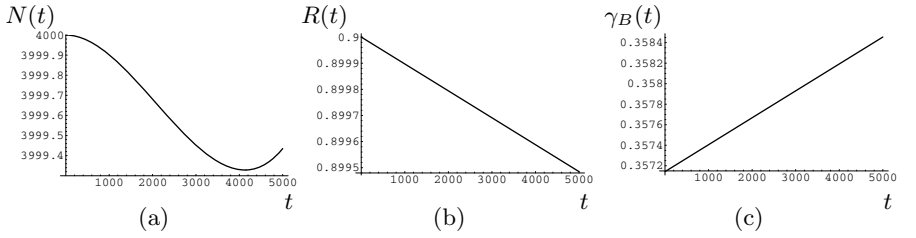


Fig. 4. Same notation as in Fig. 3, but with $R(t)$ and $\gamma_B(t)$ variable according to (8). The decay constant α is set to $4 \cdot 10^{-7}$.

parameters than in the case of γ_B fixed. This happens both for parameters denoting the environmental conditions (R) and for those characterizing the bacterial resistance to penicillin. The last benefit renders actually compatible the presence of more variants of the same bacterium in a particular biological environment.

2.4 Controlling the Ecological Loop

Following the histories of the single bacteria, starting from a random distribution of their strengths, we will attain an even more stable behavior of the population as a consequence of an indirect feedback. Actually our population is constituted by individuals working in an open loop, i.e. in the absence of a set point and feedback for attaining it. On the contrary the adaptation of the indivitual's attitude is monotone in the direction of increasing γ_B, hence without any control action reducing its value. Nevertheless curves like those in Fig. 5 denote oscillations around a metastable set point, before the occurrence of a disruptive event that either precipitates the population size to 0 or explode it.

The rough mechanism is the following:

- a γ_B increase strengthens the survival capability of an individual and rises the mean value of this parameter over the whole population. Nevertheless, this action, repeated on weaker individuals, is not sufficient to guarantee the population survival, unless its size rapidly explodes. Indeed, to avoid this explosion we need a mean strength not exceedingly high with respect to the mortality rate;
- the temporary equilibrium is guaranteed by the duplication of individuals, which occurs with greater probability on the weaker individuals, by definition. Hence, since the progeny maintain the strength of the parent (actually we admit a small increase to the weakest newborn with respect to the mean strength) the mean strength generally decreases after each duplication;
- these contrasting actions generally determine long term oscillation in the population size, where an exceeding oldness of the individuals (high $\bar{\gamma}_B$) pushes weaker individuals to reproduce, and this generates an exponential growth, since also the sons are weaker individuals. As the young individuals get old the reproduction decreases until a new run of this cycle.

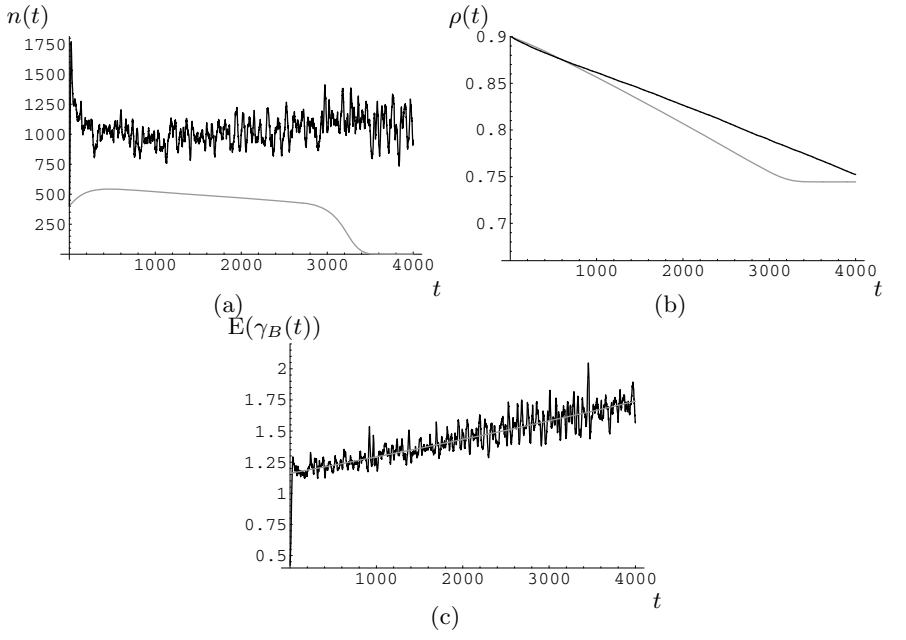


Fig. 5. Black lines: Global dynamic trajectories of (a) $N(t)$, (b) $\rho(t)$, and (c) $\mathrm{E}(\gamma(t))$ obtained setting in the simulation tool: $\mu_E = 0.2$, $\mu_D = 0.9$, $N(0) = 400$, $\rho(0) = 0.9$, $r = 10^{-7}$, $\delta = 0.022$ and the initial strenghts uniformly distributed in $[0, 0.8]$. Gray lines: corresponding trajectories computed applying a system of differential equations similar to (8), where $\gamma_B(t)$ is the function obtained by fitting the graph of $\mathrm{E}(\gamma(t))$ multiplied by a specific constant in order to prevent an exponential increment in the number of bacteria.

Putting in (8) a γ_B history that fits the average course of the individuals' strength evolution as in Fig. 5(c), we obtain a quite different process with a sharper dynamics. Actually we needed to multiply by a scale factor 0.89077 the γ_B trajectory in order to obtain the temporarily stable process described by the gray curves in Fig. 5. A slight change up or down in the last digit of the above constant would produce a sharp either explosion or implosion, respectively.

3 The Simulation Tool

The full model of the population evolution is ruled by a system of a huge set of nonlinear stochastic differential equations that does not find in general easy analytical solutions. Hence we set up a simulation tool for studying these dynamics. The basic operation is the thresholding of a uniform random number – as it is supplied by usual operating system routines – for simulating Bernoullian variables decreeing the various probabilistic step, e.g. whether a bacterium emits penicinillase or not, dies under penicillin attack or not, etc. The probabilities depend on free parameters as described in the previous section, with some technical tricks as follows:

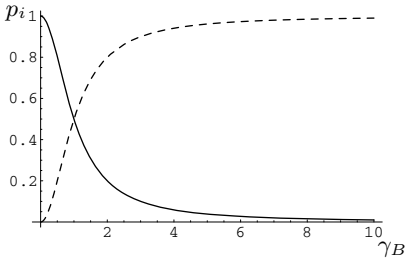


Fig. 6. Course of p_D (plain line) and p_E (dashed line) vs. γ_B

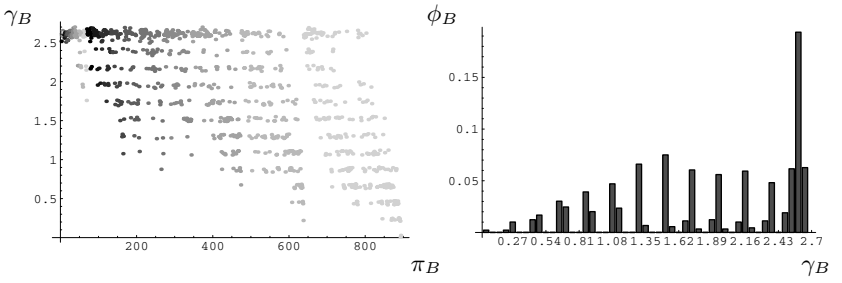
- γ_B increments. In order to smooth the duplication/emission probability variations near its 0/1 extreme values, we allow γ_B to vary without constraint and compute the duplication probability p_D as:

$$p_D = \frac{1/\gamma_B}{1/\gamma_B + \gamma_B} \tag{9}$$

  and the emitting probability p_E as $1 - p_D$ (see Fig. 6).
- Oscillation dumping. In order to smooth the $n(t)$ trajectories we constrain the γ_B variations with a hysteresis cycle. Namely, we require that after ν_u consecutive increases of its strength a bacterium must wait for ν_d steps before new strength increases.
- Newborn strengthening. Still in view of getting stability to $n(t)$ we increase the individuals' strengths at their generation if they are lower than the population average by a threshold factor τ_s.
- Ecological factor. We make the strength increments proportional to the cubic root of $n(t)$ to take into account the bacteria difficulty in finding food with the population increase.

All these facilities, together with the ruling parameters – namely, i) the amount of the single γ_B increments on regular and newborn individuals, ii) ν_u and ν_d, and within limited ranges iii) the threshold factors τ and τ_s as well, may modify the timing and shapes of $n(t)$ and γ_B's trajectories, but, according to the nonparametric value of Theorem 1 results, do not change the fate of the population dynamics, i.e. the extinction or explosion of the population. Its fate is determined by the following four factors: i) the initial amount of bacteria and ii) penicillin molecules, and the bacterial mortality rates iii) μ_D and iv) μ_E. This number obviously increases if more than one variant of the bacterium is present in the population. Moreover, relatively large changes in τ and τ_s may modify the process fate as well.

We visually follow the process evolution with two kinds of graphs: the former tracks the global dynamic parameters: $n(t)$, $\rho(t)$ and $E(\gamma_B(t))$ as seen in Fig. 5. The latter shows the spread of the strength on the population, possibly distinguishing between the individuals of different generations or also different variants (see Fig. 7).



(a) (b)

Fig. 7. (a) Graph of the strenghts distribution γ_B at iteration $t = 5000$ for each bacterium π_B, sorted from the oldest (lower abscissa) to the youngest (higher abscissa) individual. Different gray levels specify the bacterium generation time (cyclically from black to light gray). (b) Frequency ϕ_B of bacteria having strength γ_B in same intervals over the population generated at iteration $m = 5000$.

The procedure is available on web [11]. The user may submit her/his experiment by fixing the values of the free parameters in the format she/he finds in the web page. He may also ask for changing the updating rules through rules suggested by himself.

4 Simulation Examples

In this section we report a series of numerical experiments aimed at illustrating some typical behaviors of bacterial population in our model. As shown in Figs. 8(a), 9(a), 10(a), the course of $n(t)$ is more stable than in the case of constant uniform strength over the bacteria, virtually enlarging the running time of one to two orders. Moreover the balance between bacterium mortality rate and penicillin destruction rate proves generally more favorable in generating metastable courses instead of population explosion. Note that we must wait for over $40,000$ simulation steps to see the population implosion.

More interesting to see is the strength distribution within the population (see Figs. 8(c), 9(c), 10(c)). We always start with a uniform distribution between 0 and 0.8. A first effect of the updatings is the quantization of the strength, where the number of quantization levels is large (around 20 in our experiment when the system is in a metastable state, and reduces to less than half in case of implosion of the population. This phenomenom is related to the constant strength increment δ [2].

Note that quantization is a beneficial effect. Indeed, if we allow a random increment δ of the strength γ_B (see Fig. 11), so as to guarantee a higher variability in the strengths' distribution, we are no longer able to maintain the same stable bahavior of the population, like in the previous cases, thus originating a bacteria explosion also in correspondence to very high strength increments δ (as instance

[2] It can be better appreciated in the figures charachterized by having a narrower γ_B's range, due to the discretization applied in the graph construction.

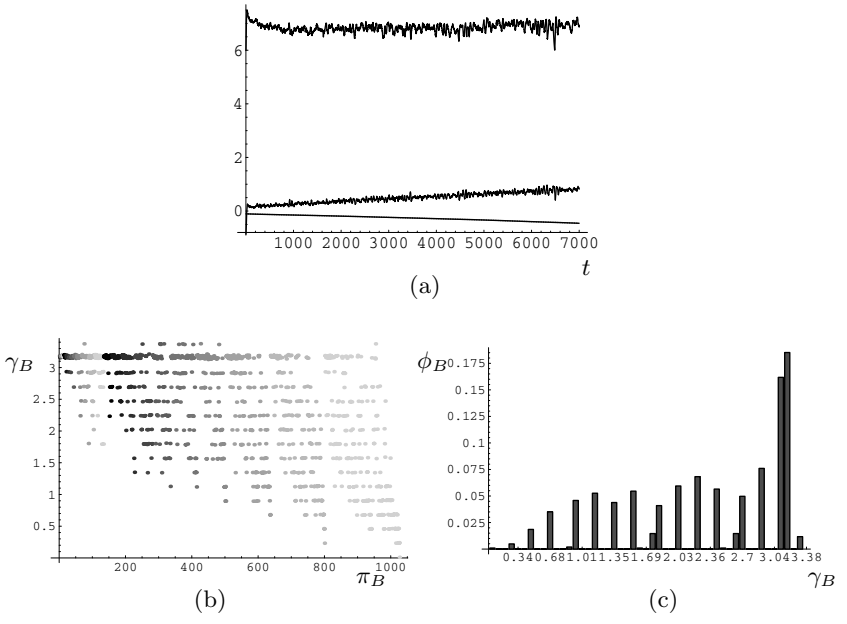


Fig. 8. A stable population behavior. (a) Course of $n(t)$, $\mathrm{E}(\gamma(t))$ and $\rho(t)$ (from top to the bottom) in logarithmic scale (natural logarithm), when the parameters are the same as in Fig. 5. (b) and (c) Same notation as in Fig. 7.

δ is a random number belonging to $[0, 0.2]$ in Fig. 11). This is due to the high concentration of weakest individuals that continue the doubling process unaware of their own fate.

Also the spread shape is characteristic (see Figs. 8(b), 9(b), 10(b)). We may recognize a certain number of gaussian distributions (for instance 3 in Fig. 8(c)) where the rightmost one is compressed toward a maximum that is typical of the experiment. The evolution of these curves precisely denote the course of the process. They remain with the same characteristics for stable behaviors. Note that Fig. 8 is the continuation of the life history of the same population in Figs. 5 and 7. The number of distributions and their variances decrease when we approach implosion. In order to improve their visualization, we show the strengths spread at iteration 5000, of a more gently reducing population, so that we have around 200 individuals still alive. In the case of explosion, on the contrary, the curves merge to such a point that it becomes difficult to recognize them, with the sole exception of the rightmost distribution that sharply concentrates in its average.

Another reading key of these graphs concerns the temporal evolution of the strengths that we may recover thanks to the gray tune changes in the points. The tunes are cyclically associated to the generation number, low abscissa values corresponding to the oldest bacteria. *Vice versa* youngest bacteria have high value of abscissa. From these graphs we may see the increase of strength

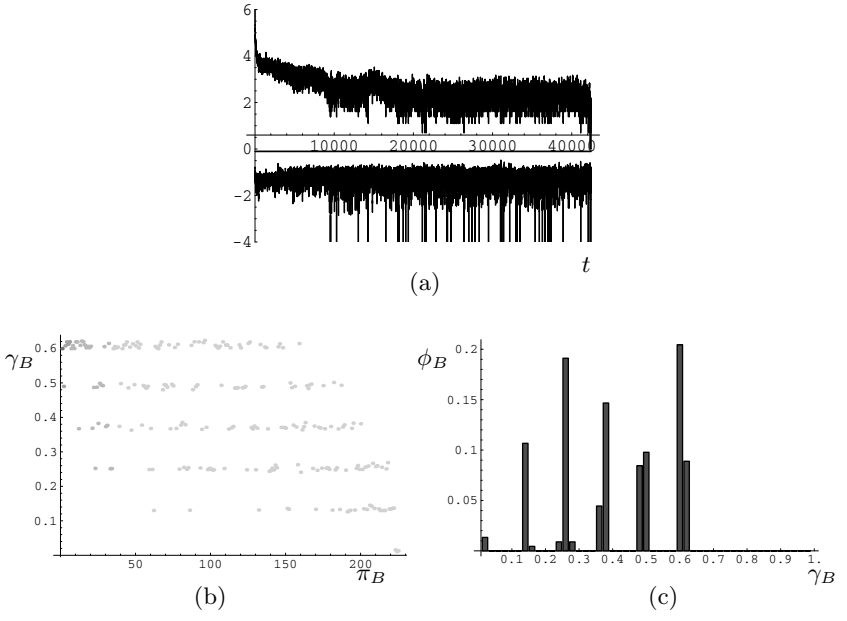
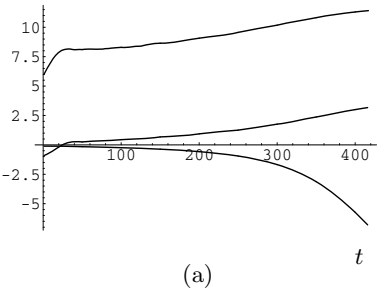


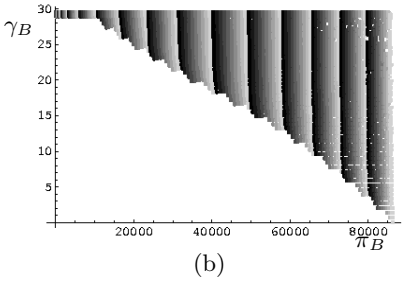
Fig. 9. A population going to implode. Same notation as if Fig. 8 (with $n(t)$, $\rho(t)$ and $\mathrm{E}(\gamma(t))$ from the top to the bottom), with: (a) $\mu_D = 0.97$, $\mu_E = 0.94$ and $\delta = 0.1$; (b)-(c) $\mu_D = 0.97$, $\mu_E = 0.79$ and $\delta = 0.08$.

with bacterium aging, as well as the high survival rate of the oldest individuals. Young individuals duplicate and die frequently, thus mantaining a low average strength. A population is vital when the newborn span all the strength spectrum. It implodes when no sufficient old individuals cover the higher part of the spectrum. In some extent, we may say that the population behaves as an aggregate of different phenotypes even if the differences are not due to different genotypes.

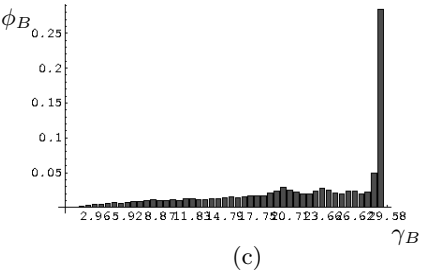
In case of different genotypes (different versions of the same bacterium), base histories are the same for each population, but they interfere in a complex way (see Fig. 12). Essentially, we have an initial overwhelming of the strongest population but a long term revenge of the weakest one. The reader may remember that we distinguish the two populations by a multiplicative factor mapping the mortality rate of one population onto the latter's. All other parameters are the same. We have a first phase where the stronger population dominates the latter getting higher values to their strength and leaving the reproduction work mainly to the weakest population. After a first metastable plateau (see Fig. 12(a)), the first population individuals die, although they had reached greatest strengths (top gray line in Fig. 12(c)), thus giving completely camp to the other population. At genotypical level we may imagine that strongest individuals are renewed by the long term mutation process along the generations giving rise to a sequence of variants' alternations like the one described by Fig. 12.



(a)

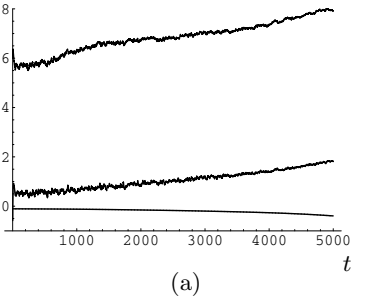


(b)

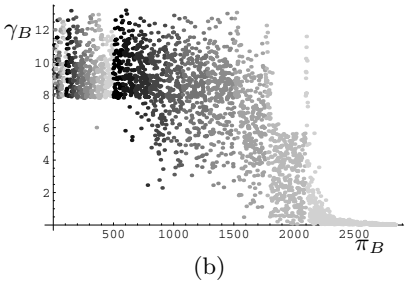


(c)

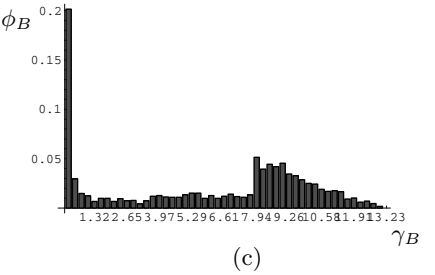
Fig. 10. A population going to explode. Same notation as if Fig. 8, with $r = 10^{-6}$ and $\delta = 0.02$.



(a)



(b)



(c)

Fig. 11. Same notation as in Fig. 8 with the sole exception that δ is randomly chosen in $[0, 0.2]$ to avoid quantization phenomenon

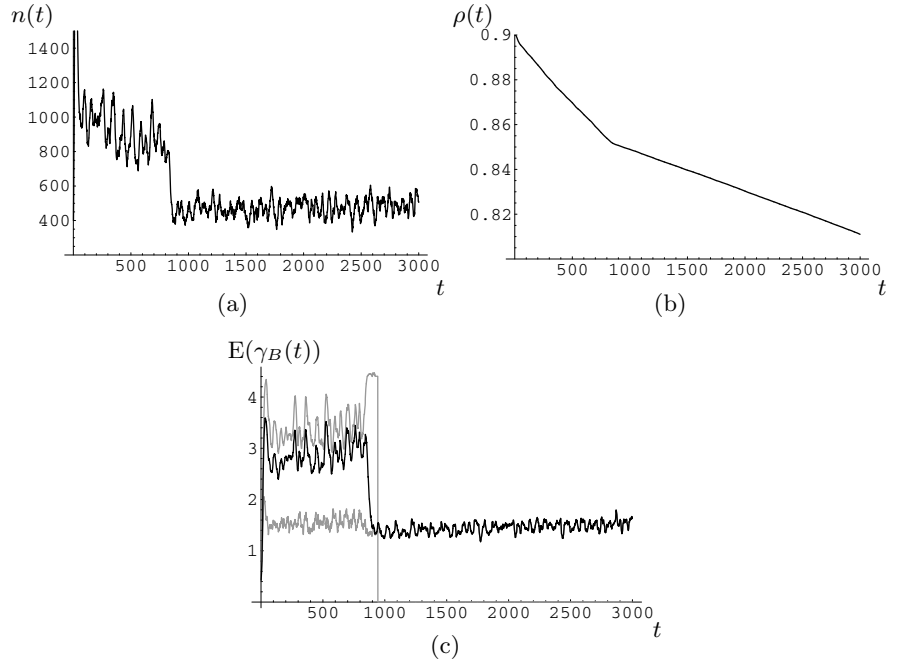


Fig. 12. A merge of two bacteria variants. Global dynamic trajectories of (a) $N(t)$, (b) $\rho(t)$, and (c) $\mathrm{E}(\gamma(t))$ in case of different versions of the same bacterium, obtained setting in the simulation tool: $N(0) = 400$, $\rho(0) = 0.9$, $r = 10^{-7}$, $\delta = 0.03$ and the initial strenghts uniformly distributed in $[0, 0.8]$. The weakest group (initially constituting the 90% of the total bacteria population) has mortality rate of $\mu'_D = 0.9$ and $\mu'_E = 0.2$, while the strongest one $\mu''_D = 0.45$ and $\mu''_E = 0.1$. (c) Gray lines: $\mathrm{E}(\gamma(t))$ of both strongest and weakest group.

5 Conclusion

We apply at its greatest extent the connectionist paradigm – elementary local computations plus a dense connection grid to generate complex behaviors [12] – to explore the principled dynamics of bacterial populations. Hence we abandon the complex ontologies attributed to the single cells by finite automata approach such as Cellular Automata [13] or Multi Agent Systems [14], since their implementation reqires a non disregardable number of boolean gates, and came to individuals endowed with the sole addition operation with a 0. Moreover we commit to the interaction with other cells the negation operation (hence the subtraction) necessary to complement any principled universal algebra. In this way we experimented two loops stabilizing the bacterial population: the phenotypical level through a balance of resistance reproduction activity in the whole population, and the genotypical level through a turn-over between variously resistant bacterial variants.

The almost unavoidable drawback of this approach is the impossibility of getting analytical synthesis of this dynamics. Rather, we elaborate a theory

that is at most extent parameter free, in the sense that some few statistics are sufficient to conditioning the long term evolution of the population, and the remaining ones prime different trajectories toward the asymptotes (a somehow intronic effect). In this sense, what we are proposing is the development of a new representation language which ensures the advantages of both the differential equation approach (macroscale) and of cellular automata (microscale) for the description of the behavior of complex systems [15]. It can focus on the global population dynamics as well as on the events occurrency at the single cell level. We have successfully tested this approach on a very simplified biological model (bacterial cells and antibiotic resistance). Future work will be directed to:

- addressing scalability issues after having configured ours as a dialect of efficient languages suitable for managing stochastic simulation [16, 17], such as *stochastic Pi-calculus* [18];
- applying the model to inspect complex systems in which events such as reciprocal interactions between microorganisms and components of the innate immune system, such as *complement* [19], are important for individual survival (idiotypic and neural networks, for example) and might result in the development of new vaccination approaches [20].

References

1. Garrett, S.: Parameter-free, adaptive clonal selection. In: Congress on Evolutionary Computing, CEC. IEEE (2004)
2. Stewart, E., Madden, R., Paul, G., Taddei, F.: Aging and death in an organism that reproduces by morphologically symmetric division. PLoS Biology **3** (2005) 295–300
3. Capasso, V.: Mathematical Structures of Epidemic Systems. Springer-Verlag, New York (1992)
4. Prikrylova, D., J'ilek, M., Waniewski, J.: Mathematical Modeling of the Immune Response. CRC Press (1992)
5. Perelson, A.S.: Modelling viral and immune system dynamics. Nature Rev. Immunol. **2** (2002) 28–36
6. McCulloch, W., Pitts, W.: A logical calculus of ideas immanent in nervous activity. Bulletin of Mathematical Biophysics **5** (1943) 115–133
7. Castiglione, F.: A network of cellular automata for the simulation of the immune system. International Journal Morden Physics C 10 (1999) 677–686
8. Blackwell, D., Girshick, M.A.: Theory of Games and Statistical Decisions. Dover Publications, Inc., New York (1979)
9. Apolloni, B., Bassis, S., Gaito, S., Malchiodi, D., Zoppis, I.: Facing indeterminacy for winning a monotone game. Information Sciences (2005) to appear.
10. Apolloni, B., Bassis, S., Gaito, S., Malchiodi, D.: Cooperative games in a stochastic environment. In Apolloni, B., Kurfess, F., eds.: From synapses to rules. Discovering symbolic rules from neural processed data. Kluwer Academic/Plenum Publishers, New York (2002) 75–86
11. Tool, B.S.M.: `http://laren.usr.dsi.unimi.it/Batteri/index.html` (2005)
12. Rumelhart, D.E.: Parallel Distributed Processing. Volume 1. MIT Press, Cambridge (1986)

13. Wolfram, S.: Theory and Applications of Cellular Automata. World Press (1986)
14. Ferber, J.: Multi-Agents Systems. AddisonWesley (1999)
15. Kleinstein, S.H., Seiden, P.E.: Simulating the immune system. IEEE Computing in Science and Engineering **2** (2000) 69–77
16. Gillespie, D.: A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. Journal of Computational Physics **22** (1976) 403–434
17. Turner, T., Schnell, S., Burrage, K.: Stochastic approaches for modelling in vivo reactions. Computational Biology and Chemistry **28** (2004) 165–178
18. Regev, A., Shapiro, E.: Cells as computation. Nature (2002) 343–419
19. Stoiber, H., Clivio, A., Dierich, M.P.: Role of complement in hiv infection. Annual Review of Immunology **15** (1997) 649–674
20. Dierich, M.P., Stoiber, H., Clivio, A.: A "complementary" aids vaccine. Nature Medicine **2** (1996) 153–155

Modelling the Immune System with Situated Agents

Stefania Bandini[1], Franco Celada[2], Sara Manzoni[1], Roberto Puzone[3], and Giuseppe Vizzari[1]

[1] Dipartimento di Informatica, Sistemistica e Comunicazione, Università degli Studi di Milano–Bicocca, Via Bicocca degli Arcimboldi, 8, 20126 Milano, Italy
{bandini, manzoni, vizzari}@disco.unimib.it
[2] Università di Genova, Viale Benedetto XV 10, 16132 Genova, Italy
celada@cba.unige.it
[3] National Institute for Cancer Research Epidemiology and Prevention, L.go Benzi 10, 16132 Genova, Italy
puzone@unige.it

Abstract. The Immune System (IS) represents the defence mechanism of higher level organisms to micro organismic threats. It is a genuinely distributed system provided with mechanisms of adaptation to unknown threats by means of the cooperation of heterogenous autonomous entities. Features of the overall systems, such as learning capabilities, possibility to tackle unknown threats in any part of the body, are a consequence of these interactions. This paper describes how a Multi-Agent approach, and more precisely the Situated Cellular Agents (SCA) model, can be suitably applied to represent specific elements and mechanisms of the IS. After a brief description of the composing parts and the internal mechanisms of the IS, the SCA model will be introduced and exploited to represent them[1].

1 Introduction

The Immune System (IS) of vertebrates constitutes the defence mechanism of these higher level organisms (i.e. fishes, reptiles, birds and mammals) to molecular and micro organismic invaders. It is made up of some specific organs (e.g. thymus, spleen, lymph nodes) and of a huge number of cells (i.e. $10^{12} - 10^{13}$ in a man) of different kinds that have or acquire distinct functions. The response of the IS to the introduction of a foreign substance (antigen) that might be harmful involves thus a collective and coordinated response of many autonomous entities.

Different models of the IS have been devised (mostly based on an analytical approach [11], and some others based on Cellular Automata (CA) [14]) for different purposes, from prediction of effects of experimental modifications on the IS,

[1] The work presented in this paper has been partially funded by the Italian Ministry of University and Research within the project 'Cofinanziamento Programmi di Ricerca di Interesse Nazionale'.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 231–243, 2006.

to educational programs [10]. To analyze and model the IS means to study natural means of detecting harmful intrusions and effectively respond to the threat. To study the IS represents thus a way to gain insight on possible methods to prevent and tackle threats to artificial systems, such as computer networks. In a scenario where computational devices and connectivity are spreading at an incredible speed, with a growing interest in mobile autonomous agents, security issues must be carefully considered and the biological metaphor could lead to novel and more effective security models (see, e.g., [6]). The latter is just one example of applications inspired by the IS, but even its adaptation and coordination mechanisms might be useful in other areas (e.g. autonomous aircraft control [8]). In fact the growing availability of inexpensive computational units, connected and distributed over a network that is more and more "un–wired" and being thus able to move, has highlighted the need of context–aware applications. The latter must thus be able to adapt to changes in the context of execution (both geographical and logical, with reference to user's profile, needs and tasks), and cooperate with remote entities to obtain information and, in general, to be able to carry out specific operations.

Most models for the IS follow an analytical approach: differential equations systems are set up to represent some part of the IS, then they are studied (generally using numerical integration). This approach, despite its popularity, has some limitations [7], but the main one, with reference to our approach, is the distance between immunologist language and the formal definition of a mathematical model of the IS.

Other approaches presented in the literature represent components and processes of interest of the IS and simulate its behaviour according to computational models. A relevant case, with reference to our approach, is the adoption of CA (see, e.g., [1][4]). In this case entities to be described are more than just variables or parameters in an equation: they can be related to data structures and even objects present in a particular body area (a cell in the CA) and interacting according to specific rules, defined by immunologists. In this approach there is a clearer correspondence between domain entities and model concepts, therefore it is easier for the immunologist to interact with it using her language, rather than the mathematical one. This change of approach even brings more insight on details of IS response process, allowing for instance to keep track of cell concentrations in specific areas (cells of the CA). The internal evolution of entities within a single cell of the CA is determined through a probabilistic mechanism that considers the possibility of every entity to interact with every other one present in the cell (i.e. the cell is a completely connected graph). A possible issue with this approach is that rules of interaction between cells and other IS entities must be defined as global, because generally all elements of a CA share the same transition rule. Therefore a computer system based on this kind of model has to know how to handle all possible interactions between different types of entities. This problem is particularly serious as research in the area of immunology is very active and the understanding of the mechanisms of the IS is still far from complete. A careful design, according to best–practices derived by

research in the area of software engineering, can help facing technical aspects of this problem, but it probably does not make the system easier to understand. A further step in the direction of an increased accessibility for immunologists provides a direct correspondence between entities of the IS and autonomous entities of the computer system. The Multi Agent System (MAS) [5] approach is a well known computational model supporting simulation (see, e.g., [9]). The main goal of this paper is to show how it was applied in the modelling of the IS system through the adoption of Situated Cellular Agents [3]. The SCA model, a particular class of Multilayered Multi Agent Situated System (MMASS [2]), allows a more detailed representation of the interaction between entities, that are inherently space dependant more than just probabilistic. The graph structure representing the environment is more precise than the previously described CA approach because a node may contain at most a cell, that will be able to interact directly only with other adjacent cells, preserving detailed spatial relations between cells. Moreover the SCA model allows the representation of action–at–a–distance that will be exploited, for instance, to perform virus and antibody diffusion. These kind of models share with CA models the limitations related to the fact that they can only manage a very limited number of cells (compared to biological numbers), to limit the computational complexity of the simulation. Accurate number scaling must and can be done, but these approaches must be considered as qualitative models of the IS response with very wide capability of representing complex interaction between many or most IS different specific actors. Analytical models, on the other hand, can be considered quantitative models of IS response. This means that they carry the precious ability to use true experimental values for their parameters, but they must strongly reduce the complexity and number of different actors, and are mostly unable to represent the individual specific behaviour of IS actors, often considered one of the core features of the IS.

In the following Section some basic concepts related to the IS will be described in order to show how a multi agent approach can be suitably applied to simulate the IS. A through description of the IS, its components and its way of reacting in order to face infections is out of the scope of this paper; more details on the subject may be obtained from [13]. Nonetheless some kinds of mechanisms and interactions between the entities of the IS will be briefly illustrated, in order to show some examples of its internal modalities of collaboration, adaptation and learning. Section 3 will briefly introduce the SCA model, then a description of how this model can be applied to represent elements and mechanisms of the IS will be given. Conclusions and future developments will end the paper.

2 Immune System Description: The Biological Model

The task of the IS of an individual (e.g. mammals) is to detect the presence of foreign antigens inside its body and react to eliminate the related threat. Examples of antigens are foreign substances (as venoms), viruses, bacteria, parasites, and so on. While many general foreign antigens' characteristics are intrinsically known and recognized by the IS (by the so called natural "aspecific" component

of the IS, evolutionary determined), many other antigens (e.g. often mutating viruses) are genetically aprioristically unknown to the IS, as well as the possible location where the infection will take place.

Therefore IS has components (the adaptive IS) that are able to tackle unknown challenges, remember them in order to efficiently face following infections of the same kind, covering all the body. The adaptive IS is composed of many specifically and aspecifically interacting actors, and how this complex interaction can lead to a general very efficient global behaviour, it is the main issue targeted by most IS simulators.

The main kinds of IS mobile entities are the following:

- *antibodies* are proteins able to bind to a specific antigen, neutralizing its possible harmful effects;
- *B lymphocytes*, also referred to as B cells, may bind to a specific antigen and are responsible for the production of antibodies related to it. They may also differentiate into memory cells, able to take part in future immune responses if the same antigen in encountered again;
- *T lymphocytes*, also referred to as T cells, can be divided into different categories according to their function: for instance *killers* are able to recognize cells infected by a specific antigen (i.e. containing it) and *helpers* enhance the production of antibodies, stimulating the proliferation of the related B cells;
- *antigen presenting cell* (APC), also called accessory cells, are a set of different kinds of cell (e.g. macrophages, dendritic cells) that are able to identify and process antigens in order to make them recognizable by T cells (e.g. a macrophage can phagocytize an antigen and present a part of it over its surface).

All these kinds of entities are able to circulate in blood, and the last three of them belong to the white blood cells category. Static parts of the IS have a role in the life–cycle of these cells: lymphocytes are generated in the bone marrow, and mature in the thymus. Both B and T cells take residence into lymph nodes, the spleen and other tissues where they can encounter antigens, proliferate and evolve (through mitosis), and mature into fully functional cells.

Leaving aside APCs, the entities described above are devoted to a specific antigen. This means that they are able to interact with a specific antigen, in a direct way (i.e. through the interaction of their membranes) or indirectly (e.g. through a direct interaction with an infected cell, or an APC that had ingested it). The fundamental element for all these interactions is the membrane of these entities: IS specific cells have a receptor (characteristic molecular configuration) repeated all over their surface. If it is chemically "compatible" with the surface of another entity they encounter they can bind with it. In order to be able to obtain B and T cells for a number of possible antigens, whose characteristics are unknown to the IS, a large number of B and T cells (10^{10} at least) with different receptors must be generated. Moreover, during the life of those cells, somatic mutations can bring the number of receptors up to the order of 10^{16}. In

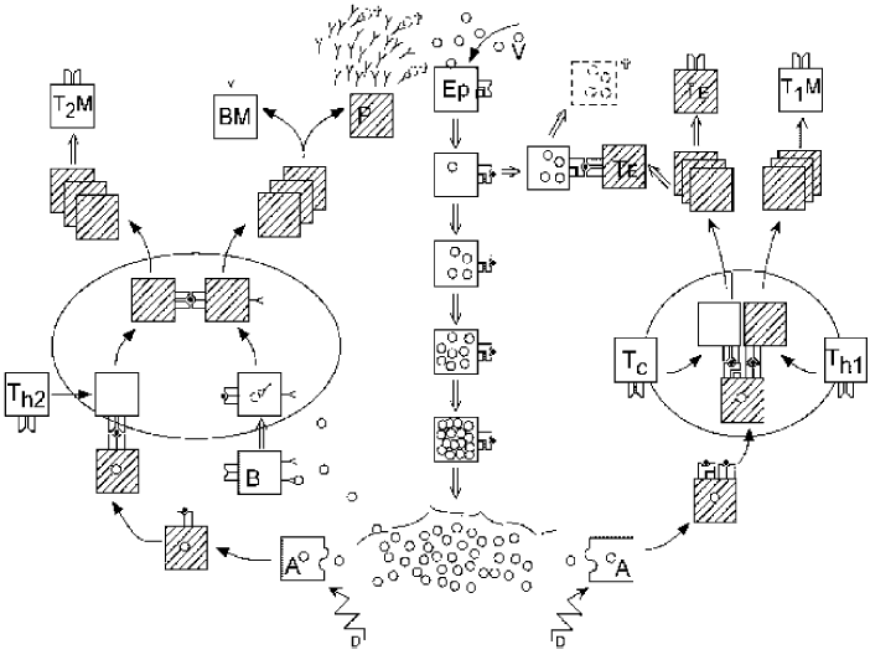


Fig. 1. Scheme of IS humoral and cellular response to a viral infection

other words, the IS does not know what specific antigens will be encountered, but the "building blocks" of their membrane are known, so combining them in a sufficient number of variants will bring reasonable protection.

Figure 1[2], shows two different kinds of response of the IS to defend the body from micro–invaders. The central column shows the evolution of a epithelial cell (EP) infected by a virus, denoted by a small circle. The latter enters the cell, proliferates inside it, and ultimately causes the burst of the cell, the release of more viruses and the cast of a damage signal. Another effect of the infection is that part of the virus is exposed over the membrane of the infected cell. There are thus two possible kinds of reaction to the infection, as the viruses to be neutralized may be found outside body's cells (e.g. freely wandering in the blood), or inside already infected cells. B cells specific for this antigen (i.e. with a receptor compatible to the virus membrane), are already able to capture and bind the free virus, but this is not enough to face the infection. Both kind of responses are triggered by an APC (denoted with A in Figure 1), that is activated by the damage signal and ingests a virus. Part of it is exposed in the membrane of the APC, and can be recognized by a helper T cell (T_{h2} in Figure) whose receptor is compatible with the virus membrane. These helpers

[2] This Figure was taken from R. Puzone, B. Kohler, P.Seiden, F. Celada, IMMSIM, a flexible model for in machina experiments on immune system responses, *Future Generation Computer Systems*, Vol. 18, No. 7, pp. 961–972, Elsevier, 2002.

can stimulate B cells to reproduce themselves, differentiating into B memory cells and plasma cells, which in turn will produce antibodies that are able to eliminate free viruses. This is generally called humoral reaction, and it is shown on the left side of Figure 1. The right side shows cellular reaction, the other kind of response to an infection whose task is to eliminate infected cells. The APC cell exposing the ingested virus can activate other helpers that will in turn activate killer T cells (denoted by T_c in Figure). The latter will proliferate and bind to infected cells, eliminating the related threat.

3 The Computational Model: Situated Cellular Agents

A system of Situated Cellular Agents can be denoted by:

$$\langle Space, F, A \rangle$$

where A is a finite set of agents, F is a finite set of fields, and *Space* is a single layered environment where agents are situated, act autonomously and interact by means of reaction or through the propagation of fields.

The possibility to define different agent types introduces heterogeneity, in other words the chance to define different abilities and perceptive capabilities. Defining T the set of types, it is appropriate to partition the set of agents in disjoint subsets corresponding to different types. The set of agents can thus be defined as

$$A = \bigcup_{\tau \in T} A_\tau$$

where $A_{\tau_i} \cap A_{\tau_j} = \emptyset$ for $i \neq j$. An agent type τ is defined by

$$\langle \Sigma_\tau, Perception_\tau, Action_\tau \rangle$$

where:

- Σ_τ defines the set of states that agents of type τ can assume;
- $Perception_\tau : \Sigma_\tau \rightarrow [\mathbb{N} \times W_{f_1}] \ldots [\mathbb{N} \times W_{f_{|F|}}]$ is a function associating to each agent state the vector of pairs representing respectively a receptiveness coefficient modulating the intensity of that kind of field and a sensitivity threshold represented by a specific field value; these functions represent the perceptive capabilities specification for that type of agent and their usage will be clarified in Section 3.2. Formally, this vector of pairs is defined as

$$\Big(c_\tau^1(s), t_\tau^1(s)\Big), \Big(c_\tau^2(s), t_\tau^2(s)\Big), \ldots, \Big(c_\tau^{|F|}(s), t_\tau^{|F|}(s)\Big)$$

  where for each i $(i = 1 \ldots |F|)$, $c_\tau^i(s)$ and $t_\tau^i(s)$ express respectively a receptiveness coefficient to be applied to the field value f_i and the agent sensibility threshold to f_i in the given agent state s.
- $Actions_\tau$ denotes the set of actions that agents of type τ can perform, and will be described in Section 3.3.

3.1 Space

The *Space* consists of a set P of sites arranged in a network (i.e. an undirected graph of sites). Each *site* $p \in P$ can contain at most one agent and is defined by $\langle a_p, F_p, P_p \rangle$ where $a_p \in A \cup \{\perp\}$ is the agent situated in p ($a_p = \perp$ when no agent is situated in p, in other words p is empty); $F_p \subseteq F$ is the set of fields active in p ($F_p = \emptyset$ when no field is active in p); and $P_p \subset P$ is the set of sites adjacent to p.

3.2 Fields

A field $f_\tau \in F$ that can be emitted by agents of type τ is denoted by

$$\langle W_\tau, Diffusion_\tau, Compare_\tau, Compose_\tau \rangle$$

where:

- $W_\tau = S \times \mathbb{N}$, where $S \subseteq \Sigma_\tau$, denotes the set of values that the field can assume; given $w_\tau \in W_\tau$, $w_\tau = \langle s_\tau, i_\tau \rangle$, where $s \in S$ represents information brought by the field (i.e. the field payload) and $i_\tau \in \mathbb{N}$ represents its intensity.
- $Diffusion_\tau : P \times W_\tau \times P^{p_d} \rightarrow (W_\tau)$ is the diffusion function for field type τ; $Diffusion_\tau(p_s, w_\tau, p_d)$ computes the value of a field on a given destination site (p_d) taking into account in which site it was emitted (p_s) and with which initial value ($w_\tau \in W_\tau$).
- $Compare_\tau : W_\tau \times W_\tau \rightarrow \{True, False\}$ is the function that compares field values. It is used by the perceptive system of agents to evaluate if the value of a certain field type is such that it can be perceived.
- $Compose_\tau : (W_\tau)^+ \rightarrow W_\tau$ expresses how field values of the same type have to be combined in order to obtain the unique value of a field type at a given site.

3.3 Situated Cellular Agents

An agent $a \in A$ is thus defined by the three–tuple $\langle s, p, \tau \rangle$, where:

- $s \in \Sigma_\tau$ denotes the *agent state* and can assume one of the values specified by its type;
- $p \in P$ is the site of the *Space* where the agent is situated;
- τ is the *agent type*, which provides the allowed states, perceptive capabilities and behavioural specification for that type of agents.

The first two elements were previously introduced, while this Section will focus on $Action_\tau$, which is made up of a set of actions and an action selection strategy. Actions can be selected from a set of primitives which include *reaction* (synchronous interaction among adjacent agents), field *emission* (asynchronous interaction among at–a–distance agents through the field diffusion–perception–action mechanism), *trigger* (change of agent state as a consequence of a perceived event) and *transport* (agent movement across the space). Every primitive will be

now briefly described specifying preconditions and effects. It must be noted that an action selection strategy is invoked when the preconditions of more than one action are verified; several possible strategies can be defined, but in this context a non–deterministic choice among possible action was adopted.

The behavior of Situated Cellular Agents is influenced by agents situated on adjacent positions and, according to their type and state agents are able to synchronously change their states. Synchronous interaction (i.e. reaction) is a two–steps process. Reaction among a set of agents takes place through the execution of a protocol introduced in order to synchronize the set of autonomous agents. When an agent wants to react with the set of its adjacent agents since their types satisfy some required condition, it starts an *agreement* process whose output is the subset of its adjacent agents that have agreed to react. An agent agreement occurs when the agent is not involved in other actions or reactions and when its state is such that this specific reaction could take place. The agreement process is followed by the synchronous reaction of the set of agents that have agreed to it. Reaction of an agent a situated in site $p \in P$ can be specified as:

$action : reaction(s, a_{p_1}, a_{p_2}, \ldots, a_{p_n}, s')$
$condition : state(s), position(p), agreed(a_{p_1}, a_{p_2}, \ldots, a_{p_n})$
$effect : state(s')$

where $state(s)$ and $agreed(a_{p_1}, a_{p_2}, \ldots, a_{p_n})$ are verified when the state of agent a is s and agents situated in sites $\{p_1, p_2, \ldots, p_n\} \subseteq P_p$ have previously agreed to undertake a synchronous reaction. The effect of a reaction is the synchronous change in state of the involved agents; in particular, agent a changes its state into s'.

Other possible actions are related to the asynchronous interaction model, related to field emission and to the perception–deliberation–action mechanism. Agent emission can be define as follows:

$action : emit(s, f, p)$
$condition : state(s), position(p)$
$effect : added(f, p)$

where $state(s)$ and $position(p)$ are verified when the agent state is s and int position is p. The effect of the emit action is a change in the active fields related to sites involved in the diffusion, according to $Diffusion_f$.

