Journal Subline

LNCS 8240

Transactions on

Computational Collective Intelligence XII

Ngoc Thanh Nguyen
Editor-in-Chief

Springer

Lecture Notes in Computer Science 8240

Ngoc Thanh Nguyen (Ed.)

Transactions on Computational Collective Intelligence XII

Springer

Editor-in-Chief

Ngoc Thanh Nguyen
Wrocław University of Technology
Institute of Informatics
Wrocław, Poland
E-mail: ngoc-thanh.nguyen@pwr.edu.pl

ISSN 0302-9743 (LNCS) e-ISSN 1611-3349 (LNCS)
ISSN 2190-9288 (TCCI)
ISBN 978-3-642-53877-3 e-ISBN 978-3-642-53878-0
DOI 10.1007/978-3-642-53878-0

Springer Heidelberg New York Dordrecht London

CR Subject Classification (1998): I.2, H.3, H.2, I.5, F.2

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Welcome to the 12th volume of *Transactions on Computational Collective Intelligence* (TCCI). This volume of TCCI includes 10 interesting and original papers that were selected via the peer-review process. The first one, entitled "*Formalisms and Tools for Knowledge Integration Using Relational Databases*" by Stanisława Kluska-Nawarecka, Dorota Wilk-Kołodziejczyk, and Krzysztof Regulski, illustrates how attribute tables can be generated from the relational databases, to enable the use of approximate reasoning in decision-making processes. The solution presented allows transferring of the burden of the knowledge integration task to the level of databases. The second paper entitled *"A Clickstream-Based Web Page Importance Metric for Customized Search Engines"* by Fatemeh Ahmadi-Abkenari and Ali Selamat considers a metric of importance based on clickstream that is independent of the link structure of the Web graph. The authors review the Web page classification approach in order to use it in website importance calculation. The next paper entitled *"Opinion Analysis of Texts Extracted from the Social Web Contributions"* by Kristína Machová focuses on automatic opinion analysis related to Web discussions. An approach to the extraction of texts from Web forum discussion contributions and their commentaries as well as filtering these texts from irrelevant and junk information is presented and a method for solving basic problems of opinion analysis is developed. In the fourth paper entitled *"Time-and Personality Based Behaviors Under Cognitive Approach to Control the Negotiation Process with Incomplete Information",* the authors Amine Chohra, Arash Bahrammirzaee, and Kurosh Madani consider the problem of an adequate negotiation strategy with incomplete information. Two approaches are presented: a bargaining process and a cognitive approach based on the five-factor model in personality. Experimental results have shown that increased conciliatory aspects lead to an increased agreement point (price) and decreased agreement time, and increased aggressive aspects lead to a decreased agreement point and increased agreement time. The fifth paper entitled *"Web Server Support for e-Customer Loyalty Through QoS Differentiation"* by Grażyna Suchacka and Leszek Borzemski is connected with the problem of offering predictive service in e-commerce Web server systems under overload. The authors propose a method for priority-based admission control and scheduling of requests at the Web server system in order to differentiate quality of service (QoS) with regard to user-perceived delays. In the sixth paper, entitled *"Applying IPC-Based Clustering and Link Analysis to Patent Analysis on Thin-Film Solar Cell"* by Tzu-Fu Chiu, the author verifies applying IPC-based clustering in an experiment. Additionally, a link diagram is used to recognize technological trends. These tools are intended for assisting decision making in real-world applications. The seventh paper by Paul Anderson, Shahriar Bijani, and Herry Herry is entitled

"Multi-agent Virtual Machine Management Using the Lightweight Coordination Calculus". It describes an LCC-based system for specifying the migration within and between data centers. The authors present example models of policies, machine allocation, and migration. In the next paper titled *"Modelling Evacuation at Crisis Situations by Petri Net-Based Supervision"*, the author, Frantisek Capkovic, presents case studies of building evacuation and endangered areas using Place/transition Petri nets. He also describes the difference between these two approaches and points out the possibilities of their mutual complementing. The ninth paper entitled *"Particle Swarm Optimization with Disagreements on Stagnation"* by Andrei Lihu, Stefan Holban, and Oana-Andreea Popescu introduces a modified particle swarm optimization (PSO) that exhibits the so-called extreme social disagreements among its wandering particles in order to resolve the stagnation when it occurs during search. The results have shown that the proposed approach may help PSO escape stagnation in most of the situations in which it was tested. The last paper, *"Evolutionary Algorithm with Geographic Heuristics for Urban Public Transportation"* by Jolanta Koszelew and Krzysztof Ostrowski, presents a new evolutionary algorithm with a special geographic heuristics. It solves the bi-criteria version of the Routing Problem in Urban Public Transportation Networks often called the Bus Routing Problem.

TCCI is a peer-reviewed and authoritative journal dealing with the working potential of computational collective intelligence (CCI) methodologies and applications, as well as emerging issues of interest to academics and practitioners. The research area of CCI has been growing significantly in recent years and we are very thankful to everyone within the CCI research community who has supported the *Transactions on Computational Collective Intelligence* and its affiliated events including the International Conferences on Computational Collective Intelligence (ICCCI). Its last event (ICCCI 2013) was held in Craiova, Romania, during September 11–13, 2013. The next ICCCI event (ICCCI 2014) will be held in Seoul in September 2014. It is a tradition that after each ICCCI event we invite authors of selected papers to extend and submit them for publication in TCCI.

We would like to thank all the authors, the Editorial Board members, and the reviewers for their contributions to TCCI. We express our sincere thanks to all of them. Finally, we would also like to express our gratitude to the LNCS editorial staff of Springer and Alfred Hofmann, for supporting the TCCI journal.

September 2013 Ngoc Thanh Nguyen

Transactions on Computational Collective Intelligence

This Springer journal focuses on research in applications of the computer-based methods of computational collective intelligence (CCI) and their applications in a wide range of fields such as Semantic Web, social networks, and multi-agent systems. It aims to provide a forum for the presentation of scientific research and technological achievements accomplished by the international community.

The topics addressed by this journal include all solutions of real-life problems for which it is necessary to use computational collective intelligence technologies to achieve effective results. The emphasis of the papers published is on novel and original research and technological advancements. Special features on specific topics are welcome.

Editor-in-Chief

Co-Editor-in-Chief

Editorial Board

Table of Contents

Formalisms and Tools for Knowledge Integration Using Relational Databases

Stanisława Kluska-Nawarecka[1], Dorota Wilk-Kołodziejczyk[2], and Krzysztof Regulski[3]

[1] Foundry Research Institute, Cracow, Poland
[2] Andrzej Frycz Modrzewski University, Cracow, Poland
[3] AGH University of Science and Technology, Cracow, Poland
nawar@iod.krakow.pl, wilk.kolodziejczyk@gmail.com, regulski@agh.edu.pl

Abstract. Until now, the use of attribute tables, which enable approximate reasoning in tasks such as knowledge integration, has been posing some difficulties resulting from the difficult process of constructing such tables. Using for this purpose the data comprised in relational databases should significantly speed up the process of creating the attribute arrays and enable getting involved in this process the individual users who are not knowledge engineers. This article illustrates how attribute tables can be generated from the relational databases, to enable the use of approximate reasoning in decision-making process. This solution allows transferring the burden of the knowledge integration task to the level of databases, thus providing convenient instrumentation and the possibility of using the knowledge sources already existing in the industry. Practical aspects of this solution have been studied on the background of the technological knowledge of metalcasting.

Keywords: attribute table, knowledge integration, databases, rough sets, methods of reasoning intelligent systems, knowledge management, decision support systems, databases, rough sets, methods of reasoning, artificial intelligence, the logic of plausible reasoning, cast.

1 Introduction

The aim of the study is to present the latest results of research on inference systems based on the use of rough sets and logic of plausible reasoning. Application of the new methodology for the acquisition of rules – semi-automatic generation of decision-making tables based on specially prepared databases - can greatly reduce the time and labour-intensive process of building a knowledge base. A novelty is here also the proposed approach to the application of the logic of plausible reasoning (LPR) to represent uncertain knowledge about a specific industrial process. In recent years, some disappointment was observed as regards the class of expert systems, to a large extent resulting from the difficulty associated with the acquisition of knowledge needed in the reasoning process. The use of the developed technique assisting the

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 1–20, 2013.

construction of knowledge bases can shed new light on the inference systems. The intention of the authors of the study is not only the disclosure of a methodology used in the representation of given domain knowledge, but also its reference to specific system solutions. The presented embodiment of modules of an information system dedicated for the foundry is currently at the stage of being tested and will be subject to further integration. The rough logic based on rough sets developed in the early '80s by Prof. Zdzislaw Pawlak [1] is used in the analysis of incomplete and inconsistent data. Rough logic enables modelling the uncertainty arising from incomplete knowledge which, in turn, is the result of the granularity of information. The main application of rough logic is classification, as logic of this type allows building models of approximation for a family of the sets of elements, for which the membership in sets is determined by attributes. In classical set theory, the set is defined by its elements, but no additional knowledge is needed about the elements of the universe, which are used to compose the set. The rough set theory assumes that there are some data about the elements of the universe, and these data are used in creation of the sets. The elements that have the same information are indistinguishable and form the, so-called, elementary sets. The set approximation in a rough set theory is achieved through two definable sets, which are the upper and lower approximations. The reasoning is based on the attribute tables, i.e. on the information systems, where the disjoint sets of conditional attributes C and decision attributes D are distinguished (where A is the total set of attributes and $A = C \cup D$.).