The effect of an agent perception of a certain field f_i can be defined as

$action : trigger(s, f_i, s')$
$condition : state(s), position(p), perceive(f_i)$
$effect : state(s')$

where $perceive(f_i)$ is verified when $f_i \in F_p$ and $Compare_\tau(c^i_\tau \cdot i_{f_i}, t^i_\tau) = true$ (in other words, field intensity modulated by an receptiveness coefficient exceeds the sensitivity threshold for that field). The effect of the trigger action is a change in agent's state according to the third parameter. The last possible action for an agent causes a change in its position and can be specified as follows:

$action : transport(p, f_i, q)$
$condition : position(p), empty(q), near(p, q), perceive(f_i)$
$effect : position(q), empty(p)$

where $empty(q)$ and $near(p, q)$ are verified when $q \in P_p$ and $q = \langle \perp, F_q, P_q \rangle$ (q is adjacent to p and it does not contain agents). The effect of a transport action is thus to change the position of the related agent.

4 Immune System Modelling with SCA

The IS is a natural system providing inherent adaptation, coordination and learning mechanisms. The fundamental IS elements to obtain adaptivity are membrane and receptors: there is no prior knowledge on possible threats, and it must base its working on really basic elements (i.e. electromagnetic forces between molecular structures), the "building blocks" of the possible interactions. Coordination mechanisms are based on the specialization of certain cells, that will become able to interact and activate their specific working when activated by the direct interaction with other entities with compatible membrane (e.g. B cells and compatible viruses, T killers and cells infected by compatible antigens). The learning mechanism provides the production of specific memory cells, that spread over the environment in order to be able to newly activate a previously performed reaction.

An important aspect is that this memory is not centralized but distributed, incrementing the overall robustness of the system, as the death of a few memory cells (that is essentially physiologic) doesn't hinder its ability to recognize a known virus. Moreover this feature enables a local and faster reaction to infections as the related signals must not reach a specific organ, travelling through the cardiovascular or lymph system, in order to activate the IS response.

A complete specification of this extremely complex system with concepts described in the SCA model cannot be given in this paper for space reasons, but some examples of how mechanisms described above can be formalized will be reported in the following.

Entities described in Section 2 can be divided in cells and other substances like viruses and antibodies. Basically the latter can be treated as signals, moving through the space (e.g. a lymph node) until they interfere (i.e. a specific antibody neutralizes a free virus) or are are perceived by another entity (i.e. a virus penetrates a cell, or is ingested by it). Interference between these entities can easily be viewed in terms of specific kind of field composition, in which two signals with certain values neutralize themselves. These considerations led to model these non–cell entities as fields. More precisely a field *humor* could be defined as follows:

$$F_h = \langle W_h, Diffusion_h, Compare_h, Compose_h \rangle$$

with $W_h = \{virus, antibody\} \times \mathbb{N}$.

Given a humor $w_k \in W_h$ we have that $w_k = \langle ht_k, m_k \rangle$, where the first element indicates the kind of humor and the second is the specification of the membrane

related to the entity. The composition of humors is only defined for compatible entities (i.e. incompatible ones simply ignore each other), and more precisely, given $w_1, w_2 \in W_h$

$$Compose_h(w_1, w_2) = \perp \quad if \quad ht_{w_1} \neq ht_{w_2} \wedge m_{w_1} = m_{w_2}.$$

In this case the $Compare_h$ function is defined as a type compatibility test, assuming value *true* only when humor types of the compared fields are the same.

While humors are represented as fields, cells are modelled as agents. This decision is based on the fact that cells have a state, a behaviour that is much more complex than that of a humor (e.g. may require cooperation with other cells), and thus require concepts related to SCA agents. To model a generic cell agent its related type must be defined:

$$Gen = \langle \Sigma_{Gen}, Perception_{Gen}, Action_{Gen} \rangle$$

where $\Sigma_{Gen} = \mathbb{N} \times \mathbb{N} \times W_h \times \{dead, alive\}$. The state s_{Gen} of agent $a_{gen} \in A_{Gen}$ is thus defined as $\langle cl, ll, m, v \rangle$ where the first element represent the current virus load, the second is the lethal load, the third represents possibly infecting virus, with the related membrane, and the last is an indication of the cell vitality. To include a humor definition will be useful when the specification of a generic cell will be modified to define other specific entities. For instance plasma cells are able to generate antibodies, whose specification will be part of their state. The perception function is simply defined as the mechanism of infection, the penetration of a virus inside the cell that triggers its reproduction inside it until the current viral load reaches the lethal level, when the cell will die and diffuse all the viruses it contains. The perception function can thus be defined as $Perception_{Gen} = (1, \langle virus, k \rangle) \; \forall \; s_{Gen} \in \Sigma_{Gen}$. A stochastic element could possibly be introduced in order to introduce a form of non–determinism.

The specification of $Action_{Gen}$ must reflect what was said above with reference to cell's behaviour, but when infected the cell must also be able to react with specific T cells whose receptor is compatible with the infecting virus. This interaction can only take place between adjacent generic and killer T cells that agreed to react.

$Action_{Gen}$ can thus be defined as follows:

- $action : trigger(s_{Gen}, f, s'_{Gen})$
  $condition : f = \langle virus, k \rangle, s_{Gen} = \langle cl, ll, v, alive \rangle, cl < ll - 1, position(p)$
  $effect : s'_{Gen} = \langle cl + 1, ll, f, alive \rangle, F_p = F_p - \{f\}$

- $action : trigger(s_{Gen}, f, s'_{Gen})$
  $condition : f = \langle virus, k \rangle, s_{Gen} = \langle cl, ll, v, alive \rangle, cl \geq ll - 1, position(p)$
  $effect : s'_{Gen} = \langle cl + 1, ll, f, dead \rangle, F_p = F_p - \{f\}$

- $action : emit(s_{Gen}, f, p)$
  $condition : s_{Gen} = \langle cl, ll, f, dead \rangle, f = \langle virus, k \rangle, cl > 0, position(p)$
  $effect : s_{Gen} = \langle cl - 1, ll, f, dead \rangle, added(f, p)$

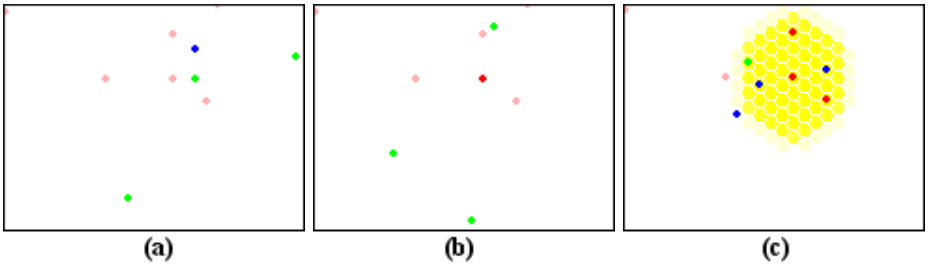


Fig. 2. In (a) a free virus is about to approach a cell; in (b) it has penetrated the cell; the virus then proliferated beyond the cell lethal load and caused its burst (c)

– $action : react(s_{Gen}, a_{TK_k}, s'_{Gen})$
$condition : s_{Gen} = \langle cl, ll, f, alive \rangle, f = \langle virus, k \rangle$
$effect : s'_{Gen} = \langle 0, ll, f, dead \rangle$

This is clearly a loose specification of humors and generic cells, as their mobility means were not indicated. The latter, and the definition a field diffusion function as well, should be designed with reference to the specific environment in which the entities are situated. If it is a blood vessel, the effect of pressure should be suitably modelled, for instance through the definition of another field influencing humor diffusion and cells mobility.

A prototype for a multi–agent system based on this specification was designed and developed using Repast [12], a Java based software framework supporting the development and execution of agent based simulations.

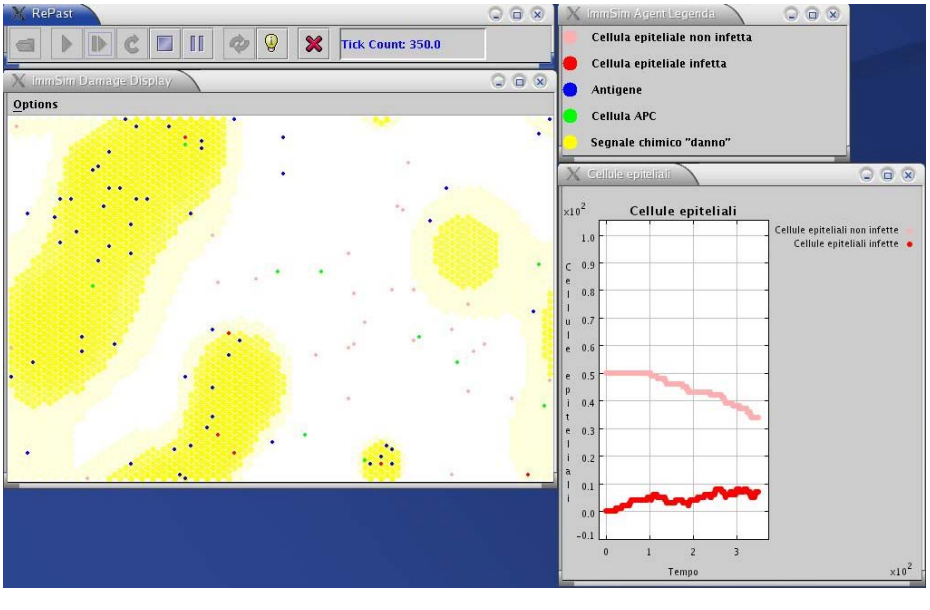


Fig. 3. A screenshots taken from the prototype of the Multi Agent IS simulation system

The membrane and receptors, previously simply defined as natural numbers, must be represented as limited data types. For this prototype we chose to adopt an 8–bit representation for receptors and membranes. Compared to the number of possible different receptors present in a fully developed and functional IS this may seem small, nonetheless this value is typically used by other CA–based simulators.

A subset of cell interactions that characterizes the IS was implemented, and the number of active entities that can be simulated is quite small compared to CA based systems, therefore the prototype should be considered as a demonstrator of the potential of this approach, especially in the educational area. One of the currently implemented interactions is cell infection performed by free viruses, cell burst and damage casting. Moreover specific APC cells (macrophages), wandering throughout the environment trying to phagocytize free viruses were also implemented. Figure 2 shows three phases of a simulation: in (a) a free virus is about to approach a cell, and in (b) it has penetrated the cell; then the virus proliferated beyond the cell lethal load and caused its burst (c). This causes the casting of a damage signal, diffusing from the burst site. Figure 3 shows an overall screenshot of the simulator, including the control panel and a graph showing the number of healthy and infected cells during the simulation.

5 Conclusions and Future Developments

IS internal mechanisms are naturally distributed, providing adaptation to unknown threats, cooperation among its composing parts, learning capabilities obtained through a decentralized memory approach. The adaptation is based on the usage of real basic elements of interaction between entities as "building blocks" for possible countermeasures to infections. Once a response was undertaken suitable memory cells are generated, in order to be able to recall it and carry it out quickly when newly necessary. Reaction is based on the cooperation of specialized entities, with precise tasks and roles in the IS working.

In this paper a Multi Agent approach to IS simulation through the adoption of the SCA model was presented. A brief description of IS composing parts and internal mechanisms was given, and the application of concepts define in the SCA model to represent them was illustrated. This work is a result of a project done in collaboration with immunologists whose goal is the development of models and instruments for immunology, and the identification of biological models and mechanisms that can be exploited in computer science. A prototype based on the SCA model for IS simulation was developed and will be evaluated by immunologists in order to define future developments and applications.

The design of robust and secure distributed systems may draw inspiration from many biological mechanisms and metaphors, but this kind of operation should be carried out with caution. In fact a partial transport of concepts and mechanisms from an area to another one could bring disappointing results.

For instance, the IS has an intrinsic adaptation mechanism, that is built on extremely basic elements, with no assumption on possible threats. To assure security in an artificial system, like a computer network, one can exploit previous

experience and knowledge on previous attacks, but new ones might use basic interaction mechanisms (i.e. the synthetic counterparts of cellular membranes) in a novel way. Defining precisely this basic level of interaction is not an easy task, as different technologies, protocols and parts of operating systems are involved, and should be considered in their context. The granularity level in monitoring and control of various operation defines the possibility to perform an effective adaptation, but even the overhead related to the various checks.

Another important feature of the IS is its distributedness: no assumption can be made on the location where the next intrusion might happen, so the monitoring must involve all the system. The extreme diffusion of personal and portable computational devices, that are more and more connected to corporate networks despite their mobility and position, makes extremely hard to define clear boundaries and interfaces. To have just a few nodes able to detect and face an intrusion will probably become an ineffective policy.

References

1. Bandini, S.: Hyper-cellular automata for the simulation of complex biological systems: a model for the immune system, in: D. Kirschner (Ed.), Special Issue on Advance in Mathematical Modeling of Biological Processes, Vol. 3, 1996.
2. Bandini, S., Manzoni, S., Simone, C.: Dealing with Space in Multi-Agent Systems: a model for Situated MAS, Proceedings of the First International Joint Conference on Autonomous Agents and Multi–Agent Sytems (AAMAS 2002), pp. 1183–1190, ACM press, 2002.
3. Bandini, S., Mauri, G., Vizzari, G.: Supporting Action–at–a–distance in Situated Cellular Agents, *Fundamenta Informaticae*, IOS press 2005 (in press).
4. Celada, F. and Seiden, P.: A computer model of cellular interactions in the immune system, *Immunology Today*, vol. 13, no. 2, pp. 56–62, Elsevier, 1992.
5. Ferber, J.: Multi-Agents Systems, AddisonWesley, 1999.
6. Kephart, J. O., Sorkin, G. B., Swimmer, M., White, S. R.: Blueprint for a Computer Immune System, in: D. Dasgupta (Ed.), Artificial Immune Systems and Their Applications, Springer–Verlag, pp. 221–241, 1999.
7. Kleinstein, S. H. and Seiden, P. E.: Simulating the Immune System, IEEE Computing in Science and Engineering, vol. 2, no. 4, pp 69–77, 2000.
8. Krishna Kumar, K. and Neidhoefer, J.: Immunized Adaptive Critics for an Autonomous Aircraft Control Application, in: D. Dasgupta (Ed.), Artificial Immune Systems and Their Applications, Springer–Verlag, pp. 242–261, 1999.
9. Moss, S., Davidsson, P. (Eds.): Multi Agent Based Simulation, Second International Workshop (MABS 2000), Boston, MA, USA, July. Revised and Additional Papers, Lecture Notes in Computer Science, Vol. 1979, Springer–Verlag, 2001.
10. Puzone, R., Kohler, B., Seiden, P., Celada, F.: IMMSIM, a flexible model for in machina experiments on immune system responses, Future Generation Computer Systems, vol. 18, no. 7, pp. 961–972, Elsevier, 2002.
11. Přiikrylová, D., Jílek, M., Waniewski, J.: Mathematical Modeling of the Immune Response, CRC Press, 1992.
12. Repast website: http://repast.sourceforge.net, 2003.
13. Roitt, I.: Essential Immunology, Blackwell, 1994.
14. Wolfram, S.: Theory and Applications of Cellular Automata, World Press, 1986.

Current Paradigms in Immunology

Eugenio Cesana, Silvia Beltrami, Antonia Emanuela Laface, Astrid Urthaler, Alessandra Folci, and Alberto Clivio

Department of Preclinical Sciences, Faculty of Medicine, University of Milano
Alberto.clivio@unimi.it

"Do antigens bear instructions for antibody specificity or do they select cell lines that arise by mutation?"

Joshua Lederberg, Science 129; 1649-1653 (1959)

Abstract. The last decade has seen a revolution in the field of Immunology. Starting from simple views on the ability of the immune system to respond to foreign antigens or to perform self/not-self discrimination, the image has become much more complex, with the realisation that autoreactive lymphocytes normally circulate in the body, without causing harm to the organism. In fact, the critical point in the development of an immune response is the activation of lymphocytes. This depends on the functional state of antigen-presenting cells and on structural features of the so-called "immune synapse". Self/not-self discrimination is therefore not as strict as previously thought: on the contrary, it has been shown that a certain degree of self-reactivity is useful, if not necessary, to the homeostasis of the organism. Furthermore, the immune system can be viewed as a network of elements which try to connect with each other to avoid death, and are endowed with emerging properties. In this review, we will make a quick summary of the "classical" paradigms in Immunology, and will discuss the dogmas (specificity, self/not-self discrimination, tolerance) as well as the new ideas to explain how the immune system works, all of them emerging from experimental observations made in the last decade of immunological research. All this may have interesting consequences both for immunologists wanting to make mathematical models of the Immune System and for those involved in the use of immune algorithms for the development of "Artificial Immune Systems" and computational applications.

1 The Classical Views

Antibody Specificity

Joshua Lederberg's two lines cited at the beginning of this article represent the clearest statement describing the dilemma researchers were faced with during the last century's fifties. Accepting selective theories and rejecting the instructionists' paradigm was at that time a question of real bravery. Evidence for the sequence-structure-function relationship in protein molecules was in fact still missing, and the belief that a somewhat strange hypermutation mechanism restricted only to the

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 244 – 260, 2006.

immune system could really exist was considered heretical and highly unlikely. For these reasons, the instructive paradigm was considered to be the most appealing and likely mechanism to explain the generation of an apparently unlimited repertoire of antibodies against a virtually infinite antigenic universe.

Approaches focused on the specificity of the immune system recognition molecules led to the idea that the interaction of antigens with antibodies was based essentially on structural complementarity. According to the well accepted instructionist paradigm, the role of antigen was to induce conformational changes in antibody molecules, therefore ensuring a more or less stable recognition. However, this idea was not supported by experimental evidence. As soon as the genetic code and the role of genes in determining the protein primary structure was uncovered, it became clear that the three-dimensional conformation of protein molecules was strictly dependent on the primary aminoacid sequence, and this was not compatible with the wide molecular flexibility required to accommodate the instructive hypotheses. However, the acceptance of the selective theory introduced another difficulty: in fact, the "one gene-one polypeptide chain" rule was clearly in conflict with the immune repertoire: there is not room enough in the genome for so many different antibody molecules.

Generation of Diversity

The breakthrough came in the mid seventies with the papers published by Susumu Tonegawa, in which an unexpected mechanism for somatic rearrangement was found to occur at specific chromosomal regions containing the genes coding for immunoglobulins (Tonegawa et al., 1974). This idea of genes not being so stable as previously supposed changed the views and paved the way to the unconditional acceptance of the idea of a somatic generation of antibody diversity, starting from a germline-encoded repertoire of smaller genetic regions. This, together with the discovery that eukaryotic genes are not continuous but split into coding and non-coding parts, allowed to explain how a virtually unlimited repertoire of antibodies can be produced starting from a limited amount of genes (see Figure 1). Therefore, the immune system was brought again into the "Darwinian" domain, with diversity being generated by a more or less random mechanism, and cells endowed with receptors able to bind the antigen being rewarded with the ability to proliferate and secrete substantial amounts of specific antibodies.

The Clonal Selection Theory and the Immunological Memory

Thus was generation of diversity explained: what about memory? It was well accepted that the immune system could "remember" previous encounters with the antigen, and that, in these cases, a stronger protective response was produced. Clearly, the Darwinian view was almost self-explaining: a selective mechanism was acting, by which the population of antibody-producing cells was continuously changing under the effect of antigenic challenge. Clones expressing a receptor able to interact with antigen are stimulated to proliferate and produce a progeny which is enriched in cells recognising the antigen. The clonal expansion lasts as long as the antigen persists, but a certain number of memory cells still remain available for future responses to the same antigen.

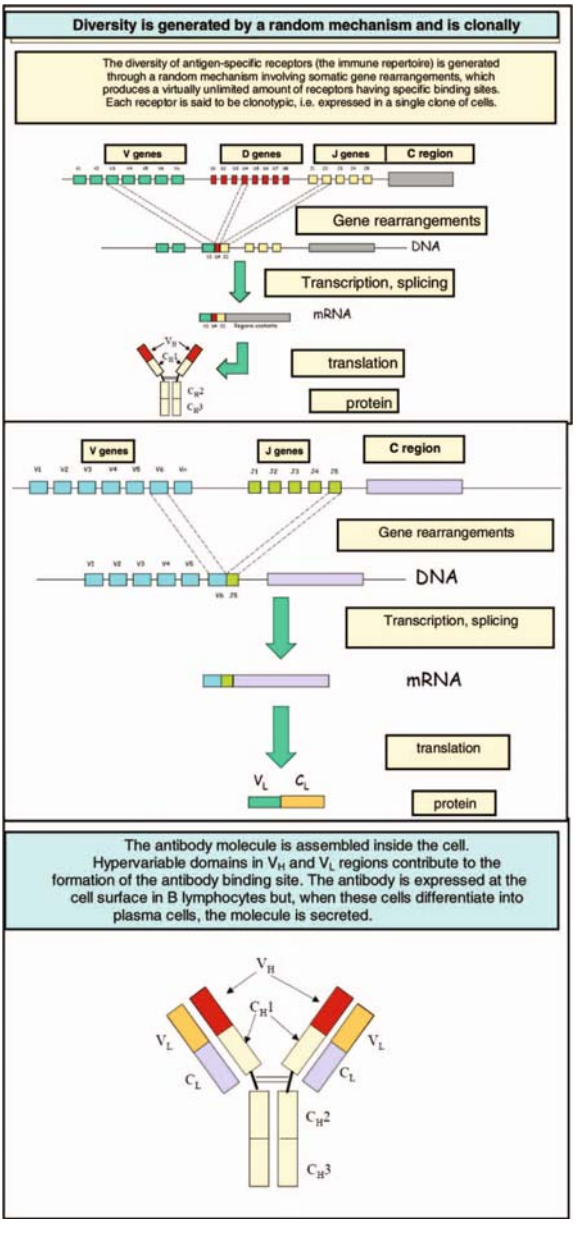


Fig. 1. The mechanism of generation of diversity of clonotypic receptors

The Clonal Deletion Paradigm for Self-not Self Discrimination and Recessive Tolerance

The Darwinian paradigm in immunology was based on the fact that the ability to "see" the antigen is strongly rewarded. However, if we consider the B cell population,

in which the generation of clonotypic receptors is a random process, it is clear that B cell clones able to bind "self" structures are continuously generated. Therefore, the first encounter of a particular B cell will most likely be one with a "self" structure instead of a "not-self" component. So, the idea of self-not self discrimination introduced by Ehrlich in 1900 ("horror autotoxicus") and further accepted by all the immunological community as a dogma had to be in some way rediscussed. In fact, why don't we reward the autoreactive B-cell clones? It became clear very soon that the model of a direct and simple link between receptor engagement and clonal expansion was indeed too "naïf". In fact, the existence of a population of functionally different lymphocytes which are not antibody producers *per se*, but which may control antibody production by B lymphocytes had already been described (Bretscher and Cohn, 1970). These cells were named "Helper T cells" and the exact mechanism of T-B cell cooperation was clarified years later (Lanzavecchia, 1985).

T cells are also endowed with a clonotypic receptor in which diversity is generated by rearrangement mechanisms which are similar to those generating B-cell diversity (Haskins et al., 1984). Therefore, the clonotypic repertoire of T cells is also virtually infinite. How do T cells avoid self-reactivity?

T lymphocytes, as opposed to B cells, which emerge from the bone marrow as functionally active cells, must undergo a maturation process in the thymus before being able to function as effector T cells. In order to explain tolerance towards self components, it was therefore postulated that T cells learn to discriminate self from not-self within the thymus (Kappler et al., 1987). Thymic cells should express a sufficiently wide repertoire of "self" antigens, so as to impair the progression of autoreactive T cells through the thymic maturation process, thus leading to the deletion of clones characterized by a strong autoreactivity. According to this view, tolerance towards self structures is a ***recessive*** phenomenon due to the "absence of help".

The Molecular Nature of Thymic Deletion

Clonal deletion at the level of the thymus requires that all the possible "self" structures, including peripheral tissue-specific antigens or antigens which only appear later in life (sex-related antigens, for example) are in some way expressed within the thymus, if T cells specific for these antigens have to be deleted from the T cell repertoire. It is necessary to hypothesize that genes coding for these peripheral or temporally-related antigens are expressed, even at low levels, in thymic epithelial cells so that clonal deletion may be accomplished. In a recent series of papers, a role in thymus deletion has been proposed for a transcriptional activator protein (AIRE) which is expressed in the thymus and is responsible for the local expression of most of the genetic repertoire of the individual (Park et al., 2003; Mathis and Benoist, 2004). This would explain how T cells bearing autoreactive T cell receptors can be deleted during thymic maturation and supports the clonal deletion paradigm for explaining tolerance towards self antigens.

Antigen-Binding Molecules and Antigen Recognition

How is the antigen "seen" by B and T cells? (see Figure 2). The antibody binding site is a three-dimensional cleft with a relatively small volume, enough for the equivalent

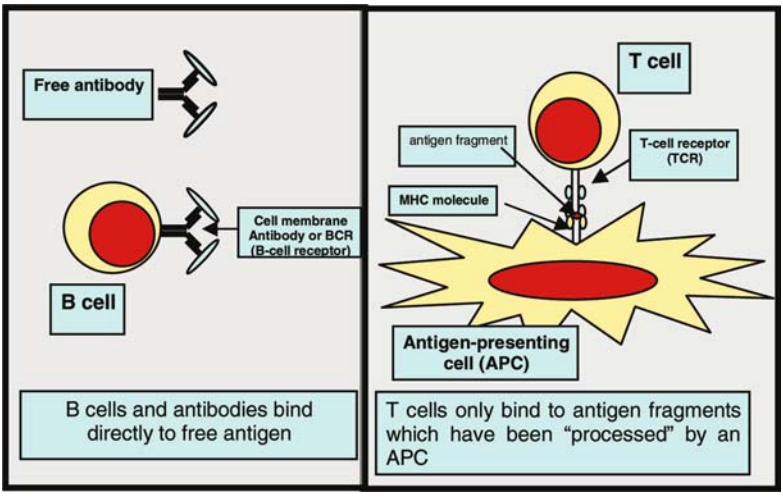


Fig. 2. Interaction of B and T cells with antigen

of five-six aminoacids; a protein may be composed of several hundreds of aminoacids, therefore any protein may bear dozens of different regions which may fit into the antigen-binding site. These regions are called *epitopes*. Therefore, the B-cell receptors (membrane-bound antibodies) and the free antibody molecules are able to bind directly to specific epitopes of a given antigen. The basis of T-cell recognition is quite different and was clarified in the seventies (Zinkernagel and Doherty, 1974). A T cell does not interact directly with a free antigen, but recognizes cells which express fragments of the antigen at their surface. In the case of helper T cells, the antigen is intracellularly processed and "presented" to them by cells endowed with endocytic capacity (now called dendritic cells). "Presentation" means that proteolytic fragments of the antigen are carried to the surface of the Antigen-Presenting Cell (APC) in the pocket of germline-encoded molecules called the Major Histocompatibility Complex (MHC) Class II molecules. It became soon clear that T cells exist which are able to directly kill virus-infected host cells, and that this recognition depends on the expression of germline-encoded MHC molecules called Class I molecules, typically present at the surface of every nucleated somatic cell. A virus-infected somatic cell can be recognized by a particular T cell clone, which undergoes helper T cell-dependent clonal expansion and ultimately kills the target cell. The interaction of the antigen-derived peptides and MHC molecules has been clarified: it appears to be a low-affinity interaction involving only a few aminoacid residues which interact with aminoacids in fixed positions, being part of the "floor" of the MHC cleft. Since the diversity of MHC molecules is not somatically generated but inherited as a Mendelian trait and is characterized by a wide genetic polymorphism, it is clear that any peptide, either self or not self, can fit the MHC groove, provided that it has the right aminoacids in the required positions. In addition, this means that a subject with a specific set of MHC alleles will present a particular subset of peptides from a defined antigen. Conversely, different subjects, having a different set of MHC alleles, will present different peptides of the same antigen to T cells: the immune response is

therefore "personalized". In 1985 it was shown that human B cells are able to present antigen to T cells, due to the fact that they specifically take up antigen by antibody-mediated endocytosis (Lanzavecchia, 1985). This is the basis of T-B cell cooperation. Consequently, antigen fragments are brought to the surface of APCs by MHC molecules and presented to the T cells. T cells expressing TCRs specific for the MHC-antigen complex can interact with APCs and be activated to proliferate and express effector functions. Therefore, as mentioned earlier, the lack of an autoimmune response is interpreted, according to this paradigm, as being due to the absence of T cells reactive against self peptides.

The Idiotypic Network

The ability of the immune system to recognize the antigenic universe is dependent on the clone dimensions. Each cell expressing a single Ig or TCR must undergo a partial clonal expansion in order to ensure that enough antigen-reactive cells are present within each clone. What drives this expansion, if the antigen is not yet present? An interesting mechanism was proposed in 1974 (Jerne, 1974): Since the clonotypic receptors are generated by a random mechanism, we should expect that all the possible binding sites are produced, and that an immune network of reciprocal interactions between clones should emerge as a consequence: the immune system, according to Jerne, is an "internal image" of the external antigenic universe (Figure 3). The network ensures that each clone is partially expanded, and this explains the presence of a sufficient number of cells per clone, even in the absence of the real antigen.

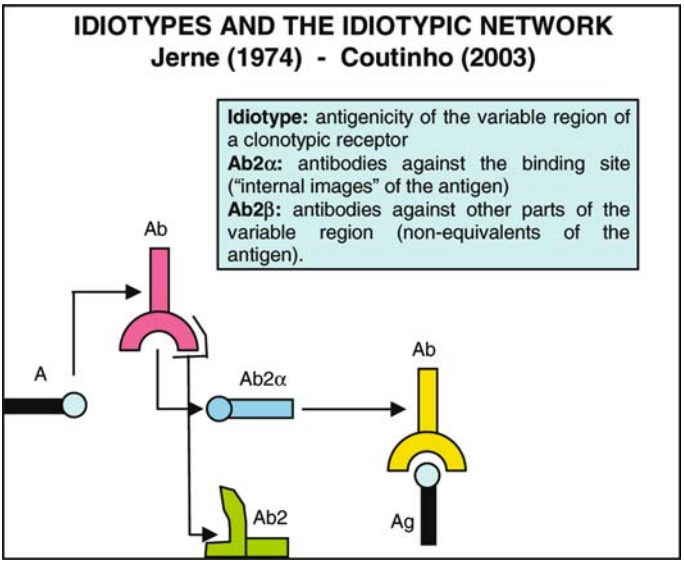


Fig. 3. Idiotypes and the concept of idiotypic network

The original idea of Jerne was further developed (Varela and Coutinho, 1991) who proposed the so-called second and third-generation networks, thus extending the concept to the B-cell level and even to the T-cell population. They considered the immune system as a tightly interconnected network of interactions in which a competition is established among immune system cells for achieving reciprocal contact: the cells which establish connections are able to survive and participate to the establishment of the network, those which fail die of "loneliness". Since new lymphocytes are constantly produced by the primary lymphoid organs, these new elements are also screened for their ability to enter the network.

This interesting view was abandoned for some time during the 90's, but is now regaining popularity, as will be discussed later (Shoenfeld, 2004). In fact, the concept of the idiotypic network is now being extended to a more generalized view which considers the immune system as an interconnected, multi-layered network of cells which includes not only elements of the immune system itself, but also cells of the nervous and endocrine system as well.

2 Changing Views

This brief survey of the historical facts related to the interpretation of how the immune system works clearly shows that there are still many points to be clarified as far as the "logic" of the immune response is concerned. We will now discuss how the views are changing in the light of new findings coming from experimental as well as clinical immunology.

The Link Between Innate and Adaptive Immunity

We must consider that invertebrates are said to perform quite nicely without a truly adaptive immune system, which is a peculiarity of vertebrates. The terms "innate" and "adaptive" refer to the fact that the innate response is immediate, being based on mechanisms which do not change over time, whereas the adaptive response develops with a certain delay, but becomes relatively specific for the current antigen. In vertebrates, both the innate and the adaptive immune systems are active and cooperate intimately, as shown by several reports in the last ten years of immunological research (reviewed in Germain, 2004). The main components of the innate immune response. are complement, phagocytes, NK cells and dendritic cells, and the effector molecules they use are germline-encoded, components, as well as non-clonotypic, pattern-recognition receptors (PRRs) (Akira and Sato, 2003), which recognise pathogen-associated molecular patterns (PAMPS) on foreign cells. Conversely, the actors of the adaptive immune system are the B and T lymphocytes endowed with the clonotypic receptors described earlier.

In vertebrates, the front line of defense that foreign invaders encounter as soon as they enter our body is mostly components of the innate immune system, such as complement, phagocytes and dendritic cells. The interaction of complement components with external invaders usually results in the physical elimination of foreign cells by lysis and in the effective recruitment of phagocytes to the site of pathogen entry, due to the diffusion of soluble compounds derived from partial

bacterial breakdown and complement activation, all of which have chemoattractant properties towards phagocytes. This ensures rapid and effective elimination of potentially harmful pathogens. The innate response is however not endowed with memory properties. Dendritic cells are currently considered as the elements linking the innate and the adaptive immunity (Mellman, 2005). They may be found in different functional states, namely immature, actively endocytic cells with a low antigen-presenting capacity but substantial mobility (they actually patrol the body taking up damaged self and not-self material), and the mature DCs, which stop capturing but acquire high antigen-presenting properties. The details of the functional link between the innate and the adaptive immune system are discussed in detail in the following chapters.

Dendritic Cells as Immune Sensors

The B cell and T cell activation processes have been shown to require complex cell-to-cell interactions: B cells and T killer cells are in fact activated by T helper cells. The latter, however, need to interact with a mature dendritic cell in order to be able to exert its helper function (Delon and Germain, 2000). Thus, activation of the adaptive immune system ultimately depends on a cell which is unable to structurally discriminate between self and not-self! Moreover, the ability of a DC to activate the adaptive immune response strictly depends on its state of maturation. What determines the transition of a DC from the immature to the mature state? How long does this temporal window of activation last? Which signals are responsible for DC maturation and activation?

As shown in Figure 4, DCs in different maturation stages express different surface markers and exclusive functions. In particular, the transition between the immature and the mature stage is characterised by the increase in co-stimulatory molecules

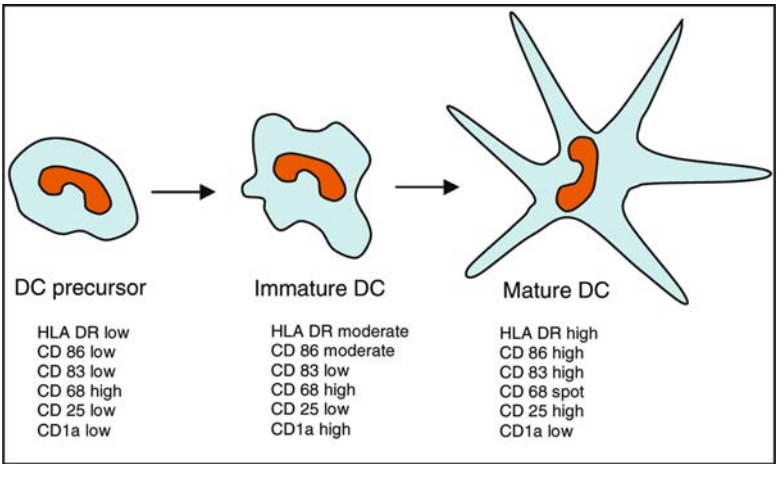


Fig. 4. Features of dendritic cells in different maturation stages

which are critical for T-cell activation. If a T cell, irrespective of its specificity (anti-self or anti-not self) interacts with an immature DC, it is anergised. On the contrary, when such a T cell interacts with a mature DC, it is brought into the activation process. It is important to notice that DCs continuously patrol the body tissues and actively take up self as well as not-self material by an endocytic process. Subsequently, they migrate to the lymphnodes, where they can interact with resident T cells and activate them.

The Danger Theory

In 1994, Polly Matzinger's proposal of a new paradigm introduced a breakthrough in the interpretation concerning the mechanisms of activation of the immune system (Matzinger, 1994). She claimed that the immune system is not really interested in discriminating self from not-self, but only "dangerous events" from non-dangerous ones. No surprise, this proposal was viewed as a heretical statement by most of the immunological community. However, several observations supported this view (Paterson, 2003; Skoberne 2004), most of them being quite familiar to every-day immunologists: for example, it is easy to generate B and T cell responses against "self" antigens, provided that a good adjuvant is used which generates a "dangerous" signal, such as a local inflammatory reaction. In addition, it was perfectly known (this was also a really historical finding by Morgenroth in 1905) that one can easily find circulating antibodies as well as B and T lymphocytes which are reactive against self antigens *in vitro*, but which do not set up an immune response *in vivo*. On the other hand, we are faced continuously with non-self antigens present in the intestinal flora, and we perfectly tolerate these "non-self" entities. Pregnancy is in this case viewed as a particular kind of graft, in which no rejection response is mounted. Thus, the idea of tissue damage being an important factor for the activation of the immune response was brought into the field. This interesting and radically innovative point of view introduced the concept of the immune system as a component which is not isolated but is actively involved in tissue maintenance and regeneration. Some of the previsions of the danger theory were entirely speculative at that time, but experimental observations were soon made in which "danger signals" produced by pathogens as well as by damaged tissues were characterised (Skoberne, 2004). The "Danger paradigm" states that, irrespective of the "self or not-self" features of the antigen, the activation of the immune response ultimately depends on cells which are sensors of danger, i.e. the dendritic cells. In fact, the role of danger signals in the process of DC maturation/activation has been partially clarified in the recent years.

Toll-Like Receptors as Danger Sensors

In parallel with Matzinger's, other "heretical" views of the immune response were proposed during the nineties as well: in particular, Charles Janeway and Ruslan Medzhitov proposed that the immune system discriminates "infectious non-self" from "non-infectious self" (Medzhitov and Janeway, 2000). The recognition involves conserved structural motifs in pathogen components (Pathogen-Associated Molecular Patterns, PAMPs), which interact with non-clonotypic, germline-encoded receptors

(Pattern-Recognition Receptors, PRRs), which were however not known at the time of the proposed model. However, the discoveries of Toll-like receptors, which are indeed able to recognise conserved pathogenic structures and are expressed at the surface of dendritic cells, gave an experimental support to the Medzhitov-Janeway paradigm. Toll-like receptors (TLRs) were described as fundamental molecules for the immune response, and they belong to a family which currently comprises at least ten different members, of which at least 6 are directly involved in the immune response (Figure 5). They are expressed as homo- or heterodimers at the cell surface, and their different combinations constitute a "Toll-like receptor code" in terms of activating or inhibiting properties (Akira and Sato, 2003). In addition to their capacity to recognise pathogen-derived structures, these receptors were found to interact with intracellular components released from damaged cells, thus giving a long-awaited experimental support to Matzinger's Danger theory. Triggering of TLRs is coupled to intracellular signal transduction pathways controlling pro-inflammatory as well as lymphocyte-activating cytokine production, thus providing a functional link between the innate and the adaptive immune system.

Therefore, the ability of DCs to activate the specific immune response (Steinman et al., 2005) depends on their state of maturation, which in turn depends on the exposure to either pathogen-derived structures, or self components released from cells dying of a "violent death". This view allows for the interpretation of auto-immune reactivity as a phenomenon which might be transient and important in tissue renewal and maintenance, but which may become long-lasting when the inflammatory state persists for a longer period and is sustained by chronic infection.

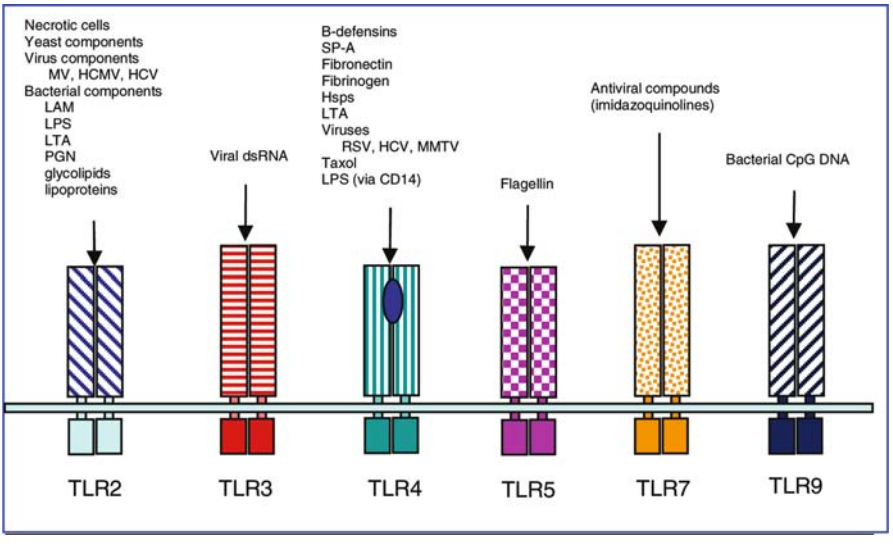


Fig. 5. Schematic representation of TLRs involved in the immune response and their main ligands. Only homodimers are shown here, but evidence is emerging for heterodimers endowed with enhancing/inhibiting activities.

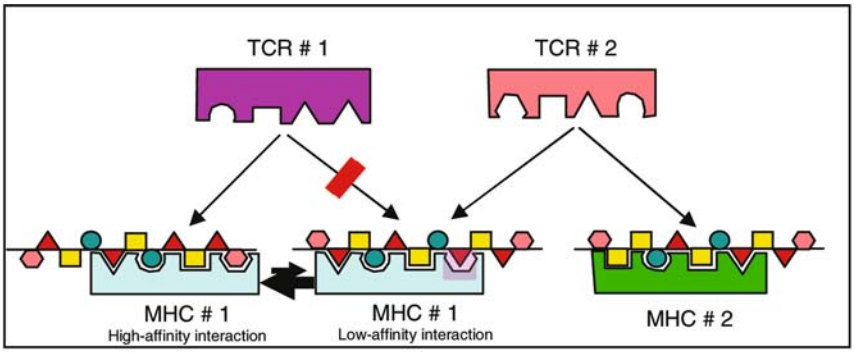
Degeneracy of the Immune Response (One Clone-Many Specificities, One Epitope-Many Clones)

One of the striking features of the immune response which is now being discussed is ***specificity***. Several authors have shown (reviewed in Edelman and Gally, 2001) that even antibodies produced by a single clone of B cells are able to recognize a wide array of different antigens and, in addition, that a single clone of T cells also shows reactivity against a variety of targets (Cohen et al., 2004). This might be due to different causes: first, the short aminoacid stretch recognised by a particular antibody or TCR might be present in different antigens ***(true cross-reactivity)***. But in most cases, it has been shown that there is no structural relationship between the antigens recognized by a single clonotypic receptor. This can only be explained by taking into account ***degeneracy***. Every clonotypic receptor, whether it is a BCR or a TCR, apparently does not recognize a specific aminoacid stretch, but instead it fits to a molecular shape which might well be reproduced by different sequences. Moreover, the recognition of different structures might be characterised by an array of different affinities. In the case of BCRs and antibodies, this was already well known: in fact several "conformation-dependent" epitopes have been described which are no longer recognised upon denaturation with anionic detergents or reducing agents. On the contrary, TCR degeneracy has been more difficult to explain, since TCRs recognize peptides fitting in the groove of MHC molecules.

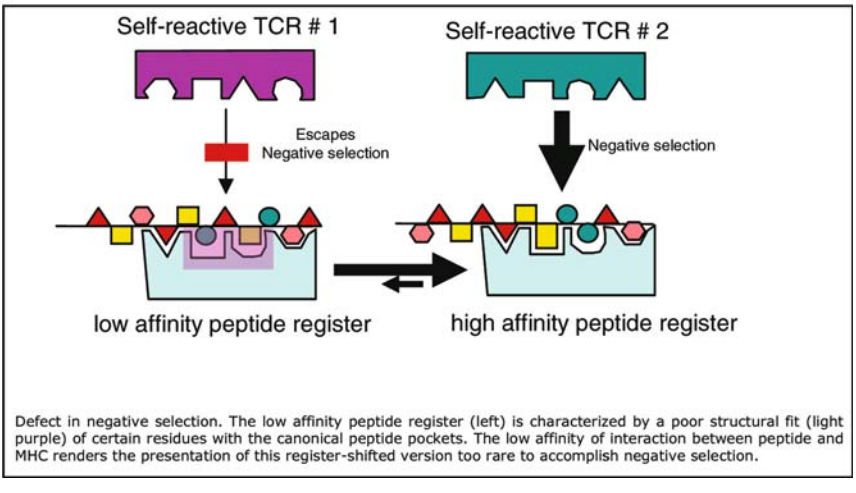
Degeneracy and Peptide Register Shifting on the MHC

An interesting explanation of the ability of a particular TCR to recognise different peptides (or, conversely, the ability of a particular peptide to stimulate different clones of T cells) has been recently proposed (Bankovich et al, 2004): it involves a shift in the frame of peptide fitting within the MHC protein groove, especially when the peptide is presented on MHC Class II molecules. For example, a 12-aminoacid peptide is usually bound to the MHC by establishing a weak interaction between particular aminoacids of the peptide and the "floor" and "walls" of the MHC binding site. Depending on how the peptide is positioned within the MHC, different aminoacids may be exposed to the TCR interface; therefore different clones of T cells might be stimulated by the same peptide (Figure 6). This leads to a degenerate response and makes it even more difficult to discriminate between "self" and "not-self" peptides or it might determine the activation of an anti-self T cell upon exposure to an external antigen, thus explaining several cases of virus-induced autoimmunity. The same mechanism might be responsible for explaining the ability of a single TCR to bind to different antigens, provided that the "TCR interface" is compatible in both cases.

Therefore, structural discrimination within the immune system no longer appears to be relevant in distinguishing self from not-self. Other phenomena must be considered, such as shape mimicry and peptide recognition with different affinities.



Frame shifting for T cell recognition. TCR #1 recognizes a self-peptide bound in the canonical register of MHC #1 (left), but not the register-shifted complex (center). The register-shifted complex is recognized by TCR #2 and resembles a non-self-peptide presented by MHC #2 (right). This example demonstrates how frame shifting can induce molecular mimicry leading to activation of an autoimmune T cell. Arrows show that there is equilibrium between the high and low affinity registers.