So far, attribute tables used for approximate reasoning at the Foundry Research Institute in Cracow were generated by knowledge engineers and based on data provided by experts, using their specialist knowledge [2]. The reasoning was related with the classification of defects in steel castings. Yet, this process was both time-consuming and expensive. Automatic or at least semi-automatic creation of attribute arrays would allow the integration of technological knowledge, already acquired in the form of e.g. relational databases, with the methods of approximate reasoning, the results of which are becoming increasingly popular in processes where it is necessary to operate on knowledge incomplete and uncertain [3, 4, 5].

Another, though only in the phase of experiment, concept of inference using attribute tables with incomplete knowledge has been based on the applied formalism of the logic of plausible reasoning (LPR). This paper presents the basic concepts of LPR and the methodology used in constructing the rules for the diagnosis of casting defects.[11]

2 Relational Data Model

2.1 Set Theory vs Relational Databases

The relational databases are derived in a straight line from the set theory, which is one of the main branches of mathematical logic. Wherever we are dealing with relational databases, we de facto operate on sets of elements. The database is presented in the form of arrays for entities, relationships and their attributes. The arrays are structured in the following way: entities – lines, attributes - columns, and relationships - attributes. The arrays, and thus the entire database, can be interpreted as relations in a mathematical meaning of this word. Also operations performed in the database are to be understood as operations on relations. The basis of such model is the relational algebra that describes these operations and their properties. If sets A_1, A_2, ... A_n are given, the term "relation r" will refer to any arbitrary subset of the Cartesian product A_1 A_2 ... A_n. A relation of this type gives a set of tuples $(a_1, a_2, ..., a_n)$, where each $a_i \in A_i$. In the case of data on casting defects, the following example can be given:

- header corresponds to the scheme of relation,
- elements of the relationship - tuples are represented by lines.

It is customary to present a model of a database - schema of relationship with ER (entity relationship) models to facilitate the visualisation. The simplest model of a database on defects in steel castings can take the form shown in Figure 1.

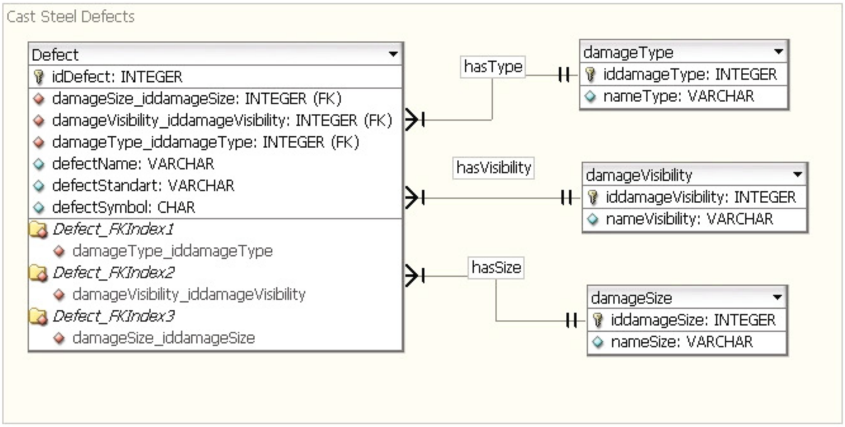


Fig. 1. A fragment of ER database model for defects in steel castings

2.2 Generating Attribute Tables Based on Relational Databases

As can be concluded from this brief characterisation of the relational databases, even their structure, as well as possible set theory operations (union, intersection, set difference, and Cartesian product) serve as a basis on which the attribute tables are next constructed, taking also the form of relationships. Lines in an attribute array define the decision rules, which can be expressed as:

$$\text{IF} \ldots \text{THEN} \ldots\text{:} \quad X \rightarrow Y \tag{1}$$

where $X= x_1 \wedge x_2 \wedge .. \wedge x_n$ is the conditional part of a rule, and Y is its decision part. Each decision rule sets decisions to be taken if conditions given in the table are satisfied. Decision rules are closely related with approximations. Lower approximations of decision classes designate deterministic decision rules clearly defining decisions based on conditions, while upper approximations appoint the non-deterministic decision rules.

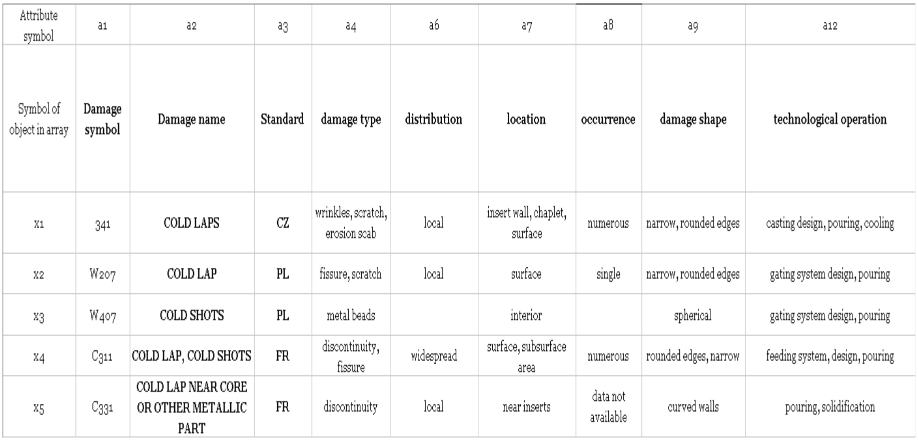
The object of this study is not to discover new knowledge, since knowledge of the relationships between variables is derived from the expert. It is the expert who shows the knowledge engineer how to combine the collected data to achieve the results, which are decision tables. It is therefore possible, to generate an array of attributes from a relational database. For this purpose, the only thing to be done is to select, based on the expert knowledge, attributes from the schema of relationships, which have to (and can) play the role of decision attributes in the array, and also the set of conditional attributes based on which the classification process will be done. This method of preparation of the decision-making tables gives considerable labour savings for both the knowledge engineer and the expert who is exempt from manual data preparation.

The attributes with an area ordered by preference are called criteria because they refer to assessment in a specific scale of preference. An example is the line in a decision table, i.e. an object with a description and class assignment.

Experiment proved that it is possible, therefore, to generate an attribute table using a relational database. The only requirement is to select from the schema of relations, basing on the expert knowledge, the attributes that should (and can) play the role of decision attributes in the table, and also a set of conditional attributes, which will serve as a basis for the classification process. Such attribute table was developed with experts from Foundry Research Institute in Cracow.

In the case of Table 1, the conditional attributes will be attributes a_4-a_{12}, and the decision attributes will be a_1-a_3, since the decision is proper classification of defect.

Table 1. Fragment of attribute table for defects in steel castings



Attribute symbol	a1	a2	a3	a4	a6	a7	a8	a9	a12
Symbol of object in array	**Damage symbol**	**Damage name**	**Standard**	**damage type**	**distribution**	**location**	**occurrence**	**damage shape**	**technological operation**
x1	341	COLD LAPS	CZ	wrinkles, scratch, erosion scab	local	insert wall, chaplet, surface	numerous	narrow, rounded edges	casting design, pouring, cooling
x2	W207	COLD LAP	PL	fissure, scratch	local	surface	single	narrow, rounded edges	gating system design, pouring
x3	W407	COLD SHOTS	PL	metal beads		interior		spherical	gating system design, pouring
x4	C311	COLD LAP, COLD SHOTS	FR	discontinuity, fissure	widespread	surface, subsurface area	numerous	rounded edges, narrow	feeding system, design, pouring
x5	C331	COLD LAP NEAR CORE OR OTHER METALLIC PART	FR	discontinuity	local	near inserts	data not available	curved walls	pouring, solidification

Creating attribute tables we are forced to perform certain operations on the database. The resulting diagram of relationships will be the sum of partial relations, and merging will be done through one common attribute. In the case of an attribute table, the most appropriate type of merging will be external merging, since in the result we expect to find all tuples with the decision attributes and all tuples with the conditional attributes, and additionally also those tuples that do not have certain conditional attributes, and as such will be completed with NULL values.

3 Classification Using Rough Set Theory

Issues concerning the theory of rough sets are widely reported in the literature, but known applications are tailored to the specific set of practical tasks. In this study, an attempt was made to show how the concepts and formalisms of rough sets can be used in the knowledge of the foundry processes providing, at the same time, tools for relevant solutions. [14, 15, 16, 17]

The basic operations performed on rough sets are the same as those performed on common sets. Additionally, several new concepts not used in common sets are introduced.