Defect in negative selection. The low affinity peptide register (left) is characterized by a poor structural fit (light purple) of certain residues with the canonical peptide pockets. The low affinity of interaction between peptide and MHC renders the presentation of this register-shifted version too rare to accomplish negative selection.

Fig. 6. Peptide register shifting and T cell degeneracy

3 The Complexity of Immunological Synapses

We have shown that the activation and regulation of a specific immune response is the result of communication between T cells and APCs, as a consequence of the molecular interactions at the cell-cell interface (the immunological synapse). The detailed description of the structure of this contact surface is now becoming a hot field of research in functional immunology (Delon and Germain, 2000; Huppa and Davis, 2003). For years, the topic was almost neglected, since particular attention was given only to TCR-MHC-peptide complexes as the main determinants of T cell activation. However, recent findings concerning differences in timing, spacing and

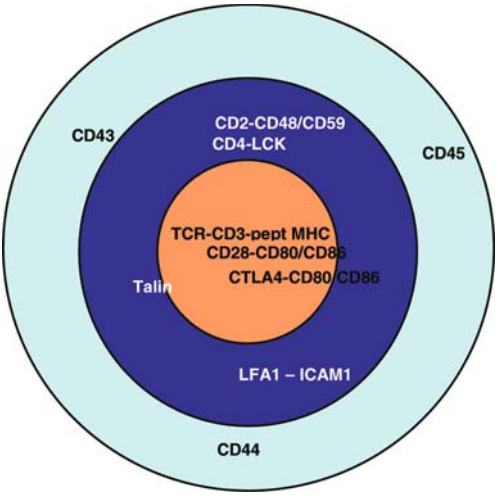


Fig. 7. Spatial arrangement of the different components of the immune synapse viewed from above. The relative position of the molecules taking part in the interaction between the T cell and the APC or the target is a prerequisite for optimal T cell activation.

molecular composition of the immunological synapse (Krummel and Davis, 2002) have contributed to realise the complexity of (if not to clarify) the features of the T helper cell-APC and T killer-target cell contacts which are relevant for the activation and regulation of the adaptive immune response. In fact, studies in the last ten years have highlighted the importance of co-stimulatory molecules, in addition to the "classic triad", in the process of T-cell activation. The finding that the immune synapse is characterised by a dynamic structure in which significant changes take place upon contact with the antigen-presenting cells or with the target cells has added a high degree of complexity to the process of cell activation within the immune response. The different molecular components taking part into the immune synapse organize themselves in a spatial dislocation which determines the outcome of the cellular interaction (see figure 7). Furthermore, other types of synapses are becoming increasingly interesting, such as the NK-target cell interface, in which the "missing self" signal is relevant, together with other activating as well as non-activating stimuli. Finally, new types of contacts still have to be discovered, such as the regulatory T cell-effector cell interaction which leads to regulatory mechanisms discussed in the following sections.

NK Cells and the "Missing Self" Hypothesis

A cellular interaction which belongs to the innate immune response against virus-infected cells and tumor cells is the NK-target cell interaction. The molecular basis of this particular interplay has been clarified during the nineties by several groups (for a recent review, see Moretta et al., 2005). Essentially, the concept is that a NK cell establishes a close contact with the target cell involving engagement of receptors which transduce negative signals (Killer Inhibiting Receptors, KIR) and others which

can potentially activate the NK cell. The outcome of the interaction, i.e. tolerance or cell lysis, depends upon a balance between the two different forms of signal (see Figure 8). MHC molecules are ubiquitously expressed by all the somatic cells, so these are tolerated by the NK population due to the negative signal delivered by KIR engagement by MHC. Virus-infected or tumor cells frequently underexpress MHC molecules and can be killed by NK cells, provided that these targets also express activating ligands.

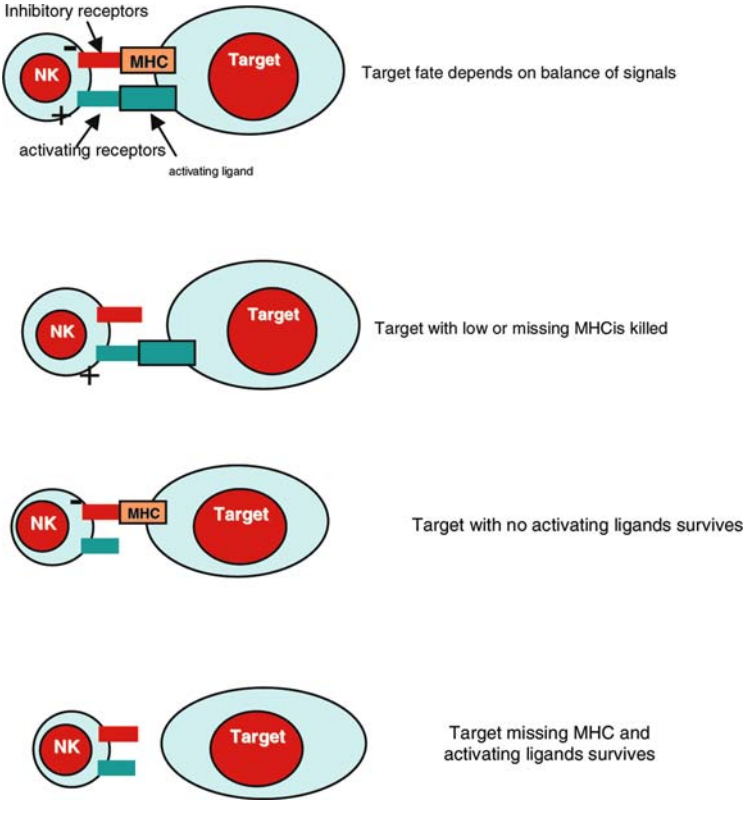


Fig. 8. The outcome of the NK-target cell interaction depends on the balance of signals delivered to the NK cell by the engagement of activating and inhibiting receptors.

Regulatory T Cells and the Concept of Dominant Tolerance

The interpretation of tolerance as a recessive phenomenon due to the absence of self-reactive T cells has been seriously questioned in the last years; The discovery (or "re-discovery") of T cell populations with specific regulatory activity against self as well as external antigens, which can transfer tolerance when injected, shows that tolerance may also be interpreted as a ***dominant*** feature (Coutinho and Haas, 2001; Coutinho et al., 2001; Nishimura et al., 2004). This experimental evidence should change the way clinicians fight autoimmune diseases and tumors: tolerance towards self antigens

might be due to the persistence of regulatory T cells: these inhibit the activation of self-reactive T cells which have escaped negative selection in the thymus. Therefore, autoreactive clones are advantageous if they are responsible for regulatory circuits. It is not yet clear, for example, which surface molecules expressed by either antigen-presenting cells or effector T cells do regulatory T cells recognise. This is currently the subject of extensive research (see von Boehmer, 2005) or which is the exact mechanism by which regulatory T cells control the immune responses (Figure 9). In any case, the clones of regulatory cells should not be deleted in the thymus: if this happens, severe autoimmune reactivity is triggered by lack of control. The existence of regulatory T cells allows for a new interpretation of the role of the thymus in shaping the immune repertoire: the decision of what to tolerate as "self" might be based on the development of a repertoire of regulatory T cells which are long-lasting and survive throughout life, thus explaining how the tolerance towards self structures lasts even after thymic involution.

Cross-Talk Between Immune, Endocrine and Nervous Systems

Current research in the fields of Immunology, neurobiology and endocrinology tend to consider these systems as highly interrelated and cross-talking systems, which share molecules such as ligands and receptors (Tracey, 2002). This is consistent withMatzinger's Danger theory, which considers the immune system as an entity which, far from being self-standing and isolated, is tightly integrated with other systems of the body. Antigen-presenting cells which control the onset of the inflammatory reaction and the immune responses are, for example, under neural control and can be rapidly de-activated through a feedback loop involving the acetylcholine receptor expressed at their surface and TNFα as well as IL-1 receptors expressed at the afferent nerves. Research in this field is very active, and deals with

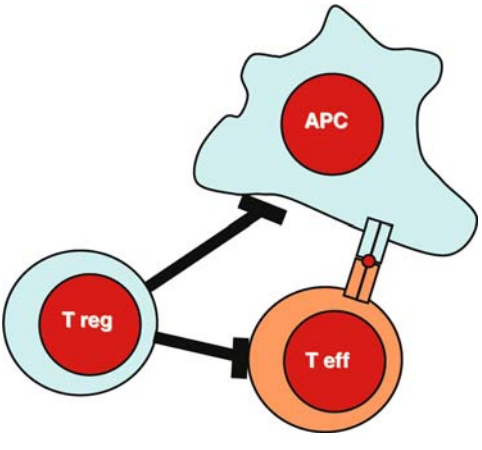


Fig. 9. Regulatory T cells might exert their effects either by inhibiting the effector T cells (helpers or killers) or by inhibiting the antigen-presenting cells

the definition of the whole repertoire of molecules and functions shared by the elements of the three systems (Szyper-Kravitz et al., 2005). In this respect, therefore, the immune system is mainly viewed in its extended function of overall maintenance of the structural/functional efficiency of the host.

References

Akira S, Sato S. 2003. Toll-like receptors and their signaling mechanisms. *Scandinavian Journal of Infectious Diseases. 35(9):555-62*

Bankovich AJ, Girvin AT, Moesta AK, Garcia KC. 2004. Peptide register shifting within the MHC groove: theory becomes reality. *Molecular Immunology. 40(14-15):1033-9*

Bretscher P, Cohn M. 1970. A theory of self-nonself discrimination *Science. 169(950):1042-9*

Burnet FM. 1976. A modification of Jerne's theory of antibody production using the concept of clonal selection *Cancer Journal for Clinicians. 26(2):119-21*

Cohen IR, Hershberg U, Solomon S. 2004. Antigen-receptor degeneracy and immunological paradigms. *Molecular Immunology. 40(14-15):993-6*

Coutinho A, Haas W. 2001. In vivo models of dominant T-cell tolerance: where do we stand today? *Trends in Immunology. 22(7):350-1*

Coutinho A, Hori S, Carvalho T, Caramalho I, Demengeot J. 2001. Regulatory T cells: the physiology of autoreactivity in dominant tolerance and "quality control" of immune responses. *Immunological Reviews. 182:89-98*

Delon J, Germain RN. 2000. Information transfer at the immunological synapse. *Current Biology. 10(24):R923-33,*-28

Edelman GM, Gally JA. 2001. Degeneracy and complexity in biological systems. *Proceedings of the National Academy of Sciences of the United States of America. 98(24):13763-8*

Germain RN. 2004. An innately interesting decade of research in immunology. *Nature Medicine. 10(12):1307-20*

Haskins K, Kappler J, Marrack P. 1984. The major histocompatibility complex-restricted antigen receptor on T cells. *Annual Review of Immunology. 2:51-66*

Huppa JB, Davis MM. 2003. T-cell-antigen recognition and the immunological synapse. *Nature Reviews. Immunology. 3(12):973-83*

Jerne NK. 1974. Towards a network theory of the immune system *Annales d'Immunologie. 125C(1-2):373-89*

Kappler JW, Roehm N, Marrack P. 1987. T cell tolerance by clonal elimination in the thymus *Cell. 49(2):273-80*

Krummel MF, Davis MM. 2002. Dynamics of the immunological synapse: finding, establishing and solidifying a connection. *Current Opinion in Immunology. 14(1):66-74*

Lanzavecchia A. 1985. Antigen-specific interaction between T and B cells *Nature 314(6011):537-9*

Mathis D, Benoist C. 2004. Back to central tolerance. *Immunity.*509-16

Matzinger P. 1994. Tolerance, danger, and the extended family. *Annual Review of Immunology. 12:991-1045*

Medzhitov R, Janeway C, Jr. 2000. The Toll receptor family and microbial recognition. *Trends in Microbiology. 8(10):452-6*

Mellman I. 2005. Antigen processing and presentation by dendritic cells: cell biological mechanisms. *Advances in Experimental Medicine & Biology. 560:63-7*

Moretta L, Bottino C, Pende D, Vitale M, Mingari MC, Moretta A. 2005. Human natural killer cells: Molecular mechanisms controlling NK cell activation and tumor cell lysis. *Immunology Letters. 100(1):7-13*

Nishimura E, Sakihama T, Setoguchi R, Tanaka K, Sakaguchi S. 2004. Induction of antigen-specific immunologic tolerance by in vivo and in vitro antigen-specific expansion of naturally arising Foxp3+CD25+CD4+ regulatory T cells *International Immunology. 16(8):1189-201*

Park Y, Moon Y, Chung HY. 2003. AIRE-1 (autoimmune regulator type 1) as a regulator of the thymic induction of negative selection *Annals of the New York Academy of Sciences. 1005:431-5*

Paterson HM, Murphy TJ, Purcell EJ, Shelley O, Kriynovich SJ, Lien E, Mannick JA, Lederer JA. 2003. Injury primes the innate immune system for enhanced Toll-like receptor reactivity *Journal of Immunology. 171(3):1473-83*

Shoenfeld Y. 2004. The idiotypic network in autoimmunity: antibodies that bind antibodies that bind antibodies. *Nature Medicine. 10(1):17-8*

Skoberne M, Beignon AS, Bhardwaj N. 2004. Danger signals: a time and space continuum. *Trends in Molecular Medicine. 10(6):251-7*

Steinman RM, Bonifaz L, Fujii S, Liu K, Bonnyay D, Yamazaki S, Pack M, Hawiger D, Iyoda T, Inaba K, Nussenzweig MC. 2005. The innate functions of dendritic cells in peripheral lymphoid tissues. *Advances in Experimental Medicine & Biology. 560:83-97*

Szyper-Kravitz M, Zandman-Goddard G, Lahita RG, Shoenfeld Y. 2005. The neuroendocrine-immune interactions in systemic lupus erythematosus: a basis for understanding disease pathogenesis and complexity. *Rheumatic Diseases Clinics of North America. 31(1):161-75*

Tonegawa S, Steinberg C, Dube S, Bernardini A. 1974. Evidence for somatic generation of antibody diversity *Proceedings of the National Academy of Sciences of the United States of America. 71(10):4027-31*

Tracey KJ. 2002. The inflammatory reflex. *Nature. 420(6917):853-9,*-26

Varela FJ, Coutinho A. 1991. Second generation immune networks. [Review] [59 refs]. *Immunology Today. 12(5):159-66,*

von Boehmer H. 2005. Mechanisms of suppression by suppressor T cells. *Nature Immunology. 6(4):338-44*

Zinkernagel RM, Doherty PC. 1974. Restriction of in vitro T cell-mediated cytotoxicity in lymphocytic choriomeningitis within a syngeneic or semiallogeneic system. *Nature. 248(450):701-2*

Supporting Collaborative Knowledge Work: A Methodology for Developing ICT Tools for Biomedical Research

Henry Linger

Knowledge Management Research Program,
Faculty of Information Technology, Monash University,
Melbourne, Australia
henry.linger@infotech.monash.edu.au

Abstract. The nature of knowledge work not only addresses the productive aspects of an activity but also the intellective work that underpins production. Support for knowledge work requires tools for both types of tasks in order to integrate these aspects and to make the cognitive tasks visible. Biomedical research is concerned with the structure and behaviour of phenomena and requires tools that support the exploration of both aspects of the phenomena. This paper presents a method for developing computer-based models that allow such investigations. The method is based on a linguistic paradigm that assumes the models express the common language that represents the phenomena and is constructed collaboratively between a biomedical researcher and a computer scientist.

1 Introduction

Biomedical research is knowledge work in that it assumes not only task performance, but also the review and evaluation of the work done in order to understand and learn from the experience. The Task-based Knowledge Management (TbKM) framework developed by Burstein and Linger [1] integrates these two aspects and presents a layered understanding of a task as an activity system through:

- the explicit models of the structure and process of a task, as conceptualised by the actor performing the task to document her understanding of the task (conceptual layer)
- models that enable the task to be efficiently performed (pragmatic layer).

The two layers of the TbKM framework can be characterised as integrating "doing" (pragmatic) and "thinking" (conceptual). It also provides tangible artefacts to support the intellective work as well as the practical actions required to produce organisationally defined outcomes. In this sense the framework supports knowledge work as defined by Iivari and Linger [2, 3] as it;

- allows the object of work to be defined (the inputs, outputs and performance)
- identifies the body of knowledge that underpins the work (the conceptual models)

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 261–271, 2006.

– allows instantiation in terms of the item of work
– supports the production of knowledge as an aspect of the outcome (learning)
– represents and inscribes the objects of work (the models) providing the artefacts with which the actor can perform her work

Studies of work practices in domains that clearly involve knowledge work (e.g. [4]) indicate that there is little support for "thinking" work. Moreover such work is generally invisible and not explicitly integrated with "doing". Development of computerised tools to support knowledge work is premised on the involvement of both a domain and information systems experts as collaborators in a process characterised by negotiation. Such collaboration is focussed on the work of the subject domain expert and is limited to the expert's interpretation of the available information about the subject. The result of the collaboration are tools that explicitly support specific activities and incorporate complex and incomplete knowledge within the narrow subject area defined by the domain expert. Most significantly, this collaboration can be characterised as a process of constructing a common language, expressed through a model that can be implemented as a computer-based artefact.

Figure 1 below shows this collaboration. It represents the collaboration as an activity that overlaps the domains and situates the participants as both experts and designers as they construct the common language. The diagram also highlights that the common language is represented as a model that accommodates feature that facilitates its implementation as a computer-based artefact.

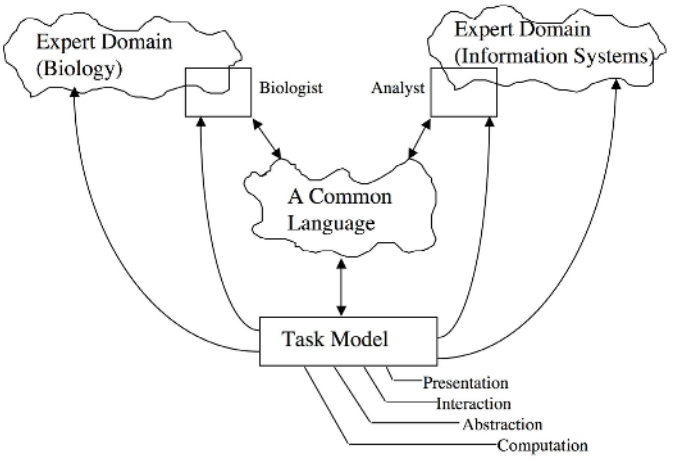


Fig. 1. A Framework for Language Construction (Source: [5])

This paper presents a method for developing computer-based tools to support knowledge work that is based on a linguistic paradigm. This method is focused on the development of simulation models to explore both the static and dynamic aspects of phenomena and is based on research that involved simulation modelling of HIV immunology [6].

2 Collaborative Development Model (CDM)

The process of constructing models of a subject domain requires domain experts to develop a language to express the phenomena that are being explored. The models are abstract representations of the conceptual understanding of the domain. However, to construct computer-based artefacts of that language requires collaboration between a subject expert and an information systems expert to derive a common language that draws on both domains. This focus on language, and particularly the negotiations over elements of the language, highlights the process of how the experts share their understanding of the phenomena being modelled and what can be achieved with the models. Collaboration informs both experts and provides insights into their respective domains. The evolving common language ensures a constructive conversation between the experts about the phenomena being modelled and the intended use of the model. Thus participation in this process, in itself, is potentially a significant benefit to the participants and insights gained through the process are explicitly incorporated into the construction of the common language.

The common language enables knowledge to be expressed in a manner that is accessible to all potential users and substantiates different aspects of the phenomena. It focuses on what scientists actual do as opposed to what scientific theory say is done or observers of science suppose is done [7, 8, 1] and allows the phenomena to be richly described. Such representations are collectively referred to as inscriptions by Latour [9]

> *"...everything from the largest galaxy to the smallest particle is processed in the laboratory so that it can be captured on paper. Unlike the scientific phenomena themselves, the paper representation or inscriptions can be read, superimposed, synthetised, integrated and transmitted. Above all they can be manipulated".*

Computing opens the possibilities for new forms of inscriptions with the innovative characteristic of being executable; they are capable of symbolic activity, and interaction. These inscriptions are the basis of *in-machina* [1] experimentation. The common language represents the experts in their role as co-designers and expresses the negotiated outcome that links domain inscriptions and the formalisms of computer science (usually a graph).

The models, based on the common language, are an explicit statement of the goals and intent that motivates the investigations being undertaken. The complexity of the knowledge required to make sense of the phenomena means that multiple models are needed to support defined activities that enable a comprehensive exploration of the phenomena. As each model is linked directly to a particular activity, they can be used independently and specific knowledge can be incorporated into each model. In this way the models are dynamic and

[1] The term *In-Machina* was originally coined by Celada and Seiden and identified in their paper in Immunology Today [10]. An analogous term, *In-Numero* was used by Mussio [11] and refers to the same type of experimentation.

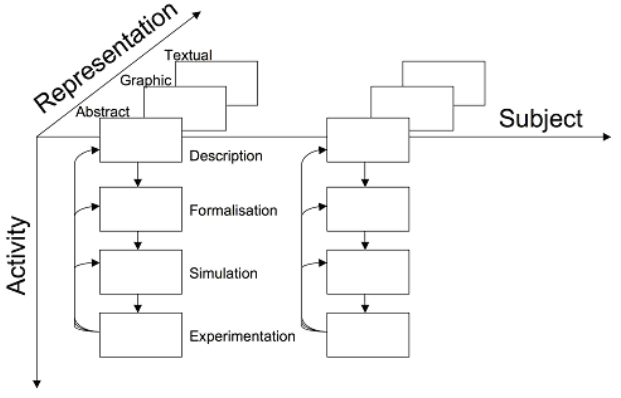


Fig. 2. The Modelling Space of CDM

evolving products that provide different perspectives of the phenomena being investigated. However, such partitioning of the subject is inherently a reductionist approach which has the potentially to trivialise the phenomena being modeled [23, 24, 25]. This is especially the case in areas that are complex and not fully understood. When the focus of modelling is the complexity itself, the simplifications of the reductionist approach are antithetical to the work being done. Thus the challenge is to assemble the models in a manner that avoids the inherent problems of reductionism.

The Collaborative Development Model (CDM) shown in Figure 2 above conceptualises the expression of the common language as a series of models that together articulates the complexity of the phenomena being studied. It is a three-dimensional modeling space in which individual models are located. This provides the necessary granularity of the subject to accommodate the complexity and support different perspectives of the phenomena. Computerising the models provides an environment for their integration and establishing their interdependence. Presenting the whole modelling space as a computer-based artefact allows users to browse and explore the multiple perspectives of the subject and maintain its complexity.

2.1 Activity Dimension

The Activity dimension describes the nature of the models and their implied goals. The nature of language implies that the development of the different models in this dimension cannot be random but must be progressive from the Descriptive model to the Experimentation model. The development process in this dimension allows the experts to gain a shared understanding of the phenomena [12, 13] and allows the experts to communicate effectively [14]. It is difficult to imagine that two experts can gain the fluency in their common language to construct an Experimentation model without approaching this through the other models.

The construction of the various models is a transformation process that evolves the common language to accommodate the specific activity supported by the model. Each model extends the lexicon, grammar, syntax and semantics of the previous model. The current model differs from the previous model in respect to the knowledge that is represented and/or the intended use of the model. This evolution of the common language is an aspect of the dynamic nature of CDM but can create language inconsistencies between models. To avoid this, any changes to the language in the current model is incorporated, where appropriate, in previous models.

The representation of the common language also presents a challenge. A predefined representation formalism disenfranchises the subject domain expert. In formal modelling it is the modeller who verifies the model by interpreting solutions for domain meaning that are then presented to the domain expert ([15] and in particular [16, 17, 18]). From the linguistic perspective, such a formal model would lack both semantic and visual clarity required by the subject domain expert to "read" the model. This raises the issues of whether the content of such a model would adequately represent the subject matter, the ability of the subject domain expert to assess the accuracy and appropriateness of the model and, most importantly, the subject domain expert's confidence in the model.

An important assumption in CDM is that the models are idiosyncratic and hypothetical as they represent the knowledge of the collaborators rather than objective domain knowledge. The modelling process allows the experts to think through the implications of their hypotheses and explicitly declare them. Areas of uncertainty can be highlighted and can inform the design of experiments to disambiguate that uncertainty.

Description Activity. The function of the Descriptive model is to declare the scope of the Activity dimension by identifying the entities and phenomena of interest. The main usage of the Descriptive model is essentially browsing and information dissemination. It is also the simplest expression of the common language between the domain and computing experts.

However, negotiation is necessary to ensure that both the subject and the questions to be investigated are feasible in terms of size of model, choice of (iconic) expressions of the common language, ability to express concepts unambiguously in the common language and relevance to the research activity in the subject domain.

Formalisation Activity. The function of the Formal model is to investigate the static properties of the phenomena. The transformation of the Descriptive model to a Formal model is a process of formalisation of the semantics of the common language of the Descriptive model. Such formalisation is the basis of defining the common language as executable objects. CDM is not based on a specific formalism but collaborators are encouraged to adapt any suitable existing formalism that can adequately express their common language.

The transformation process of the Descriptive model to the Formal model is one of expressing the phenomena with more precision and finer granularity.

This requires additional information to be incorporated into the model that could result in the structure of the model being altered or the iconic element of the common language extended to reflect more of the detailed semantics. The precision of the model requires constraints on its syntax and the information required to express that precision. The common language also evolves to support the use of the Formal model. The language requires facilities to express the analytical functions.

Simulation Activity. The function of the Simulation model is to support the exploration of the dynamic behaviour of the phenomena that is limited to the manipulation of parameters. This allows researchers to use the model to explore aspects of the domain but within the structures defined by the model. Such investigations are non-trivial and often allow experiments that cannot be practically performed outside the computer.

The distinction between the Formal and Simulation models is the additional quantitative knowledge required for meaningful simulation and the level of detail and granularity of information. The common language is extended by additional iconic elements and the process can also change the structure of the model. The computing environment is extended to accommodate facilities to support interactive simulation and the logging of results [2]. These extensions are made from the perspective of the common language but are formalised to enable computer simulation to be conducted.

With complex and incomplete knowledge it is often difficult to adequately define the Formal and Simulation model with existing data. In such situations it is feasible to initially bypass the Formal model and transform the Descriptive model directly into the Simulation model. With limited information about the subject, quantitative information can be derived from simulation to deduce facts about the Formal model. On the other hand, hypotheses or best guesses can be inferred from the Formal model for inclusion in the Simulation model. In practice, since both models are computer-based, the testing of hypotheses incorporated in the Formal and Simulation models is facilitated by the ability to conduct analyses and/or simulations. This provides a means by which the model can be fine-tuned and is a process by which the validity of the model is continually challenged.

The interaction between the Formal and Simulation provides a constructive means to address incomplete data. In these situations it is the models themselves that are used to explore the domain to gain insight into the phenomena that are represented. Such usage establishes a very close nexus between the *in-machina* investigations and the experimental work conducted by the subject domain expert. Analyses and simulation generate results that superficially have domain significance because they are based on hypotheses not substantiated by published data. Experimental protocols need to be constructed to investigate whether the reported phenomena reflect reality or are an artefact of the model.

Experimentation Activity. The function of the Experimentation model is to support hypothesis formulation and testing. Unlike the Simulation model where

[2] For an example of an implementation of such a computer environment see [19, 11, 20].

investigations are framed in terms of varying parameter values, the Experimental model assumes that the hypothesis will entail structural changes to the model, semantic changes in the iconic language or both. Support for such usage requires computational tools and an environment in which to use these tools: a Computational Laboratory. Existing software packages do not support such functionality but these concepts have been pursued by other research projects for the last 15 years (see for example [19, 11, 20, 10]).

Unlike the transformations of the previous models, the construction of the Experimentation model is a process of extending the common language so as to allow direct interaction with the model to explicitly declare the hypothesis through structural changes to the model. This includes facilities to prune the model removing those branches that are outside the scope of the proposed experiment. In effect, the Experimentation model is a complex and sophisticated environment in which the Simulation model is embedded.

2.2 Representation Dimension

The Representation dimension allows information to be modelled using different forms and/or media. To make models accessible to users, who might not be the modellers, representations draw on the traditional domain inscriptions. Implementing these inscriptions as computer models implicitly extends the inscription. At a minimum, such models can be implemented as a hypertextual that allows a complex network of links between elements within a model and between models that together form the computer model. Moreover, such implementation can utilise different media more suited to the type of symbolic activity to be performed with those inscriptions.

The Representation dimension also provides opportunities to construct multiple models as dialects of the common language or semantically differentiated set of expressions. An example would be the representation of temporal knowledge. A dialect of the common language would involve additional marking on the diagram that would be utilised when the model is executed. Incorporating temporal knowledge in the formal definition of the common language however changes the semantics of the formalism and results in a structurally different model. In both cases the new model would be justified on the basis of the different perspectives of the reality being represented.

The models in the Representation dimension need to conform to the same purpose and goal as defined by the Activity dimension. The models can be constructed using any available media, draw on any appropriate domain inscription and can be expressed formally or descriptively. The models can represent the same knowledge in different forms or different aspects of that knowledge. The models, individually and collectively, define what phenomena are explored.

2.3 Subject Dimension

The Subject dimension allows for different aspects of the domain to be modelled separately. This dimension exists as a direct consequence of the linguistic

paradigm. The common language relies on the ability to define a limited lexicon that adequately expresses the necessary complexity, in terms of concepts and depth, to support research. To construct models that reflect this complexity within a reasonable timeframe, the scope of the model must be well defined and very narrow. This necessitates the partitioning the domain into a series of narrowly focussed models. Of particular interest is the possibility of modelling the same subject but from a different perspective.

The differences in the models in this dimension are particularly important from an experimental position in that they address a different research agenda and consequently a different set of experiments. This dimension allows the modellers to effectively partition the subject of their research to address their diverse experimental needs. This means that models along this dimension do not necessarily share the same common language since the language needs to adequately represent the phenomena being modelled. This is in contrast to the traditional approaches of translating the subject matter into a predefined and constrained formalism, irrespective of the subject matter or purpose of the model. While such flexibility important aspect of the linguistic basis of CDM, it presents a challenge to the formal integration of models in this dimension.

The rationale for models in this dimension is based on the modellers recognising that their subject is too broad to be effectively modelled. This occurs early in the process as the modellers spend their time on the breadth of the model and find it difficult to include the range of detail, expressing the range of concepts uniformly and the model becoming too dense. The result is that the readability of the model is impaired.

An alternative impetus for modelling in this dimension is the use of existing models that suggest new areas of investigation; the exploration of existing models offer insights into the phenomena that are based on factors beyond the scope of the existing model.

3 Artefact Construction

The construction of a computer-based artefact involves assembling multiple models within an environment that enables the phenomena to be explored interactively. To this end, many of the presentation and usage aspects of the models are embedded in the computer environment. This blurring of the distinction between the model and the medium in which it is represented is deliberate as the computer and its facilities are explicitly exploited in the construction of the model. The design of such artefacts is an integral part of CDM as it provides the means to represent the complexity of the phenomena through the integration of the models and their interactions.

CDM models are explicitly idiosyncratic and are not objectively validated but rely on the authority of the expert. Thus the artefact is directly related to the experts' research and must support the diverse experimentation required to address that research agenda. A versioning facility is required to provide the expert with control over model variations and their instances so that *in-machina*

experimentation is rigorous and can be interwoven with usual research practice. Such facilities are also important to provide outputs that would enable such research to be peer reviewed. An example of such an artefact is shown in Figure 3 below. It shows an aspect of HIV immunology in three representations: as a formal model expressed in the semantics of hierarchical colour Petri net; as an graphic model that shows the schematic representation of biological entities and their dynamic behaviour through animation; and as a text model that annotates the other models through bibliographic references and the domain expert's comments.

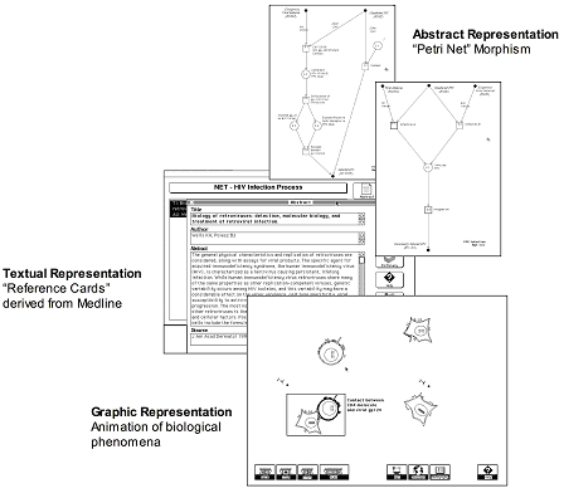


Fig. 3. A Montage of Models

Such artefacts can be conceptualised as a hypertextual document that includes the models, the facilities for the interactive use of the integrated models. Such a document needs to be dynamic so as to accommodate the evolution of the document itself. This constitutes a computational laboratory as referred to by Mussio [11] that facilitates the practical implementation of CDM. The importance of a computational laboratory is the ability to re-use its structure and environment to conduct the research other area of the domain. An alternate conceptualisation of such an artefact is provided by Mussio [21] with his proposed e-document that incorporates various representations, annotations and interactivity within a shared computer-based artefact.

4 Exploiting CDM

As a declaration of the "system in focus" [22], the Descriptive model is not envisaged for the direct use by the subject domain expert. Its main use is for information dissemination to non-specialist users. The Descriptive model can be supplemented by additional models along the Representation dimension that

would be expressed in terms accessible by such users. Most significantly, Descriptive models can be used as a foundation for didactic applications, including CBT, and electronic publishing.

Both the Formal and Simulation models are aimed at the subject domain experts' workplace as they are idiosyncratic designed for use by the domain expert. However such models, with modifications, could be utilised to construct facilities for research training in the domain provided the nature of the information is appropriate to such usage. Simulation model can inform the research agenda of the domain and has the capacity to be developed as another research modality within the domain, provided appropriate computational tools are also available.

The construction of the Experimentation model effectively provides the computational tools to support a new research modality in the subject domain. For such a modality to gain wide currency, the Experimentation model would need to be constructed so as to provide an environment for general use by other workers in the domain.

5 Conclusion

The CDM creates a space to model and explore phenomena that is central to the research agenda in a subject domain. In this sense it is an implementation of the conceptual layer of TbKM and provides the means to develop the tools needed to support knowledge work. The models and artefacts that are developed in this space are clearly designed to provide experts with the tools for "thinking". This allows the experts to be innovative in articulating assumptions and hypotheses and testing these positions by simulating the phenomena. The insights gained in such exploration provide the basis for more traditional activity within the domain that can verify the hypotheses. It is this interaction between the traditional work practices and the new modality of work that integrates the "doing" and "thinking" that together defines knowledge work.

References

1. Burstein, F., Linger, H.: Supporting post-fordist work practices: A knowledge management framework for dynamic intelligent decision support. Journal of IT&P special issue on KM **16** (2003) 289–305
2. Iivari, J., Linger, H.: The characteristics of knowledge work: A theoretical perspective. In: Proceedings of Americas Conference on Information Systems, AMCIS'2000, Long Beach California, August. (2000)
3. Iivari, J., Linger, H.: Knowledge work as collaborative work: A situated activity theory view. In: Collaborative Technology mini-track, Hawaiian International Conference on Systems Science (HICSS'32). (1999)
4. Aarons, J., Burstein, F., Linger, H.: What is the task? – applying the task-based km framework to weather forecasting. In: Proceedings of the Australian Conference on Knowledge Management and Intelligent Decision Support (ACKMIDS). (2005)
5. Linger, H.: Towards a linguistic paradigm for information systems development. In: Proceedings of 5th International Conference on Information Systems Development, ISD'96; Sopot, Poland. (1996)

6. Linger, H., Komleva, N., Newnham, J.: Collaborative development of support tools for biological research. In: EWHCI'94, East West Conference on Human Computer Interfaces, St. Petersburg, Russia. (1994)
7. Charlesworth, M., Farrall, L., Stokes, T., Turnbull, D.: Life amongst the scientists: an anthropological study of an Australian scientific community. Oxford University Press, Melbourne (1989)
8. Turnbull, D., Stokes, T.: Manipulable systems and laboratory strategies in a biomedical institute. In Grand, H.L., ed.: Experimental Inquiries, Netherlands, Kluwer Academic Publishers (1990)
9. Latour, B.: Visualization and cognition: Thinking with eyes and hands. Knowledge and Society: Studies in the Sociology of Culture Past and Present **6** (1986) 1–40
10. Celada, F., Seiden, P.: A computer model of cellular interactions in the immune system. Immunology Today **13** (1992) 56–62
11. Mussio, P., Pietrogrande, M., Protti, M.: Simulation of hepatological models: A study in visual interactive exploration of scientific problems. Journal of Visual languages and Computing **2** (1991) 75–95
12. Lyytinen, K.: Different perspectives on information systems: Problems and solutions. ACM Computing Surveys **19** (1987) 5–46
13. Wootton, A.: Dilemmas of Discourse. George Allen & Unwin Ltd., London (1975)
14. Tondl, L.: Problems of Semantics. Reitel Publishing Company, Dordrecht, Holland (1981)
15. Perelson, A., Weisbuch, G.: Theoretical and experimental insights into immunology. In: Proceedings of the NATO Advanced Research Workshop on Theoretical Immunology Paris Nato ASI Series, Berlin, Springer-Verlag (1991)
16. Segal, L.A.and Perelson, A., Hyman, J., Klaus, S.: Rash theory. In: Perelson & Weisbuch 1991. (1991)
17. Agur, Z., G., M., I., M.: Mimicking the strategy of the immune system: Insights gained from mathematics. In: Perelson & Weisbuch 1991. (1991)
18. McLean, A.R.: T memory cells in a model of t-cell memory. In: Perelson & Weisbuch 1991. (1991)
19. Mussio, P., Finadri, M., Gentini, P., Colombo, F.: A bootstrap approach to visual user interface design and development. The Visual Computer **8** (1992) 75–93
20. Mussio, P., Pietrogrande, M., Brambilla, F., Caserta, C., Cefalo, A., Randetti, M., Bottoni, P., Dioguardi, N.: Modelling of biological and medical systems: a systemic strategy. BioSystems **26** (1991) 31–43
21. Mussio, P.: e-Documents as tools for the humanized management of community knowledge. In Linger et al., eds.: Constructing the Infrastructure for the Knowledge Economy: Methods and Tools; Theory and Practice, Kluwer Academic Publishers (2004)
22. Beer, S.: The Brain of Firm. John Wiley & Sons, New York (1981)
23. Eaves, D.: Professionals, Professions and the Curiosity of Clinical Practice. Working Paper 20/93, Department of Information Systems, Monash University (1993)
24. Morik, K.: Sloppy Modelling. In Morik, K., ed.: Knowledge Representation and Organisation in Machine Learning, Berlin, Springer-Verlag (1989)
25. Klein, H.K. and Hirschheim, R.A.: A Comparative Framework of Data Modelling Paradigms and Approaches. The Computer Journal. 30 (1987) 8-15

Profiling Network Attacks Via AIS

Anastasia Pagnoni and Andrea Visconti

Department of Computer Science and Communication, University of Milan, Milano, 20135, Italy

Abstract. The paper extends the intrusion detection methodology proposed by Tarakanov et al. in [8] to k-dimensional shape spaces, for k greater or equal 2. k real vectors, representing antibodies, are used to recognize malicious (or, non-self) connection logs. We suggest a method for recognizing antigens (generating such antibodies) via Singular Value Decomposition of a real-valued matrix obtained by preprocessing a database of connection logs [9]. New incoming connection requests are recognized by the antibodies as either self (normal request), or non-self (potential attack), by (a) mapping them into a k-dimensional shape space, and (b) evaluating the minimum Hamming distance between their image and that of a known attack logs. It is easy to see that using a shape space of dimension greater than 2 significantly reduces false positives.

1 Introduction

Biological immune systems are quite successful at protecting living organisms against a vast variety of dangerous intruders. This complex defense system includes a first line of defense, consisting in chemical and physical barriers — skin, mucus secretions, stomach pH, etc., —, and two more specific defense systems, the innate and the acquired immune system. A number of authors [1, 6, 3, 2] suggested biological immune systems as a new, promising paradigm for the design of powerful computer security systems.

Researchers designed artificial immune systems inspired by both the workings of the adaptive immune system [8, 7, 4], and the innate immune system [5, 11]. The adaptive immune system is the part of the biological immune system capable of adapting to an ever-changing environment so as to recognize, and fight, already-met antigens better. The innate immune system is the part of the biological immune system in charge of protecting the body from its birth.

In [8] Tarakanov et al. presented the formal model for an intrusion detection system inspired by the workings of an artificial immune system, or more precisely, of the acquired immune system. This IDS uses a real-valued matrix L, obtained by preprocessing a training database, consisting of an extensive record of previous connection logs. Matrix L is used to map preprocessed connection logs, old and new, into a real 2-dimensional vector space, in a way that lumps together "similar" network attacks. This space is called *shape space*.

This method relies on the heuristic choice of two singular vectors (antibodies) of the SVD (Singular Value Decomposition) of matrix L, and fails for unfortunate choices.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 272–277, 2006.

In the sequel, we suggest how to improve the performance of such an IDS by using a shape space of higher dimension, and by choosing the our antibodies in a deterministic way. The suggested method will be formally introduced in Section 3.

2 Problem Description

As we mentioned, biological immune systems draw up several lines of defense to protect living organisms from all kinds of potentially dangerous antigens. Computer networks too rely on several lines of defense to protect themselves against unwanted intrusions: firewalls, access control systems, intrusion detection systems (IDSs). Indeed, firewalls can be seen as network equivalents of the skin, as they filter external requests of access, and block all connection attempts that violate certain criteria. Login policies correspond to physiological barriers, as they let regular physiological users into the network, while blocking the access of outsiders, and so on. IDSs are used to detect unauthorized use of computer systems, either by internal, or external network users. An IDS usually relies on set of profiles of normal user behavior, and defines so-called misuse signatures, based on known intrusion techniques, or specific system vulnerabilities. An IDS detects intrusions by identifying significant deviations from normal behavior, and checks whether connection logs exhibit misuse signatures [10].

Our problem was to design an IDS capable of recognizing attack signatures in a log file, signatures that characterize a certain type of attack. Looking for a specific attack log, — for a specific antigen — in a long log file requires knowing which, if any, set of features characterizes this attack. Such a set of features is usually called an *antigen signature*. Some such signatures are both well-known and certain, and can easily be recognized automatically, while other are completely unknown, or just less well-defined.

Our approach is based on the assumption that network attacks are often variants of known attacks, and hence have the same signatures.

In this paper we suggest how to recognize antigen signatures in a file of connection logs. We assume that our file has been obtained, and preprocessed, via a suitable log parser. For our testing we used the database of KDD Cup 1999 Data Set [9], which is a log file used for The Third International Knowledge Discovery and Data Mining Tools Competition. This database contains a preprocessed standard set of data, which includes a wide variety of intrusions simulated in the framework of a military LAN.

3 Signature Recognition

The proposed AIS routinely performs the following four macro-steps: (a) Antigen identification. "Old" logs are partitioned into $k + 1$ sets, k of which correspond to different network attacks, one to regular user activity. This is done at first by a human supervisor. (b) Learning setup. In this phase, the immune system learns how to automatically extract attack features that uniquely characterize

attacks the k attack types above, or mutations of them. (c) Antigen definition step. k antigen signatures, one for each attack type, are extracted. (d) Antigen recognition step. New connections requests are processed, and recognized as being either self or non-self.

Let us introduce the for macro-steps in detail.

Let L be a $m \times n$ real-valued matrix obtained by parsing and preprocessing a large file of security-relevant connection logs of a certain network.

Let $l_j = [l_{j_1}, l_{j_2}, \ldots, l_{j_n}]^T$ be the j-th row vector of matrix L.

Assume the supervisor has identified k different attacks types, $A_1, A_2, \ldots, A_k$ among the m rows of L. It is easy to see that every attack of type A_i will be characterized by the recurrence of a specific pattern in a number of its row components. We will call this pattern the signature of attack A_i, and denote with a_i the number of row components it consists of.

This way, the set of rows of matrix L is partitioned in k subsets L_i, such that the elements the L_i, and only them, exhibit signature A_i. In this paper we will assume the number k to be a priori known.

3.1 Antigen Identification

During the antigen identification phase, an expert partitions matrix L into $k+1$ sub-matrices $S, L_1, \ldots, L_k$, so that sub-matrix L_i is made up exactly of the rows belonging to the same attack i, while the rows of S are associated to non-attack logs. The ordering of rows is preserved in the sub-matrices.

3.2 Learning Setup

In the learning phase, the AIS learns how to extract from sub-matrix the signature of the i-th attack. This is done by first applying the singular value decomposition (or, SVD) to matrix L_i.

The SVD method is very well-known matrix decomposition method [12], based on the following theorems.

Theorem 1. *Let L be a $m \times n$ matrix defined on a field F. $m \times m$ matrix $L \times L^T$ and $n \times n$ matrix $L^T \times L$ have the same set of non-null eigenvectors.*

Theorem 2. *For both matrix $L \times L^T$ and matrix $L^T \times L$ exactly one maximal set of orthonormal eigenvectors exist.*

Definition 1. *The square roots of the above eigenvalues are called the* singular values *of matrix L. Orthonormal eigenvectors of matrix $L \times L^T$, and matrix $L^T \times L$, are respectively called left, and right,* singular vectors *of L.*

Theorem 3 (SVD). *Any $m \times n$ matrix L, defined on a field F, is the product of three matrices: an $m \times m$ matrix P, an $m \times n$ diagonal matrix C, and an $n \times n$ R. The entries on the main diagonal of matrix C, are the singular values of L, arranged in decreasing order. The columns of matrix P build a set of left singular vectors of L, ordered as the corresponding singular values. The columns of matrix R build a set of right singular vectors of L, similarly ordered.*

$$\begin{pmatrix} L_{m\times n} \end{pmatrix} = \begin{pmatrix} P^T_{m\times m} \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \\ 0 & \cdots & 0 & 0 \\ 0 & \cdots & 0 & 0 \end{pmatrix} \begin{pmatrix} R_{n\times n} \end{pmatrix} \tag{1}$$

By applying the SVD [12] decomposition method to attack matrix L_i we get

$$SVD(L_i) = P_i^T C_i R_i \tag{2}$$

In the sequel we will denote by $e_i = [\lambda_1, \ldots, \lambda_n]$ the vector of singular values of the matrix L_i, and always assume that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$. The entries of vector e_i are singular values of matrix L_i, stored in decreasing order.

3.3 Antigen Definition Step

Every $m \times n$ matrix L on a vector space V can be defined as a mapping from vector space V^m to vector space V^n. The SVD decomposition of a real-valued matrix L, decomposes L into the product of three matrices, each of which can be interpreted as an elementary mapping. More precisely, if $L = P^T CR$ is the SVD of L, matrix P can be interpreted a rotation of the axes of vector space $\Re^m$ around its origin; matrix C can be interpreted a mapping from vector space $\Re^m$ to vector space $\Re^n$ that just "stretches" coordinates (equivalently, as just a unit change on the existing axes), change of unit on the axes of vector space $\Re^m$ around its origin; matrix R can again be interpreted a rotation of the axes of vector space $\Re^n$ around its origin.

As the singular values appear in decreasing order in the main diagonal of C, the SVD detects the most relevant information components of the mapping represented by L, and hence of L itself.

Based on this observation, we can extract k antigen signatures, one for each attack type, from the corresponding L_i matrices. To do this, we first compute the SVD of all matrices L_i, their k vectors of singular values e_i, and normalize the e_i. Then, we set a number between 0 and 1 as a common threshold σ, as the lower bound for all our singular values. Singular values above σ — that is, singular values with information content over our threshold — will be used to extract the antigen signatures. Is will be done as follows.

Definition 2. *Let e_i the vector of normalized singular values*

$$e_i = \left[\frac{\lambda_1}{\lambda_1}, \ldots, \frac{\lambda_{min}}{\lambda_1}\right] = [1, \lambda_2, \ldots, \lambda_{min}], \tag{3}$$

and let σ be a threshold, with $\sigma \in [0, 1]$. We define E_i the following subset of singular values:

$$E_i = \{\lambda_f : \lambda_f \in e_i, \quad and \quad 1 - \lambda_f \leq \sigma, \quad \forall \lambda_f \in e_i\}. \tag{4}$$

With the threshold σ, we define the number of singular values that belong to the subset E_i.

Definition 3. *Let n_i be the cardinality of the subset E_i.*

The cardinality of the subset E_i is a characteristic of attack i. In this way, we can define the n_i number of singular values more important for an attack: the *antigen signature*.

In matrix C_i, we set the $n - n_i$ smallest singular values equal to 0. We can say that the matrix product $P_i^T C_i R_i$ will be close to L_i.

$$P_i^T C_i R_i \simeq L_i. \tag{5}$$

3.4 Antigen Recognition Step

A strong recognition of self/non-self connection logs is key too the performance of any AIS. We suggest applying an extension of the method suggested in [8]. There A. Tarakanov et al. suggested representing preprocessed connection logs as formal proteins, and discriminating non-self from self logs by mapping them onto a 2-dimensional shape space. This is a particular Cartesian plane, the axes of which are two right singular vectors of log matrix L, introduced above. This way, connection logs are mapped onto the shape space. The coordinates of the image of log vector l are defined as:

$$\begin{cases} w_{1i} = \frac{1}{\lambda_1} l^T R_{1i} \\ w_{2i} = \frac{1}{\lambda_2} l^T R_{2i} \end{cases} \tag{6}$$

where R_{1i} and R_{2i} are called antibody probes.

This approach is based on the assumption that this mapping will group "similar" non-self logs together, in clusters called recognition balls, and characterized by their signature. Recognition of a new connection log l is done by mapping it into the shape space, and detecting a recognition ball the signature of which has minimal distance from it. A log "close enough" to a certain signature is recognized as non-self, and identified as the corresponding attack.

It is easy to see, and Tarakanov acknowledges, that the choice of the two antibody probes is crucial to the performance of this recognition method. In fact, picking the "wrong" antibodies could lead to a poor, even empty, set of recognition balls.

To solve this problem we suggest: (a) perform recognition on the attack level, that is on the attack matrices instead than on the whole log matrix L; (b) using more than two antibody probes, if necessary. More precisely, choosing as many antibody probes as there are maximal singular values of L_i — such maximal singular values being defined as in (4). (c) Antibody probes for attack A_i will be the right singular vectors of L_i corresponding to the maximal singular values of L_i.

If the matrix of i-th attack type has n_i maximal singular values, we will take the corresponding right vectors, $R_{n_i i}$, as antibody probes for this type of attack. Such antibodies will be used to map new connections logs into an n_i-dimensional shape space. More precisely, new connection log l, will be mapped into shape space point $w_{1i}, w_{2i}, \ldots, w_{n_i i}$ the components of which are:

$$\begin{cases} w_{1i} &= \frac{1}{\lambda_1} l^T R_{1i} \\ w_{2i} &= \frac{1}{\lambda_2} l^T R_{2i} \\ & \cdots \\ w_{n_i i} &= \frac{1}{\lambda_{n_i}} l^T R_{n_i i} \end{cases} \tag{7}$$

Hamming distance is used to evacuate detect the recognition ball closest to it. If the distance of the new connection log l is under a certain threshold, l will be added to the matrix L_i, as an additional row.

4 Conclusions

The method we presented is to be considered work in progress, and is been tested on a variety of security-related log files. It has proven quite effective in denying foreign processes access to the server, and in blocking DoS (denial of service) and port scanning attacks. Its efficiency in learning to detect new, unknown network attacks has shown promising testing results, though we are still studying how to set optimal signature thresholds.

References

1. L.N. De Castro, J. Timmis J, Artificial Immune System: a new computational intelligence paradigm, New York: Springer-Verlag, 2002
2. P. D'haeseleer, S. Forrest, P. Helman, An immunological approach to change detection: algorithms, analysis and implication, Proceedings of the 1996 IEEE Symposium on Computer Security and Privacy, IEEE Computer Society Press, 1996
3. F. Esponda, S. Forrest, P. Helman, Positive and Negative Detection, IEEE Transaction on System, Man, and Cybernetics, in press, 2004 downloaded from the internet, at http://www.cs.unm.edu/∼forrest/isa_papers.htm
4. S. Forrest, S. Hofmeyr, A. Somayaji, Computer immunology, Communication of ACM, Vol. 40, No. 10, 1997, pp. 88–96
5. S. Forrest, S. Hofmeyr, A. Somayaji, T. Longstaff, A sense of self for UNIX processes, Proceedings of the 1996 IEEE Symposium on Research in Security and Privacy, IEEE Press, 1996
6. S. Forrest, A. Perelson, L. Aleen, R. Cherukuri, Self-nonself discrimination in a computer, Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy, Los Alamos, CA, IEEE Computer Society Press, 1994
7. S. Hofmeyr, S. Forrest, Architecture for an Artificial Immune System, Evolutionary Computation, Vol. 7, No. 1, Morgan-Kaufmann, San Francisco, CA, 2000
8. A.O. Tarakanov, V.A. Skormin, S.P. Sokolova, Immunocomputing: Principles and Applications, New York: Springer-Verlag, 2003
9. KDD Cup 1999 Data Set, downloaded from the internet, at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
10. B. Mykerjee, T.L. Heberlein, K.N. Levitt, Network Intrusion Detection, IEEE Network, Vol. 8, No. 3, 1994.
11. A. Pagnoni, A. Visconti, An Innate Immune System for the Protection of Computer Networks, In Information and Communication Technologies, B.R.Baltes et al., eds., ACM International Conference Proceedings Series, 2005, ISBN 0-9544145-6-X.
12. R. Horn, Johnson, Matrix Analisys, Cambridge University Press, 1986.

Fuzzy Continuous Petri Net-Based Approach for Modeling Immune Systems*

Inho Park, Dokyun Na, Doheon Lee**, and Kwang H. Lee

Department of BioSystems, KAIST,
373-1, Guseong-dong, Yuseong-gu, Daejeon, Republic of Korea
dhlee@biosoft.kaist.ac.kr

Abstract. The immune system has unique defense mechanisms such as innate, humoral and cellular immunity. These mechanisms are closely related to prevent pathogens from spreading in the host and to clear them effectively. To get a comprehensive understanding of the immune system, it is necessary to integrate the knowledge through modeling. Many immune models have been developed based on differential equations and cellular automata. One of the most difficult problem in modeling the immune system is to find or estimate appropriate kinetic parameters. However, it is relatively easy to get qualitative or linguistic knowledge. To incorporate such knowledge, we present a novel approach, fuzzy continuous Petri nets. A fuzzy continuous Petri net has capability of fuzzy inference by adding new types of places and transitions to continuous Petri nets. The new types of places and transitions are called fuzzy places and fuzzy transitions, which act as kinetic parameters and fuzzy inference systems between input places and output places. The approach is applied to model helper T cell differentiation, which is a critical event in determining the direction of the immune response.

1 Introduction

The immune system is our defense mechanism against pathogens or abnormal cells. The immune system is composed of innate and adaptive immune system. The innate immune system refers to antigen-nonspecific defense mechanisms which a host uses immediately or within several hours after exposure to antigen. In this type of response, immune cells belonging to the innate immune system recognize a few highly conserved structures, which are called pathogen associated molecular patterns(PAMPs), present in many different microorganisms. The adaptive immune system consists of humoral immunity and cellular immunity. Humoral immunity produces antibodies, which are our main chemical weapons against pathogens. Cellular immunity leads to killing of cells infected by viruses [1].

* This work was supported by National Research Laboratory Grant (2005-01450) from the Ministry of Science and Technology. We would like to thank CHUNG Moon Soul Center for BioInformation and BioElectronics and the IBM-SUR program for providing research and computing facilities.

** Corresponding author.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 278–285, 2006.

The immune system is a complex system composed of nearly 10^{12} immune cells and characterized by complex interactions, between these elements, which are highly distributed over the body. Although molecular immunologists have been very successful in revealing the molecular basis of specific immunological phenomena, in order to get a comprehensive understanding of the immune system, it is necessary to integrate the knowledge through modeling studies. With immune system models, we can make *in silico* experiments and predict the result of *in vivo* or *in vitro* experiments for further studies [2].

Several models based on either on cellular automata (CA) [3, 12, 13] or on differential equations [4, 5, 10, 11] have been developed to explain immunological phenomena. For example, IMMSIM is a representative CA-based immune-specific simulator. Although CA-based models can show the individual properties of each immune cell, they could not afford to simulate the immune system as a whole, because of computational complexity [6]. Unlike CA-based models, differential equation based models can show averaged properties of immune cells by expressing the rates of change of immune cells or antibodies at the population level. However, it is not easy to solve or analyze the equations when many variables and nonlinearity are involved in a model.

To integrate immune system models, we applied continuous Petri Nets to immune system modeling[7]. Petri Nets have several advantages over other methods for immune system modeling. Petri Nets have intuitive graphical representations with strong theoretical background and have been successfully used for concurrent, asynchronous, and parallel system modeling in many fields. Recently, they have been widely used to represent biological pathways or processes because of their properties, which are relevant for biological systems[8, 9]. Although we could easily integrate immune system models with continuous Petri Nets, it is difficult to find the appropriate parameter values needed to complete the model.

Here we present a novel approach for incorporating qualitative knowledge when constructing immune system models by using fuzzy inference systems. To demonstrate the usefulness of this approach, we applied it to helper T cell differentiation model a very important process in the regulating of the immune system.

2 Modeling Approach

A Petri net is a graphical and mathematical modeling tool. In many fields, Petri nets have been successfully used for concurrent, asynchronous, and parallel system modeling. Especially, Petri nets are used to represent biological pathways or processes. Following is the definition of basic Petri nets[14, 15].

Definition 1. *A Petri net is a 5-tuple* $R =< P, T, F, W, M_0 >$ *where* $P = \{p_1, p_2, \cdots, p_n\}$ *is a finite set of places,* $T = \{t_1, t_2, \cdots, t_n\}$ *is a finite set of transitions. The set of places and transitions are disjoint,* $P \cap T = \emptyset$. $F \subseteq (P \times T) \cup (T \times P)$ *is a set of arcs.* $W : F \rightarrow \{1, 2, 3, \cdots\}$ *is a weight function. And* $M_0 : P \rightarrow \{0, 1, 2, 3, \cdots\}$ *is the intial marking.*

The behavior of a Petri net is described in terms of changes of tokens in places according to the firing of transitions. If all of the input places of a transition have tokens more than the weight of arc between the transition and the place, the transition is enabled. Of the enabled transitions, only one transition can fire. After a transition is fired, the tokens of input places are removed as many as the weight of the arc. And the tokens of output places are added as many as the weight of the arc.

Because of discreteness of the basic Petri net, it is not sufficient for immune system modeling. So we used a continuous Petri net which is an extension of the basic Petri net. The differences between the basic Petri net and the continuous Petri net are following. In the continuous Petri net, places can have real value marking and transitions fire continuously with some velocity. The velocity of transition is affected by the marking of places. Following is the definition of the continuous Petri net[16].

Definition 2. *A continuous Petri net is a 6-tuple* $R =< P, T, V, F, W, M_0 >$ *where* P, T, F, W, M_0 *are same to the basic Petri net.* $V : T \rightarrow V(p_1, p_2, p_3, \cdots) \in R^+$ *is the firing speed function.*

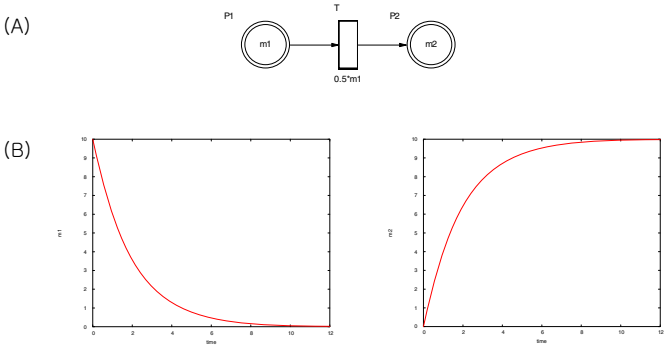


Fig. 1. Continuous Petri net. In the graphical representation of the Petri net (A), a circle represents a place and a rectangle represents a transition. Two graph (B) show the changes of tokens of places with respect to time.

In our immune system model, places represent immune cells or external materials (e.g. antigen or virus) and transitions represent interactions (e.g. B cell activation by antigen).

In mathematical modeling and simulation, we usually need appropriate values of kinetic parameters of the system. However, it is difficult to find parameter values of the system in the literatures. Therefore, researchers use various methods to estimate unknown parameters. On the contrary, it is relatively easy to get linguistic or qualitative knowledge. To incorporate this knowledge into a model, we employed fuzzy inference system and we could inference kinetic parameters from qualitative knowledge.

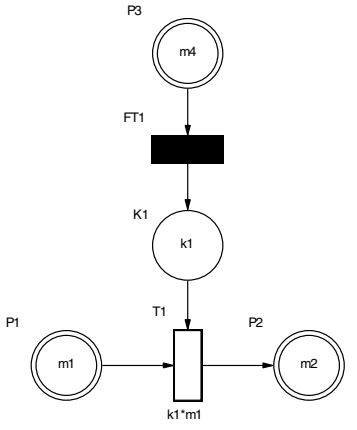


Fig. 2. A fuzzy continuous Petri net. In the figure, black transition represents fuzzy transition and single circle represents fuzzy place. Fuzzy transition FT1 inferences k_1 value from fuzzy if-then rules. And k_1 used in the firing speed function of continuous transition, T1.

To include fuzzy inference functionality in the continuous Petri Nets model, we added new types of transitions and places, fuzzy transitions and fuzzy places. The role of fuzzy transitions are to infer parameter from fuzzy if-then rules between input places and output place. And only fuzzy places can be the output place of fuzzy transition. Fuzzy places act as reaction parameters of continuous transitions. The Fig. 2 shows a simple example of fuzzy continuous Petri net.

3 Modeling Example: Helper T Cell Differentiation

3.1 Helper T Cell Differentiation

Helper T cells regulate adaptive immune response by producing various cytokines upon antigen stimulation. The regulatory functions of helper T cells are promoted by their differentiation into two distinct subsets, Th1 and Th2 cells which have been defined on the basis of their pattern of cytokine secretion[17]. Th1 cells are involved in inducing cellular immune response by activating cytotoxic T cells. Th2 cells trigger B cells to produce antibodies. In general, cellular immune responses remove intracellular microorganisms by killing infected cells, whereas humoral immune responses eliminate extracellular pathogens by antibody. Because cellular and humoral immune response have quite different roles in protecting the host from foreign substances, helper T cell differentiation is a crucial event in the immune response.

The decision of a naive helper T cell to differentiate into Th1 or Th2 is mainly controlled by cytokines such as interleukins(e.g. IL-2, IL-10, IL-12) and

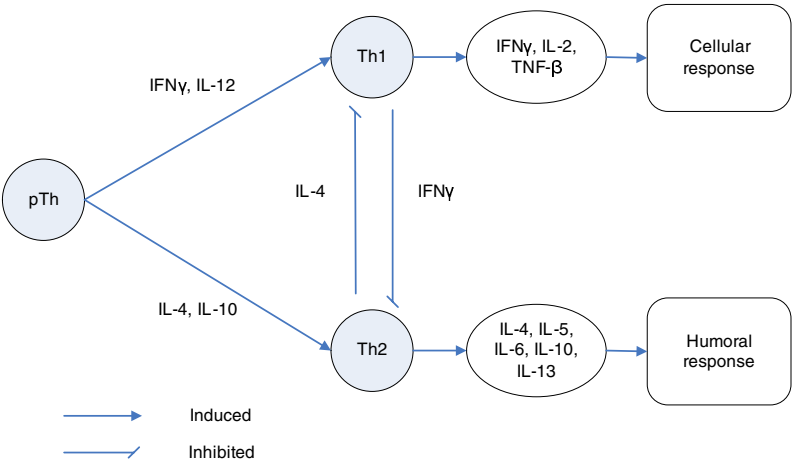


Fig. 3. A schematic diagram of helper T cell differentiation

interferon(e.g. IFN-γ). The cytokines form a signalling network, regulate helper T cell differentiation, and are produced by helper T cells and other immune system components[18]. Although both Th1 and Th2 cells secrete a large number of cytokines, only Th1 cells secrete IL-2 and IFN-γ, while only Th2 cells produce IL-4 and IL-10. Fig. 3 shows a simple schematic diagram of helper T cell differentiation. To understand the mechanism of helper T cell differentiation with cytokine network, many mathematical models based on ordinary differentiation equations have been proposed [18, 20, 22].

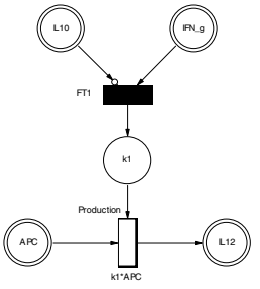
3.2 A Fuzzy Continuous Petri Net Model

We represented a simple Th1/Th2 differentiation model with a fuzzy continuous Petri Net model. Fig. 4 shows several reactions in the model. Each fuzzy transition has if-then fuzzy rules between input places and output place.

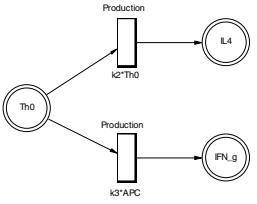
In the fuzzy transition, FT1 has three rules:

1. The concentration of IL-12 is higher than $0.5ng/ml$ and that of IL-4 is lower than $10ng/ml$, the differentiation rate is about $4day^{-1}$.
2. The concentration of IL-4 is higher than $10ng/ml$, the differentiation rate is about $2day^{-1}$.
3. The concentration of IL-4 is higher than $100ng/ml$, the differentiation rate is nearly 0.

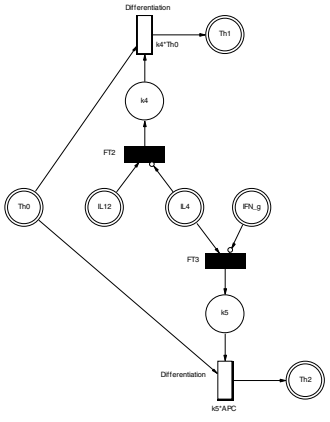
Although the if-then rules are not complete, the gaps could be filled during fuzzification process of rules. And also newly discovered knowledge can be easily incorporated in the fuzzy rule system[21].



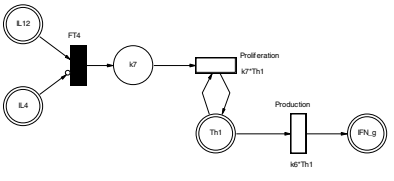
(a) APC cytokines production



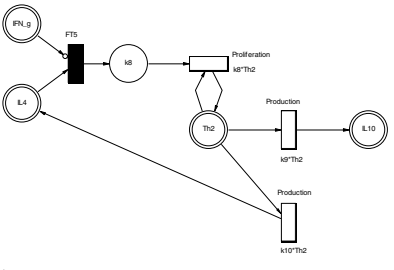
(b) Th0 cytokines production



(c) Th1/Th2 differentiation



(d) Th1 proliferation and cytokine production



(e) Th2 proliferation and cytokine production

Fig. 4. Important reactions of Th cell differentiation

4 Conclusion and Discussion

In conclusion, we propose a new approach to immune system modeling, a fuzzy continuous Petri net. The advantage of the modeling method is that we can make use of qualitative or linguistic knowledge which is relatively easy to get. Fuzzy inference systems have been successfully used in expert systems in various domains, and we show that it can also used in immune system modeling. To be more useful, we need to develop more robust methods for constructing fuzzy rules from immunological knowledge. Moreover, we also have to develop analysis technique such as reachability and boundedness for traditional Petri net.

The challenge of recent research in immune system modeling is how to integrate models of molecular transcriptional networks and cellular communications via cytokines[23]. Hierarchical petri net approach could be a good candidate formalism for integrating knowledge from different levels.

References

1. Janeway, C.A., Travers, P., Walport, M., Shlomchik, M., Immunology: The Immune System in Health and Disease. Taylor and Francis Inc., London 2001
2. Alan Aderem and Leroy Hood, Immunology in the post-genomic era, Nat. Immunol. 2001 May;2(5):373–375
3. Castiglione, F., A network of cellular automata for the simulation of the immune system. Int. J. Morden Physics C 10, 1999:677–686.
4. Rundell, A., DeCarlo, R., HogenEsch, H., Doerschuk, P., The humoral immune response to *Haemophilis influenzae* type b:a mathematical model based on T-zone and germinal center B-cell dynamics. J. Theor. Biol., Vol. 228, No. 2, May 2004
5. Perelson, A.S., Modelling viral and immune system dynamics. Nature Rev. Immunol. 2 (2002) 28–36
6. Puzone, R., Kohler, B., Seiden, P., Celada, F., IMMSIM, a flexible model for in machine experiments on immune system responses. Future Generation Computer Systems 18 (2002) 961–972.
7. Dokyun Na, Inho Park, Doheon Lee, and Kwang H. Lee, Integration of Immune Models Using Petri Nets, ICARIS 2004, LNCS 3239, pp. 205-216.
8. Matsuno H, Doi A, Nagasaki M, and Miyano S., Hybrid Petri net representation of gene regulatroy network, Pac. Symp. Biocompute., 2000, pp. 341-52
9. Mor Peleg, Iwei Yeh, and Russ B. Altman, Modelling biological processes using workflow and Petri Net models, Bioinformatics, 18(6):825–837.
10. Simeone Marino, Denise E. Kirschner., The human immune response to *Mycobacterium tuberculosis* in lung and lymph node. J. Theor. Biol., Vol. 227, No. 4, April 2004.
11. G. A. Bocharov and A. A. Romanyukha., Mathematical Model of Antiviral Immune Response III. Influenza A Virus Infection., J. Theor. Biol., Vol, 167, No. 4, April 1994.
12. Steven H. Kleinstein and Philip E. Seiden., Simulation the immune system. Computing in Science and Engineering, Vol.2, No.4, July 2000.
13. R. M. Z. dos Santos and S. Coutinho., Dynamcis of HIV infection: A Cellular Automata Approach. Phys. Rev. Letters, Vol.87, No.16, October 2001.

14. J. L. Peterson., Petri net theory and the modeling of systems. Prentice Hall, englewood Cliff, NJ, 1981
15. T. Murata., Petri nets: Properties, analysis and applications, Proc. IEEE Vol. 77, No. 4, April 1989
16. H. Alla, R. David., A modeling and analysis tool for discrete event systems: continuous Petri net. Performance Evaluation, Vol. 33, No. 3, August 1999
17. Street, N. E. and T. R. Mosmann., Functional diversity of T lymphocytes due to secrection of different cytokine patterns., FASEB. J. Vol. 5, 1991, 171–177
18. Claudia Bergmann and J. Leo Van Hemmen, Th1 or Th2: How an Approate T Helper Response can be Made, Bulletin of Mathematical Biology, 63:405–430, 2001.
19. Dennis L. Chao, Miles P. Davenport, Stephanie Forrest, Alan S. Perelson, A stochastic model of cytotoxic T cell responses. J. Theor. Biol., Vol. 228, No. 2, May 2004
20. Andrew Yates, Claudia Bergmann, J. Leo Van Hemmen, JarStark, and Robin Callard, Cytokine-modulated Regulation of Helper T Cell Populations, J. theor. Biol., 206:539–560, 2000.
21. Kwang H. Lee, First Course on Fuzzy Theory and Applications, Springer, 2005
22. Michael A. Fishman and Alan S. Perelson, Th1/Th2 Differentiation and Cross-regulation, Bulletin of Mathematical Biology, 61:403–436, 1999.
23. Andrew Yates, Robin Callard, Jaroslav Stark., Combining cytokine signalling with T-bet and GATA-3 regulation in Th1 and Th2 differentiation: a model for cellular decision-making., Jour. Theor. Biol., 231 (2004) 181-196

A General Learning Rule for Network Modeling of Neuroimmune Interactome

D. Remondini[1,3,4], P. Tieri[1,2], S. Valensin[1,2], E. Verondini[1,4], C. Franceschi[1,2], F. Bersani[1,4], and G.C. Castellani[1,3,4]

[1] "L.Galvani" Interdipartimental Center for Biophysics, Bioinformatics and Biocomplexity
[2] Department of experimental Pathology
[3] DIMORFIPA
[4] Physics Department and INFN, University of Bologna
daniel.remondini@unibo.it, gastone.castellani@unibo.it

Abstract. We propose a network model in which the communication between its elements (cells, neurons and lymphocytes) can be established in various ways. The system evolution is driven by a set of equations that encodes various degrees of competition between elements. Each element has an "internal plasticity threshold" that, by setting the number of inputs and outputs, determines different network global topologies.

1 Introduction

The network description of Immune System (IS) began in the 70's [15]. The network theory of IS was further developed over the past 30 years with several approaches[17, 8, 3]. The network approach on IS was influenced by Neural Network theories and modelling that, in their original formulation, started in the 40's and was developed along various lines and perspectives [12, 2]. Among the various learning rules we will consider one special class : the energy based learning rule [4], where the weight updating is obtained by minimization of a "Risk" or "Energy Function"[12, 14]. During the last years the so called "Complex network theory" showed several progresses in theoretical and applicative fields[22, 21]. In these models the evolution of links is obtained by various "rules" [22]. In the contemporary BioMedicine, concepts like transcriptome, interactome, etc are network-inspired. The effect of connection topology on immune and neural networks is still not completely understood, and the the existing models are not completely characterized on the basis of the generated connectivity patterns [9]. One of the present challenges is a better understanding of how microscopic relations between and within network elements can lead to macroscopic phenomena such as adaptation, learning and memory, and how genetic and environmental information fluxes interact. This work deals with the *connectivity* problem, which is not new in the frame of IS models [6, 20]. Among

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 286–292, 2006.

various neural network the so called BCM (Bienenstock, Cooper and Munro) neural network (BCM NN)[2, 14, 7, 5] shows an interesting similarity in term of stability of "learning rule" solutions with a class of immune network models[3]. The relation between immune and nervous system looks very intriguing since both systems can be described, from a mathematical-systemic point of view, by similar equations based on a "network" structure [10, 15]. Recent studies suggest that the similarity between the two systems is not purely formal [11]. Among others, a surprising result was [1] that some fundamental molecules for learning and memory in the nervous system (like CaMK II and Calcineurin) play a fundamental role also in the IS, leading to a new scenario in which molecules and pathways are shared by the two systems and conserved during evolution. In this paper we present a learning rule, that can be applied to a wide class of networks to model some emergent properties such as memory and learning.

2 Model

We propose a model in which generic units (neurons, lymphocytes etc) can communicate by links among them. Each unit is specified by its inputs, outputs and an internal parameter: a history dependent threshold. This threshold depends on the time average of incoming and outgoing signals and maps the history of each unit to a plasticity function. This assumption is justified by the fact that, in terms of internal activity and methabolic energy, the receiving and sending activities are "active" processes. The "plasticity"function Φ is responsible for the strengthening and weakening of links between nodes. The definition of Φ is choosing according to statistical considerations and mathematical simplicity, but the general results are true for a wide class of changing sign functions [14]. The evolution equation are:

$$\frac{du_{ij}}{dt} = u_{ij}\Phi_{ij} \qquad i \neq j, 1 \ldots, n. \tag{1}$$

$$\Phi_{ij} = u_{ij} - \Theta_{ij} \qquad i \neq j, 1 \ldots, n. \tag{2}$$

$$\Theta_{ij} = \sum_{i \neq j \in \Omega_{ij}} u_{ij}^2 \tag{3}$$

where u_{ij} are the link values and Ω_{ij} are appropriate subsets of the network links. The choice of the Ω subsets is crucial, since the development of different network structures critically depend on this choice.

3 Results

Our results are obtained by varying the number of elements (input and output) in the definition of the threshold Θ. In the case of a global threshold where all

the units interact each other we obtain a directed network in which one unit gets all the incoming (or outgoing) links (see Fig. 1). This is an extreme case of "unfair" link distribution. By weakening this assumption we are able to get intermediate cases in which few units get most of the links, with a "fat-tailed" connectivity degree distribution (see Fig. 3). If the competition is restricted

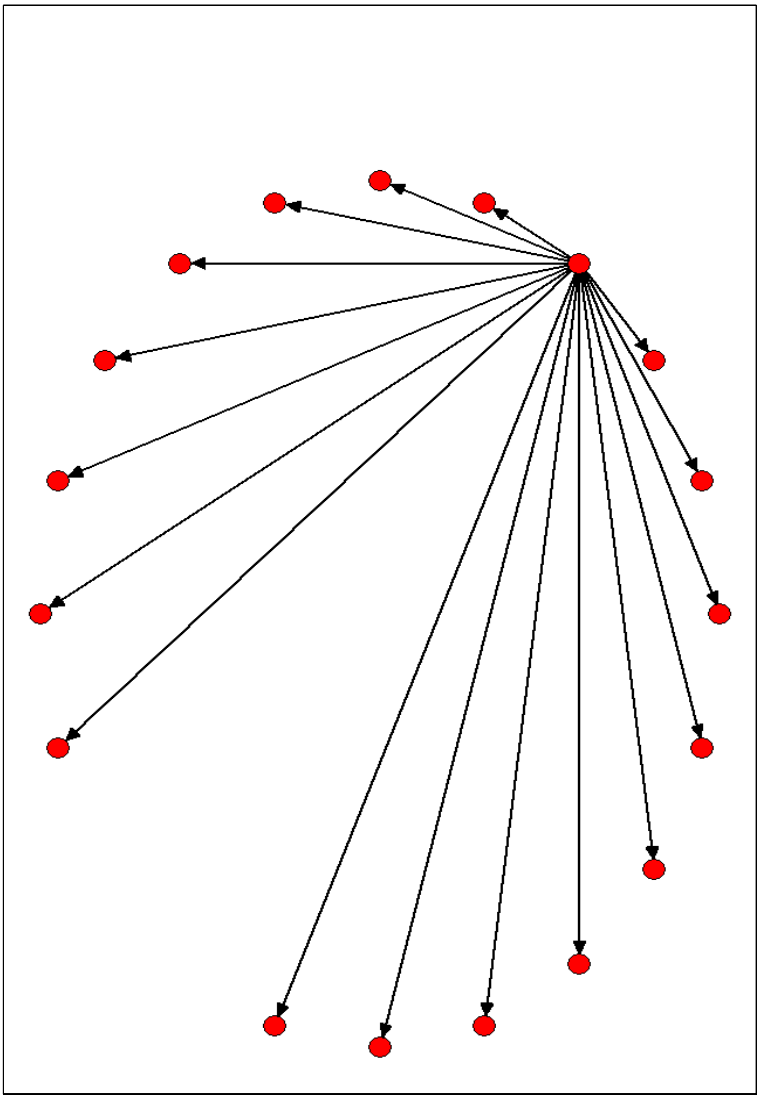


Fig. 1. Directed graph generated from the evolution rule with a threshold such that one unit wins all

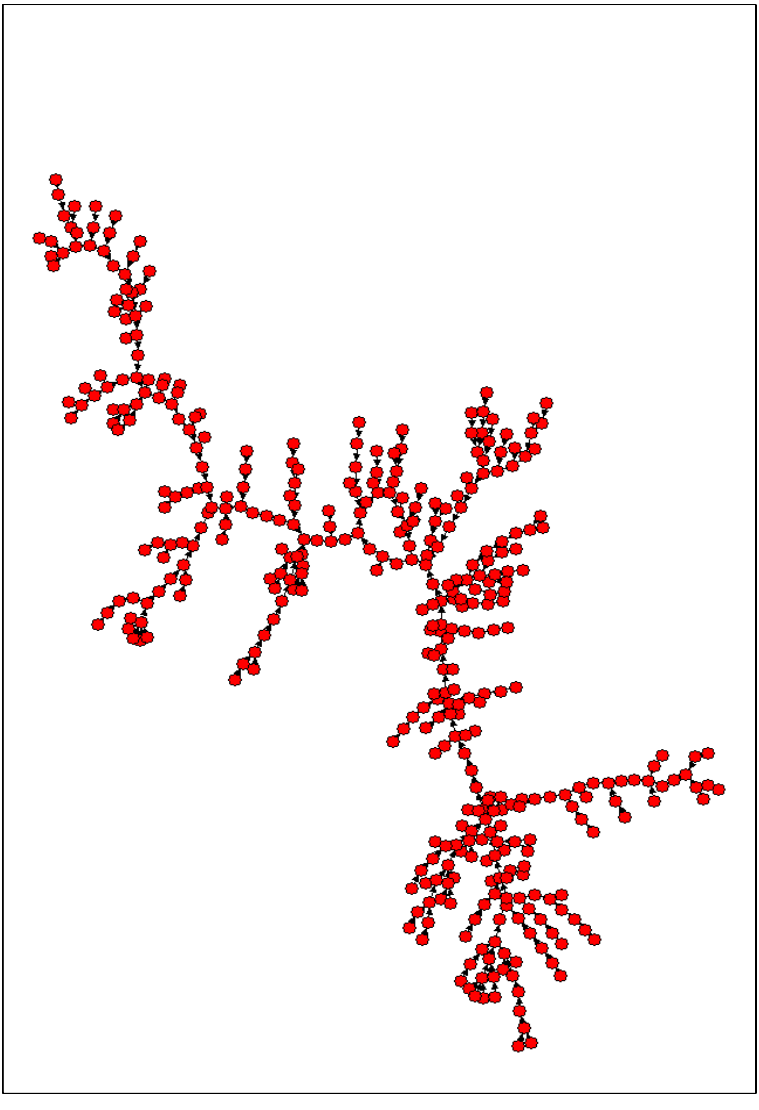


Fig. 2. Directed graph where each unit has only one incoming link and randomly distributed outgoing links

to all of the incoming (or outgoing) links for each unit, we obtain complete selectivity (only one directed link) for the input but leaving the output unconstrained, and viceversa (see Fig. 2). We remark that all these networks are different from a random generated network, because they embody multiple structures that can be further analyzed and experimentally tested in terms of biological interpreteation.

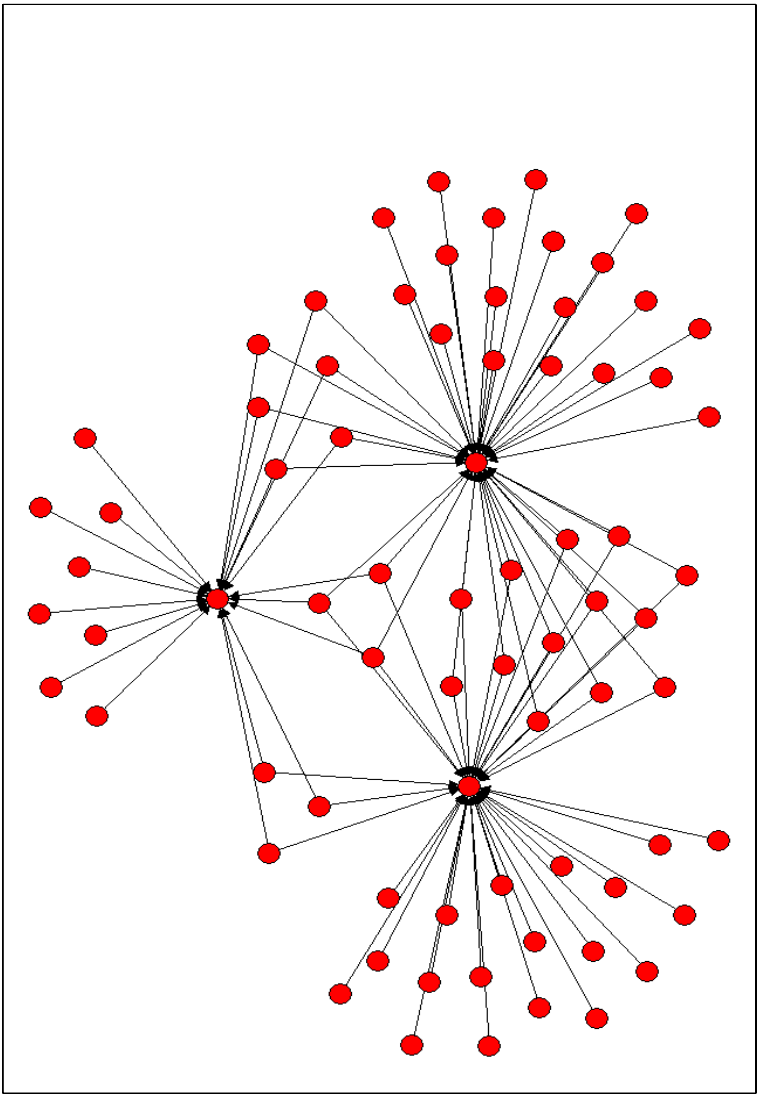


Fig. 3. Directed graph where the "one unit wins all" rule has been weakened

4 Conclusion

This analysis is one of the first attempts to build a network based on a learning rule that can produce different topologies. Our network is characterized by generic units that can be neurons, lymphocytes or generic cells. The learning rule has a direct interpretation in the development of network connectivity because the history dependent threshold has a biological counterpart in biological

phenomena such as memory induction in neurons and lymphocytes. Our numerical results show that we can tune the network topology, from random to scale-free-like connectivity degree distributions, by changing the links contained in the definition of the threshold Θ. We think that this is a first step towards more realistic neural and immune models taking into account the internal structure [18, 19] of cell communication and the role of signalling molecules in the induction of learning and memory.

Acknowledgments. This work has been supported by Italian FIRB grant, INFN FB11 grant and a Bologna University grant (ex 60 %).

References

1. J. D. Bui, S. Calbo, K. Hayden-Martinez, L. P. Kane, P. Gardner, and S. M. Hedrick. A role for CaMkII in T cell memory. *Cell*, 10 (2000):457–467.
2. E. L. Bienenstock, L. N. Cooper, and P. W. Munro. Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *J. Neurosci.*, 2:(1982)32–48.
3. G. C. Castellani, C. Giberti, C. Franceschi, and F. Bersani. Stable state analysis of an immune network model. *Int. J. Chaos Bif.*, 8(6):(1998) 1285–1301.
4. A. Bazzani, D. Remondini, N. Intrator, and G. C Castellani. The effect of noise on a class of energy-based learning rules. *Neural Comput.*, 15(7): (2003) 1621-40.
5. G. C Castellani, N. Intrator, H. Shouval, and L. Cooper. Solution of the bcm learning rule in a network of lateral interacting nonlinear neurons. *Network*, 10: (1999) 111–121.
6. A. Coutinho. Beyond clonal selection and network. *Immunol. Rev.*, 110:(1989) 63–87.
7. GC. Castellani, EM. Quinlan, , LN Cooper, and HZ. Shouval. A biophysical model of bidirectional synaptic plasticity: Dependence on ampa and nmda receptors. *PNAS*, 98(22): (2001) 12772–12777.
8. R. J. De Boer. Symmetric idiotypic networks: Connectance and switching stability and suppression. *Bull. Math. Biol.*, 55:(1993) 745–780.
9. Remondini D, Bazzani A, Franceschi C, Bersani F, Verondini E, Castellani G. Role of connectivity in immune and neural network models: memory development and aging. *Riv.Biol.*, 96(2):(2003) 225-39.
10. G. W. Hoffmann, T.A. Kion, R.B. Forsyth, K.G. Soga, and A. CooperWillis. The n-dimensional network. In A.S. Perelson, editor, *Theoretical Immunology, Part 2*, p. 291. Addison-Wesley, 1988.
11. K. Hayden-Martinez, L. P. Kane, and S. M. Hedrick. Effects of a constitutively active form of calcineurin on t cell activation and thymic selection. *J. Immunol.*, 165:(2000) 3713–3721.
12. J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proc Natl Acad Sci U S A.*, 79(8):(1982) 2554-8.
13. G. W. Hoffmann. A neural network model based on the analogy with the immune system. *J. Theor. Biol.*, 122:(1986) 33–67.
14. N. Intrator and L. N Cooper. Objective function formulation of the BCM theory of visual cortical plasticity: Statistical connections, stability conditions. *Neural Networks*, 5:(1992) 3–17.

15. N. K. Jerne. Towards a network theory of immune system. *Annu. Immunol.*, 125:(1974)373–389.
16. A.S. Perelson. Immune network theory. *Immunol. Rev.*, 110:(1989) 5–36.
17. A.S. Perelson. Mathematical approaches in immunology. In S. I. Andersson, editor, *Theory & Control of Dynamical Systems*, pp 200–230. World Scientific, 1992.
18. P. Tieri, S. Valensin, C. Franceschi, C. Morandi, G.C. Castellani. Memory and Selectivity in Evolving Scale-Free Immune Networks. In Artificial Immune Systems Springer-Verlag LECTURE NOTES IN COMPUTER SCIENCE, pp 93-101, 2003.
19. P. Tieri, S. Valensin, V. Latora, G.C. Castellani, M. Marchiori, D. Remondini, C. Franceschi: Quantifying the relevance of different mediators in the human immune cell network. *Bioinformatics*, 21:(2005) 1639-1643.
20. J. Stewart and F. J. Varela. Exploring the meaning of connectivity in the immune network. *Immunol. Rev.*, 110:(1989) 37–61.
21. D.J.Watts , S.H. Strogatz : Collective dynamics of "small-world" networks. *Nature*, 393 (1998) 440-442.
22. A.L. Barabasi, R. Albert : Statistical mechanics of complex networks, *Rev of Mod Phys*, 74, (2002) 48-94.

On Diversity and Artificial Immune Systems: Incorporating a Diversity Operator into aiNet

Paul S. Andrews[1] and Jon Timmis[2]

[1] Department of Computer Science, University of York, UK
psa@cs.york.ac.uk
[2] Departments of Electronics and Computer Science, University of York, UK
jtimmis@cs.york.ac.uk

Abstract. When constructing biologically inspired algorithms, important properties to consider are openness, diversity, interaction, structure and scale. In this paper, we focus on the property of diversity. Introducing diversity into biologically inspired paradigms is a key feature of their success. Within the field of Artificial Immune Systems, little attention has been paid to this issue. Typically, techniques of diversity introduction, such as simple random number generation, are employed with little or no consideration to the application area. Using function optimisation as a case study, we propose a simple immune inspired mutation operator that is tailored to the problem at hand. We incorporate this diversity operator into a well known immune inspired algorithm, aiNet. Through this approach, we show that it is possible to improve the search capability of aiNet on hard to locate optima. We further illustrate that by incorporating the same mutation operator into aiNet when applied to clustering, it is observed that performance is neither improved nor sacrificed.

1 Introduction

The development of algorithms in the field of Artificial Immune Systems (AIS) has grown rapidly over the past 10 years. It is only recently, however, that the research methodologies employed to design AIS have begun to be examined [1, 2, 3]. Highlighted by each of these methodologies, is the need to consider carefully the target applications and biological inspirations when designing AIS. By doing this, it is hoped that the designer will maximise their chance of producing an improved AIS. As a preliminary piece of work, we revisit an existing AIS called aiNet [4], and consider how it introduces diversity into its population of candidate solutions for the task of function optimisation. We then incorporate into aiNet, a diversity operator that has been tailored for function optimisation and inspired by an immune process called receptor editing. Function optimisation experiments are carried out to assess the performance of the algorithm with, and without, the diversity operator. The results show improved performance with the diversity operator. This process of adapting a bio-inspired algorithm to a particular problem, via the introduction of additional functionality such as the diversity operator outlined in this paper, is a common task within the evolutionary computation community. However, the adaptations made to improve

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 293–306, 2006.

performance in one problem domain may not be suitable when applied to a different domain. Thus, consideration must again be taken when attempting to re-apply specialised algorithm modifications for different purposes. To highlight this, we incorporate the diversity operator into a version of aiNet designed for data clustering. Experiments show that the performance of aiNet for clustering is not affected by the diversity operator for the chosen test case. This results is to be expected as the diversity operator, which was designed for a different problem area, was copied blindly without consideration for the clustering application.

2 Artificial Immune Systems

AIS have been defined by de Castro and Timmis [1] as being:

> "adaptive systems, inspired by theoretical immunology and observed immune functions, principle and models, which are applied to problem solving."