3.1 Indiscernibility Relations

For each subset of attributes, the pairs of objects are in the relation of indiscernibility if they have the same values for all attributes from the set B, which can be written as:

$$IND(B) = \{x_i, x_j \in U : \forall b \in B,\ f(x_i, b) = f(x_j, b)\} \tag{2}$$

The relation of indiscernibility of elements x_i and x_j is written as x_i IND(B) x_j. Each indiscernibility relation divides the set into a family of disjoint subsets, also called abstract classes (equivalence) or elementary sets. Different abstract classes of the indiscernibility relation are called elementary sets and are denoted by *U/IND (B).* Classes of this relation containing object xi are denoted by *[x_i]IND(B).* So, the set *[x_i]IND(B)* contains all these objects of the system *S,* which are indistinguishable from object xi in respect of the set of attributes *B* [6]. The abstract class is often called an elementary or atomic concept, because it is the smallest subset in the universe *U* we can classify, that is, distinguish from other elements by means of attributes ascribing objects to individual basic concepts.

The indiscernibility relationship indicates that the information system is not able to identify as an individual the object that meets the values of these attributes under the conditions of uncertainty (the indeterminacy of certain attributes which are not included in the system). The system returns a set of attribute values that match with certain approximation the identified object.

Rough set theory is the basis for determining the most important attributes of an information system such as attribute tables, without losing its classificability as compared with the original set of attributes. Objects having identical (or similar) names, but placed in different terms, make clear definition of these concepts impossible. Inconsistencies should not be treated as a result of error or information noise only. They may also result from the unavailability of information, or from the natural granularity and ambiguity of language representation.

To limit the number of redundant rules, such subsets of attributes are sought which will retain the division of objects into decision classes the same as all the attributes. For this purpose, a concept of the reduct is used, which is an independent minimal subset of attributes capable of maintaining the previous classification (distinguishability) of objects. The set of all reducts is denoted by RED (A).

With the notion of reduct is associated the notion of core (kernel) and the interdependencies of sets. The set of all the necessary attributes in B is called kernel (core) and is denoted by core (B). Let $B \subseteq A$ and $a \in B$. We say that attribute a is superfluous in B when:

$$\mathrm{IND(B)= IND(B - \{a\})} \tag{3}$$

Otherwise, the attribute a is indispensable in B. The set of attributes B is independent if for every a$\in$B attribute a is indispensable. Otherwise the set is dependent.

The kernel of an information system considered for the subset of attributes $B \subseteq A$ is the intersection of all reducts of the system

$$\mathrm{core(B)}=\cap\ \mathrm{RED(A)} \tag{4}$$

Checking the dependency of attributes, and searching for the kernel and reducts is done to circumvent unnecessary attributes, which can be of crucial importance in optimising the decision-making process. A smaller number of attributes means shorter dialogue with the user and quicker searching of the base of rules to find an adequate procedure for reasoning. In the case of attribute tables containing numerous sets of unnecessary attributes (created during the operations associated with data mining), the problem of reducts can become a critical element in building a knowledge base. A completely different situation occurs when the attribute table is created in a controlled manner by knowledge engineers, e.g. basing on literature, expert knowledge and/or standards, when the set of attributes is authoritatively created basing on the available knowledge of the phenomena. In this case, the reduction of attributes is not necessary, as it can be assumed that the number of unnecessary attributes (if any) does not affect the deterioration of the model classificability.

3.2 Query Language

Query language in information systems involves rules to design questions that allow the extraction of information contained in the system. If the system represents information which is a generalisation of the database in which each tuple is the realisation of the relationship which is a subset of the Cartesian product (data patterns or templates), the semantics of each record is defined by a logical formula assuming the form of [8]:

$$_i=[A_1=a_{i,1}] \wedge [A_2=a_{i,2}] \wedge \ldots \wedge [A_n=a_{i,n}] \tag{5}$$

The notation $A_j=a_{i,j}$ means that the formula φ_i is true for all values that belong to the set $a_{i,j}$. Hence, if $a_{i,j}=\{a_1, a_2, a_3\}$, $A_j=a_{i,j}$ means that $A_i=a_1 \vee A_i=a_2 \vee A_i=a_3$, while the array has a corresponding counterpart in the formula:

$$\Psi= \varphi_1 \vee \varphi_2 \vee .. \vee \varphi_m \tag{6}$$

If the array represents some rules, the semantics of each line is defined as a formula:

$$_i=[A_1=a_{i,1}] \wedge [A_2=a_{i,2}] \wedge \ldots \wedge [A_n=a_{i,n}] \Rightarrow [H=h_i] \qquad (7)$$

On the other hand, to the array of rules is corresponding a conjunction of formulas describing the lines.

The attribute table (Table 1.) comprises a set of conditional attributes $C=\{a_4, a_5, a_6, a_7, a_8, a_9\}$ and a set of decision attributes $D= \{a_1, a_2, a_3\}$. Their sum forms a complete set of attributes $A=C\cup D$. Applying the rough set theory, it is possible to determine the elementary sets in this table. For example, for attribute a4 (damage type), the elementary sets will assume the form of:

- $E_{\text{wrinkles}} = \{\emptyset\}$; $E_{\text{scratch}} = \{\emptyset\}$; $E_{\text{erosion scab}} = \{\emptyset\}$; $E_{\text{fissure}} = \{\emptyset\}$;
- $E_{\text{wrinkles, scratch, erosion scab}} = \{x_1\}$;
- $E_{\text{cold shots}} = \{x_3\}$;
- $E_{\text{fissure, scratch}} = \{x_2\}$;
- $E_{\text{discontinuity}} = \{x_5\}$;
- $E_{\text{discontinuity, fissure}} = \{x_4\}$;
- $E_{\text{wrinkles, scratch, erosion scab, fissure}} = \{\emptyset\}$; $E_{\text{wrinkles, scratch, erosion scab, fissure, cold shots}} = \{\emptyset\}$;
- $E_{\text{wrinkles, scratch, erosion scab, cold shots}} = \{\emptyset\}$; $E_{\text{discontinuity, fissure, cold shots}} = \{\emptyset\}$;
- $E_{\text{discontinuity, fissure, wrinkles, scratch, erosion scab}} = \{\emptyset\}$;
- etc.

Thus determined sets represent a partition of the universe done in respect of the relationship of indistinguishability for an attribute “distribution”. This example shows one of the steps in the mechanism of reasoning with application of approximate logic. Further step is determination of the upper and lower approximations in the form of a pair of precise sets. Abstract class is the smallest unit in the calculation of rough sets. Depending on the query, the upper and lower approximations are calculated by summing up the appropriate elementary sets.

The sets obtained from the Cartesian product can be reduced to the existing elementary sets.

Query example: t1= (damage type, {discontinuity, fissure}) ˙ (distribution, {local})

When calculating the lower approximation, it is necessary to sum up all the elementary sets for the sets of attribute values which form possible subsets of sets in a query:

$\underline{S}(t_1)$ = (damage type, {discontinuity, fissure}) · (distribution, {local}) + (damage type, {discontinuity}))] · (distribution, { local})

The result is a sum of elementary sets forming the lower approximation:

$$E_{\text{discontinuity, local}} \cup E_{\text{discontinuity, fissure, local}} = \{x_5\}$$

The upper approximation for the submitted query is:

$\underline{S}(t_1)$ = (damage type, {discontinuity, fissure}) · (distribution, {local}) + (damage type, {discontinuity}))] · (distribution, {local}) + (damage type, {discontinuity, scratch}))] · (distribution, {local})

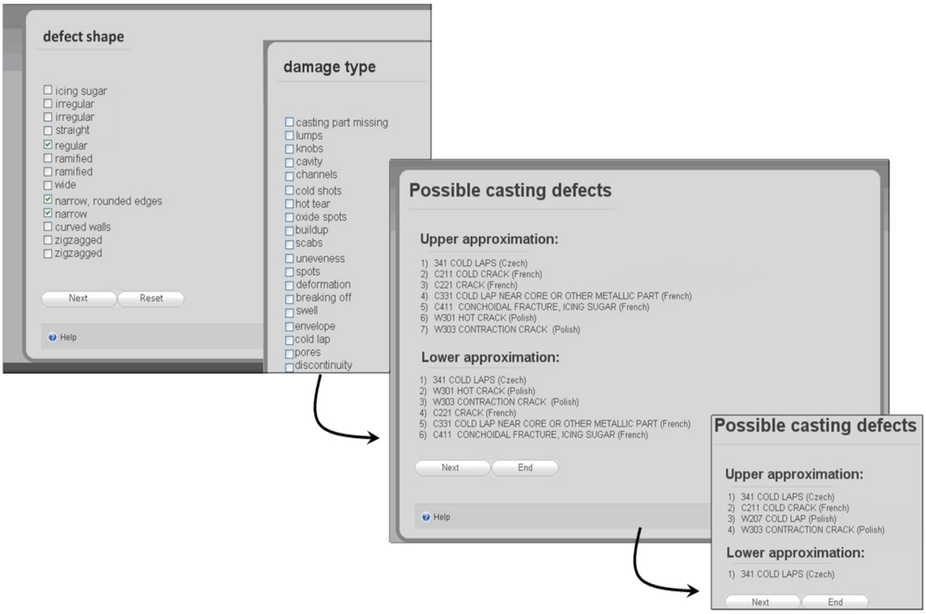
The result is a sum of elementary sets forming the upper approximation:

$$E_{\text{discontinuity,local}} \cup E_{\text{fissure,scratch,local}} \cup E_{\text{discontinuity,fissure,local}} = \{ x_{2,}\ x_{5,} \}$$

3.3 Reasoning Using RoughCast System

The upper and lower approximations describe a rough set inside which there is the object searched for and defined with attributes. The practical aspect of the formation of queries is to provide the user with an interface such that its use does not require knowledge of the methods of approximate reasoning, or semantics of the query language.

It was decided in the Foundry Research Institute in Cracow to implement the interface of a RoughCast reasoning system in the form of an on-line application leading dialogue with the user applying the interactive forms (Fig. 2a). The system offers functionality in the form of an ability to classify objects basing on their attributes [7]. The attributes are retrieved from the database, to be presented next in the form of successive lists of the values to a user who selects appropriate boxes. In this way, quite transparently for the user, a query is created and sent to the reasoning engine that builds a rough set. However, to make such a dialogue possible without the need for the user to build a query in the language of logic, the system was equipped with an interpreter of queries in a semantics much more narrow than the original Pawlak semantics. This approach is consistent with the daily cases of use when the user has to deal with specific defects, and not with hypothetical tuples. Thus set up inquiries are limited to conjunctions of attributes, and therefore the query interpreter has been equipped with one logical operator only. The upper and lower approximations are presented to the user in response.



defect shape

- ☐ icing sugar
- ☐ irregular
- ☐ irregular
- ☐ straight
- ☑ regular
- ☐ ramified
- ☐ ramified
- ☐ wide
- ☑ narrow, rounded edges
- ☑ narrow
- ☐ curved walls
- ☐ zigzagged
- ☐ zigzagged

damage type

- ☐ casting part missing
- ☐ lumps
- ☐ knobs
- ☐ cavity
- ☐ channels
- ☐ cold shots
- ☐ hot tear
- ☐ oxide spots
- ☐ buildup
- ☐ scabs
- ☐ uneveness
- ☐ spots
- ☐ deformation
- ☐ breaking off
- ☐ swell
- ☐ envelope
- ☐ cold lap
- ☐ pores
- ☐ discontinuity

Possible casting defects

Upper approximation:

1) 341 COLD LAPS (Czech)
2) C211 COLD CRACK (French)
3) C221 CRACK (French)
4) C331 COLD LAP NEAR CORE OR OTHER METALLIC PART (French)
5) C411 CONCHOIDAL FRACTURE, ICING SUGAR (French)
6) W301 HOT CRACK (Polish)
7) W303 CONTRACTION CRACK (Polish)

Lower approximation:

1) 341 COLD LAPS (Czech)
2) W301 HOT CRACK (Polish)
3) W303 CONTRACTION CRACK (Polish)
4) C221 CRACK (French)
5) C331 COLD LAP NEAR CORE OR OTHER METALLIC PART (French)
6) C411 CONCHOIDAL FRACTURE, ICING SUGAR (French)

Possible casting defects

Upper approximation:

1) 341 COLD LAPS (Czech)
2) C211 COLD CRACK (French)
3) W207 COLD LAP (Polish)
4) W303 CONTRACTION CRACK (Polish)

Lower approximation:

1) 341 COLD LAPS (Czech)

Fig. 2. Forms selecting attribute values in the RoughCast system, upper and lower approximations calculated in a single step of reasoning and the final result of dialogue for the example of "cold lap" defect according to the Czech classification system

The RoughCast system enables the exchange of knowledge bases. When working with the system, the user has the ability to download the current knowledge base in a spreadsheet form, edit it locally on his terminal, and update in his system. The way the dialogue is carried out depends directly on the structure of decision-making table and, consequently, the system allows reasoning using arrays containing any knowledge, not just foundry knowledge.