This very general definition reflects the fact that the term AIS is used to describe a collection of systems taking inspiration from different ideas and processes in immunology. Typically, however, most AIS have been inspired by the notion that the immune system protects the body from potentially harmful micro-organisms called pathogens, and by the immune mechanisms used to achieve this. One such mechanism utilises the production of molecules called antibodies (Ab) that can recognise other molecules, called antigens (Ag), via a process of chemical binding. An Ag can either be part of a pathogen or of the organism's own cells, known as self Ag. The recognition of self Ag by Ab in the body is considered undesirable, and so additional immune mechanisms exist ensuring only Ab that recognise non-self Ag are produced. In addition to taking inspiration from such immune mechanisms, AIS designers typically use the immunological terminology to describe the components of the system. Examples of AIS include those based on the selection of immune cells in thymus and the negative selection principle [5], the maturation of immune cells via the process of clonal selection [6], and the interaction of immune cells described by the immune network theory [4]. A comprehensive overview of AIS and immunology is beyond the scope of this paper and the interested reader is referred to [1] and [7] for more details.

2.1 Designing AIS

As the field of AIS has matured, a better understanding of how they work is currently being sought. From these investigations, a number of ideas have arisen concerning how AIS should be designed. de Castro and Timmis [1] introduced a layered framework for engineering AIS that identifies three basic system design elements: a representation for the components of the system, a set of mechanisms (affinity measures) to evaluate the interactions of individuals within the environment and with each other, and procedures for adaptation. Additionally, Freitas and Timmis [2] highlighted the need to consider the problem for which the AIS

is being designed, and note that the selection of a representation scheme and evaluation functions in AIS can bias the results. Consequently, when designing a suitable AIS, it needs to be tailored to the problem at hand. Similarly, Hart and Ross [8] investigated the effects of different affinity measurement mechanisms with respect to immune networks and highlight the importance of getting the shape-space and matching rule right when designing AIS. Recently, Stepney et al. [3] proposed that bio-inspired algorithms, such as AIS, are best developed within a conceptual framework. This framework takes an interdisciplinary approach to the problem, and involves designing AIS through a series of observational and modelling stages in order to identify the key characteristics of the immunological process on which the AIS will be based. The same authors also highlight five areas for consideration when designing bio-inspired algorithms that can affect their behaviour: openness, diversity, interactions, structure and scale.

Common to all of the works above, is the requirement on the designer to consider both the biology on which the AIS is based, and the AIS application area. By doing this, it is hoped that the designer will take a more principled approach to algorithm design, leading to a better suited and performing algorithm. In this paper, we have chosen to revisit the design of an existing AIS, the Artificial Immune NETwork (aiNet) that was first introduced by de Castro and Von Zuben [4] in 2000. The design of this algorithm follows the ideas later presented as the layered framework for engineering AIS [1]. By taking into account additional AIS design ideas highlighted above, we investigate mechanisms for introducing population diversity in aiNet for the specific task of function optimisation, and take inspiration from diversity mechanisms in the human immune system.

2.2 aiNet

aiNet was originally designed for the task of data clustering [4], but has also been adapted for multi-modal function optimisation [9]. The algorithm is an example of a discrete immune network model in which network elements, called Ab, adapt to match a population of input elements, called Ag, that are to be recognised. Typically, the Ab and Ag elements are represented as D-dimensional vectors of real numbers, and are matched for similarity using a distance metric such as Euclidean distance. aiNet proceeds by an iterative procedure where each member of the Ag population is presented to each Ab in turn. Members of the Ab population then adapt to better match the Ag according to a mechanism inspired by Burnet's clonal selection principle. Once the Ab have adapted to the Ag, the same distance metric is used to calculate the similarity between each of the Ab. Ab that are similar to each other are eliminated due to a pre-defined threshold. It is this interaction between the Ab elements that forms the immune network, which can produce a dynamic behaviour of its own without the need for Ag interactions. Diversity is introduced into the population of Ab at the end of each iteration by the addition of a number of new Ab that have been randomly generated to a legal value in the problem's search space. de Castro and Timmis [1] provide the following pseudocode for a generalised version of the aiNet algorithm:

1. *Initialisation*: create an initial random population of network Ab;
2. *Antigenic Presentation*: for each Ag, do:
   (a) *Clonal Selection and Expansion*: for each Ab, determine its affinity with the Ag presented. Select a number of high affinity Ab and reproduce (clone) them proportionally to their affinity;
   (b) *Affinity Maturation*: mutate each clone inversely proportional to affinity. Re-select a number of the highest affinity clones and place them into a clonal memory set;
   (c) *Metadynamics*: eliminate all memory clones whose affinity with the Ag is less than a pre-defined threshold;
   (d) *Clonal Interactions*: determine the network interactions (affinity) among all the Ab of the clonal memory set;
   (e) *Clonal Suppression*: eliminate those memory clones whose affinity with each other is less than a pre-specified threshold;
   (f) *Network Construction*: incorporate the remaining clones of the clonal memory with all network Ab;
3. *Network Interactions*: determine the similarity between each pair of network Ab;
4. *Network Suppression*: eliminate all network Ab whose affinity is less than a pre-specified threshold;
5. *Diversity*: introduce a number of new randomly generated Ab into the network;
6. *Cycle*: repeat Steps 2 to 5 until a stopping condition is reached.

The version of aiNet for data clustering (clust-aiNet) is presented in [10], and follows the above pseudocode. The function optimisation aiNet version (opt-aiNet) is presented in [9], and differs from the pseudocode in a number of ways. In opt-aiNet, there is no distinct Ag population for the Ab to interact with, but an optimisation function instead, and therefore the Ab will represent candidate solutions to the optimisation function. As there is no Ag population, step 2 of the pseudocode above is updated to perform a localised search whereby all Ab are evaluated against the optimisation function, and are cloned proportionally to their fitness. This local search stops when the average Ag population fitness doesn't change significantly from one iteration to the next dependant on a threshold parameter.

3 Diversity

As highlighted in step 5 of the aiNet pseudocode above, diversity is introduced into the Ab population by randomly generating Ab vectors over the possible search space. This process is similar to the main diversity mechanism according to the clonal selection principle from which aiNet takes inspiration. Within the immune system, however, a mechanism called receptor editing exists that introduces diversity to the Ab population in a slightly different way.

3.1 Receptor Editing

In the human immune system, Ab are Y-shaped molecules formed of 2 linked molecular chains, called the light chain and the heavy chain. Each of these chains consists of a constant region, similar for all Ab, and a variable region, which is unique to each Ab. This variable region is the primary site where binding, and hence recognition, of Ag occurs. So in order for the immune system to be able to recognise unseen Ag, a large number of random Ab are initially generated. This generation process uses gene libraries and DNA rearrangement to produce a vast number of different Ab with unique variable regions. Any of these Ab that react with self Ag are then rejected, leaving a population of Ab that can go through an adaptation process of cloning and mutation of their variable regions to produce Ab that better match an Ag. The receptor editing mechanism can take Ab that were initially rejected for recognising self Ag, and apply a process whereby the variable region of the light chain of the Ab will re-arrange. The variable region of the heavy chain, however, stays the same, therefore the new Ab receptor is different to the original one, whilst retaining part of the old structure that is known to be able to recognise Ag [7].

In a computational sense, we can view the receptor editing process as re-using information that is known to be beneficial. Comparing this to the generation of Ab at random, we see that receptor editing can bias the occurrence of a new solution to an area of the search space where beneficial results are known to exist. Based on this observation, we extrapolate from this idea and develop an operator that re-uses the information stored within an existing Ab, whilst adding a biased variability.

3.2 A Diversity Operator

As our application is function optimisation, we investigated the sort of diversity operator that could introduce new Ab individuals into aiNet that beneficially bias the search, thus improve performance. By inspecting the results achieved with the standard opt-aiNet algorithm, it was apparent that the optima on the boundaries of the function are the most difficult to find, therefore our diversity operator was designed so that it would increase the chance of locating these optima. This property is present in a non-uniform mutation operator presented by Michalewicz in [11] for use in genetic algorithms. The operator is called non-uniform as it works dependant on the current algorithm iteration, producing less variability the higher the iteration number, thus allowing better local exploration in the later stages of the algorithm. The operator is described by equations 1 and 2 below:

$$x' = \begin{cases} x + \Delta(t, u - x) & \text{if a random digit is 0} \\ x - \Delta(t, x - l) & \text{otherwise} \end{cases} \tag{1}$$

$$\Delta(t, y) = y \left(1 - r^{\left(1 - \frac{t}{T}\right)^b}\right) \tag{2}$$

where x is a real number between a lower bound, l, and an upper bound, u, t is the iteration number, T is the maximum number of iterations, b is a parameter

determining the degree of non-uniformity, and r is a uniformly generated random number in the range [0,1]. The iteration dependent nature of this operator is not advantageous to a diversity operator in an AIS as it is being used for a different purpose. Mutation and local exploration are carried out by the 2nd step in the aiNet pseudocode presented above, and so we do not wish the diversity of new Ab to be restricted at the end of the algorithm's operation. So, by removing the iteration dependant nature of function $\Delta(t, y)$ we are left with the following uniform operator described by equations 3 and 4:

$$x' = \begin{cases} x + \Delta(u - x) & \text{if a random digit is 0} \\ x - \Delta(x - l) & \text{otherwise} \end{cases} \tag{3}$$

$$\Delta(y) = ry \tag{4}$$

This operator has the advantage of biasing the generated number towards the boundary of the values of x. This is illustrated by comparing Figure 1(a) with Figure 1(b), which show the distribution of values generated with this operator for $x = 0$ and $x = 7$ respectively, and $l = -10$ and $u = 10$. Here, as 0 is in the middle of the range of values x can be in, there is an equal probability for any value between -10 and 10 occurring. With $x = 7$, however, the chance of moving between 7 and 10 is a lot higher than between -10 and 7.

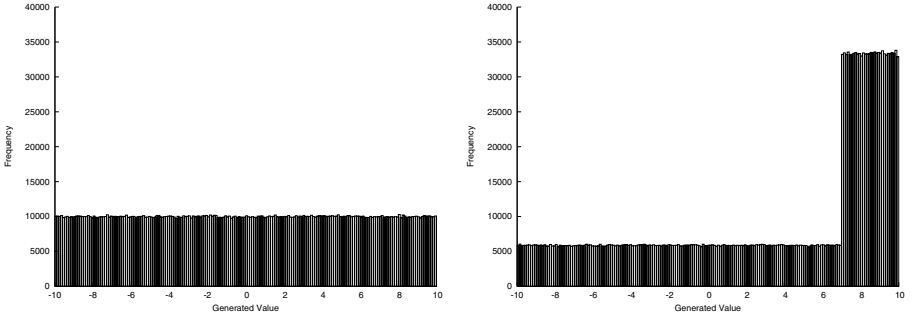
We have taken this operator and adjusted it further taking inspiration from the receptor editing process. The new operator is given by equations 5 and 6:

$$x' = \begin{cases} x + \Delta(u - x) & \text{if a random digit is 0} \\ x - \Delta(x - l) & \text{otherwise} \end{cases} \tag{5}$$

$$\Delta(y) = \begin{cases} ry & \text{if } y < p(u - l) \\ rp(u - l) & \text{otherwise} \end{cases} \tag{6}$$

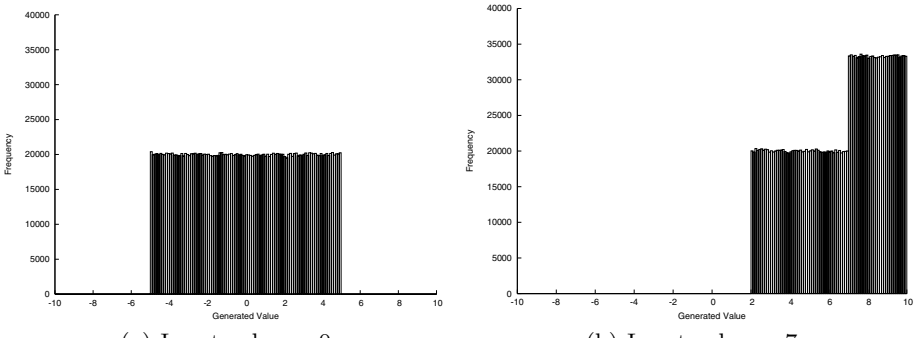
where p is a parameter that sets the range of values x' can be to a proportion of $[l,u]$. To illustrate this, Figure 2(a) shows the distribution of values generated when $x = 0$, $l = -10$, $u = 10$ and $p = 0.25$. Here we see that the new value will have an equal probability of falling between -5 and 5, but cannot be further away. Figure 2(b) shows the distribution of values generated when $x = 7$, demonstrating that the new operator still biases the search towards the boundaries of x when it is nearby. It is noted that this operator is only suitable on static functions where the boundaries of the function dimensions are known.

This diversity operator, described in equations 5 and 6, was incorporated into aiNet at the diversity introduction stage of the algorithm (stage 5 in the pseudocode above). The new diversity mechanism works by randomly selecting existing Ab vectors from the Ab network, cloning them, and then subjecting each dimension of the clone to the diversity operator. A user defined parameter is used to set the ratio of new Ab vectors to be introduced by the diversity operator and the original random introduction respectively.



(a) Input value = 0 (b) Input value = 7

Fig. 1. Histograms showing frequency of generated values using the uniform Michaelwicz operator



(a) Input value = 0 (b) Input value = 7

Fig. 2. Histograms showing frequency of generated values using the diversity operator

4 Function Optimisation

To test our approach at designing a diversity operator for opt-aiNet, the modified algorithm was applied to two function optimisation problems using the following three different ratios of diversity operator and random Ab introduction:

1. 100% Random Ab
2. 50% Random Ab, 50% Diversity Operator
3. 100% Diversity Operator

In the following results, we refer to these as versions 1, 2 and 3 of the algorithm respectively. For all experiments, the parameter p is set to an empirically determined value of 0.25, and all other algorithm parameters remain constant. Note that version 1 is identical to the original aiNet algorithm, and is used for comparison purposes to ascertain the effectiveness of the diversity operator.

When applied to a multi-modal function optimisation problem, opt-aiNet is capable of finding not only the global optimum, but a large number of the local

optima. Therefore, to evaluate the performance of opt-aiNet, we measure the number of optima found, and the speed at which it finds them. In accordance with [9] and [12], we calculate the number of optima found by recording the number of Ab in the final network. However, we differ from [9] and [12] by measuring the algorithm's performance by recording the number of times the objective function is evaluated during a run, rather than algorithm iterations. For the following experiments, our hypothesis is that we will be able to better locate the hard to find optima using the diversity operator when compared to the original random Ab introduction, as we are biasing the search in areas of the search space where opt-aiNet appears to struggle.

4.1 Experiments

We experimented with two 2-dimensional function optimisation problems, thus the Ab vectors consist of 2 values, x and y. This means that the fitness landscape of the functions can be represented on a 3 dimensional plot. The first optimisation function, used in accordance with [9], is given by equation 7:

$$f(x, y) = x.sin(4\pi x) - y.sin(4\pi y + \pi) + 10 \tag{7}$$

where x and y are in the range [-2,2]. A graphical depiction of this function is given in Figure 3, which shows that there are 100 maxima in this function.

For the second optimisation function, we have adapted $f(x, y)$ to introduce an irregular fitness landscape with a large area containing no local maxima. This type of irregular fitness landscape should provide a different challenge for the opt-aiNet algorithm. This function, shown graphically in Figure 4, is given by:

$$g(x, y) = \begin{cases} f(x, y) & \text{if } x < 0.5 \\ -x^2 + 7 & \text{otherwise} \end{cases} \tag{8}$$

where x and y are in the range [-2,2]. There are 70 maxima for this function.

Two experiments were performed with functions $f(x, y)$ and $g(x, y)$ using the 3 versions of opt-aiNet mentioned above. The first experiment used the algorithm

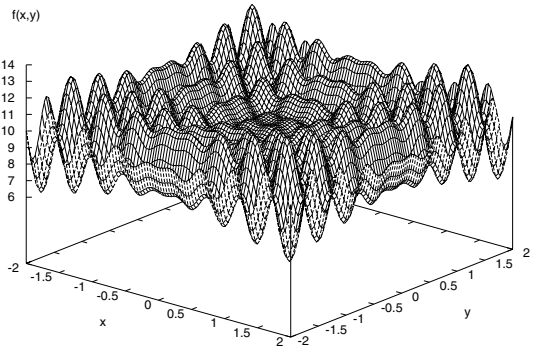


Fig. 3. Regular multi-modal function $f(x, y)$

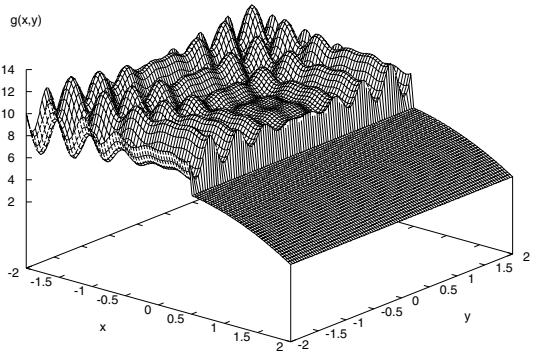


Fig. 4. Irregular multi-modal function $g(x, y)$

stopping condition described in [9], where the algorithm stops when the size of the Ab network does not change between iterations. This experiment was run 100 times for each algorithm version, and the mean, median and range (lowest and highest) values for the number of maxima found and function evaluations performed were recorded. The second experiment sets the stopping condition to a fixed number of maxima, so the algorithm only stops when this number of maxima have been found. For the function $f(x, y)$, 100 algorithm runs were made with the stopping conditions 90 maxima and 100 maxima and for $g(x, y)$, 100 runs were made with the stopping condition set to 70 maxima. Again the mean, median and range values were recorded.

4.2 Results

The results for the first experiment with $f(x, y)$ are shown in Table 1. These show that the more the diversity operator is used, the more maxima the algorithm will find, however, this is at the expense of function evaluations, with the number of evaluations increasing the more the diversity operator is used. Dividing the mean number of function evaluations by the mean number of maxima found for each algorithm version, we see that the number of evaluations required to find each maximum works out at around the same value ($\approx$ 4600 evaluations). This

Table 1. Results showing number of maxima and function evaluations for 100 runs of opt-aiNet using the original stopping condition when applied to $f(x, y)$

Algorithm Version	Maxima Mean	Maxima Median	Maxima Range	Evals Mean	Evals Median	Evals Range
1	87.68	89	[68 : 98]	400131	393019	[195932 : 663058]
2	91.13	92	[76 : 100]	416715	405020	[198396 : 649462]
3	92.12	94	[65 : 99]	428251	437162	[151635 : 689975]

Table 2. Results showing number of maxima and function evaluations for 100 runs of opt-aiNet using the original stopping condition when applied to $g(x, y)$

Algorithm Version	Maxima Mean	Maxima Median	Maxima Range	Evals Mean	Evals Median	Evals Range
1	51.86	55	[12 : 65]	435999	449537	[26136 : 812966]
2	55.83	58	[10 : 68]	378820	382250	[21032 : 767019]
3	58.59	62	[8 : 70]	316343	321035	[11792 : 625460]

Table 3. Results showing number of evaluations for 100 runs of opt-aiNet to find 90 maxima when applied to $f(x, y)$

Algorithm Version	Evals Mean	Evals Median	Evals Range
1	356807	346148	[245333 : 578523]
2	364084	348931	[261624 : 576818]
3	350657	343552	[235554 : 576444]

Table 4. Results showing number of evaluations for 100 runs of opt-aiNet to find 100 maxima when applied to $f(x, y)$

Algorithm Version	Evals Mean	Evals Median	Evals Range
1	2215077	1947781	[897314 : 9415846]
2	1344824	1154989	[660264 : 3911380]
3	966430	910349	[449955 : 2279970]

Table 5. Results showing number of evaluations for 100 runs of opt-aiNet to find 70 maxima when applied to $g(x, y)$

Algorithm Version	Evals Mean	Evals Median	Evals Range
1	1675100	1642025	[677820 : 2761473]
2	924242	888019	[488257 : 1823107]
3	643913	612656	[307362 : 1122979]

shows that the main effect of the diversity operator for this experiment is to delay the termination of opt-aiNet, allowing more maxima to be found but at cost the of function evaluations. Looking at the range values for the number of maxima and evaluations, we see that there is a large disparity between the lowest and the highest number, and thus the performance of the algorithm is widely varying. This observation agrees with observations of [12]. The results for the first experiment with $g(x, y)$ shown in Table 2, enhance and exaggerate those seen with $f(x, y)$. Again, we see that the the more the diversity operator is used, the

more maxima are found. However, in contrast to the $f(x, y)$ results, it is observed that the number of function evaluations required to find these maxima reduces the more the diversity operator used, making the average number of evaluations to find each maxima smaller. This function also enhances the unpredictable nature of when the algorithm will stop, with the difference between the lowest and highest number of maxima found being very large for all algorithm versions.

The results for the second experiment with $f(x, y)$ in Tables 3 and 4 and show conclusively where the benefit of the diversity operator lies. When the stopping condition is set at 90 maxima, all three algorithm versions perform virtually identically. However, when the algorithm is required to find all 100 maxima, the more the diversity operator is used, the lower the number of function evaluations required, with algorithm version 1 requiring over twice as many function evaluations on average than algorithm version 3. Upon visual inspections of the maxima found using the 90 maxima stopping condition, it was seen that the 10 maxima not found were located on the boundary of the search space. From this we conclude that the diversity operator enhances the ability of opt-aiNet to find these difficult maxima, thus showing our approach at designing a biased mutation operator was effective. The results of the second experiment to find all 70 maxima in $g(x, y)$ are shown in Table 5. Again, this shows clearly that the advantage of the diversity operator, with algorithm version 1 requiring over two and half times more function evaluations than algorithm version 3 to find all the maxima.

From these experiments, we can see that the diversity operator gives us more benefit on the function $g(x, y)$ than $f(x, y)$. This is due to the large area of the function $g(x, y)$ that contains no local maxima, so there is a far smaller chance of introducing a new Ab into this area with the diversity operator than there is with random introduction. We can also see that our hypothesis that the diversity operator will improve the performance of opt-aiNet was shown to be correct for these two functions.

5 Data Clustering

Having established the advantages of the diversity operator for function evaluation, we blindly incorporated it into clust-aiNet and applied it to a data clustering problem to observe its performance in a problem domain it was not explicitly designed for. The task of clustering data differs in many ways to the task of function optimisation, thus the experiments and performance measures suitable for clust-aiNet also differ. clust-aiNet is an unsupervised learning algorithm that performs a data compression to produce a map of the data representing the centres of data clusters. After running the algorithm, the final Ab network represents an internal image of the data set to which it was exposed [10]. Thus to determine whether this internal image is a representative solution, de Castro and Von Zuben [10] analyse their solutions using a number of statistical and graph theory techniques. We feel that, although valid, this approach is beyond the scope of this paper, so we make an assumption that the final output of

our experiments with clust-aiNet are all equally valid. We have verified this assumption by visually inspecting a subset of the output of our experiments. This assumption means that we can focus on assessing whether the diversity operator improves the performance of clust-aiNet by measuring speed of convergence of the different algorithm versions.

5.1 Experiments

For the clust-aiNet experiments, the same three algorithm versions were used with p set to 0.25 and all other parameters remaining constant. The task was to cluster a set of randomly generated data shown in Figure 5. This data consists of six clusters of fifty 3-dimensional data items, where each dimension falls in the range [-10,10]. Each of the 3 versions of clust-aiNet was run 100 times, with the algorithm stopping after 50 iterations. During each run, the number of Ab in the Ab network was recorded at each iteration. The mean number of Ab for each iteration was then calculated for the 3 versions of clust-aiNet.

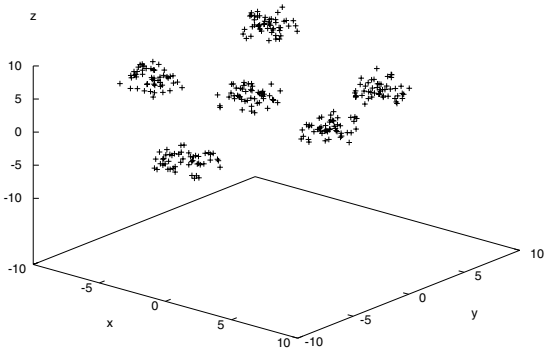


Fig. 5. Randomly generated clustered data points

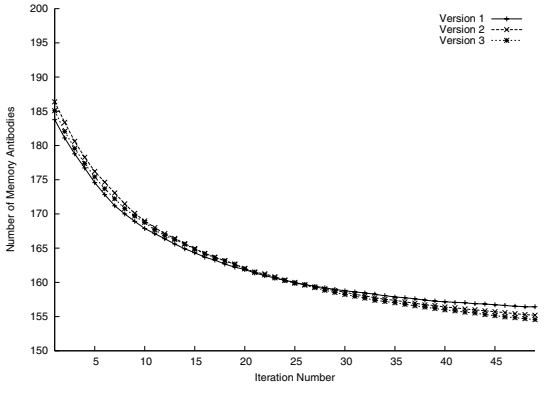


Fig. 6. Results showing mean number of memory Ab per iteration for 100 runs with each version of clust-aiNet when applied to data clustering

5.2 Results

The results of the experiment with clust-aiNet shown in Figure 6, show virtually no difference in the speed of convergence of the algorithm for all three versions of the algorithm. Although the experiments with clust-aiNet were not extensive, these results show that the diversity operator neither enhances nor harms its performance. This is not surprising, however, as the operator was designed to improve the performance of function optimisation, not data clustering.

6 Conclusion

There are a number of general conclusions we can draw from our results. Our original intention was to take a structured approach at designing a diversity operator to improve the performance of aiNet for the task of function optimisation. The results with opt-aiNet clearly show that our diversity operator does this, and so we conclude that our design approach was successful. By incorporating the diversity operator into clust-aiNet and using it for a purpose it was not designed for, we were able to show that the bias involved in the design of the diversity operator was not appropriate in improving performance for this problem domain. This type of result is to be expected and backs up the suggestion of taking a problem oriented approach [2] in the design of AIS for a specific engineering problems. We have also been successful in showing that the way diversity is introduced into AIS does affect the performance of the algorithm, as highlighted by Stepney et al. [3]. As a final comment, we suggest that if the fitness landscape of the problem to which you are applying an AIS is unknown, an appropriate method for introducing diversity is to use a mixture of randomly introduced Ab and an alternative diversity mechanism that tries to bias the newly introduced individuals dependant on the nature of the problem.

References

1. de Castro, L.N., Timmis, J.: Artificial Immune Systems: A New Computational Intelligence Approach. Springer-Verlag (2002)
2. Freitas, A.A., Timmis, J.: Revisiting the foundations of artificial immune systems: A problem-oriented perspective. In: Proceedings of the 2nd International Conference on Artificial Immune Systems (ICARIS 2003), Springer (2003) 229–241
3. Stepney, S., Smith, R.E., Timmis, J., Tyrell, A.M.: Towards a conceptual framework for artificial immune systems. In: Proceeding of the 3rd International Conference on Artificial Immune Systems (ICARIS 2004), Springer (2004) 53–64
4. de Castro, L.N., Von Zuben, F.J.: An evolutionary immune network for data clustering. In: Proceeding of the IEEE Brazilian Symposium on Artificial Neural Networks. (2000) 84–89
5. Forrest, S., Perelson, A., Allen, L., Cherukuri, R.: Self-nonself discrimination in a computer. In: Proceedings of the IEEE Symposium on Research in Security and Privacy. (1994) 202–212

6. de Castro, L.N., Von Zuben, F.J.: Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation **6** (2002) 239–251
7. Janeway, C.A., Travers, P., Walport, M., Shlomchik, M.: Immunobiology: The Immune System in Health and Disease. 5th edn. Garland Publishing (2001)
8. Hart, E., Ross, P.: Studies on the implications of shape-space models for idiotypic networks. In: Proceedings of the 3rd International Conference on Artificial Immune Systems (ICARIS 2004), Springer (2004) 413–426
9. de Castro, L.N., Timmis, J.: An artificial immune network for multimodal function optimization. In: 2002 Congress on Evolutionary Computation. (2002) 699–704
10. de Castro, L.N., Von Zuben, F.J.: ainet: An artificial immune network for data analysis. In Abbass, H.A., Sarker, R.A., Newton, C.S., eds.: Data Mining: A Heuristic Approach. Idea Group Publishing (2002) 231–259
11. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag (1996)
12. Timmis, J., Edmonds, C.: A comment on opt-ainet: An immune network algorithm for optimisation. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2004). (2004) 308–317

Lipschitzian Pattern Search and Immunological Algorithm with Quasi-Newton Method for the Protein Folding Problem: An Innovative Multistage Approach

A.M. Anile[1], V. Cutello[1], G. Narzisi[1], G. Nicosia[1], and S. Spinella[2]

[1] Department of Mathematics and Computer Science, University of Catania,
V.le A. Doria 6, 95125 Catania, Italy
{vctl, narzisi, nicosia}@dmi.unict.it
[2] University of Calabria,
Ponte P. Bucci 17/B, 87036 Arcavata di Rende (CS), Italy
sspinella@unical.it

Abstract. In this work we show an innovative approach to the protein folding problem based on an hybrid Immune Algorithm (IA) and a quasi-Newton method starting from a population of promising protein conformations created by the global optimizer DIRECT. The new method has been tested on Met-Enkephelin peptide, which is a paradigmatic example of *multiple-minima problem*, 1POLY, 1ROP and the three helix protein 1BDC. The experimental results show as the multistage approach is a competitive and effective search method in the conformational search space of real proteins, in terms of quality solution and computational cost comparing the results of the current state-of-art algorithms.

Keywords: Deterministic Search, DIRECT, Immune Algorithms, Clonal Selection Algorithms, Quasi-Newton method, Hybrid methods, Protein Folding, Protein Structure Prediction, Bioinformatics.

1 Introduction

Real proteins quickly fold into a complicated three-dimensional (3D) structure. Evolutionary algorithms have been used to predict the native structure with the lowest energy conformation of a given primary sequence. Proteins are known to have many important functions in the cell, such as enzymatic activity, storage and transport of material, signal transduction, antibodies and more. The amino acids composition of a protein will usually uniquely determine its 3D structure, to which the protein's functionality is directly related.

Successful structure prediction requires a free energy function sufficiently close to the true potential for the native state, as well as a method for exploring conformational space. Protein structure prediction is a challenging problem because current potential functions have limited accuracy and the conformational space is vast. Several algorithmic approaches have been applied to the Protein Folding (PF) problem in the last 50 years: molecular dynamics [Lev83], Monte

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 307–323, 2006.

Carlo methods [SKHB97], simulated tempering [HO97], evolutionary algorithms [BE94]. In spite of all these efforts, Protein Folding remains a challenging and computationally open problem.

Potential energy function. In order to evaluate the structure of a molecule we use energy functions. It would be "nice" to use quantum mechanics, but it is too computationally complex to be practical to model large systems (long proteins). So, we use classical physics to come up with energy functions. Sometimes called potential energy functions or force fields, these functions return a value for the energy based on the conformation of the molecule. They provide information on what conformations of the molecule are better or worse. The lower the energy value, then the better should be the conformation. Most typical energy functions have the form:

$$E(R) = \sum_{bonds} B(\boldsymbol{R}) + \sum_{angles} A(\boldsymbol{R}) + \sum_{torsions} T(\boldsymbol{R}) + \sum_{non-bonded} N(\boldsymbol{R}) \tag{1}$$

where $\boldsymbol{R}$ is the vector representing the conformation of the molecule, typically in cartesian coordinates or in torsion angles.

The literature on cost functions (often called energy functions) is enormous. In this work, in order to evaluate the conformation of a protein, we use the CHARMM (version 27) energy function. CHARMM (Chemistry at HARvard Macromolecular Mechanics) is a popular all-atom force field used mainly for the study of macromolecules. It is a composite sum of several molecular mechanics equations: stretching, bending, torsion, Urey-Bradley, impropers, van-der-Walls, electrostatics. The CHARMM [FM:00] energy function has the form:

$$\begin{aligned} E_{charmm} = & \sum_{bonds} k_b(b-b_0)^2 + \sum_{UB} k_{UB}(S-S_0)^2 + \sum_{angles} k_\theta(\theta-\theta_0)^2 \\ & + \sum_{torsions} k_\chi[1+\cos(n\chi-\delta)] + \sum_{impropers} k_{imp}(\phi-\phi_0)^2 \\ & + \sum_{nonbond} \varepsilon_{ij}\left[\left(\frac{Rmin_{ij}}{r_{ij}}\right)^{12} - \left(\frac{Rmin_{ij}}{r_{ij}}\right)^{6}\right] + \frac{q_i q_j}{e r_{ij}} \end{aligned} \tag{2}$$

where, in order, b is the bond length, b_0 is the bond equilibrium distance and k_b is the bond force constant; S is the distance between two atoms separated by two covalent bonds (1,3 distance), S_0 is the equilibrium distance and k_{UB} is the Urey Bradley force constant; θ is the valence angle, θ_0 is the equilibrium angle and K_θ is the valence angle force constant; χ is the dihedral or torsion angle, k_χ is the dihedral force constant, n is the multiplicity and δ is the phase angle; ϕ is the improper angle, ϕ_0 is the equilibrium improper angle and k_{imp} is the improper force constant; ε_{ij} is the Lennard Jones (LJ) well depth, r_{ij} is the distance between atoms i and j, $Rmin_{ij}$ is the minimum interaction radius, q_i is the partial atomic charges and e is the dielectric constant. Typically, ε_i and $Rmin_i$ are obtained for individual atom types and then combined to yield ε_{ij} and $Rmin_{ij}$ for the interacting atoms via combining rules. In CHARMM, ε_{ij} values are obtained via the geometric mean $\varepsilon_{ij} = \sqrt{\varepsilon_i \varepsilon_j}$, and $Rmin_{ij}$ via the arithmetic mean, $Rmin_{ij} = (Rmin_i + Rmin_j)/2$.

Finally the energy function CHARMM (equation 2) represents our minimization *objective*, the torsion angles of the protein are the *decision variables* of the optimization problem, and the constraint regions are the *variable bounds*. To evaluate the CHARMM energy function we use routines from TINKER[1] Molecular Modeling Package [HSP99]. First, the protein structure in internal coordinates (torsion angles) is transformed in cartesian coordinates using the PROTEIN routine. Then the conformation is evaluated using the ANALYZE routine, that gives back the CHARMM energy potential of a given protein structure.

The metrics: DME and RMSD. To evaluate how similar is the predicted conformation to the native one, we employ two frequently used metrics: Root Mean Square Deviation (RMSD) and Distance Matrix Error (DME). RMSD is calculated by the formula:

$$RMSD(a,b) = \sqrt{\frac{\sum_{i=1}^{n} |r_{ai} - r_{bi}|^2}{n}} \tag{3}$$

where r_{ai} and r_{bi} are the positions of atom i of structure a and structure b, respectively, and where structures a and b have been optimally superimposed. Fitting was performed using the McLachlan algorithm [M:82] as implemented in the program ProFit[2].

DME is calculated by the formula:

$$DME(a,b) = \frac{\sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} (|r_{ai} - r_{aj}| - |r_{bi} - r_{bj}|)^2}}{n} \tag{4}$$

which does not require the superposition of coordinates. For a particular pair of structures, the RMSD, which measures the similarity of atomic positions, is usually larger than DME, which measures the similarity of inter-atomic distances.

2 DIRECT Global Search

The DIRECT method is a global search algorithm for bound constrained optimization based on Lipschitz constant estimation [JPS93]. It is classified as a *pattern search method*. The method is based on three operations:

1. Lipschitz constant estimation,
2. choice for potential optimality of domain subregions, and
3. domain subdivision.

The method focuses on the ratio between *local* exploration and *global* exploration. The estimation of Lipschitz constant is interpreted as a scheme to assign the suitable priorities to a set of rectangles which are a partition of the search domain. This priority is used to plan a stratified sampling in order to

[1] http://dasher.wustl.edu/tinker/
[2] http://www.bioinf.org.uk/software/profit/

balance exploration and convergence, which gives rise to fast convergence with a broad domain coverage. Moreover, unlike some other methods, Lipschitz global optimization requires only few parameters.

The choice for potential optimality is the following:

Definition 1 ***[Potentially optimal rectangle].*** *Given f objective function, let $\mathcal{S}$ be the set of hyperrectangles generated by the algorithm after k iterations, and let $fmin$ the current best function value. An hyperrectangle $\tilde{R} \in \mathcal{S}$ with center $\mathbf{c}_{\tilde{R}}$ and measure $\alpha(\tilde{R})$ is said potentially optimal if there exists a Lipschitz constant $K > 0$ such that*

$$f(\mathbf{c}_{\tilde{R}}) - K\alpha(\tilde{R}) \leq f(\mathbf{c}_R) - K\alpha(R) \quad \forall R \in \mathcal{S} \tag{5}$$

$$f(\mathbf{c}_{\tilde{R}}) - K\alpha(\tilde{R}) \leq fmin - \varepsilon|fmin|. \tag{6}$$

where $\varepsilon \sim 10^{-4}$ is a constant to control the clustering *during the search [JPS93].*

The above definition gives a heuristic rule to choose hyperrectangles which are potentially optimal in the sense of theirs increasing efficiency of the objective function. Equation (6) can be interpreted as controlling the clustering nearby the optimal points.

The algorithm can be structured in a main procedure (Algorithm 1) with a main loop which contains: the selection of potentially optimal rectangles, the patterned sampling, the subdivision (Algorithm 2) of these rectangle and the updating variable statements.

Algorithm 1. DIRECT pseudocode

Require: Set of rectangles S
1: $n \leftarrow 0$ {number of function calls}
2: **while** $n < TotCalls$ **do**
3: Choose $P \subseteq S$, set of potential optimal rectangles;
4: Sample the rectangles $r \in P$ updating the counter n;
5: Subdivide the rectangles of P. Let subdivision be $D_P = \{R_1, R_2, \ldots, R_m\}$
6: $S = S \setminus P \cup D_P$
7: **end while**
8: return the best minimum;

If an hyperrectangle R is potential optimal then it will be sampled in the points $\mathbf{c}_R \pm \delta \mathbf{e}_i$, $i = 1 \ldots N$, where $\mathbf{c}_R$ is the center point of the hyperrectangle, δ is one-third the side length of the hyperrectangle, and $\mathbf{e}_i$ is the ith unit vector.

Afterwards the DIRECT algorithm chooses to leave the best function values in the largest space, therefore we define

$$w_j = \min(f(\mathbf{c}_R + \delta \mathbf{e}_j), f(\mathbf{c}_R - \delta \mathbf{e}_j)) \tag{7}$$

and divide the dimension with the smallest w_j into thirds, so that $c_i \pm \delta e_j$ are the centers of the new hyper-rectangles. This pattern is repeated for all dimensions

on the "center hyper-rectangle", choosing the next dimension by determining the next smallest w_j. This strategy increases the attractiveness of searching near points with good function values (since large hyperrectangles are preferred for sampling, everything else is equal) [F03].

Algorithm 2. DIRECT Subdivision pseudocode [JPS93]

Require: A potentially optimal rectangle R with center $\mathbf{c}_R$

1: Identify the set I of dimension with the maximum side length. Let δ equal one-third of this maximum side length.
2: Sample the function at points $\mathbf{c}_R \pm \delta \mathbf{e}_i \forall i \in I$, where $\mathbf{e}_i$ is the ith unit vector.
3: Divide the rectangle into thirds along the dimension I, starting with the lowest value of $w_j = \min(f(\mathbf{c}_R + \delta \mathbf{e}_j), f(\mathbf{c}_R - \delta \mathbf{e}_j))$ and continuing to the dimension with the highest w_j.

The whole procedure can be simplified in its exponential complexity, which is due to the recursive subdivision of rectangle, stating a limit in the maximum number of rectangle subdivisions. But in this case cannot be guarantee the global convergence and it will be reached only approximated solutions.

Without the previous subdivision limit, the algorithm guarantees the convergence to the global minimum although is not possible to predict how many iterations are required. The hypothesis for this claim are the continuity of the objective function after which the infinity number of iterations give a *dense* sampling of the domain.

3 The Immune Algorithm

In this research paper, we use an internal coordinate representation (torsion angles): each residue type requires a fixed number of torsion angles to fix the 3D coordinates of all atoms. Bond lengths and angles are fixed at their ideal values. The degrees of freedom in this representation are the backbone and sidechain torsion angles (ϕ, ψ, ω, and χ_h). The number of χ angles depends on the residue type. Angles, measured on degree, are represented by real numbers approximated to the third decimal digit.

The theory of clonal selection, suggests that among all possible cells, B and T lymphocytes, with different receptors circulating in the host organism, i.e. the ones who are actually able to recognize the antigen (the fittest cells), will start to proliferate by cloning. Hence, when a B cell is activated by binding an antigen, it produces many clones, in a process called *clonal expansion*. The resulting cells can undergo *somatic hypermutation*, creating offspring B cells with mutated receptors. Antigens compete for recognition with these new B cells, their parents and with other clones. The higher the affinity of a B cell to available antigens, the more likely it will clone. This results in a Darwinian process of variation and selection, called *affinity maturation*. The increase in size of those populations and the production of cells with longer expected *lifetimes* assure the

organism a higher specific responsiveness to that antigenic attack, establishing a defense over time (immune memory). This principle inspired a new class of artificial immune systems, the clonal selection algorithms [NCBT04].

The clonal selection algorithms (or Immune Algorithms) like Genetic Algorithms, Evolution Strategies, Evolutionary Programming and Genetic Programming are methods that use simulated evolution processes to solve, or approximate, complicated computational problems and NP-complete problems that are difficult to attack with deterministic and conventional methods [NCBT04].

The immune algorithm uses an evolutionary approach using different evolutionary operators, cloning, hypermutation, hypermacromutation and aging operators, to find the low-energy conformations of peptides.

We have also made use of a Quasi-Netwon procedure in the approach that refines solutions by using local optimization.

The Immune Algorithm based on Clonal Selection Principle [NPC04a] uses a different solution representation, each structure of a given protein is encoded as a set of dihedral angles (ϕ, ψ, ω, and χ_h). In the real code representation, each B cell at time step t is a vector of real variables:

$$\boldsymbol{x}^t = (x_1, x_2, \ldots, x_n) \in \Re^n.$$

Two hypermutation and hypermacromutation operators [NCBT04, NPC04a] were used together in the immune algorithm for the HP model of proteins.

Usually, variation operators mutate a individual by adding a Gaussian distributed random vector of mean zero and predefined deviation (σ) to it as follows:

$$\boldsymbol{x}'^{(t)} = \boldsymbol{x}^t + \boldsymbol{u}$$

where the mutation vector $\boldsymbol{u}$ is computed from

$$\boldsymbol{u} = (u_1, u_2, \ldots, u_n), \qquad u_j = N_j(0, \sigma).$$

where $N_j(0, \sigma)$ is a real random number generated by a gaussian distribution of mean 0 and standard deviation σ.

The new hypermutation operator performs a local search of the conformational space. It will perturb a torsion angle $\in (\phi, \psi, \omega, \chi_h)$ of a randomly chosen residue with the law:

$$\boldsymbol{x}_i^{t,\prime} = \boldsymbol{x}_i^t + N(0, \sigma)$$

with $\sigma = 1.0$.

The new hypermacromutation operator may change the conformation dramatically. When this operator acts on a peptide chain, a torsion angle of a randomly chosen residue is reselected from its corresponding constrained region (i.e., corresponding domain of the selected torsion angle).

4 Methods

Given a sequence of amino acids, and the corresponding secondary structure assignmenet (using SSpro[3] to predict protein secondary structure prediction

[3] http://www.ics.uci.edu/∼baldig/scratch/

Algorithm 3. Immune Algorithm for protein folding

Require: $ProteinSequence, d, dup, \tau_B, I$
1: $Nc \leftarrow d \times dup$;
2: $t \leftarrow 0$;
3: **if** I **then**
4: $P^{(t)} \leftarrow$ Random_Initial_Pop($ProteinSequence$);
5: **else**
6: $P^{(t)} \leftarrow$ DIRECT($ProteinSequence$);
7: **end if**
8: Evaluate($P^{(0)}$);
9: **while** $\neg$ Termination_Condition() **do**
10: $P^{(clo)} \leftarrow$ Cloning ($P^{(t)}, Nc$);
11: $P^{(hyp)} \leftarrow$ Hypermutation ($P^{(clo)}$);
12: Evaluate ($P^{(hyp)}$);
13: $P^{(macro)} \leftarrow$ Hypermacromutation (P^{clo});
14: Evaluate ($P^{(macro)}$);
15: $(P_a^{(t)}, P_a^{(hyp)}, P_a^{(macro)}) \leftarrow$ Aging($P^{(t)}, P^{(hyp)}, P^{(macro)}, \tau_B$);
16: $P^{(t+1)} \leftarrow (\mu + \lambda)$-Selection ($P_a^{(t)}, P_a^{(hyp)}, P_a^{(macro)}$);
17: $t \leftarrow t + 1$;
18: **end while**
19: BFGS($P^{(t)}$)

[BP03]), we represent protein by a sequence of torsion angles. The backbone angles (ϕ, ψ) are selected in the favorable Ramachandran region appropriate for the given secondary structure type; we set the last backbone angle ω to the standard value of 180°. Sidechain torsion angles are constrained in regions derived from the backbone-independent rotamer library of Roland L. Dunbrack[4] [DC97]. Sidechain constraint regions are of the form: $[m - \sigma,\ m + \sigma]$; where m and σ are the mean and the standard deviation for each sidechain torsion angle computed from the rotamer library. Under these constraints the conformation is still highly flexible and the structure can take on various shapes that are vastly different from the native shape.

The immune algorithm designed in this paper can use two different initialization procedure for the first population of conformations; random initialization of population, the standard approach, and an initial population of conformations given by DIRECT global search procedure.

In the first method, the protein sequences are conformations of torsion angles randomly selected from their corresponding constrained regions. In the latter, the initial population is constituted by a set of conformations created by DIRECT algorithm [JPS93]. DIRECT balances local and global search by selecting *potentially optimal rectangles* to be further explored. We are interested in using such an interesting feature of the DIRECT algorithm. DIRECT procedure produces an initial population of promising candidate solutions inside a potentially optimal rectangles of the funnel landscape of the protein folding problem. This

[4] http://www.fccc.edu/research/labs/dunbrack/

scheme reduces the number of fitness function evaluations of the overall search process to the lowest energy value. This is the first known application of DIRECT algorithm to the protein folding problem. Considering the two different initialization procedures (devised by boolean variable I) for the population at initial generation ($t = 0$), we have the IA starting from a random initial population, and the IA-DIRECT approach for the IA starting from a population of promising protein conformations created by the global optimizer DIRECT. From the current population, a number *dup* of clones will be generated, producing the population $Pop^{(clo)}$, which will be mutated into $Pop^{(hyp)}$ by the new hypermutation operator, and into $Pop^{(macro)}$ by the new hypermacromutation operator. Algorithm 3 shows the pseudo-code of the Immune Algorithm for the structure prediction of real proteins.

Finally, it is important to note as the IA output is improved by a deterministic local search, the BFGS scheme. The BFGS (*Broyden-Fletcher-Goldfarb-Shanno*) rule is a classical formula of the quasi-Newton methods. It is widely adopted in many unconstrained optimization algorithms which compute at every step a good approximation of the inverse Hessian matrix A^{-1}. In this work was used the "limited memory" implementation in [BNS94] [DS83] which uses few correction vectors in order to reconstruct A^{-1}.

The immunological operators, DIRECT and BFGS procedure act on the internal representation, torsion angles, whereas the CHARMM potential uses the external representation, Cartesian coordinates.

5 Results

The Met-enkephalin peptide. The Immune Algorithm has been applied to determine the three dimensional structure of the pentapeptide Met-enkephalin (TYR-GLY-GLY-PHE-MET amino acids). It is a very short polypeptide, only 5 amino acids, twenty-two variable backbone and side-chain torsion (or dihedral) angles ($n = 22$) and 75 atoms. From a optimization point of view the Met-enkephalin polypeptide is a paradigmatic example of *multiple-minima problem*. It is estimated to have more than 10^{11} locally optimal conformations. Nevertheless, the Met-enkephalin (1MET) has defined structure, and an apparent global minimum (with conformational energy of $-12.90 kcal/mol$ based on ECEPP/2 routine) first located by a Monte Carlo-minimization method in 4 hours only, in 1987 by Scheraga and Purisima [PS87]. For all these reasons, this peptide is an obvious "test bed", for which a substantial amount of in silico experiments has been done [PS87, LS88, JLM97, BHM98]. For frequent convergence to the global minimum, in [LS88] the authors set a maximum number of energy function evaluations $T_{max} = 10^6$.

Using the multistage approach, the initial population of 10 individuals was generated by global optimization procedure ran for $T^1_{max} = 11171$ energy function evaluations. The immune algorithm starting from this *ad hoc* initial conformation was executed for 500 generations with a population size of $d = 10$ individual, duplication parameter $dup = 2$, expected life time parameter $\tau_B = 5$, for a maximum number of energy function evaluation equal to $T^2_{max} = 20000$.

Fig. 1. In the left plot the backbone structure of 1MET protein obtained by IA; in middle plot the optimal conformation obtained by Scheraga [RS96]; in the right plot the superimposition of the two structure. The RMSD of the two structure is 2.835Å, the $RMSD_{c_\alpha}$ is 0.490Å. In terms of DME measure for the two structures we have $DME = 2.211$Åand $DME_{c_\alpha} = 0.454$Å. The figure was made using the program UCSF Chimera [HCPF:96].

Hence, considering the computational costs of the DIRECT procedure and the IA we have a overall $T_{max} = T^1_{max} + T^2_{max} = 31171$ maximum number of energy function evaluations for frequent convergence to the best energy value close to the global minimum. For 1MET peptide we omit to run BFGS procedure to perform a further local search. The best conformation with the lowest energy value of $-20.56 kcal/mol$ obtained by Immune Algorithm is reported in the left plot of figure 1, middle plot shows the backbone conformations of the Met-enkephalin peptide obtained by Scheraga and Li [LS88], and the right plot shows superimposition of the two structure using a full atom resolution. Computing the RMSD and DME metrics with respect the C_α atoms, and all the atoms of the two structures, we have the following results: $RMSD = 2.835$Å$(mean = 2.95, \sigma = 0.061$ on five indipendent runs), and $RMSD_{c_\alpha} = 0.501$Å, while in terms of DME measure $DME = 2.142$Å and $DME_{c_\alpha} = 0.468$Å.

In figure 2 we show the comparisons, in terms of Energy values (Kcal/mol) and Fitness Function Evaluations, among the three approaches used: DIRECT global optimization procedure, Immune Algorithm and the hybrid approach,

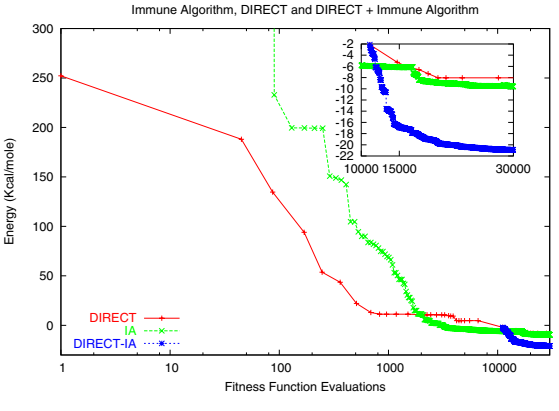


Fig. 2. Energy values versus Fitness Function Evaluations for DIRECT global search procedure, Immune Algorithm and the hybrid approach, Immune Algorithm with DIRECT, DIRECT+IA (axis x in log scale)

Immune Algorithm with DIRECT (DIRECT+IA). Both individual methods reach protein conformations with similar energy values with approximately 30000 energy function evaluations. The hybrid approach, Immune Algorithm starting from a population of promising protein conformations with energy values equal to $-2.11 kcal/mol$ created by the global optimizer DIRECT using 11171 energy function evaluations, is the only one to be able to yield the protein conformation near the native state ($E = -20.47 kcal/mol$).

Table 1 shows the comparisons of the designed hybrid immune algorithm with other evolutionary algorithms. For each algorithm, the table reports (when available from literature) the mean and standard deviation energy values, the energy function used, the best RMSD measure with respect the Scheraga's conformation, the accepted optimal conformation, and the maximum number of fitness function evaluations (FFE). It is important to note that the optimal conformation for ECEPP/2 and CHARMM energy functions are different. From a structural similarity point of view is more significant to consider the RMSD. In terms of RMSD value, the IA-DIRECT obtains the structure more similar to the accepted optimal conformation of Scheraga with a computational cost of $T_{max} = 31171$ energy function evaluations. The algorithms designed in [JLM97] were allowed to reach 150000 evaluations, while in [BHM98] this information is not available.

Finally, in figure 3 we report the dynamics of backbone angles ϕ and ψ (omiting ω and the sidechain angles χ_h) obtained by the two methods DIRECT (upper plot of figure 3) and Immune Algoritms (lower plot of figure 3). As described in section 4 and in the pseudo-code (see algorithm 3) the immune algorithm starts from a promising regions, that is, a promising conformation (a vector of torsion angles) previously determined by DIRECT procedure after 10^4 energy function evaluations.

After the first $5 \times 10^3 - 10^4$ energy function evaluaions, angles dynamics of DIRECT reaches a plateau region, there are not significant improvement. Starting by this conformation (center point of the hyper-rectangle) Immune Algorithm runs a local search inside the hyper-rectangle. The sequence of smooth

Table 1. Immune Algorithms versus Folding algorithms for *1MET*

Algorithm	Energy ($kcal/mol$)	RMSD	FFE
Scheraga's MC [LS88]	-12.90 ECEPP/2	n.a.	10^6
IA-DIRECT	-20.47 ± 1.54 CHARMM	**2.835Å**	$\mathbf{3 \times 10^4}$
REGAL (real cod.) Tight constr. [JLM97]	-23.55 ± 1.69 CHARMM	3.23Å	1.5×10^5
Lamarkian (binary cod.) [JLM97]	-28.35 ± 1.29 CHARMM	3.33Å	1.5×10^5
Baldwinian (binary cod.) [JLM97]	-22.57 ± 1.62 CHARMM	3.96Å	1.5×10^5
REGAL (real cod.) Loose constr. [JLM97]	-22.01 ± 2.69 CHARMM	4.25Å	1.5×10^5
SGA (binary cod.)[JLM97]	-22.58 ± 1.57 CHARMM	4.51Å	1.5×10^5
REGAL (real cod.)[JLM97]	-24.92 ± 2.99 CHARMM	4.57Å	1.5×10^5
standard GA [BHM98]	-3.17 ± 0.37 ECEPP/2	n.a.	n.a.
GA with sterical constraint [BHM98]	-2.35 ± 0.33 ECEPP/2	n.a.	n.a.
DGA [HMTO]	-11 ECEPP/2	n.a.	1.9×10^6
dualDGA [HMTO]	-11 ECEPP/2	n.a.	1.9×10^6
multicanonical algorithm [HO:93]	-12.1 ECEPP/2	n.a.	65412 MC steps

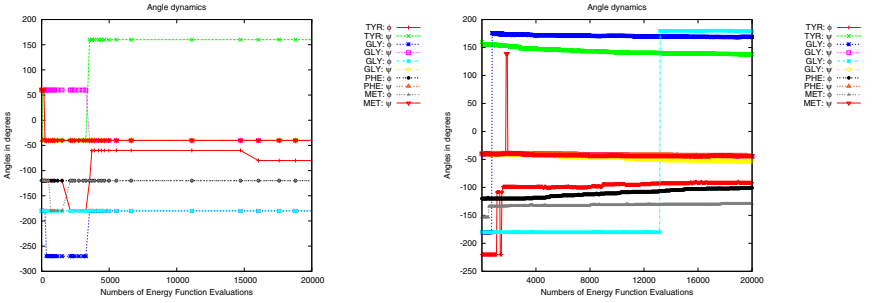


Fig. 3. Angles dynamics. Left plot: backbone angles (ϕ, ψ) dynamics versus energy function evaluations obtained by DIRECT procedure. Right plot: angles dynamics obtaited by Immune Algorithm starting from promising conformation previously determined by DIRECT.

perturbations performed by hypermutation operator and large angle variantion performed by hypermacromutation operator (e.g. figure 3 lower plot at 500 and 13000 energy function evaluations) drives Immune Algorithm to obtain 3D conformation near to the native state for 1MET peptide. Finally, the angle dynamics reported show the folding process of Met-Enkephelin peptide.

1POLY peptide. The 1POLY polypeptide is a 14 residue model of Polyalanine, its native structure is an α-helix. Polyalanine is a homogeneous molecule made up of 14 residues of the amino acid alanine (ALA), with 56 dihedral angles and 143 atoms ($ALA_1, \ldots ALA_{14}$ amino acids). Immune Algorithm using DIRECT procedure definitely formed the expected α-helix secondary structure (figure 4).

Using the multistage approach, the initial population was generated by DIRECT global optimization procedure ran for $T^1_{max} = 10183$ energy function evaluations. The IA starting from the *ad hoc* initial conformation was executed for 500 generations with a population size of $d = 10$ individual, duplication parameter $dup = 2$, expected life time parameter $\tau_B = 5$, for a maximum number of energy function evaluation equal to $T^2_{max} = 20000$. Hence, considering the computational costs of the DIRECT procedure and the IA we have a overall $T_{max} = T^1_{max} + T^2_{max} = 30183$ maximum number of energy function evaluations for frequent convergence to the lowest energy value close to the global minimum (E = 279.4 $kcal/mole$).

Fig. 4. 1POLY polypeptide structure obtained by Immune Algorithm. Left plot: the full atom resolution. Right plot: ribbon representation. The figure was made using the program UCSF Chimera [HCPF:96].

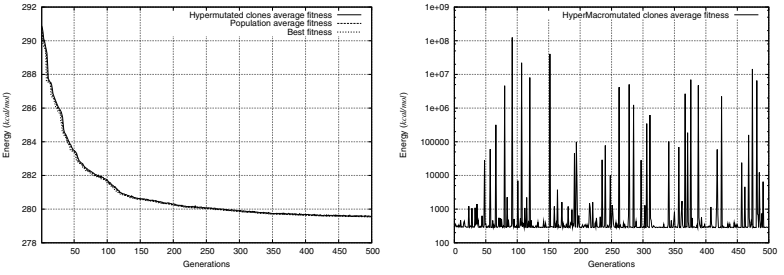


Fig. 5. Immune Algorithm dynamics, energy values versus generations, for Polyalanine peptide. Left plot: average energy value of hypermutated conformations, average energy value of current population and the best energy value. Right plot: energy values of hypermacromutated conformations versus generations.

The multistage approach needs of approximately 3×10^4 average number of energy evaluations to lowest energy value of Polyalanine while other approaches (e.g., evolutionary algorithms [JLM97] and hybrid methods [BHM98]) need at least 1.5×10^5.

In figure 5 we show the immune algorithm dynamics to describe hypermutations versus hypermacromutations phenomena. Figure 5 (left plot) reports average energy value of hypermutated candidate solutions, average energy value of current population and the best energy value; all curves decrease monotonically. The right plot shows conformations perturbated by hypermacromutation; spikes correspond to new conformation(s) with unfeasible 3D structure (for example, atoms too much close). In fact, hypermacromutation operator may change a given conformation dramatically (e.g. spikes of the right plot of figure 5), a torsion angle of a randomly chosen residue is reselected from its domain.

Table 2 shows best energy found, average energy and standard deviation values for Met-enkephalin and Polyalanine polypeptides obtained by immune algorithm starting from the promising region determined by DIRECT. All reported values are averaged on five indipendent runs.

1ROP protein. In this paragraph we show the prediction capability of the multistage approach for the 1ROP protein, a polypeptide of 56 amino acids with a 3D structure formed by 2 parallel α-helix (see figure 6). Figure 7 shows the predicted and the native structure of 1ROP. The RMSD of the two structure is 3.589 Å, and the DME is 1.89 Å. For 1ROP protein DIRECT ran for 10^5 energy

Table 2. Minimum, Mean and Standard deviation energy on five independent runs for Met-enkephalin and Polyalanine

Protein	min (kcal/mol)	mean (kcal/mol)	σ (kcal/mol)
Met-enkephalin(5 aa)	-24.85	-22.80	1.26
Polyalanine(14 aa)	279.54	279.55	0.01

Fig. 6. In the left plot the 3D structure of 1ROP protein obtained by IA; in the right plot the native structure of 1ROP. The RMSD of the two structure is 3.589 Å, and the DME is 1.89 Å. The figure was made using the program UCSF Chimera [HCPF:96].

```
1ROP:
1-56        .........1.........2.........3.........4.........5......
Sequence    MTKQEKTALNMARFIRSQTLTLLEKLNELDADEQADICESLHDHADELYRSCLARF
Predicted   CHHHHHHHHHHHHHHHHHHHHHHHHHHHHHCCCHHHHHHHHHHHHHHHHHHHHHHH
Native        HHHHHHHHHHHHHHHHHHHHHHHHHHTT HHHHHHHHHHHHHHHHHHHHHHH
```

Fig. 7. Comparison between predicted secondary structures (Predicted) and the X-ray-elucidated secondary structure (Native) for 1ROP with SSpro

function evaluations while IA ran for 2×10^4 energy functions calls. Hence we have a overall 1.2×10^5 maximum number of energy function evaluations for frequent convergence to the lowest energy value close to the global minimum ($E = -1117.249\ kcal/mole$).

Figure 8 reports two folding pathways for 1ROP protein in terms of energy value versus rmsd; it is possible to detect two distinct folding pathways (folding pathway 1 and folding pathway 2) with different final result. Folding pathway 1 obtains a final structure with lower RMSD value.

Finally, for 1ROP protein we show a folding process (see figure 9). Starting from a promising conformation obtained by DIRECT procedure the figure shows the intermediate and native-like conformations.

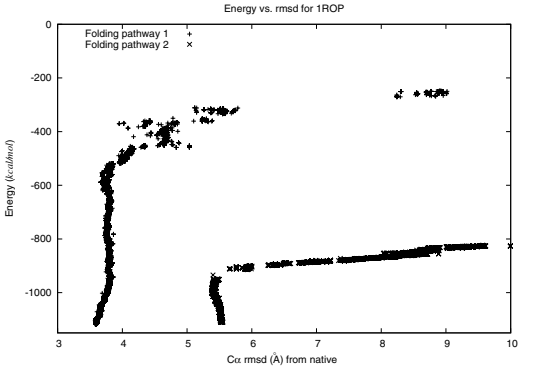
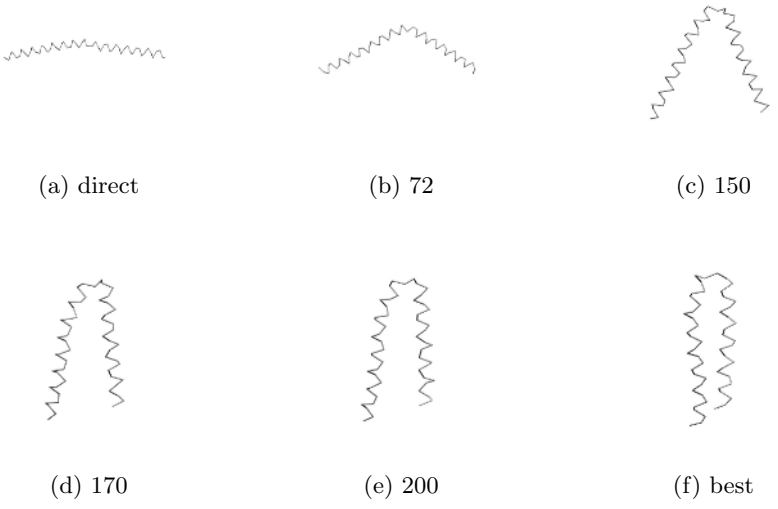


Fig. 8. Two Folding pathways for 1ROP protein



(a) direct (b) 72 (c) 150

(d) 170 (e) 200 (f) best

Fig. 9. Folding process for 1ROP Protein

Fig. 10. Predicted and native structure for 1BDC protein. The RMSD of the two structure is 8.780 Å, the $RMSD_{c_\alpha}$ is 7.368 Åand for core region [8-54] RMSD = 5.776 Å. In terms of DME measure for the two structures we have DME = 6.15 Å and DME_{c_α} =5.28 Å. The figure was made using the program UCSF Chimera [HCPF:96].

1BDC Protein. The last in silico experiment was performed on 1BDC protein, a three-helix bundle protein (see figure 11). 1BDC protein is formed by 60 residues. This hard protein instance has been chosen to assess the final prediction capability of the designed multistage approach. It is important to note that is the number of helices and the packing of the secondary structure elements, rather than the size of protein, that seems to cause difficulties.

For 1BDC protein DIRECT ran for 10^5 energy function evaluations while IA ran for 2×10^4 energy functions calls. Moreover to improve the final protein conformation has been called the quasi-Newton method BFGS procedure for 10^4 energy function evaluations to perform a further local search. Hence for 1BDC we have a overall 1.3×10^5 maximum number of energy function evaluations for frequent convergence to the lowest energy value close to the global minimum (E = -1428.96 $kcal/mole$).

Figure 10 shows the predicted and native 3D structures for 1BDC protein. The RMSD of the two structure is 8.780 Å, the $RMSD_{c_\alpha}$ is 7.368 Å. In terms

```
1BDC:
1-60        .........1.........2.........3.........4.........5.........6
Sequence    TADNKFNKEQQNAFYEILHLPNLNEEQRNGFIQSLKDDPSQSANLLAEAKKLNDAQAPKA
Native         S   SS HHHHHHHTTTTTS HHHHHHHHHHHHH  STHHHHHHHHHHHHHTTS
Predicted   CCCCCCCHHHHHHHHHHHCCCCCCHHHHHCHHHECCCCCHHHHHHHHHHHHHHHCCCCCC
```

Fig. 11. Comparison between predicted secondary structures (Predicted) and the X-ray-elucidated secondary structure (Native) for 1BDC with SSpro

of DME measure for the two structures we have $DME = 6.15$ Å and DME_{c_α} =5.28 Å. The RMSD for the core region (amino acids [8-54]) is equal to 5.776 Å.

6 Conclusions

We have developed an energy-based method to generate a variety of folds searching the conformational space with a multistage approach. DIRECT procedure produces an initial population of promising candidate solutions inside a potentially optimal rectangles of the funnel landscape of the protein folding problem. Hence, Immune Algorithm starts from a population of promising protein conformations created by the global optimizer DIRECT. This scheme reduces the number of fitness function evaluations of the overall search process to the lowest energy value. In fact, for Met-Enkephelin and Polyalanine peptides are needed 3×10^4 average number of energy function evaluations and for 1ROP and three-helix protein 1BDC are needed at most 1.3×10^5 energy function evaluations to produce native-like folds with low energy.

This is the first known application of DIRECT algorithm to the protein folding problem.

Considering the experimental results obtained by the hybrid Immune Algorithm, we can claim that it is competitive and effective search method in the conformational search space of real proteins, in terms of quality solution and computational cost. At our knowledge, the approach designed obtains the best result for the Met-Enkephelin peptide comparing the results of the current state-of-art algorithms.

Finally, the protein instances analyzed have been guided by DIRECT global search procedure and Immune Algorithm to folds resembling its crystal structure.

References

[Lev83] M. Levitt. Protein folding by restrained energy minimization and molecular dynamics. *J. Mol. Biol.*, 170:723–764, 1983.

[SKHB97] K. T. Simons, C. Kooperberg, E. Huang, and D. Baker. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring function. *J. Mol. Biol.*, 306:1191–1199, 1997.

[HO97] U. H. Hansmann and Y. Okamoto. Numerical comparisons of three recently proposed algorithms in the protein folding problem. *J. Comp. Chem.*, 18:920–933, 1997.

[BE94] J. U. Bowie and D. Eisemberg. An evolutionary approach to folding small alpha-helical proteins that uses sequence information and an empirical guiding fitness function. *Proc. Natl Acad Sci USA*, 91:4436–4440, 1994.

[HSP99] S. E. Huang, R. Samudrala, and Jay W. Ponder. Ab initio folding prediction of small helical proteins using distance geometry and knowledge-based scoring functions. *J. Mol. Biol.*, 290:267–281, 1999.

[JPS93] D. R. Jones, C. D. Perttunen, and B. E. Stuckman. Lipschitzian optimization without the lipschitz constant. *J. of Optimization Theory and Application*, 79:157–181, 1993.

[F03] D.E. Finkel: DIRECT Optimization Algorithm User Guide. Technical Report, CRSC N.C. State University, March 2003. (ftp://ftp.ncsu.edu/pub/ncsu/crsc/pdf/crsc-tr03-11.pdf)

[NCBT04] G. Nicosia, V. Cutello, P. Bentley, and J. Timmis. *Proceedings of the Third International Conference on Artificial Immune Systems*. Springer-Verlag, Berlin, Germany, 2004.

[NPC04a] V. Cutello, G. Nicosia, and M. Pavone. Exploring the capability of immune algorithms: A characterization of hypermutation operators. In *Proc. of the Third International Conference on Artificial Immune Systems (ICARIS'04)*, pages 263–276, Catania, Italy, September 2004.

[DC97] R. L. Dunbrack, Jr. and F. E. Cohen. Bayesian statistical analysis of protein sidechain rotamer preferences. *Protein Science*, 6, pp. 1661-1681, 1997.

[PS87] E. O. Purisima and H. A. Scheraga. An approach to the multiple-minima problem in protein folding by relaxing dimensionality. test on enkephalin. *J. Mol. Biol.*, 196:697–709, 1987.

[LS88] Z. Li and H. A. Scheraga. Structure and free energy of complex thermodynamics systems. *J. Mol. Stru.*, 179:333–352, 1988.

[JLM97] C. E. Kaiser Jr., G. B. Lamont, L. D. Merkle, G. H. Gates Jr., and R. Patcher. Polypeptide structure prediction: Real-valued versus binary hybrid genetic algorithms. In *Proc. of the ACM Symposium on Applied Computing (SAC)*, pages 279–286, San Jose, CA, March 1997.

[BHM98] E. Bindewald, J. Hesser, and R. Männer. Implementing genetic algorithms with sterical constrains for protein structure prediction. In *Proc. of Int. Conf. on Parallel Problem Solving from Nature (PPSN V)*, pages 959–967, Amsterdam, Netherland, September 1998.

[RS96] A. A. Rabow and H. A. Scheraga. Improved genetic algorithm for the protein folding problem by use of cartesian combination operators. *Protein Science*, 5:1800–1815, 1996.

[BP03] P. Baldi and G. Pollastri. The Principled Design of Large-Scale Recursive Neural Network Architectures-DAG-RNNs and the Protein Structure Prediction Problem. *Journal of Machine Learning Research*, 4:575–603, 2003.

[DS83] Dennis, J.E., and Schnabel, R.B. Numerical Methods for Unconstrained Optimization and Nonlinear Equations Englewood Cliffs, NJ: Prentice-Hall, 1983

[BNS94] R. H. Byrd, J. Nocedal and R. B. Schnabel. Representation of quasi Newton matrices and their use in limited memory methods. *Mathematical Programming.* 63(4):129-156, 1994

[M:82] A. D. McLachlan. Rapid Comparison of Protein Structures, *Acta Cryst.*, A38:871-873, 1982.

[HCPF:96] C.C Huang, G. S. Couch, E. F. Pettersen and T. E. Ferrin. Chimera: An Extensible Molecular Modeling Application Constructed Using Standard Components. *Pacific Symposium on Biocomputing.* 1:724 , 1996.

[HMTO] T. Hiroyasu, M. Miki, T. Iwahashi, Y. Okamoto, Dual Individual Distributed Genetic Algorithm for Minimizing the Energy of Protein Tertiary Structure, BioGEC-2003, *Biological Applications of Genetic and Evolutionary Computation, Genetic and Evolutionary Computation Conference 2003 (GECCO-2003)* July 12-16, 2003, Chicago, Illinois, USA.

[HO:93] U.H.E. Hansmann and Y. Okamoto: Prediction of Peptide Conformation by the Multicanonical Algorithm, in D.P. Landau, K.K Mon and H.-B. Schttler (Eds.), *Computer Simulation Studies in Condensed Matter Physics VI*, Springer Verlag 1993, 168.

[FM:00] Foloppe N, MacKerell A. D., Jr: All-Atom Empirical Force Field for Nucleic Acids: I. Parameter Optimization Based on Small Molecule and Condensed Phase Macromolecular Target Data. J. Comput. Chem., 21, pp. 86-104, (2000).

A Clonal Selection Algorithm for Coloring, Hitting Set and Satisfiability Problems

Vincenzo Cutello and Giuseppe Nicosia

Department of Mathematics and Computer Science, University of Catania,
V.le A. Doria 6, 95125 Catania, Italy
{vctl, nicosia}@dmi.unict.it

Abstract. In this keynote paper we present an Immune Algorithm based on the Clonal Selection Principle to explore the combinatorial optimization capability. We consider only two immunological entities, antigens and B cells, three parameters, and the cloning, hypermutation and aging immune operators. The experimental results shows how these immune operators couple the clonal expansion dynamics are sufficient to obtain optimal solutions for graph coloring problem, minimum hitting set problem and satisfiability hard instances, and that the IA designed is very competitive with the best evolutionary algorithms.

Keywords: Immune Algorithms, Clonal Selection Algorithms, Evolutionary Computation, Information Gain, Combinatorial Optimization, Graph coloring problem, Minimum Hitting Set problem, Satisfiability.

1 Introduction

The biological Immune System (IS) has to assure recognition of each potentially dangerous molecule or substance, generically called *antigen* (Ag), that can infect the host organism. The IS first recognizes it as *dangerous* or extraneous and then mounts a response to eliminate it. To detect an antigen, the IS activates a recognition process. Moreover, the IS only has finite resources and often very little time to produce *antibodies* (Ab) for each possible antigen [26].

In the last years we have witnessed an increasing number of algorithms, models and results in the field of Artificial Immune Systems [9, 11]. Natural Immune System provide an excellent example of bottom up intelligent metaphor [30], in which adaptation operates at the local level of cells and molecules, and useful behavior emerges at the global level, the immune humoral response.

The Immune Algorithm (IA) designed is based on the *theory of the clonal selection* first stated by Burnet and Ledeberg in 1959 ([3]). This theory suggests that among all the possible cells with different receptors circulating in the host organism, only those who are actually able to recognize the antigen will start to proliferate by duplication (cloning). The increase of those population and the production of cells with longer expected life-time assures the organism a higher specific responsiveness to that antigenic pattern, establishing a defense over time (immune memory). In particular, on recognition, B and T memory

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 324–337, 2006.

cells are produced. Plasma B cells, deriving from stimulated B lymphocytes, are in charge of the production of antibodies targeting the antigen.

This mechanism is usually observed when looking at the population of lymphocytes in two subsequent antigenic infections. The first exposition to the antigen triggers the *primary response*. In this phase the antigen is recognized and memory is developed. During the *secondary response*, that occurs when the same antigen is encountered again, a rapid and more abundant production of antibodies is observed, resulting from the stimulation of the cells already specialized and present as memory cells. The *hypermutation* phenomenon observed during the immune responses is a consequence of the fact that the DNA portion coding for the antibodies is subjected to mutation during the proliferation of the B lymphocytes. This provides the immune system with the ability to generate diversity.

From an information processing point of view [15] the IS can be seen as a problem learning and solving system. The Ag is the problem to solve, the Ab is the generated solution. At the beginning of the primary response the antigen-problem is recognized by poor candidate solution. At the end of the primary response the antigen-problem is defeated-solved by good candidate solutions. Consequently the primary response corresponds to a training phase while the secondary response is the testing phase where we will try to solve problems similar to the original presented in the primary response [24].

The paper is organized as follows: in section 2 we describe the proposed Immune Algorithm, in 3, 4, and 5 we define the combinatorial optimization problems, the benchmarks, and the relative fitness functions where we verify the IA's performance and report experimental results. In section 6 we report the conclusions.

2 The Immune Algorithm

We work with a simplified model of the natural IS. We will see that the IA presented in this work is very similar to De Castro, Von Zuben's algorithm, CLONALG [10, 12] and to Nicosia *et al.* immune algorithm [24, 25]. We consider only two entities: Ag and *B cells*. Ag is the problem and the B cell receptor is the candidate solution. Formally, Ag is a set of variables that models the problem; and, B cells are defined as strings (strings of integers or bits) of finite length ℓ. The input is the antigen–problem, the output is basically the candidate solutions–B cells that solve–recognize the Ag.

By $P^{(t)}$ we will denote a population of d individuals of length ℓ, which represent a subset of the space of feasible solutions of length ℓ, S^{ℓ}, obtained at time t. The initial population of B cells, i.e. the initial set $P^{(0)}$, is created randomly. After initialization, there are three different phases. In the *Interaction phase* the population $P^{(t)}$ is evaluated.

The *Cloning expansion phase* is composed of two steps: *cloning* and *hypermutation*. The cloning expansion events are modeled by *cloning potential* V and *mutation number* M, which depend upon fitness function f. If we exclude all the adaptive mechanisms [13] in evolutionary algorithms (EAs) (e.g., adaptive

mutation and adaptive crossover rates which are related to the fitness function values), the immune operators, contrary to standard evolutionary operators, depend upon the fitness function values [20]. Cloning potential is a truncated exponential:

$$V(f(\boldsymbol{x})) = e^{-k(\ell - f(\boldsymbol{x}))},$$

where the parameter k determines the sharpness of the potential. The cloning operator generates the population P^{clo}. In a minimization problem, the mutation number is :

$$M(f(\boldsymbol{x})) = (1 - opt/f(\boldsymbol{x}))\beta$$

where $opt \leq f(\boldsymbol{x}) \leq U_B$, with opt the global minimum (or the lowest value known), and U_b the upper bound; $\beta = c\ell$ models the shape of curve.

This function indicates the number of mutations in the candidate solution $\boldsymbol{x}$. The hypermutation operator from population P^{clo} generates the population P^{hyp}. The cell receptor mutation mechanism is modeled by the mutation number M. The cloning expansion phase triggers the growth of a new population of high–value B cells centered around a higher fitness function value.

In the *Aging phase,* after the evaluation of P^{hyp} at time t, the algorithm eliminates old B cells. Such an elimination process is stochastic, and, specifically, the probability to remove a B cell is governed by an exponential negative law with parameter τ_B, (expected mean life for the B cells):

$$P_{die}(\tau_B) = (1 - e^{(-\ln(2)/\tau_B)}).$$

Finally, the new population $P^{(t+1)}$ of d elements is produced. It is possible to use two kinds of Aging phases: *pure aging phase* and *elitist aging phase*. In the elitist aging, when a new population for the next generation is generated, we do not allow the elimination of B cells with the best fitness function. While in the pure aging the best B cells can be eliminate as well. In the present research work we will use the elitist aging phase because we obtain overall performance better. We observe that the exponential rate of aging, $P_{die}(\tau_B)$, and the cloning potential, $V(f(\boldsymbol{x}))$, are inspired by biological processes [28].

2.1 Termination Condition by Information Gain

To analyze the learning process, we use the notion of *Kullback information*, also called *information gain* [7], an entropy function associated to the quantity of information the system discovers during the learning phase. To this end, we define the B cells distribution function $f_m^{(t)}$ as the ratio between the number, B_m^t, of B cells at time t with fitness function value m, (the distance m from the antigen–problem) and the total number of B cells:

$$f_m^{(t)} = \frac{B_m^t}{\sum_{m=0}^{h} B_m^t} = \frac{B_m^t}{d}.$$

It follows that the information gain can be defined as:

$$K(t, t_0) = \sum_m f_m^{(t)} \log(f_m^{(t)}/f_m^{(t_0)}).$$

Table 1. Pseudo–code of Immune Algorithm

```
Immune Algorithm(d, dup, τ_B)
1. t := 0;
2. Initialize P^(0) = {x_1, x_2, ..., x_d} ∈ S^ℓ
3. while ( dK/dt ≠ 0 ) do
4.     Interact(Ag, P^(t));                    /* Compute P^(t) fitness */
5.     P^clo := Cloning (P^(t), dup);          /* Cloning expansion */
6.     P^hyp := Hypermutation (P^clo);         /* Variation operator */
7.     Evaluate (P^hyp);                       /* Compute P^hyp fitness */
8.     P^(t+1):=Aging(P^hyp ⊔ P^(t), τ_B);     /* Aging Phase */
9.     K(t, t_0):=InformationGain(P^(t+1));    /* Compute K(t, t_0) */
10.    t := t + 1;
11. end_while
```

The gain is the amount of information the system has already learned from the given Ag–problem with respect to initial distribution function (the randomly generated initial population $P^{(t_0=0)}$). Once the learning process starts, the information gain increases monotonically until it reaches a final steady state. This is consistent with the idea of a *maximum information-gain principle* of the form $\frac{dK}{dt} \geq 0$. Since $\frac{dK}{dt} = 0$ when the learning process ends, we use it as a termination condition for the IA. Finally, we note that our experimental protocol can have other termination criteria, such as maximum number of evaluations or generations.

2.2 The Selective Pressure, Dup

We can define the ratio $S_p = \frac{1}{dup}$ as the *selective pressure* of the algorithm: when $dup = 1$, obviously we have that $S_p = 1$ and the selective pressure is low, while increasing *dup* we increase the IA's selective pressure. Experimental results show that high values of d denote high clones population average fitness and, in turn, high population diversity but, also, a high computational effort during the evolution. We recall that dup is similar to the temperature in Simulated Annealing [18]. Low values of dup corresponds to a system that cools down slowly and has a high average number of evaluations to solutions (AES).

In evolutionary computation the selection of an appropriate population size is important and could greatly affect the effectiveness and efficiency of the optimization performance. For this reason EA's with dynamic population size achieve better convergence rate and discover as well any gaps or missing tradeoff regions at each generation [29]. All evolutionary algorithms need to set an optimal population size in order to discover the solutions. If the population size is too small, EAs may suffer from *premature convergence*, while if the population size is too large, undesired computational overwork may be incurred and the waiting time for a fitness improvement may be too long in practice. In the IA designed, the correct setting of parameters d and *dup* allows to discover the candidate solutions with high fitness values without using dynamic population. The IA presented use

three parameters: d, *dup* and τ_B; how we will see the first two parameters are crucial for the IA's dynamics, while the B cell's mean life is not a crucial parameter [5], hence when it is not reported we assume that $\tau_B = 25.0$ a typical value with general good performances.

3 Graph Coloring Problem

Recent studies show that when one faces the Graph Coloring Problem (GCP) with evolutionary algorithms, the best results are often obtained by hybrid EAs with local search and specialized crossover [16]. In particular, the random crossover operator used in a standard genetic algorithm performs poorly for combinatorial optimization problem and, in general, the crossover operator must be designed carefully to identify important properties, building blocks, which must be transmitted from parents population to offspring population. Hence the design of a *good* crossover operator is crucial for the overall performance of the EAs. The drawback is that is might happen to recombine good individuals from different regions of the search space, having *different symmetries*, producing poor offspring [21]. IAs do not have a crossover operator, and the crucial task of designing an appropriate crossover operator is avoided at once. The IA we will propose makes use of a particular mutation operator without having to incorporate specific domain knowledge.

For sake of clarity, we recall some basic definitions. Given an undirected graph $G = (V, E)$ with vertex set V, edge set E and a positive integer $K \leq \mid V \mid$, the Graph Coloring Problem asks whether G is K–colorable, i.e. whether there exists a function $f : V \rightarrow \{1, 2, ..., K\}$ such that $f(u) \neq f(v)$ whenever $\{u, v\} \in E$. The GCP is a well-known NP–complete problem [17]. Exact solutions can be found for simple or medium instances [22, 4]. Coloring problems are very closely related with *cliques* [19] (complete subgraphs). The size of the maximum clique is a lower bound on the minimum number of colors needed to color a graph, $\chi(G)$. Thus, if $\omega(G)$ is the size of the maximum clique: $\chi(G) \geq \omega(G)$.

For the GCP the B cells are modeled as strings of integer of length $\ell = \mid V \mid$. Hence, $f(\boldsymbol{x}) = m$ is the fitness function value of B cell receptor $\boldsymbol{x}$. Hence for the GCP, the fitness function $f(\boldsymbol{x}) = m$ indicates that there exists a m–coloring for G, that is, a partition of vertices $V = S_1 \cup S_2 \cup \ldots \cup S_m$ such that each $S_i \subseteq V$ is a subset of vertices which are pairwise not adjacent (i.e. each S_i is an *independent set*).

$$f_{gcp}(\boldsymbol{x}) = m = \ \# \text{ of coloring}(G, \boldsymbol{x})$$

To assign colors, the vertices of the solution represented by a B cell are examined and assigned colors, following a deterministic scheme based on the order in which the graph vertices are visited. In details, vertices are examined according to the order given by the B cell and assigned the first color not assigned to adjacent vertices.

Hence, for the GCP $M(f(\boldsymbol{x}))$ indicates the number of swaps between vertices in $\boldsymbol{x}$. The mutation operator chooses randomly $M(f(\boldsymbol{x}))$ times two vertices i and j in $\boldsymbol{x}$ and then swaps them. This method is very simple. In literature there

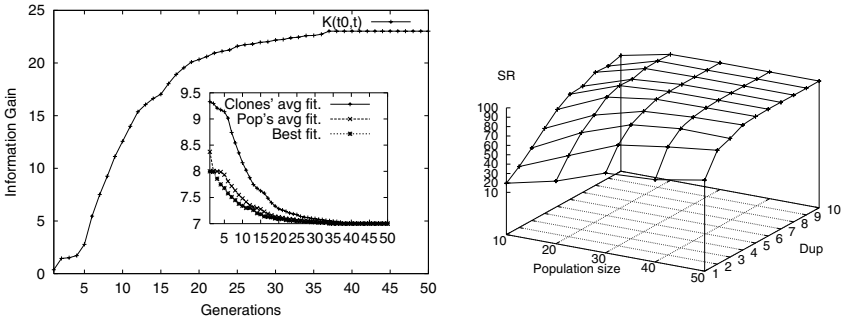


Fig. 1. Left plot: Information Gain versus generations for the GCP instance QUEEN6_6. Right plot: 3D plot of d, dup versus Success Rate (SR).

are more complicated and effective methods [16, 21, 19]. We do not use those methods because we want to investigate the learning and solving capability of our IA. In fact, the IA described does not use specific domain knowledge and does not make use of problem-dependent local searches. Thus, our IA can be improved simply including *ad hoc* local search and immunological operators using specific domain knowledge. In figure 1 (left plot) we show the information gain when the IA faces a simple Graph Coloring Problem (GCP) instance, QUEEN6_6, with vertex set $\mid V \mid = 36$, edge set $\mid E \mid = 290$ and optimal coloring 7. In particular, in the inset plot one can see the corresponding average fitness of population P^{hyp}, the average fitness of population $P^{(t+1)}$ and the best fitness value. All the values are averaged on 100 independent runs. In figure 1 (right plot) we show the values of parameters d and dup as functions of the Success Rate (SR). Each point has been obtained averaging 1000 independent runs. How we can see there is a certain relation between d and dup in order to reach a $SR = 100$. For the QUEEN6_6 GCP instance, for low values for the population we need a high value of dup to reach $SR = 100$. For $d = 10$, $dup = 10$ is not sufficient to obtain the maximum SR. On the other hand, as the population number increases, we need smaller values for dup. Small values of dup are a positive factor.

In figure 2 we show the fitness function value dynamics. In both plots, we show the dynamics of average fitness of population P^{hyp}, $P^{(t+1)}$, and the best fitness value of population $P^{(t+1)}$. Note that the average fitness of P^{hyp} shows the diversity in the current population, when this value is equal to average fitness of population $P^{(t+1)}$, we are close to a premature convergence or in the best case we are reaching a sub–optimal or optimal solution. It is possible to use the difference between P^{hyp} average fitness and $P^{(t+1)}$ average fitness,

$$\mid avg_{fitness}(P^{hyp}) - avg_{fitness}(P^{(t+1)} \mid = Pop_{div}$$

as a standard to measure population diversity. When Pop_{div} rapidly decreases, this is considered as the primary reason for premature convergence. In the left plot we show the IA dynamic when we face the DSCJ250.5.COL GCP instance ($\mid V \mid = 250$ and $\mid E \mid = 15,668$). We execute the algorithm with population size

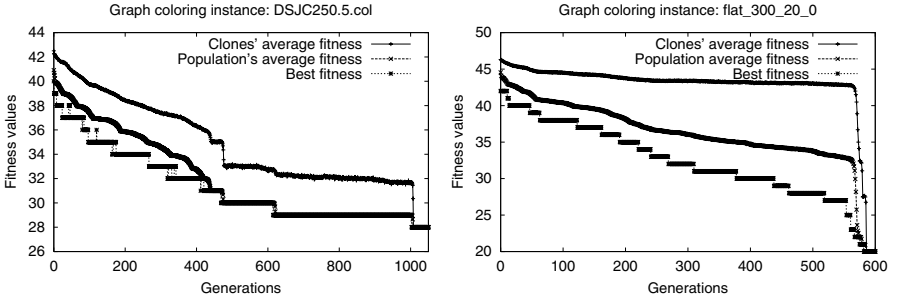


Fig. 2. Average fitness of population P^{hyp}, average fitness of population $P^{(t+1)}$, and best fitness value vs generations. Left plot: IA with pure aging phase. Right plot: IA with elitist aging.

Table 2. Experimental results on subset instances of Queen, Mycielsky, DSJC, Leighton graph instances. We fixed $\tau_B = 20.0$, and the number of independent runs 10. *OC* denotes the Optimal Coloring.

Instance G	$\mid V \mid$	$\mid E \mid$	OC	(d,dup)	Best Found	AES
Queen6_6	36	580	7	(50,5)	7	3750
Queen7_7	49	952	7	(60,5)	7	11,820
Queen8_8	64	1,456	9	(100,15)	9	78,520
Queen8_12	96	2,736	12	(500,30)	12	908,000
Queen9_9	81	1,056	10	(500,15)	10	445,000
School1_nsh	352	14,612	14	(1000,5)	15	2,500,000
School1	385	19,095	9	(1000,10)	14	2,950,000
DSJC125.1	125	736	5	(1000,5)	5	1,200,000
DSJC125.5	125	3,891	12	(1000,5)	18	1,500,000
DSJC125.9	125	6,961	30	(1000,5)	44	2,250,000
DSJC250.1	250	3,218	8	(400,5)	9	1,600,000
DSJC250.5	250	15,668	13	(500,5)	28	2,250,000
DSJC250.9	250	27,897	35	(1000,15)	74	3,950,000
le450_15a	450	8,168	15	(1000,5)	15	5,650,000
le450_15b	450	8,169	15	(1000,5,5)	15	5,800,000
le450_15c	450	16,680	15	(1000,15)	15	9,850,000
le450_15d	450	16,750	9	(1000,15)	16	10,200,000

$d = 500$, duplication parameter $dup = 5$, expected mean life $\tau_B = 10.0$. For this instance we use *pure aging* and obtain the optimal coloring. In the right plot we tackle the FLAT_300_20 GCP instance ($\mid V \mid = 300$ and $\mid E \mid = 21,375$), with the following IA's parameters: $d = 1000$, $dup = 10$, $\tau_B = 10.0$. For this instance the optimal coloring is obtained using *elitist aging*. In general, with elitist aging the convergence is faster, even though it can trap the algorithm in a local optimum.

Table 3. IA versus Evolve_AO Algorithm. The values are averaged on 5 independent runs.

Instance G	$\chi(G)$	Best–Known	Evolve_AO	IA	Difference
DSJC125.5	12	12	17.2	18.0	+ 0.8
DSJC250.5	13	13	29.1	28.0	**-0.9**
flat300_20_0	≤ 20	20	26.0	20.0	**-6.0**
flat300_26_0	≤ 26	26	31.0	27.0	**-4.0**
flat300_28_0	≤ 28	29	33.0	32.0	**-1.0**
le450_15a	15	15	15.0	15.0	0
le450_15b	15	15	15.0	15.0	0
le450_15c	15	15	16.0	15.0	**-1.0**
le450_15d	15	15	19.0	16.0	**-3.0**
mulsol.i.1	–	49	49.0	49.0	0
school1_nsh	≤ 14	14	14.0	15.0	+1.0

Table 4. IA versus Hao *et al.*'s HCA algorithm. The number of independent runs is 10.

Instance G	HCA's Best–Found and (SR)	IA's Best–Found and (SR)
DSJC250.5	28 (90)	28 (**100**)
flat300_28_0	31 (60)	32 (**90**)
le450_15c	15 (60)	15 (**100**)
le450_25c	26 (100)	**25** (**100**)

Although, with pure aging the convergence is slower and the population diversity is higher, our experimental results indicate that elitist aging seems to work well. We worked with classical benchmark graph [19]: the Mycielski, Queen, DSJC and Leighton GCP instances. Results are reported in Table 2. In these experiments the IA's best found value is always obtained $SR = 100$. For all the results presented in this section, we used *elitist aging*. In tables 3 and 4 we compare our IA with two of the best evolutionary algorithms, respectively Evolve_AO algorithm [2] and the HCA algorithm [16]. For all the GCP instances we ran the IA with the following parameters: $d = 1000$, $dup = 15$, and $\tau_B = 20.0$. For these classes of experiments the goal is to obtain the best possible coloring, no matter the value of AES. Table 3 shows how the IA outperform the Evolve_AO algorithm, while is similar in results to HCA algorithm and better in SR values (see table 4).

4 The Minimum Hitting Set Problem

An instance of the Minimum Hitting Set problem (MHSP) consists of a collection S of subsets of a finite set U and a positive integer $k \leq \mid U \mid$. Question: Is there a subset $U \subseteq U$, with $\mid U \mid \leq k$, such that U contains at least one element

from each subset in S? This problem is NP-complete. Indeed, membership to NP can be easily observed, since a guessed proposed hitting set U can be checked in polynomial time. NP-hardness is obtained by polynomial reduction from the Vertex Cover [17].

For the MHSP the B cells are modeled as binary strings of length $\ell =\mid U \mid$. We work with a fitness function that allows us to allocate feasible and unfeasible candidate solutions.

$$f_{hs}(\boldsymbol{x}) = Cardinality(\boldsymbol{x}) + (\mid S \mid -Hits(\boldsymbol{x}))$$

The candidate solution must optimize both terms. Each population member $\boldsymbol{x}$ must minimize the size of set U and maximize the number of hit sets. If $(\mid S \mid -Hits(\boldsymbol{x})) = 0$, $\boldsymbol{x}$ is a hitting set, that is a feasible solution.

The used fitness gives equal opportunity to the evolutionary process to minimize both terms. For example, if we have a collection S of 50000 sets and the following two individuals: $\boldsymbol{x}_1$, with $Cardinality(\boldsymbol{x}_1) = 40$, $Hits(\boldsymbol{x}_1) = 49997$, $f_{hs}(\boldsymbol{x}_1) = 43$; $\boldsymbol{x}_2$, with $Cardinality(\boldsymbol{x}_2) = 42$, $Hits(\boldsymbol{x}_2) = 49999$, $f_{hs}(\boldsymbol{x}_2) = 43$, it is difficult to decide which individual is the best, the choice is crucial and strongly influences the subsequent search in the landscape.

We test our IA by considering randomly generated MHSP instances [6, 8], denoted by "hs$\mid U \mid$-$\mid C \mid$"[1]. Our experimental results are reported in table 5. One can see name instance, parameter values, best hitting set known and best hitting set found by the our IA for each instance, and AES. For each problem instance we performed 100 independent runs. By inspecting the results, one can also see that increasing the values of d and *dup*, the average number of fitness evaluations increases.

In figure 3 (left plot), we show the 3D graphic of a set of numerical simulation. To understand the algorithm's behavior when changing the parameters, we performed a set of experiments on a simple Hitting Set instance, "hs100-1000", in which d and *dup* are given several values. The problem solving ability of the algorithm depends heavily upon the number of individuals we "displace"on the space $S^{d\times l}$ and on the duplication parameter *dup*. The population size varies from 100 to 200 while *dup* from 10 to 55. The Z axis shows the average number of evaluations to solutions (we allowed $T_{max} = 100000$ evaluations and performed 100 independent runs). The population size d strongly influences the number of solutions found. Indeed, when increasing d the number of solutions found increases. On the other hand, the average number of fitness function evaluation also increases. The value of *dup* influences the convergence speed. Thus, the parameters d and *dup* are crucial parameters for the IA. In figure 3 (right plot), we can see the fitness function values for the optimization process in action during the first 70 generations. The minimum hitting set is obtained at generation 60. Subsequently, other B cells with the same cardinality are discovered, completing the set of (found) solutions.

[1] All MSHP instances are available at http://www.dmi.unict.it/~ nicosia/cop.html

Table 5. Minimum Hitting Set Instances

MHSP instance	(d,dup)	best known	best found	AES
hs50-1000	(10,5)	7	7	360
hs50-10000	(10,5)	10	10	510
hs100-1000	(100,30)	6	6	18000
hs100-10000	(200,15)	9	9	27200
hs100-50000	(200,15)	39	39	98200
hs250-1000	(200,30)	10	10	60200
hs250-10000	(200,30)	12	12	72200
hs500-1000	(350,30)	10	10	105350
hs500-10000	(500,30)	16	16	240500
hs500-50000	(100,50)	21	21	105100

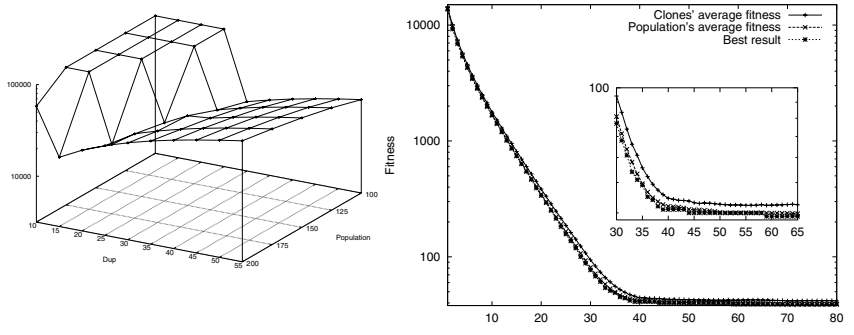


Fig. 3. Left plot: 3D representation of experimental results, with dimensions: d, dup and AES. Right plot: Fitness function values versus generations for the hs100-50000 Minimum Hitting set Problem instance. Y axis is in log scale and origin is fixed at X-value 39.

5 The 3-SAT Problem

3-SAT is a fundamental NP-complete problem in propositional logic [17]. An instance of the 3-SAT problem consists of a set V of variables, a collection C of clauses over V such that each clause $c \in C$ has $\mid c \mid = 3$. The problem is to find a satisfying truth assignment for C. The 3-SAT, and in general K-SAT, for $K \geq 3$, is a classic test bed for theoretical and experimental works. For the 3-SAT the B cells are modeled as binary strings of length $\ell = \mid V \mid$. The fitness function for 3-SAT is very simple, it computes only the number of satisfied clauses

$$f_{sat}(\boldsymbol{x}) = \ \#\text{SatisfiedClauses}(C, \boldsymbol{x})$$

For our experiments, we used A. van Gelder's 3-SAT problem instance generator, MKCNF.C [2]. The program MKCNF.C generates a "random" constant clause length

[2] The van Gelder's mkcnf.c code (Cnfgen.tar) is available by anonymous ftp at ftp://dimacs.rutgers.edu/pub/challenge/satisfiability/contributed/UCSS/instances

Table 6. 3-SAT Instances

Case	Instance name	Randseed	$\mid V \mid$	$\mid C \mid$	(d,dup)	AES
i1	sat30-129	83791	30	129	(50,20)	14632
i2	sat30-129	83892	30	129	(50,30)	292481
i3	sat30-129	83792	30	129	(50,20)	11468
i3	sat40-172	62222	40	172	(50,30)	24276
i4	sat50-215	82635	50	215	(75,15)	50269

Table 7. 3-SAT Instances. IA versus Eiben *et al.*'s evolutionary algorithm.

Case	$\mid V \mid$	$\mid C \mid$	Randseed	SR	AES	SR	AES
1	30	129	83791	1	**2708**	1	6063
2	30	129	83892	0.94	**22804**	0.96	78985
3	30	129	83792	1	**12142**	1	31526
4	40	172	83792	1	**9078**	1	13328
5	40	172	72581	0.82	37913	1	2899
6	40	172	62222	**1**	**37264**	0.94	82031
7	50	215	87112	0.58	**17342**	1	28026
8	50	215	82635	1	**42137**	1	60160
9	50	215	81619	0.26	**67217**	0.32	147718
10	100	430	87654	**0.32**	**99804**	0.06	192403
11	100	430	78654	0.04	**78816**	0.44	136152
12	100	430	77665	0.32	**97173**	0.58	109091

CNF formula and can force formulas to be satisfiable. For accuracy, we perform our tests in the transition phase, where the hardest instances of the problem are located [23]. In this case we have $\mid C \mid = 4.3 \mid V \mid$. The experimental results for four 3-SAT instances are shown in table 6. The random seed (Randseed) reported in third column is the MKCNF.C's input parameter useful to reproduce the same formula. We underline that for each instance we performed 50 independent runs. We note that the first three instances involve different formulae with the same number of variables and clauses. We can observe that among difficult formulae, there are even more difficult ones (at least for our algorithm), as the different number of evaluations proves. Such a phenomenon can be observed also for formulae with a higher number of variables and clauses. In determining a truth assignment the *AES* grows proportionally to the difficulty of satisfying the formula.

In table 7 we show the result of the last set of experiments. In this case we use the immune algorithm with a SAW (stepwise adaptation of weights) mechanism [14]. A weight is associated with each clause, the weights of all clauses that remains unsatisfied after T_p generation are incremented ($\delta w = 1$). Solving a constraint with a high weight gives high reward and thus the more pressure is given on satisfying those constraints, *the hardest clauses.* The last two columns refer to experimental results in [1] performed with an evolutionary algorithms

with SAW mechanism. The table shows that the IA with SAW mechanism, outperforms in many cases the results in [1], both in terms of success rate and computational performance, i.e. a lower number of fitness evaluations. Last set of experimental results shows how the IA designed coupled with a mechanism for adaptive recognition of hardest constraints, is sufficient to obtain optimal solutions for any K-Sat problem instances.

6 Conclusions

The Immune Algorithm designed represent a straightforward usage of the ideas upon which the theory of the clonal selection is stated [3]. The clonal selection is itself a Darwinian evolution process so that similarities with Evolution Strategies and Genetic Algorithms are natural candidates. Our IA, instead, does not use proportional cloning to fitness value. Moreover, there is neither a *birth operator* to introduce diversity in the current population nor a mutation rate (p_m) to flip a bit, B cells memory, nor threshold m_c to clone the best cells in the present population. Finally, one can observe similarities between our IA and the $(\mu + \lambda)$ evolution strategies using ranking selection [27]. We consider relevant for the searching ability: the size of the recognizing clones (the parameter *dup*), since it determines the size of the fitness landscape explored at each generation; and the number of individuals (d) since it determines the problem solving capacity.

One relevant disadvantage of our IA is that the search process may stop with a local minimum, when you are working with a small population size and a duplication parameter not sufficiently large. Moreover, the computational work increases proportionally to the size of these parameters. This slows down the search process in the space of feasible solutions although it gives better chances of finding good approximations to optimal solutions. The aging operator makes use of elitism, which on one hand speeds up the search, but on the other hand, may force the population to get trapped around a local minimum, reducing the diversity. Using simple fitness functions we have investigated the IA's learning and solving capability. The experimental results show how the proposed IA is comparable to and, in many combinatorial problems instances, outperforms the best evolutionary algorithms.

References

1. Bäck, T., A. E. Eiben and M. E. Vink. A superior evolutionary algorithm for 3-SAT. In V. William Porto, N. Saravanan, Don Waagen, and A. E. Eiben, editors *Proceedigns of the 7th Annual Conference on Evolutionary Programming*, Lecture Notes in Computer Science, 1998, Berlin, Springer-Verlag.
2. Barbosa, V. C., C. A. G. Assis and J. O. do Nascimento. Two Novel Evolutionary Formulations of the Graph Coloring Problem. *Journal of Combinatorial Optimization*, in press.
3. Burnet, F. M. *The Clonal Selection Theory of Acquired Immunity.* Cambridge University Press, Cambridge, UK, 1959.

4. Caramia, M. and P. Dell'Olmo. Iterative Coloring Extension of a Maximum Clique. *Naval Research Logistics*, 48:518–550 2001.
5. Cutello, V., G. Nicosia, M. Pavone. A Hybrid Immune Algorithm with Information Gain for the Graph Coloring Problem. accepted to *Genetic and Evolutionary Computation Conference (GECCO)* July 12–16, 2003, Chicago, Illinois, USA.
6. Cutello, V. and G. Nicosia. An Immunological approach to Combinatorial Optimization Problems. *Lecture Notes in Computer Science/Lecture Notes in Artificial Intelligence*, Springer Verlag, 2527:361–370, 2002.
7. Cutello, V. and G. Nicosia. Multiple Learning using Immune Algorithms. In Proceedings of the 4th International Conference on Recent Advances in Soft Computing, 2002.
8. Cutello, V., F. Pappalardo and E. Mastriani. An Evolutionary Algorithm for the T-constrained variation of the Minimum Hitting Set Problem. In *Proceedigns of the IEEE World Congress on Computational Intelligence, Congress on Evolutionary Computation (CEC 2002)*, Honolulu, HI (2002).
9. Dasgupta, D. editor. *Artificial Immune Systems and their Applications*. Springer-Verlag, Berlin Heidelberg New York, 1999.
10. De Castro, L. N. and F. J. Von Zuben. The Clonal Selection Algorithm with Engineering Applications. In Proceedings of GECCO 2000, Workshop on Artificial Immune Systems and Their Applications, 2000.
11. De Castro, L. N. and J. Timmis. *Artificial Immune Systems: A New Computational Intelligence Paradigm*. Springer-Verlag, UK 2002.
12. De Castro, L. N. and F. J. Von Zuben. Learning and optimization using the clonal selection principle. *IEEE Transaction on Evolutionary Computation*, 6(3):239–251, 2002
13. Eiben, A. E., R. Hinterding and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transaction on Evolutionary Computation*, 3(2):124–141, 1999.
14. Eiben, A. E., J. K. van der Hauw and J. I. van Hemert. Graph coloring with adaptive evolutionary algorithms. *Journal of Heuristics*, (4):25–46, 1998.
15. Forrest, S. and S. A. Hofmeyr. Immunology as Information Processing. In L. A. Segel and I. R. Cohen, editors, *Design Principles for Immune System & Other Distributed Autonomous Systems*, Oxford University Press, New York, 2000.
16. Galinier, P. and J. Hao. Hybrid Evolutionary Algorithms for Graph Coloring. *Journal of Combinatorial Optimization*, 3(4):379–397, December 1999.
17. Garey, M. R. and D. S. Johnson. *Computers and Intractability: a Guide to the Theory of NP-completeness*. Freeman, New York 1979.
18. Johnson, D. R., C. R. Aragon, L. A. McGeoch and C. Schevon. Optimization by simulated annealing: An experimental evaluation; part II, graph coloring and number partitioning. *Operations Research* 39:378–406, 1991.
19. Johnson, D. S. and M. A. Trick editors. *Cliques, Coloring and Satisfiability: Second DIMACS Implementation Challenge*. American Mathematical Society, Providence, RI 1996.
20. Leung, K., Q. Duan, Z. Xu, and C. W. Wong. A New Model of Simulated Evolutionary Computation – Convergence Analysis and Specifications. *IEEE Transaction on Evolutionary Computation*, 5(1):3–16, 2001.
21. Marino, A. and R. I. Damper. Breaking the Symmetry of the Graph Colouring Problem with Genetic Algorithms. In D. Whitley, D. Goldberg, E. Cantu-Paz, L. Spector, I. Parmee, and H.-G. Beyer, editors, *Workshop Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'00)*, Las Vegas, NV, 2000. Morgan Kaufmann.

22. Mehrotra, A. and M. A. Trick. A Column Generation Approach for Graph Coloring. *Journal of Computing*, 8:344–354, 1996.
23. Mitchell, D., B. Selman and H. J. Levesque. Hard and easy distributions of SAT problems. In Proceedigns of the AAAI, San Jose, CA, pages 459–465 1992.
24. Nicosia, G., F. Castiglione, and S. Motta. Pattern Recognition by primary and secondary response of an Artificial Immune System. *Theory in Biosciences*, 120(2):93–106, 2001.
25. Nicosia, G., F. Castiglione and S. Motta. Pattern Recognition with a Multi–Agent model of the Immune System. In International NAISO Symposium (ENAIS'2001), 2001.
26. Perelson, A. S., G. Weisbuch and A. Coutinho editors. *Theoretical and Experimental Insights into Immunology*. Springer-Verlag, New York, 1992.
27. Rogers, A. and A. Prügel-Bennett. Genetic Drift in Genetic Algorithm Selection Schemes. *IEEE Transactions on Evolutionary Computation*, 3:298–303, 1999.
28. Seiden P. E. and F. Celada. A Model for Simulating Cognate Recognition and Response in the Immune System. *Journal of Theoretical Biology*, 158:329–357, 1992.
29. Tan, K. C., T. H. Lee and E. F. Khor. Evolutionary Algorithms with Dynamic Population Size and Local Exploration for Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation* 5(6):565–588, 2001.
30. Timmis, J., T. Knight, L. N. de Castro and E. Hart. An Overview of Artificial Immune Systems. In Ray Paton, Hamid Bolouri, Mike Holcombe, Howard Parish and Richard Tateson editors, *Computation in Cells and Tissues: Perspectives and Tools of Thought*, Springer-Verlag, Natural Computation Series, Heidelberg, 2003, in press.

Artificial Immune-Based Optimization Technique for Solving Economic Dispatch in Power System

Titik Khawa Abdul Rahman, Saiful Izwan Suliman, and Ismail Musirin

Faculty of Electrical Engineering, Universiti Teknologi Mara,
40450, Shah Alam, Selangor, Malaysia
takitik@streamyx.com

Abstract. This paper presents an Artificial Immune-based optimization technique for solving the economic dispatch problem in a power system. The main role of electrical power utility is to ensure that electrical energy requirement from the customer is served. However in doing so, the power utility has also to ensure that the electrical power is generated with minimum cost. Hence, for economic operation of the system, the total demand must be appropriately shared among the generating units with an objective to minimize the total generation cost for the system. Economic Dispatch is a procedure to determine the electrical power to be generated by the committed generating units in a power system so that the total generation cost of the system is minimized, while satisfying the load demand simultaneously. The proposed technique implemented Clonal Selection algorithm with several cloning, mutation and selection approaches. These approaches were tested and compared in order to determine the best strategy for solving the economic dispatch problem. The feasibility of the proposed techniques was demonstrated on a system with 18 generating units at various loading conditions. The results show that Artificial Immune System optimization technique that employed adaptive cloning, selective mutation and pair-wise tournament selection has provided the best result in terms of cost minimization and least execution time. A comparative study with λ-iteration optimization method and Genetic Algorithm was also presented.

1 Introduction

Electrical power system are designed and operated to meet the continuous variation of power demand. The power demand is shared among the generating units and economic of operation is the main consideration in assigning the power to be generated by each generating units. The procedure to ensure for economic operation of a power system is called Economic Dispatch(ED). Economic Dispatch problem is an optimization problem that determines the optimal output of online generating units so as to meet the load demand with an objective to minimize the total generation cost[1].

Various mathematical methods and optimization techniques have been employed to solve for ED problems. Among the conventional methods that were previously employed, include lambda iteration method, base point and participation method and the gradient method[2]. These numerical methods assumed the incremental cost curves of the generating units are monotonically increasing piecewise linear

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 338 – 345, 2006.

functions. However, this assumption may cause these methods to be infeasible because of the nonlinear characteristics of the actual systems. In reference [2] dynamic programming (DP) method was implemented for solving the ED problem. Nevertheless, the DP method may cause the dimensions of the ED problem becomes extremely large, hence requires massive computational effort.

In the past decade, global optimization techniques such as simulated annealing (SA), genetic algorithms (GAs) and evolutionary programming (EP) have been increasingly used to solve for power system optimization problems [3]. The SA method is an optimization technique that employs probabilistic approach in accepting candidate solutions in the search process so that it can jump out from the local optimal solutions to approach the near global solution [4]. On the other hand, it is difficult to appropriately set the control parameters of the SA based algorithm and in addition, the speed of the algorithm is slow when applied to real power system problems [5]. GAs have been successfully employed to solve for economic dispatch problem due to its ability to model any kind of constraints using various chromosome coding scheme according to specific problem [6]. However, long execution time and non-guaranteed in convergence to the global optimal solution contribute the main disadvantages of GAs [6]. Similarly, evolutionary programming optimization technique has also received increasing attention by many researchers due to its ability to look for near global optimal solution. It has been applied successfully to solve optimization problems in power system such as economic dispatch, unit commitment, optimal power flow and many others. However, the results obtained from evolutionary programming technique are occasionally just near the global optimum. In addition, its long execution time posed its main disadvantage [7].

Artificial immune systems can be defined as metaphorical systems developed using ideas, theories and components, extracted from the natural immune system [8]. The natural immune system is a very complex system with several mechanisms for defense against pathogenic organisms. The main purpose of the immune system is to recognize all cells within the body and categories those cells as either self or nonself. The immune system learns through evolution to distinguish between dangerous foreign antigens and the body's own cells or molecules. From an information-processing perspective, the immune system is a remarkable parallel and distributed adaptive system. It uses learning, memory and associative retrieval to solve recognition, classification and optimization tasks [9]. A few computational models have been developed based several principles of the immune system such as immune network model, negative selection algorithm, positive selection algorithm and clonal selection principle [10,11].

In this paper, a new method for solving ED problem based on the Artificial Immune System (AIS) is presented. The developed AIS adopted the Clonal Selection algorithm in determining the optimal active power to be generated by the generating units in a power generation system. The total generation cost is taken to be the affinity measure for the AIS-based ED problem. Individual with lower total generation cost is considered to have higher affinity. The aim of the clonal operator is to produce a variation in the population around the parents according the affinity [12]. Hence, the searching area is enlarged and therefore the problem can be solved better, avoiding premature convergence. Several loading scenarios on a practical system having 18 generating units with a number of inequality and equality constraints were investigated in order to demonstrate the robustness and feasibility of the proposed technique.

The results obtained from the proposed technique were also compared with those obtained from the GA optimization and lamda iteration methods in order to assess for the solution quality and computational efficiency.

2 Economic Dispatch Mathematical Formulation

Solving the Economic Dispatch problem is to solve an optimization problem with an objective to minimize the total cost of generation. The solution gives the optimal generation output of the online generating units that satisfy the system's power balance equation under various system and operational constraints. The Economic Dispatch problem can be formulated mathematically as follows:

Equation 1 is the total generation cost to be minimized and therefore the objective function to the problem. Its value is taken to be the affinity to individual in the AIS optimization technique.

$$F_T = \sum_{i=1}^{N} F_i(P_i) \tag{1}$$

$$F_i(P_i) = a_i + b_i P_i + c_i p_i^2 \tag{2}$$

The cost of power generation for each generating unit is given by equation 2. Parameters a_i , b_i , c_i in the equation symbolizes constants on the Input-Output Curve of a generating unit.

The ED problem considered in this paper is the classic ED, whereby the losses are neglected. Hence, the power balance equation is given by equation 3; which is the equality constraint that has to be satisfied. While equation 4 is the inequality constraint, indicating the generation limits for each generating unit in the system .

$$P_1 + P_2 + P_3 + \ldots.. + P_N = P_D \tag{3}$$

$$P_{i\,min} <= P_i <= P_{i\,max} \tag{4}$$

P_D is the total load demand (Megawatts). $P_{i,min}$ is the lower limit of power generated by unit i. $P_{i,max}$ is the upper limit of power generated by unit i. N is the number of generators in the power generation system.

3 Artificial Immune System (AIS)

The natural immune system is a complex pattern recognition system that defends the body from foreign pathogens. In a simple manner, it recognizes all the body's own cells within the body as the self-cells and the foreign disease causing elements or the antigens as the non-self-cells. The non-self cells are further categorized in order to activate the suitable defense mechanism and it is unique with respect to a particular antigen. At the same time, the immune system also developed a memory to enable more efficient responses in case of further infection by the similar antigen. The process taken place in the immune system can be looked as a distributed task force that has the intelligence to take action from a local and global perspective using a network of chemical messengers for communication [11].

In order to understand the theoretical concept of AIS, the biological process of the immune system need to be appreciated. B-lymphocytes and T- lymphocytes are the two main components in the lymphocyte structure responsible for the defense mechanism in the body immune system. The B-lymphocytes are cells produced by the bone marrow and the T-lymphocytes are cells produced by the thymus. B-lymphocyte will produce only one antibody that is placed on its outer surface that acts as a receptor. When our body is exposed to an antigen, B-lymphocytes would respond to secrete antibodies specific to the particular antigen.

The portion on the antigen recognized by the antibody is called epitope that acts as an antigen determinant. Therefore each type of antibody has its specific antigen identifier called idiotope. The matching antibody-antigen are bounded and a second signal from the T-helper cells, would then stimulates the B-lymphocyte to proliferate and maturate in order to form a large clone of the plasma cells. Plasma cells are non-dividing cells that secrete large amount of antibodies. Since the lymphocytes can only make only one antibody, therefore the antibodies secreted by the plasma cell will be identical to that which was originally acted as the lymphocyte receptor. Besides maturing into plasma cells, the lymphocytes also differentiate into long-lived B-memory cells. These memory cells circulate through the blood, lymph and tissue, so that when exposed to a second antigenic stimulus, they will differentiate into large lymphocytes that are capable to produce high affinity antibodies to fight against the same antigens that stimulated the first response [11].

4 Clonal Selection Algorithm

The immune process described in section 3.0 is known as Clonal Selection Algorithm. Only only those cells that recognized the specific antigens would be selected to proliferate and thus go through the process of affinity maturation. In the selection stage, B cells with high affinity with respect to an antigen i.e recognized the antigen; are activated and stimulated to proliferate producing a large number of clones. In the maturation process, these clones mutate and turn into plasma cells which then that secrete large number of antibodies. Some of the B cells clones maturate into memory cells that have the memory of the antigenic pattern for future infections. The antibodies secreted from the second response would have higher affinity than those of the earlier response. In the computational point of view, this strategy suggests that the process perform a greedy search, where the individual will be locally optimized and the newcomers would yield a broader exploration of the search space. This characteristics makes the clonal selection algorithm is suitable for solving multi-modal optimization problems.

When the clonal selection algorithm is implemented for solving optimization problem, a few adaptations have to be made as follows [11] :-

a) There is no explicit antigen to be recognized, but an objective function is to be optimized. Therefore the affinity of an antibody refers to the evaluation of the objective function.
b) All antibodies are to be selected for cloning.
c) The number clones generated by the antibodies are equal.

However, this study investigated the effect of varying the number of clones according to the affinity and the results were compared to that obtained from standard cloning process.

5 Implementation of Clonal Selection-Based AIS for Solving Economic Dispatch

The developed AIS optimization technique using Clonal Selection algorithm was implemented to solve the economic dispatch problem on a practical system having 18 generating units [6]. Real number was used to represent the attributes of the antibodies. Each antibody attribute will be in a form of a pair of real valued vector (x_i, η_i), $\forall i \in \{1,\ldots, \mu\}$, where η_i is a strategy parameter[12]. Each antibody will go through the mutation process according to the expression given by equations 5 and 6.

$$\eta_i'(j) = \eta_i(j)\, exp(\tau' N(0,1) + \tau N_j(0,1)), \tag{5}$$

$$x_i'(j) = x_i(j) + \eta_i'(j)\, N_j(0,1), \tag{6}$$

where,

$N(0,1)$ is a normally distributed random number with zero mean and standard deviation equals to one.

$$\tau = ((2(n)^{1/2})^{1/2})^{-1}$$

$$\tau' = ((2n)^{1/2})^{-1}$$

n = number of attribute for an antibody

The clonal selection was implemented according to the following procedures:

a) Initial population is formed by a set of randomly generated real numbers. Each antibody was tested for any constraint violation using equations (3) and (4). Only antibodies that satisfy the constraint are included in the population set.
b) The affinity value of each antibody in the population set is evaluated using equation (1).
c) Clone the individuals in the population, giving rise to a temporary population of clones.
d) The population of clones undergoes maturation process through genetic operation i.e mutation. The fitness of the mutated clones are evaluated.
e) A new population of the same size as the initial population is selected from the mutated clones based on their affinity.
f) The new population will undergo the same process as stated in steps a – e.
g) The process is repeated until the solution converged to an optimum value.

Two other mutation techniques were also tested namely Cauchy mutation that used Cauchy distributed random number [13] and selective mutation that select the best individual resulted from Gaussian and Cauchy mutation. Finally, two selection strategies namely ranking and pair-wise tournament were implemented for comparison.

As other optimization technique, several parameters have to be determined before its implementation such as the population size and number of clones generated by each antibody (for fixed clones size). Based on the simulation results, the following

parameters are found to be suitable: 20 members in a population pool and the number of proliferated clones is 10 for standard cloning. The equation for adaptive cloning process was developed based on the fittest antibody will produce more clones compared to weaker ones. Equation (7) is implemented to determine the number of clones according the affinity measures or the fitness.

$$\text{No. of clones}_i = \left(1 - \frac{f_i}{\sum_{i=1}^{i-20} f_i} \right) x200 \tag{7}$$

f_i = fitness value
Σf_i = sum of fitness in a population

6 Results

The generator operating limits and quadratic cost function coefficients for the 18 generators in the test system are as in reference 6. The maximum total power output P of the generators is 433.22 MW. Various tests were made with varying percentage of the maximum power as the power demand.

The results from the simulation are summarized in Table 1. These results give the minimum generation cost for each approach after 20 runs. It was observed from theruns, the variation in the cost of generation is very small. This table shows the minimum generation cost obtained by each AIS technique when the demand is 70%

Table 1. Minimum Generation Cost From the Simulation

Parameter Representation	Cloning	Mutation	Selection	Fitness (Minimum Generation Cost) /$h^{-1}
Real number	Standard	Gaussian	Ranking	19993.1401
			Tournament	20039.0749
		Cauchy	Ranking	19926.5094
			Tournament	20088.0232
		Selective	Ranking	19894.9573
			Tournament	19893.3673
	Adaptive	Gaussian	Ranking	19957.0999
			Tournament	19911.1857
		Cauchy	Ranking	19957.0999
			Tournament	19971.9569
		Selective	Ranking	19909.2739
			Tournament	19888.3696

Table 2. Comparison in Minimum Generation Cost For Various Load

Demand	λ-iteration (cost/$)	Binary GA (cost/$)	Real GA (cost/$)	Real AIS (cost/$)
95% ***P***	29731.05	29733.42	29731.05	28911.25
90% ***P***	27652.47	27681.05	27655.53	26937.50
80% ***P***	23861.58	23980.24	23861.58	23383.98
70% ***P***	20393.43	20444.68	20396.39	19888.37

of the total maximum power output. It can be observed that the minimum generating cost is obtained by the real number AIS technique that implemented adaptive cloning, selective mutation strategy and tournament selection with total generation cost of $19888.37. The average execution time for all of AIS techniques is 2 s.

The results obtained from the proposed real number AIS technique implementing adaptive cloning, selective mutation strategy and tournament selection were compared with those obtained from the classical optimization technique namely λ-iteration method and Genetic Algorithm technique as shown in Table 2. From the results, it shows that the proposed technique has performed much better than the classical optimization technique, binary-coded GA and also real-coded GA for various power demands[6]. The proposed AIS optimization technique was much faster than the other techniques. It takes only 2 seconds to provide the optimal solution. The tests were carried out a Pentium IV personal computer.

7 Conclusion

A new approach of using clonal selection based AIS to solve for economic dispatch in power system is presented. The developed AIS technique is capable to determine the power to be generated by each generating unit in power system so that the cost of generation could be minimized while satisfying some operating constraints. Real number representation of the antibody attributes was implemented that represent the optimal output of the generating units. Several modifications were made on the cloning, mutation and selection schemes of the developed AIS optimization technique. The results obtained from various combinations of schemes were compared. It was found that the developed AIS optimization technique with real number representation of the antibody attribute employing adaptive cloning scheme, selective mutation and tournament selection gives the best result. A comparative study was carried out between the proposed technique, the λ-iteration method and also Genetic Algorithm technique. The results has shown that the proposed real number AIS optimization technique with adaptive cloning scheme, selective mutation and tournament selection is capable to provide better results with reduced computation time. Hence, the study shows that AIS could be a promising technique for solving complicated optimization problems in power system operation.

References

[1] J. Tippayachai, W. Ongsakul and I. Ngamroo, "Parallel Micro Genetic Algorithm for Constrained Economic Dispatch", *IEEE Trans. Power Syst.*, vol.17, pp. 790 – 797, Aug 2002.

[2] F.N. Lee and A.M Breipohl, "Reserved Constrained Economic Dispatch With Prohibitive Operating Zones", *IEEE Trans. Power Syst.*, vol.8, pp. 246 – 254, Feb 1993.

[3] K.P. Wong and Y.W. Wong, "Genetic and Genetic/Simulated – Annealing Approaches to Economic Dispatch ", *Proc. IEE Gen. Trans. Dist.*, vol.141, no. 5, pp. 507 – 513, Sept 1994.

[4] P. Attavriyanupp, H. Kita, T. Tanaka and J. Hasegawa, "A Hybrid EP and SQP for Dynamic Economic Dispatch With Nonsmooth Fuel Cost Function", *IEEE Trans. Power Syst.*, vol.17, pp. 411 – 416, May 2002.

[5] A.G. Bakirtzis, V. Petridis and S. Kazarlis, "Genetic Algorithm Solution to The Economic Dispatch Problem", *Proc. IEE Gen. Trans. Dist.*, vol.141, no. 4, pp. 377 – 382, July 1994.

[6] J. G. Damousis, A.G. Bakirtzis and P.S. Dokopoulos, "Network-Constrained Economic Dispatch Using Real-Coded Genetic Algorithm", *IEEE Trans. Power Syst.*, vol.18, pp. 198 – 205, Feb 2003.

[7] B. N. S. Rahimullah, E.I. Ramlan and T.K.A. Rahman, "Evolutionary Approach for Solving Economic Dispatch in Power System", In *Proceedings of the IEEE/PES National Power Engineering Conference*, vol.1, pp. 32 – 36, Dec 2003.

[8] L. N. de Castro and J. Timmis, "Artificial Immune Systems : A Novel paradigm to Pattern Recognition", In *Artificial Neural Networks in Pattern Recognition*, SOCO-2002, University of Paisley, UK, pp. 67-84, 2002.

[9] D. Dasgupta and N. A. Okine, "Immunity-Based Systems: A Survey", In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol.1, pp. 369 – 374, Oct 1997.

[10] L.N. de Castro and F.J. Von Zuben. "Learning and Optimisation Using the Clonal Selection Principle", *IEEE Trans. Evolutionary Computation.*, vol.6, pp. 239 – 251, June 2002.

[11] L.N. de Castro and F.J. Von Zuben. "Artificial Immune System : Part 1 – Basic Theory and Applications", *Technical report*, TR-DCA 01/99, Dec 1999.

[12] Y. Matsumura, K. Okhura and K. Ueda, "Evolutionary Programming with Non-Coding Segments for Real Valued Function Optimization", In *Proceedings of the IEEE International Conference on Systems, Man and Cybernatics 1999*, vol.4, pp. 242 – 247, Oct 1999.

[13] X.Yao and Y. Liu, "Fast Evolutionary Programming", Proc. Of the 5th Annual conference on Evolutionary Programming, pp 451 – 460, MIT Press 1996.

Securing IPv6-Based Mobile Ad Hoc Networks Through an Artificial Immune System

Julian L. Rrushi

Dipartimento di Informatica e Comunicazione,
Università degli Studi di Milano,
Via Comelico 39/41, 20135 Milano, Italy
jlr@cert-it.dico.unimi.it

Abstract. Mobile Ad Hoc Networks are vulnerable to security attacks that aim at disrupting routing information, exhausting nodes resources, maliciously manipulating data traffic etc. Construction of security mechanisms for Mobile Ad Hoc Networks is complicated by the fact that they lack a network infrastructure and a central authority for authentication and distribution of cryptographic keys.

In this paper is presented a theoretical model of an Artificial Immune System inspired by the vertebrate immune system. The objective of the proposed AIS is protection and reaction against known and unknown dysfunctions or attacks in a Mobile Ad Hoc Network that uses IPv6 addresses. This AIS also includes features inspired by Danger Theory, one of the latest immunological findings and a point of hot debate in the area of immunology.

Keywords: Mobile Ad Hoc Network, Artificial Immune System, Network Security, IPv6.

1 Introduction

The field of `Mobile Ad Hoc Networks` (MANETs) is rapidly growing and it is likely that such networks will see widespread use within the next few years [5]. MANETs are often used in environments such as mobile battlefields, law enforcement, emergency services, corporations and government agencies, where the information exchanged through the network is highly sensitive. On the other hand, MANETs are easy targets of security attacks that aim at disrupting routing information, exhaust nodes resources, maliciously manipulate data traffic etc. The radio frequency transmission media makes MANETs susceptible to eavesdropping. The highly mobile nature of their nodes also creates serious physical security challenges as the compromise of a legitimate node or the insertion of a malicious node could become source of attacks against the whole network. In Mobile Ad Hoc Networks without appropriate security provisions, it is relatively simple to snoop network traffic, replay transmissions, manipulate packet headers, and redirect routing messages [4]. MANETs' vulnerability is due to their dynamically changing topology, absence of conventional security infrastructures and open medium of communication.

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 346–354, 2006.

Attacks against MANETs are generally categorized as follows:

- Attacks against the routing protocol, where the attacker advertises wrong routing information, sends false route error packets, performs a selective routing, drops or does not forward routing protocol packets etc.
- Attacks that aim at exhausting resources of other nodes in the MANET. The attacker might achieve this by sending many RREQs for bogus destinations, selectively dropping packets thus causing affected nodes to send a large number of control packets and RREQs, etc.
- Cooperative attacks, where malicious nodes strategically cooperate with each other to cause harm.

There are many sophisticated techniques for performing these attacks and others. This motivated us to design a protocol independent security system that could be able to protect MANETs from known and unknown dysfunctions or attacks.

In this paper are described wanted functions, structure and organization of an Artificial Immune System (AIS) designed for securing MANETs using IPv6 addresses [3]. When considered from a computational point of view, the immune system can be considered to be a rich source of inspiration as it displays learning, adaptability, is self-organizing, highly distributed and displays a memory [1]. Functionalities of the AIS proposed in this paper include recognizing and responding to attacking or selfish nodes, maintaining diversity, tuning recognition of a specific attack etc. Another source of inspiration has been the `Danger Theory` (DT) [6] [7], that allows our AIS to detect and respond to self but harmful nodes in a MANET.

The AIS model presented in this paper has been intended to be indipendent of routing protocols. However, there are some important parameters that need a per-protocol tuning in order to capture the whole security context.

The remaining part of this paper is organized as follows. Section 2 gives an overview of Mobile Ad Hoc Networks. Section 3 gives an overview of other works related to security architectures in Mobile Ad Hoc Networks and inspired by the Human Immune System. Section 4 presents the Artificial Immune System designed in this work. Section 5 summarizes the contribution of this work and concludes the paper.

2 Preliminaries

A Mobile Ad Hoc Network is a collection of mobile nodes that are arbitrarily located in such a manner that the interconnections between them are capable of changing on a continual basis [5]. MANET nodes are equipped with wireless transmitters and receivers using antennas, and communicate with each other via wireless radio frequency links. MANETs have no supporting infrastructure, i.e. there are no fixed routers [5]. Nodes function as both terminals and routers for traffic among other nodes which need to communicate but are not in the reciprocal radio-range. A wireless connectivity in the form of a random, multi hop graph or ad hoc network exists between the nodes[4].

Salient characteristics of MANETs, which also affect the organization and design of their security mechanisms, are the following [4]:

- Nodes are free to move as they wish, thus the network topology may change in an unpredictable way at unpredictable times.
- MANETs are bandwidth-constrained as wireless links have a low capacity. This capacity is additionally lowered by natural factors like fading, noise and interference.
  Wireless channels have a broadcast nature, with two consequences:
  1. Unless security is enforced with external mechanisms, communications are inherently insecure because can be overheard by all nodes in the transmitter communication range, not only by the destination.
  2. Because of the contention on the shared medium, collisions are possible that further decrease the available bandwidth.
- Nodes rely on exhaustible means for their energy. In these cases design of routing or security protocols should take into account the energy conservation as an optimizing factor.
- The decentralized nature of network control in MANETs makes them more prone to physical security attacks.
  On the other hand solutions for MANETs must be distributed, as each node at any time could go out of communication range of any other node as a consequence of either movements or battery discharge. Thus, there should not exist any node having a key role over the other nodes.

Existing ad hoc routing protocols rely on trustworthy collaboration between nodes, that are expected to behave according to routing protocol specifications.

3 Related Work

In [12] is described a security architecture composed of monitor agents thought as T cells, decision agents thought as B cells and killer agents thought as antibodies. A monitor agent resides on each node and monitors all other nodes that are in its radio-range. It codes the behavior of a node using numbers according to the following: 1–RREQ sent, 2–RREQ received, 3–RREP sent, 4–RREP received, 5–PRRER sent, 6–RRER received, 7–DATA sent, 8–DATA received. The monitor agent sends these coded behaviors to a decision agent, that in turn takes a decision by referring to its security policies and its immune memory. If the decision agent recognizes a node as malicious, it will produce killer agents to surround the invader and isolate it.

In [13] is presented an AIS for detecting nodes misbehaviors in a MANET. The main elements of a HIS are mapped as follows:

Body: MANET, Self-Cells: well behaving nodes, Non-Self Cells: misbehaving nodes, Antigen: Sequence of observed DSR protocol events. Antibody: A pattern with the same format as the compact representation of antigen Chemical Binding: binding of antibody to antigen is mapped to a matching function. Negative Selection and Bone Marrow: Antibodies are created during an offline learning

phase. The bone marrow is mapped to a network with only certified nodes. In [13] a detection and classification algorithm is also defined.

In [8], an Intrusion Detection System (IDS) is presented, that examines the system calls executed by each application. This IDS tries to learn the normal behavior of applications by collecting system call traces during time intervals when the system is not under attack. Then the IDS extracts all possible sub traces containing six consecutive system calls and inserts them into a data base. These steps are referred to as the *learning phase*. A sub trace that does not match any of the entries of the database is tagged as abnormal. In the *monitoring phase*, when the IDS monitors applications in order to detect intrusions, the IDS collects traces of the system calls executed by each application and measures the degree of abnormality of each individual call trace as the number of abnormal sub traces it contains.

In [9] an evasion technique is presented, that consists of replacing the arguments of system calls that are part of the normal behavior of an application with values that represent operations the attacker wants to perform. This way, the legitimate system call executed by the application, will execute on behalf of the attacker, taking into account that the IDS considers system calls only and ignores their parameters [9].

4 Anatomy of Our AIS

The AIS proposed in this paper is inspired by CI and DT, and by the work presented in [8]. It is intended for securing Mobile Ad Hoc Networks using IPv6 [3] addresses. This AIS uses `Cryptographically Generated Addresses` (CGA) [2] as its basic instrument for grounding the immune response, i.e. linking it directly to the attacker.

In the model presented in this paper, when a node is in distress it sends signal 0, that represents a `"danger signal"`, directly to T lymphocyte. Consequently `"danger signal"` and `"this antigen really is dangerous"` are both represented by signal 2.

All the terms mentioned in this section and that belong to HIS, are intended in the artificial sense.

The proposed AIS consists of `agents` present on each MANET node, and is there to protect the MANET like HIS protects the body. AIS agents cooperate with each other in order to recognize and respond to pathogen nodes, i.e. attacking, selfish or dysfunctioning nodes.

The AIS presented in this paper is composed of an `Artificial Innate Immune System` (AIIS) and an `Artificial Adaptive Immune System` (AAIS). Both AIIS and AAIS are distributed through the MANET and are the union of individual contributions of each MANET node.

Both pathogen nodes and self (or non-dangerous) nodes, are characterized by antigens. In this work an antigen is defined as an ordered sequence of routing protocol events. The protocol event representation defined by related works and discussed in the previous section, has been extended. This was done to learn the lesson from the evasion technique in [9], also discussed in the previous section.

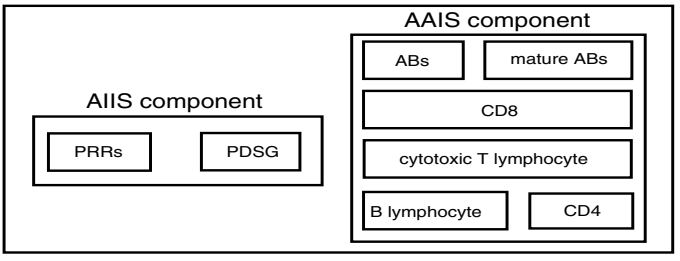


Fig. 1. Structure of an AIS Agent

Taking as an instance AODV[10], a routing protocol is a combination of values of fields given in the table below, for instance type 1 packet with the `join` flag set sent. The proposed AIS builds an antigen, which characterizes a neighbor node, by monitoring this node during `monitoring time intervals` of duration `t` s. Thus, an antigen that characterizes a node, is a trace of protocol events that took place at this node during a monitoring time interval.

B lymphocytes of the AIS described in this paper, process antigens in order to produce MHCs that B and T lymphocytes can recognize.

A B lymphocyte extracts from an antigen all possible subtraces containing `x` consecutive routing protocol events. In the AIS presented in this paper each of these subtraces is called a `gene`. A MHC is built as a sequence of `y` genes. `t`, `x` and `y` are parameters that may be protocol specific and are to be experimentally defined and tuned. In this work, an AB is a sequence of genes, that has a variable length. An AB recognizes an MHC in a `perfect way` if there is an one-to-one equality correspondence between `y` genes of the AB and the `y` genes of the MHC. An AB recognizes an MHC in a `partial way` if there is an one-to-one equality correspondence between `z` genes of this AB and `z` genes of this MHC, where `z` is less than y but greater than a predefined threshold.

The AIIS of each individual node is instructed to recognize abnormal states of the node where it resides. This could be done by defining some parameters related to main memory consumption, traffic generated per time unit, routing dysfunctions or any other state related factor. If properly predefined threshold values are exceeded, AIIS will make the node broadcast a `danger packet`, signaling to neighbor nodes that it is under attack or dysfunctioning. We call this a `positive` danger signal. If for any reason a node sends a danger signal when

Table 1. Fields of a routing protocol event in our AIS

packet type	flags	flag status	action
1	join	set	sent
2	repair	reset	received
3	gratuitous	-	-
4	destination only	-	-
data	unknown sequence number	-	-
-	repair	-	-
-	acknowledgement	-	-
-	no delete	-	-

it is not under attack, the AIS is such that it can deal with this fact using the laws of lymphotics, as described below. In (Fig. 1) `positive danger signal generator` (PDSG) is responsible for monitoring values of these parameters and deciding whether the node is in an abnormal state.

On the other hand, AIIS can be instructed to detect behaviors of any of its neighbor nodes, indicating that the neighbor is under attack. We call this a `negative` danger signal. These behaviors can be represented by antigens. If a node notices the presence of such an antigen in a neighbor node, then it can realize that the neighbor node is under attack or is not functioning properly. Each node needs to be fed with ABs, that have been built especially to recognize any MHCs created from an antigen indicating that a node is in distress.

AIIS could also be instructed to recognize known attacks on first encounter. In order to do this, each node needs to be fed with ABs that have been built especially to recognize any MHCs characterizing a known attack. In this context we can think of these ABs and MHCs as PRRs and PAMPs, respectively. These ABs could also clone and mature so as to recognize variations or combinations of known attacks.

In (Fig. 1) PRRs are ABs responsible for both detecting negative signals from their neighbors if these are in distress, and recognizing known attacks on first encounter.

The components of AAIS are B lymphocytes, CD4, CD8, cytotoxic T lymphocytes, ABs and mature ABs. A B lymphocyte is continuously refreshed with newly generated ABs. On the other hand those B lymphocytes that do not recognize any antigen for a long time, die. Thus B lymphocytes are subject to a kind of clonal selection. When a B lymphocyte recognizes an antigen, it either gets mature or is killed by CD8, depending on whether it detected a dangerous node or an honest node. In (Fig. 2), node N represents nodes that are in the radio-range of the malicious node, and node H represents nodes that are in the radio-range of the victim node. Malicious node M attacks victim node V, that is not in the radio-range of M. If the AIIS of victim node V will be able to notice the damage to V, it will periodically send in broadcast danger packets, till the state of V returns to normality. The B lymphocyte at node N transforms the antigen, say antigen a1, of the malicious node M into MHCs, and if this is the first time the MANET encounters a pathogen node with this kind of antigen, their ABs recognize these MHCs in a partial way. At this point we say that the B lymphocyte at node N receives signal 1. CD8 identifies the B lymphocyte whose ABs recognized the MHCs produced by an antigen of M's, and marks it for an eventual deletion. B lymphocyte asks to node N to send a `suspect report` packet that holds the MHCs in question, to the victim node V. Although the suspect report packet is destined to the victim, it is to be processed by any neighbor of the victim node, that receives it and this neighbor node should not forward it to the victim node. In our example the suspect packet is received and processed by node H. If the antibodies of CD4 at node H recognize the MHCs held in the previously received suspect report packet, we say that node H receives signal 1. On the other hand, if node H receives a `danger packet` broadcast by the victim

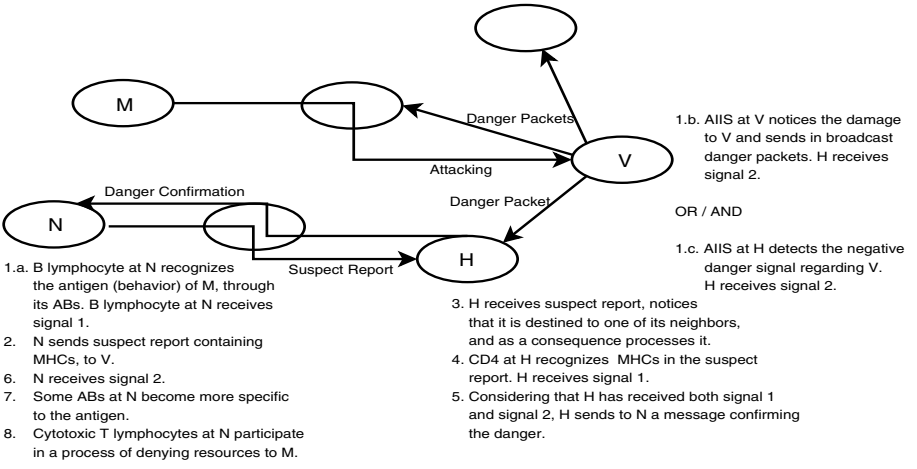


Fig. 2. AIS in action when there really is an attack

node V, i.e. node H receives a positive danger signal from node V, we say that node H receives signal 2. Node H receives signal 2 also if its AIIS senses the presence of a negative danger signal from the victim node V.

If node H receives both signal 1 and signal 2, it sends to node N a message confirming the danger, and the set of ABs at H that recognized the antigen a1 of malicious node M will reproduce in order to match it better. If node N receives such a message, we say that this node receives signal 2. If there really is an attack against node V, H will receive signal 2 and it will let node N receive signal 2 (Fig. 2). If node N receives both signal 1 and signal 2, also the ABs of the B lymphocyte at N that recognized the antigen a1 of malicious node M will reproduce in order to match it better. Each time this process is repeated, ABs of this B lymphocyte reproduce and get better. Highly matured ABs become `mature ABs`. Furthermore, the CD8 at node N cancels the deletion mark from this B lymphocyte. Cytotoxic T lymphocyte of node N, then, kill malicious node M, i.e. participate in a process of denying resources to malicious node M. In this case AIS performs a primary response.

When node N is in the radio-range of any node that is characterized by an antigen a1, its mature ABs will quickly recognize in a perfect way the MHCs created from antigen a1. If this is the case, node N does not send any MHCs to any neighbor of the victim node, and does not need to receive signal 2 in order to have its cytotoxic T lymphocyte kill the pathogen node.

If this is not the case, MHCs created from antigen a1 will be recognized by ABs of the B lymphocyte that recognized them when antigen a1 was encountered for the first time. In this case node N needs to receive signal 2 to proceed with killing malicious node M, and we say that AIS performs a secondary response.

When node N is in the radio-range of any node that is characterized by an antigen a2, that is structurally similar but not identical to antigen a1, some ABs from the B lymphocyte that recognized MHCs created from antigen a1, will

recognize MHCs created from antigen a2. In this case node N needs to receive signal 2, and AIS performs a cross-reactive response.

In the case node H does not receive signal 2, it ignores the suspect report packet it received from node N. It does the same if the antibodies of its CD4 do not recognize the MHCs held in this suspect report packet, or it receives no suspect report packet at all. If node N does not receive signal 2, it means that those ABs have matched antigens of well behaving nodes. In this case CD8 deletes the B lymphocyte marked for deletion earlier. CD8 at both N and H, cache for some defined time the MHCs created from the innocent antigen. If any of the ABs, that are created in a B lymphocyte, recognize any of these MHCs, CD8 deletes that B lymphocyte immediately.

Thus, ABs are subject to a light negative selection process performed by CD8. This is because negative selection shows scaling problems when it is applied to real network traffic [11]. On the other hand, in a large area of systems to protect such as MANETs, it may be difficult to find computationally efficient detectors that cover self and non-self adequately.

When both malicious node M and victim node V are in the radio-range of each other, any other well behaving node in this radio-range can receive signal 2 directly from the victim node. Thus, no intermediate node is needed for confirming the attack occurrence.

An important functionality of the AIS proposed in this paper is the `network vaccination`. During this process each node that has met an antigen `p` that identifies an attack `q`, sends this antigen to all the other nodes in the MANET. The other receiving nodes will create ABs that are specific to this antigen, and will be able to recognize better and respond faster to any malicious node performing attack `q`, even though they have never encountered the antigen `p` before.

5 Conclusion

In this paper has been presented a theoretical model of an Artificial Immune System inspired by the vertebrate immune system, danger theory and the work in [8]. There was defined the structure of this AIS and described the wanted functionalities of two main components of its, that are: Artificial Innate Immune System and Artificial Adaptive Immune System. Efforts were made to map a set of concepts from HIS to AIS, such as antigen, gene, major histocompatibility complex, antibodies etc, reflecting a lesson learned from an evasion technique described in [9].

This AIS model will be refined and completely specified since some policies and parameters have yet to be defined. Then it will be implemented in a network simulator and its behavior will be analyzed in order to verify its completeness and soundness.

Acknowledgments

I wish to thank Prof. Elena Pagani and Prof. Philip Grew for useful discussions and their helpful comments in preparing the paper.

References

1. Timmis J., Knight T., de Castro L. N., Hart E., "An Overview of Artificial Immune Systems", "Computation in Cells and Tissues: Perspectives and tools for thought", Natural Computation Series, pp 51-86, November 2004.
2. Aura T., "Cryptographically Generated Addresses (CGA)", RFC 3972, Work in Progress, March 2005.
3. Hinden R., Deering S,. "Internet Protocol Version 6 (IPv6) Addressing Architecture", April 2003.
4. Corson S., Macker J., "Mobile Adhoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", RFC 2501, Work in Progress, January 1999.
5. Royer E. M., Toh C., "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", IEEE Personal Communications , April 1999
6. Matzinger P, "The Danger Model in Its Historical Context", Scandinavian Journal of Immunology, 2001.
7. Matzinger P, "Tolerance, Danger and the Extended Family", Annual Review of Immunology, 1994.
8. Hofmeyr S., Forrest S., Somayaji A., "Intrusion Detection Using Sequences of System Calls", Journal of Computer Security, 1998.
9. Wagner D. and Soto P., "Mimicry Attacks on Host-based Intrusion Detection Systems", Proc. of the 9th ACM Conference on Computer and Communications Security, 2002.
10. Perkins C., Belding-Royer E., Das S., "Ad hoc On-Demand Distance Vector (AODV) Routing", RFC 3561, Work in Progress, July 2003.
11. Kim J, Bentley P, "Evaluating Negative Selection in an AIS for Network Intrusion Detection", Genetic and Evolutionary Computation Conference, pp 1330-1337, 2001.
12. Ping Y., Yan Y., Yafei H, Yiping Z., Shiyong Z., "Securing Ad Hoc Networks through mobile agent", Proceedings of the 3rd international conference on Information security, pp 125-129, 2004, Shanghai, China.
13. Le Boudec J., Sarafijanovic S., "An Artificial Immune System Approach to Misbehavior Detection in Mobile Ad-Hoc Networks", In Proceedings of Bio-ADIT 2004 (The First International Workshop on Biologically Inspired Approaches to Advanced Information Technology), pp. 96-111, January 29-30 2004, Lausanne, Switzerland.

Challenges for Artificial Immune Systems

Jon Timmis

Department of Computer Science and Department of Electronics,
University of York, Heslington, York, YO10 5DD
jt517@ohm.york.ac.uk
http://www-users.cs.york.ac.uk/jtimmis

Abstract. In this position paper, we argue that the field of Artificial Immune Systems (AIS) has reached an impass. For many years, immune inspired algorithms, whilst having some degree of success, have been limited by the lack of theorectical advances, the adoption of a limited immune inspired approach and the limited application of AIS to hard problems.

1 Introduction

The UK research community has proposed a number of Grand Challenges for Computer Science research[1] and ambitious plans for the development of a variety of research areas. Grand Challenge 7 (GC-7)[2] [1] addresses the area of *Non-Classical Computation*, which includes exploring areas of both biologically inspired paradigms, and the exploitation of the natural world (for example, DNA computing and quantum computing) to develop new notions of computation. GC7 consists of a number of *journeys* of which one is concerned with Artificial Immune Systems. In the spirit of GC7, this position paper proposes a number of challenges to the AIS community, and initial thoughts on how we might go about addressing those challenges. We begin by exploring the current *immunological mind set* of the AIS practitioner with a simple overview of the immunology that has served as inspiration to date. We then provide a brief overview of the area of AIS, and describe AIS in terms of a simple framework (as defined by De Castro and Timmis [2]). We then argue that rather than viewing the immune system in isolition from other bioligical systems, consideration should be given to immune, neural and endocrine interactions and how this may impact on the development of immune inspired approaches. We then conclude with a number of *challenges* to the AIS community. It should be said, that many of these ideas are not new, indeed not my own, but have come from many discussions with people in the AIS community. This paper is intended to be both a review and position paper that will hopefully draw together many ideas and stimulate further discussion and research.

[1] http://www.nesc.ac.uk/esi/events/Grand_Challenges/
[2] http://www.cs.york.ac.uk/nature/gc7/index.htm

B. Apolloni et al. (Eds.): WIRN/NAIS 2005, LNCS 3931, pp. 355–367, 2006.

2 The Immune System: From an Artificial Immune Systems Perspective

Recent developments within AIS, have focussed on three main immunological theories: clonal selection [3], immune networks [4] and negative selection [5]. Researchers in AIS have concentrated, for the most part, on the *learning* and *memory* mechanisms of the immune system (typically taking clonal selection and immune network theories as a basis), and the selection of detectors for identifying anomalous entities (typically undertaken with negative selection theory).

It is becoming increasingly apparent that the biological inspiration behind AIS has been somewhat naive and that the mechanisms and processes within the immune system (not to mention the role of the immune system) exploited by the AIS community has taken a limited perspective. Indeed, as argued by [6] AIS, and the majority of bio-inspired paradigms, have been guilty of a *reasoning by metaphor* approach. I would agree with this, and despite the success to date of AIS (and this should not be ignored) the restricted view of the immune system adopted by the AIS practitioners will no doubt limit the success of AIS.

In this section, we review the immunology, that has served as the foundations for much of AIS. We outline the main immunological theories that have acted as a source of inspiration, notably: clonal selection, immune networks and negative selection theories, and provide a brief explanation of how these have been exploited within AIS. A full review of these can be found in [2].

2.1 Immunity

The vertebrate immune system is composed of diverse sets of cells and molecules that work together with other systems, such as neural and endocrine, in order to maintain a *steady state* within the host. One traditionally held view on the role of the immune system is to protect our bodies from infectious agents such as viruses, bacteria, fungi and other parasites. On the surface of these agents are antigens that allow the identification of the invading agents (pathogens) by the immune cells and molecules, thus provoking an immune response. There are two basic types of immunity, innate and adaptive. Innate immunity [7] is not directed towards specific invaders, but against general pathogens that enter the body. The innate immune system plays a vital role in the initiation and regulation of immune responses, including adaptive immune responses. Specialized cells of the innate immune system evolved so as to recognize and bind to common molecular patterns found only in microorganisms. However, the innate immune system is by no means a complete solution to protecting the body.

Adaptive or acquired immunity [8], allows the immune system to launch an attack against any invader that the innate system cannot remove. The adaptive system is directed against specific invaders, and is modified by exposure to such invaders. The adaptive immune system mainly consists of lymphocytes, which are white blood cells, more specifically B and T-cells. These cells aid in the process of recognizing and destroying specific substances. Any substance that is capable of generating such a response from the lymphocytes is called an antigen or immunogen. Antigens are not the invading microorganisms themselves;

they are substances such as toxins or enzymes in the microorganisms that the immune system considers foreign. Adaptive immune responses are normally directed against the antigen that provoked them and are said to be antigen-specific.

2.2 Clonal Selection

A large part of AIS work has been based on the clonal selection theory. When antibodies on a B-cell bind with an antigen, the B-cell becomes activated and begins to proliferate. New B-cell clones are produced that are an exact copy of the parent B-cell, but then undergo somatic hypermutation [9] and produce antibodies that are specific to the invading antigen. The clonal selection principle [3] is the term used to describe the basic properties of an adaptive immune response to an antigenic stimulus. It establishes the idea that only those cells capable of recognizing an antigenic stimulus will proliferate, thus being selected against those that do not. Clonal selection operates on both T-cells and B-cells. The B-cells, in addition to proliferating or differentiating into *plasma cells*, can differentiate into long-lived B *memory cells*. Memory cells circulate through the blood, lymph and tissues. When exposed to a second antigenic stimulus they commence differentiating into large lymphocytes (plasma cells) capable of producing high affinity antibody.

In order for the immune system to be protective over periods of time, antigen recognition is insufficient. In the normal course of the evolution of the immune system, an organism would be expected to encounter a given antigen repeatedly during its lifetime. The initial exposure to an antigen that stimulates an adaptive immune response (an immunogen) is handled by a small number of B-cells, each producing antibodies of different affinity. Storing some high affinity antibody producing cells from the first infection, so as to form a large initial specific B-cell sub-population (clone) for subsequent encounters, considerably enhances the effectiveness of the immune response to secondary encounters. These are referred to as memory cells. Rather than starting from a *tabula rosa* every time, such a strategy ensures that both the speed and accuracy of the immune response becomes successively stronger after each infection.

Computationally then, this has led to the development of clonal selection based algorithms. There are many in the literature, most of which have focussed on developing optimisation approaches, such as the work by De Castro and Von Zuben on CLONALG [10], the work by Nicoisa et al. [11] and the work by Garrett on parameter free clonal selection [12]. Clonal selection has also formed the basis of learning algorithms (mainly supervised) such as AIRS by Watkins [13] and a parallel and a distributed version of AIRS [14], a distributed version of CLONALG by Watkins et al [15], the work by Kim and Bentley with DynamicCS applied to computer security [16], work by Secker et al, on email filtering [17] and many, many more. From a computational perspective, application of the clonal selection theory leads to algorithms that evolve (through a cloning, mutation and selection phase), candidate solutions in terms of optimisation or pattern detectors in terms of learning. Each of these algorithms have populations of *B-cells* (candidate solution) that match against *antigens* (function to be optimised).

These B-cells then undergo cloning (usually in proportion to the strength of the match) and mutation (usually, inversely proportional to the strength of the match). High affinity B-cells are then selected to remain in the population, some low affinity cells are removed and new random cells are generated. In essence, this is a high level abstraction of the clonal selection process. Through this process, good solutions can be found, and in terms of dynamic environments (such as [16, 17]) these solutions can be maintained over long periods of time.

2.3 Immune Networks

In the mid 1980's there was a great interest in the idea that B-cells formed an *idiotypic network* [4]. In that paper, Jerne proposed that the immune system is capable of achieving immunological memory by the existence of a mutually reinforcing network of B-cells. These cells not only stimulate each other but also suppress connected B-cells, though to a lesser degree. This suppression function is a mechanism by which to regulate the over stimulation of B-cells in order to maintain a stable memory. This network of B-cells occurs due to the ability of paratopes, located on B-cells, to match against idiotopes on other B-cells. The binding between idiotopes and paratopes has the effect of stimulating the B-cells. This is because the paratopes on B-cells react to the idiotopes on similar B-cells, as it would an antigen. However, to counter the reaction there is a certain amount of suppression between B-cells which acts as a regulatory mechanism.

From a computational perspective, this has led to the development of a number of immune network algorithms. Work by De Castro et al. on aiNET [18, 19], by Timmis et al on AINE [20, 21], by Neal on a self-stabilising immune network [22], by Bersini [23] and by Ishida [24]. There is little to unify these immune network algorithms. The main unifying theme is that they are all, in one way or another, based on the ideas of clonal selection (as outlined above). The main addition is that the candidate solutions (or detectors) interact based on stimulation and suppression mechanisms employed from the immune network. Immune network algorithms have tended to suffer a great deal from a large overhead of computational complexity, which has made their applicability to date, rather limited.

2.4 Negative Selection

Negative selection is a process of *selection* that takes place in the thymus gland. T-cells are produced in the bone marrow and before they are released into the lymphatic system, undergo a maturation process in the thymus gland. The maturation of the T-cells is conceptually very simple. T-cells are exposed to self-proteins in a binding process. If this binding activates the T-cell, then the T-cell is killed, otherwise it is allowed into the lymphatic system.

This process was one of the first to really catch the eye of the AIS community. Pioneering work by Forrest [25, 26] and more recently [27] viewed the negative selection process analogous to the production to detectors that can detect a change in the normal behavior of the system. The negative selection principle is used to filter randomly produced *change detectors* (filtered against what is considered

to be normal behavior) to populate a system that monitors a data stream. Deviations from the normal behavior are then flagged by the system. Whilst this is intuitively appealing, and indeed, has demonstrated some success, there are issues to do with scalability of the approach [28], rather than applicability.

3 Artificial Immune Systems

In an attempt to create a common basis for AIS, work in [2] proposed the idea of a framework for AIS. The authors argued the case for proposing such as framework from the standpoint that in the case of other biologically inspired approaches, such as artificial neural networks (ANN) and evolutionary algorithms (EAs) such a basic idea exists and helps considerably with the understanding and construction of such systems. For example, de Castro and Timmis [2] consider a set of artificial neurons, which can be arranged together so as to form an artificial neural network. In order to acquire knowledge, these neural networks undergo an adaptive process, known as learning or training, which alters (some of) the parameters within the network. Therefore, the authors argued that in a simplified form, a framework to design an ANN is composed of: a set of artificial neurons, a pattern of interconnection for these neurons, and a learning algorithm. Similarly, the authors argued that in evolutionary algorithms, there is a set of artificial chromosomes representing a population of individuals that iteratively suffer a process of reproduction, genetic variation, and selection. As a result of this process, a population of evolved artificial individuals arises. A framework, in this case, would correspond to the genetic representation of the individuals of the population, plus the procedures for reproduction, genetic variation, and selection. Therefore, the authors adopted the viewpoint that a framework to design a biologically inspired algorithm requires, at least, the following basic elements:

- A representation for the components of the system (known as shape space);
- A set of mechanisms to evaluate the interaction of individuals with the environment and each other. The environment is usually simulated by a set of input stimuli, one or more fitness function(s), or other mean(s) and;
- Procedures of adaptation that govern the dynamics of the system, i.e., how its behavior varies over time.

The framework can be thought of as a layered approach. In order to build a system, one typically requires an application domain or target function. From this basis, the way in which the components of the system will be represented will be considered. For example, the representation of network traffic may well be different than the representation of a real time embedded system. Once the representation has been chosen, one or more affinity measures are used to quantify the interactions of the elements of the system. There are many possible affinity measures (which are partially dependent upon the representation adopted), such as Hamming and Euclidean distances. The final layer involves the use of algorithms, which govern the behavior (dynamics) of the system. Here, in the original

framework proposal, algorithms based on the following immune processes were presented: negative and positive selection, clonal selection, bone marrow, and immune network algorithms.

3.1 Building Artificial Immune Systems

When constructing an AIS, there are many computational and practical issues to consider. The first is computational complexity of the approach. This relates to the time and space required to generate the suitable number of detectors (members of a population) that are required for the job [29]. For example, there are a number of works that outline the unacceptable computational complexity of the negative selection approach from AIS [30, 28, 31] as there is an exponential relationship between the size of the data set to be used, and the number of detectors that it is possible to generate. However, other approaches within AIS, such as clonal selection and immune networks, seem not to suffer quite the same problem. The second aspect to consider is the data to be used. In the context of embedded systems for example, if one abstracts away from the system components and uses state machines, then one has to be careful that there is an accurate mapping between the state machine and the actual system, and ensure that the state machine adequately scopes the space to be immunised [29]. Consideration here also has to be given to the way in which data is represented. The shape space paradigm proposes varying ways of data representation and interaction. However, when dealing with discrete values, such as those found in embedded systems, the method of defining affinity (i.e. seeing how similar one item is to another) is not as clear-cut as it may seem. This is coupled with the fact that mutation, even what might be thought of as a small amount, could have a huge impact on the meaning of the data. Should a binary shape space be employed, the mere flipping of one bit could indicate a huge shift in meaning of the state, rather than the small shift that may be desired. In both of these situations, domain knowledge can play a pivotal role in the success or failure of such as system [29]. For example, recent studies by [32], point out that the main effect on immune network algorithms may well be the way in which interaction is defined. Through the development of a simple model Hart and Ross demonstrate the evolution of various immune network structures which are considerably affected by the choice of affinity measure between two B-cells, which in turn effects how B-cells interact with each other. Whilst no concrete conclusions are drawn here, the message is clear: think before you design.

4 Reflections on AIS

The original AIS were with an interdisciplinary slant. For example, Bersini [23, 33, 34] pays clear attention to the development of immune network models, and then applies these models to a control problem characterised by a discrete state vector in a state space. Bersini's proposal relaxes the conventional control strategies, which attempt to drive the process under control to a specific zone of

the state space; he instead argues that the metadynamics of the immune network is akin to a meta-control whose aim is to keep the concentration of the antibodies in a certain range of viability so as to continuously preserve the identity of the system.

There are other examples of interdisciplinary work, such as the development of immune gene libraries and ultimately a bone marrow algorithm employed in AIS [35], and the development of the negative selection algorithm and the first application to computer security [25]. However, in more recent years, work on AIS has drifted away from the more biologically-appealing models and attention to biological detail, with a focus on more engineering-oriented approach. This has led to systems that are examples of reasoning by metaphor [6]. These include simple models of clonal selection, immune networks and negative selection algorithms as outlined above. For example, the clonal selection algorithm (CLONALG), whilst intuitively appealing, lacks any notion of interaction of B-cells with T-cells, MHC or cytokines. In addition, the large number of parameters associated with the algorithm, whilst well understood, make the algorithm less appealing from a computational perspective. aiNET, again, whilst somewhat affective, does not employ the immune network theory to a great extent. Only suppression between B-cells is employed, whereas in the immune network theory, there is suppression and stimulation between cells. With regards to negative selection, the simple random search strategy employed, combined with using a binary representation, makes the algorithm computational so expensive, that it is almost unusable in a real world setting [28, 31].

4.1 Problem Oriented Perspective

Freitas and Timmis [36] outline the need to consider carefully the application domain when developing AIS. They review the role AIS have played in the development of a number of machine learning tasks, including that of classification. However, Freitas and Timmis point out that there is a lack of appreciation for possible inductive bias within algorithms and positional bias within the choice of representation and affinity measures. Indeed, this observation is reinforced by the work of Hart and Ross [32] with the development of their simple immune network simulator with various affinity metrics. They make the argument that seemingly generic AIS algorithms, are maybe not so generic after all, and each has to be tailored to specific application areas. This may be facilitated by the development of more theoretical aspects of AIS, which will help us to understand how, when and where to apply various AIS techniques.

It should be noted that there have been some previous attempts at providing *design principles* for immune systems, such as work by Segal et al. [37] and Bersini and Varela [34]. However, work by Segal, whilst extremely interesting, focussed primarily on network signalling, and did not provide a comprehensive set of general design principles, or provide any test application areas for those principles. Work by Bersini, focussed on the immune network and *self assertion* ideas of the immune system to create his design principles and whilst being more concrete, are still quite high level:

- Principle 1: The control of any process is distributed around many operators in a network structure. This allows for the development of a self-organising system that can display emerging properties.
- Principle 2: The controller should maintain the viability of the process being controlled. This is keeping the system within certain limits and preventing the system from being driven in one particular way.
- Principle 3: While there may be perturbations that can affect the process, the controller learns to maintain the viability of the process through adaptation. This learning and adaptation requires two kinds of plasticity: a parametric plasticity, which keeps a constant population of operators in the process, but modifies parameters associated with them; and a structural plasticity which is based on the recruitment mechanism which can modify the current population of operators.
- Principle 4: The learning and adaptation are achieved by using a reinforcement mechanism between operators. Operators interact to support common operations or controls.
- Principle 5: The dynamics and metadynamics of the system can be affected by the sensitivity of the population.
- Principle 6: The system retains a population-based memory, which can maintain a stable level in a changing environment.

These are potentially useful principles, that should be refined in light of immunological advances and possibly taken on board (to some degree) by the community. These need to be tested in various application areas, and refined to allow for the creation of not only a generic set of AIS design principles that are useful to the community, but also specific ones for specific application areas. With this, may come a better understanding of how to apply AIS, and not fall into the traps highlighted by Freitas and Timmis.

4.2 Theoretical Perspective

There is very limited work on the more theoretical aspects of AIS. In terms of convergence proofs, one paper, [38], presents a complete proof for a specific multi-objective clonal selection algorithm using markov chains [39]. As pointed out by Hone and Kelsey [40] a useful and valued avenue to explore would be into the dynamics of immune algorithms based on nonlinear dynamical systems inspired by biological models [41], and stochastic differential equations [42]. Given the use of clonal selection based algorithms within AIS, a great deal could be gained by the community with further theoretical investigations such as, the role of mutation operators, which could be used to provide information for optimal mutation rates for specific functions. There is some advancement in the theoretical aspects of negative selection, with a firmer understanding of the role of affinity measures such as *r-chunk* matching and the scalability issues relevant to certain shape-spaces employed with negative selection [28, 27].

4.3 The Immune System Is Not an Island

Homeostasis is the ability of an organism to achieve a steady state of internal body function in a varying environment [43]. This is achieved via complex interactions between a number of processes and systems within the organism, in particular the immune, neural and endocrine systems. We propose that by examining these systems, and their interactions, it should be possible to gain insight into how organisms achieve homeostasis, and therefore exploit these interactions in the realm of computer science and engineering. There is a large body of work investigating the interactions of these systems, but little (to be honest hardly any) work in the area of AIS has payed any attention to this, however for a more detailed overview and for some initial ideas see [44].

Whilst the immune system is clearly an interesting system to investigate, if viewed in isolation, many key emergent properties arising from interactions with other systems will be missed. Such systems do not operate in isolation in biology, therefore, consideration should to be given to the interactions of the immune, neural and endocrine systems, and how, together, they allow for emergent properties to arise [45, 46, 47]. Immune, neural and endocrine cells express receptors for each other. This allows interaction and communication between cells and molecules in each direction. It appears that products from immune and neural systems can exist in lymphoid, endocrine and neural tissue at the same time. This indicates that there is a bi-directional link between the nervous system and immune system. Therefore, it would seem that both endocrine and neural systems can affect the immune system. There is evidence to suggest that by stimulating areas of the brain it is possible to affect certain immune responses, and also that stress (which is regulated by the endocrine system) can suppress immune responses: this is also reciprocal in that immune cells can affect endocrine and neural systems. The action of various endocrine products on the neural system is accepted to be an important stimulus of a wide variety of behaviors. These range from behaviors such as flight and sexual activity to sleeping and eating [44].

Computationly then, what does this have to say to AIS? It should be possible to explore the role of interaction between these three systems. One interesting avenue would be to design an AIS to help select the types of components which will be most useful when added to a control system at any moment (differentiation) and to remove components when they are proving harmful to the control system (apoptosis). The biological immune system cells select which action to perform by detecting properties of the cells and chemical environment through molecular interactions at membrane receptors. In an artificial system, similar properties can be detected by looking at activation states of artificial neurons and endocrine cells as well as global state information such as current consumption and battery levels. Thus the artificial immune system components can make similar decisions within the artificial context.

4.4 Methodology

Work in [6] proposes a conceptual framework that allows for the development of more biologically grounded AIS, through the adoption of an interdisciplinary

approach. Metaphors employed have typically been simple, but somewhat effective. However, as proposed in [6], through greater interaction between computer scientists, engineers, biologists and mathematicians, better insights into the workings of the immune system, and the applicability (or otherwise) of the AIS paradigm will be gained. These interactions should be rooted in a sound methodology in order to fully exploit the synergy. Modelling techniques can be employed such as Cellular automata [47] , petri nets [48, 49] and process calculi such as π-calculus [50], stochastic π-calculus [51], and ambient calculus [52]. These techniques exist and a great deal of work has been done in this area to understand the workings of numerous biological systems: as a community of AIS practitioners, it is time to take a look.

5 Conclusions: Challenges for Artificial Immune Systems

Based on the above observations, there are a number of challenges that are not easy to solve in a day:

- **Novel and Accurate Metaphors.** Typically naive approaches to extracting metaphors from the immune system have been taken. This has occurred as an accident of history, and AIS has slowly drifted away from its immunological roots. Time is now ripe for greater interaction with immunologists and mathematicians to undertake specific experimentation and create useful models, all of which can be used as a basis for abstraction into powerful algorithms;
- **Integration of Immune and Other Systems.** The immune system does not work in isolation. Therefore, attention should not only be paid to the potential of the immune system as inspiration, but also other systems with which the immune system interacts, in particular the immune, neural and endocrine systems. This will pave the way for a greater understanding of the role and function of the immune system and develop a new breed of immune inspired algorithms.
- **Theoretical basis for AIS.** Much work on AIS has concentrated on simple extraction of metaphors and direct application. Despite the creation of a framework for developing AIS, it still lacks significant formal and theoretical underpinning. AIS have been applied to a wide variety of problem domains, but a significant effort is still required to understand the nature of AIS and where they are best applied. For this, a more theoretical understanding is required;
- **Application of AIS.** Work to date in the realm of AIS has mainly concentrated on *what other paradigms do*, such as simple optimisation, learning and the like. this has happened as an accident of history, and whilst productive, the time is here to look for the *killer application* of AIS, or if not that radical, then applications where the benefit of adopting the immune approach is clear;

Acknowledgments. This paper is a result of many useful interactions with a wide variety of people in the AIS community and in particular during meetings that have taken place under the aegis of the EPSRC funded ARTIST[3]

[3] http://www.artificial-immune-systems.org/artist.htm

network in the UK. Particular thanks to: Mark Neal, Emma Hart, Susan Stepney, Andy Tyrrell, Andy Greenstead, Andy Hone, Hugo Van-den-Berg, Adrian Robins, Jamie Tycross, Al Lawson, Colin Johnson, Qi Chen and Paul Andrews for stimulating discussions.

References

1. Stepney, S., Clark, J., Tyrrell, A., Johnson, C., Timmis, J., Partridge, D., Adamatsky, A., Smith, R.: Journeys in non-classical computation: A grand challenge for computing research. Grand Challenge Report 7, National E-Science Centre, University of Edinburgh (2003)
2. de Castro, L., Timmis, J.: Artificial Immune Systems: A New Computational Intelligence Approach. Springer-Verlag (2002)
3. Burnet, F.: The Clonal Selection Theory of Acquired Immunity. Cambridge University Press (1959)
4. Jerne, N.K.: Towards a network theory of the immune system. Annals Immunol. **125C** (1974) 373–389
5. Aberola-Ila, J., Hogquist, K.A., Swan, K.A., Bevan, M.J., Perlmutter, R.M.: Positive and negative selection invoke distinct signaling pathways. J. Exp. Med **184** (1996) 9–18
6. Stepney, S., Smith, R., Timmis, J., Tyrrell, A.: Towards a Conceptual Framework for Artificial Immune Systems. In: LNCS 3239, Springer (2004) 53–64
7. Janeway, C., Medzhitov, R.: Innate immune recognition. Ann. Rev. Immunol. **20** (2002) 197–216
8. Janeway Jr., C.A., Paul, T.: Immunobiology: The Immune System in Health and Disease. Third edn. Garland Publishing, New York (1997)
9. Berek, C., Ziegner, M.: The maturation of the immune response. Immunology Today **14** (1993) 200 – 402
10. de Castro, L., Von Zuben, F.J.: Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation, Special Issue on Artificial Immune Systems **6** (2002) 239–251
11. Cutello, V., Nicosia, G., Parvone, M.: Exploring the capability of immune algorithms: A characterisation of hypermutation operators. In: LNCS 3239, Springer (2004) 263–276
12. Garrett, S.: Parameter-free, Adaptive Clonal Selection. In: Congress on Evolutionary Computing, CEC, IEEE (2004)
13. Watkins, A.: An Artificial Immune Recognition System. Mississippi State University. MSc Thesis. (2001)
14. Watkins, A., Timmis, J.: Exploiting Parallelism Inherent in AIRS, an Artificial Immune Classifier. In et al, G.N., ed.: Third International Conference on Artificial Immune Systems. Number 3239 in LNCS, Springer (2004) 427–438
15. Watkins, A., Xintong, B., Phadke, A.: Parallelizing an Immune-Inspired Algorithm for Efficient Pattern Recognition. In: Intelligent Engineering Systems through Artificial Neural Networks: Smart Engineering System Design: Neural Networks, Fuzzy Logic, Evolutionary Programming, Complex Systems and Artificial Life, ASME Press (2003) 224–230
16. Kim, J., Bentley, P.J.: Immune memory in the dynamic clonal selection algorithm. In Timmis, J., Bentley, P.J., eds.: Proceedings of the 1st International Conference on Artificial Immune Systems ICARIS, University of Kent at Canterbury, University of Kent at Canterbury Printing Unit (2002) 59–67

17. Secker, A., Freitas, A., Timmis, J.: AISEC: An Artificial Immune System for E-mail Classification. In: Proceedings of the Congress on Evolutionary Computation, IEEE (2003) 131–139
18. de Castro, L.N., Von Zuben, F.J.: aiNet: An Artificial Immune Network for Data Analysis. In:. Idea Group Publishing, USA (2001) 231–259
19. de Castro, L., Timmis, J.: Hierarchy and convergence of immune networks: Basic ideas and preliminary results. In Timmis, J., Bentley, P.J., eds.: Proceedings of the 1st International Conference on Artificial Immune Systems ICARIS, University of Kent at Canterbury, University of Kent at Canterbury Printing Unit (2002) 231–240
20. Timmis, J., Neal, M.J.: A Resource Limited Artificial Immune System for Data Analysis. Research and Development in Intelligent Systems XVII (2000) 19–32 Proceedings of ES2000, Cambridge, UK.
21. Knight, T., Timmis, J.: AINE: An immunological approach to data mining. In: Proc. of the IEEE International Conference on Data Mining. (2001) 297–304
22. Neal, M.: An Artificial Immune System for Continuous Analysis of Time-Varying Data. In Timmis, J., Bentley, P., eds.: Proceedings of the First International Conference on Artificial Immune Systems ICARIS, Canterbury, UK, UKC (2002) 76–85
23. Bersini, H.: Immune network and adaptive control. In: Proc. First European Conference on Artificial Life, MIT Press (1991) 217–226
24. Ishida, Y.: Active diagnosis by self-organisation : An approach by the immune network metaphor. In: Proceedings of the International Joint Conference on Artificial Intelligence, Nagoya, Japan., IEEE (1997) 1084–1089
25. Forrest, S., Perelson, A., Allen, L., Cherukuri, R.: Self-Nonself Discrimination in a Computer. In: Proc. of the IEEE Symposium on Research in Security and Privacy. (1994) 202–212
26. Forrest, S., Hofmeyr, S., Somayaji, A.: Computer Immunology. Communications of the ACM **40** (1997) 88–96
27. Esponda, F., Forrest, S., Helman, P.: A formal framework for positive and negative detection schemes. IEEE Transactions on Systems, Man and Cybernetics Part B **34** (2004) 357–373
28. Stibor, T., Bayarou, K., Eckert, C.: An investigation of R-chunk detector generation on higher alphabets. In: LNCS 3102. (2004) 26–30
29. Timmis, J., de Lemos, R., Ayara, M., Duncan, R.: Towards immune inspired fault tolerance in embedded systems. In Wang, L., Rajapakse, J., Fukushima, K., Lee, S., Yao, X., eds.: Proceedings of 9th International Conference on Neural Information Processing, IEEE (2002) 1459–1463
30. Kim, J., Bentley, P.J.: A model of gene library evolution in the dynamic clonal selection algorithm. In Timmis, J., Bentley, P.J., eds.: Proceedings of the 1st International Conference on Artificial Immune Systems ICARIS, University of Kent at Canterbury, University of Kent at Canterbury Printing Unit (2002) 182–189
31. Stibor, T., Mohr, P., Timmis, J., Eckert, C.: Is negative selection algorithm suitable for anomaly detection? In: Proceedings of GECCO 2005. LNCS. (2005)
32. Hart, E., Ross, P.: Studies on the implications of shape-space models for idiotypic networks. In: LNCS 3239, Springer (2004) 413–427
33. Bersini, H.: Reinforcement and recruitment learning for adaptive process control. In: Proc. Int. Fuzzy Association Conference (IFAC/IFIP/IMACS) on Artificial Intelligence in Real Time Control. (1992) 331–337
34. Bersini, H., Varela, F. In: The Immune Learning Mechansims: Recruitment, Reinforcement and their Applications. Chapman Hall (1994)

35. Hightower, R.R., Forrest, S.A., Perelson, A.S.: The evolution of emergent organization in immune system gene libraries. In Eshelman, L.J., ed.: Proc. 6th Int. Conf. Genetic Algorithms, Morgan Kaufmann (1995) 344–350
36. Freitas, A., Timmis, J.: Revisiting the Foundations of Artificial Immune Systems: A Problem Oriented Perspective. In: LNCS 2787, Springer (2003) 229–241
37. Segal, L., Cohen, I., eds.: Design Principles for the Immune System and Other Distributed Systems. Oxford University Press (2001)
38. Villalobos-Arias, M., Coello, C.A.C., Hernandez-Lerma, O.: Convergence analysis of a multiobjective artificial immune system algorithm. In: Lecture Noted in Computer Science 3239. (2004) 226 – 235
39. Grimmett, G.R., Stirzaker, D.R.: Probability and Random Processes. Oxford University Press (1982)
40. Hone, A., Kelsey, J.: Optima, extrema and artificial immune systems. In: Lecture Notes in Computer Science, Springer (2004) 89–98
41. Farmer, J.D., Packard, N.H., Perelson, A.S.: The Immune System, Adaptation, and Machine Learning. Physica **D** (1986) 187–204
42. Brzezniak, Z., Zastawniak, T.: Basic Stochastic Processes. Springer (1999)
43. Besendovsky, H.O., del Ray, A.: Immune-Neuro-Endocrine Interactions: Facts and Hypotheses. Nature **249** (1996) 356–358
44. Neal, M., Timmis, J.: : Once more unto the breach... towards artificial homeostasis? In: Recent Advances in Biologically Inspired Computing. IGP (2004)
45. WR, S.: Qualitative mathematical models of endocrine systems. Am. Journal of Physiology **245** (1983) 473–477
46. Liu, B., Deng, G.: An improved mathematical model of hormone secretion in the hypothalamo-pituitary-gonadal axis in man. J Theor Biol **150** (1991) 51–8
47. Sieburg, H.B., Clay, O.K.: The cellular device machine development system for modeling biology on the computer. Complex Systems **5** (1991) 575–601
48. Peleg, M., Yeh, I., Altman, R.B.: Modeling biological processes using workflow and petri net models. Bioinformatics **18** (2002) 825–837
49. Matsuno, H., Tanaka, Y., Aoshima, H., Doi, A., Matsui, M., Miyano, S.: Biopathways representation and simulation on hybrid functional petri net. In Silico Biology **3** (2003) 389–404
50. Kuttler, C., Niehren, J., Blossey, R.: Gene regulation in the pi calculus: Simulating cooperativity at the lambda switch. In: Proc Concurrent Models in Molecular Biology (Bioconcur 04). (2004)
51. Phillips, A., Cardelli, L.: A correct abstract machine for the stochastic pi-calculus. In: Proc Concurrent Models in Molecular Biology (Bioconcur'04), ENTCS. (2004)
52. Regev, A., Panina, E.M., Silverman, W., Cardelli, L., Shapiro, E.: Bioambients: An abstraction for biological compartments. Theoretical Computer Science (Special Issue on Computational Methods in Systems Biology) **325** (2004) 141–167

Author Index

Andrews, Paul S. 293
Anile, A.M. 307
Apolloni, Bruno 147, 214
Arca, Stefano 120

Bandini, Stefania 231
Barriga, Angel 157
Bassis, Simone 214
Baturone, Iluminada 157
Bede, Barnabás 188
Beltrami, Silvia 244
Bersani, F. 286
Bertazzoni, Stefano 95
Bertoni, Alberto 31
Bianchini, M. 38
Bose, J. 44
Brox, María 157
Brox, Piedad 157
Brunelli, U. 65

Camastra, Francesco 1
Campadelli, Paola 120
Campolo, Maurizio 171
Cannataro, M. 127
Casiraghi, Elena 120
Castellani, G.C. 286
Celada, Franco 231
Cesana, Eugenio 244
Ciaramella, Angelo 49
Clara, Narcís 163, 171
Clements, Mark A. 140
Clivio, Alberto 214, 244
Cutello, Vincenzo 307, 324

Di Gesù, Vito 179
Di Nola, Antonio 188

Esposito, Anna 70
Esposito, Antonietta M. 70

Fardis, Masoum 83
Filippone, Maurizio 207
Folci, Alessandra 244
Franceschi, C. 286
Friedman, Jerome H. 179
Furber, S.B. 44

Gaglio, Salvatore 113
Gaito, Sabrina 214
Galily, Mehdi 83, 90
Gentile, Antonio 140
Gersnoviez, Andrés 157
Giudicepietro, Flora 70
Gori, M. 38
Guzzi, P.H. 127

Hussein, Nashaat M. 157

Iannizzi, D. 147

Jalaie, Behzad 90

La Foresta, Fabio 78
Laface, Antonia Emanuela 244
Lanzarotti, Raffaella 120
Lee, Doheon 278
Lee, Kwang H. 278
Li, Jinyu 140
Linger, Henry 261
Lojacono, Roberto 95
Longo, Giuseppe 49

Malchiodi, Dario 147, 214
Mammone, Nadia 78
Manzoni, Sara 231
Martino, Gabriele 197
Masiello, Stefano 70
Masulli, Francesco 207
Mauri, Giancarlo 132
Mazza, T. 127
Mencattini, Arianna 95
Micheli, Alessio 10
Moniaci, Walter 95, 105
Montesino, Federico 157
Morabito, Francesco Carlo 78, 171
Moreno-Velo, Francisco 157
Motisi, Ignazio 113
Muselli, Marco 18, 23
Musirin, Ismail 338

Na, Dokyun 278
Narzisi, G. 307
Nicosia, Giuseppe 307, 324

Pagnoni, Anastasia 272
Paladino, Nicla 188
Parisi, Raffaele 57
Park, Inho 278
Pasero, Eros 95, 105
Pedrycz, W. 147
Petrosino, Alfredo 197
Piazza, V. 65
Pignato, L. 65
Pilato, Giovanni 113, 140
Pozzi, Sergio 132
Pugliese, Luca 70
Puzone, Roberto 231

Rahman, Titik Khawa Abdul 338
Remondini, D. 286
Roudsari, Farzad Habibipour 83, 90
Rovetta, Stefano 207
Rrushi, Julian L. 346

Salmeri, Marcello 95
Sánchez-Solano, Santiago 157
Sarti, L. 38
Scarpetta, Silvia 70
Scarselli, F. 38
Sestito, Antonio S. 10
Shapiro, J.L. 44
Siniscalchi, Sabato M. 140
Sorbello, Filippo 65, 140
Spinella, S. 307
Staiano, Antonino 49
Suliman, Saiful Izwan 338

Tagliaferri, Roberto 49
Tieri, P. 286
Timmis, Jon 293, 355
Tradigo, G. 127

Uncini, Aurelio 57
Urthaler, Astrid 244

Valensin, S. 286
Valentini, Giorgio 31
Vassallo, Giorgio 140
Vella, Filippo 113
Veltri, P. 127
Verondini, E. 286
Vigliano, Daniele 57
Visconti, Andrea 272
Vitabile, S. 65
Vizzari, Giuseppe 231

Yazdian, Ali 83

Zoppis, Italo 132