The issue determining to what extent the system will be used is how the user can acquire a knowledge base necessary to operate the system. So far, as has already been mentioned, this type of a database constructed in the form of an array of attributes was compiled by a knowledge engineer from scratch. However, the authors suggest to develop a system that would enable acquiring such an array of attributes in a semi-atomatic mode through, supervised by an expert, the initial round of queries addressed to a relational database in a SQL language (see 2.2)

4 The Logic of Plausible Reasoning

The logic of plausible reasoning (abbreviated as LPR) was proposed by A. Collins and R. Michalski in 1989. [11] LPR is the result of analysis of a large number of descriptions of reasoning used by people when answering the questions. LPR creators were able to identify certain patterns of reasoning used by man [12] and create a formal system of inference based on the variable-valued logic calculus [11], allowing a representation of those patterns. The test results confirming the possibility of representing "human reasoning" in the LPR have been presented in [13].

The essence of the LPR is to create a formalised description of relationships that exist between the concepts occurring in a certain hierarchy (e.g. in the form of graphs). The vertices represent classes of objects, and objects (groups of defects, defects) or a manifestation of objects (the occurrence of a specific cause of defects). The edges represent relationships between concepts [13].

The LPR introduces a set of concepts and relationships that express the properties of the described knowledge.

Terms, values and statements describe various concepts in a given hierarchy. For a group of defects in metal objects these can assume the following form:

Term A - top of a hierarchy, B - argument of a term

Form of term A (B): (8)

A sample entry

```
castingDefect(coldCrack)
```

This entry indicates that one of the defects in castings is a cold crack, where the concept of defect, which is a vertex of the term, is the concept referring to all the defects, while argument of the term is one of the defects called cold crack.

Value - a concept or a set of concepts.

A sample entry

```
Cause(shrinkageCavity)=hotSpots
```

where hot spots is the name of one of the causes of the formation of the defect called shrinkage cavity, and in terms of LPR, this is the concept representing a value of a set that is attributed to the statement

```
Cause(shrinkageCavity).
```

The statements express a relationship that exists between the term and a value

```
defect(hotCrack)=absent (statement)
```

this statement indicates that in the case under discussion, the defect hot crack is absent

```
defect(coldCrack)≠blue (negative statement)
```

This entry indicates that among the values attributed to the term defect (coldCrack) there is no word blue.

Other examples of formulation of the statements referred to the defects in cast products:

```
defectCause(mechanicalDamage)={wrongRemovalGates, wrongRemovalRisers,
wrongTrimming,(possibly another cause) wrongRemovalOther-
TechnologicalAllowances}
defectCause(coldCrack)={damageDuringTrimming, DamageDu-
ringManipulation}
```

The arguments of terms are here the names of individual defects, while the sets of values of the terms are causes of these defects.

The relationships that define mutual relations between concepts in the hierarchy are GEN and SPEC

$$A_1 \text{ GEN } A_2 \text{ in CX } (A,D(A)) \tag{9}$$

This entry indicates that the concept A1 is placed in the hierarchy above A_2 in the context (A, D (A))

$$A_2 \text{ SPEC } A_1 \text{ in CX } (A,D(A)) \tag{10}$$

This entry indicates that the concept A2 is placed in the hierarchy below A_1 in the context (A, D(A))

A_2, A_1 – the described concepts,

CX (A,D(A)) - defines the context in which the dependencies are considered,

A – the concept lying in the hierarchy above A_2, A_1

D(A) – the term specifying the properties characteristic of A,

For example, the entry:

```
castingDefects GEN internalDefects in CX (metalProducts,
damage(metalProducts))
```

means that the concept casting defects is placed in the hierarchy above the concept internal defects (it concerns the internal defects in castings), but the dependence is considered in the context of damage of metal products.

By contrast, the statement below indicates that the defect of shape is placed in the hierarchy below the casting defect, which means that it is a kind of casting defect.

```
shapeDefect SPEC castingDefect
```

Relationships SIM and DIS describe the similarity and dissimilarity of concepts:

$$A_1 \text{ SIM A2 in CX(B,D(B))} \tag{11}$$

This entry means that the concept A_1 is similar to the concept A_2 in the context (A, D(A))

$$A_1 \text{ DIS } A_2 \text{ in CX(B,D(B))} \tag{12}$$

This entry means that the concept A_1 is dissimilar to the concept A_2 in the context (A, D (A))

Correspondingly, we can write:

```
shapeDefect SIM continuityBreaks in CX (castingDefect,
defects(castingDefect))
```

This example indicates that the defect of shape is similar to the defect called break of continuity. These defects are similar to each other in this meaning that both are casting defects (as mentioned by the context of similarities).

The next group includes mutual dependences and mutual implications determining the degree of concept equivalence. In the interrelation:

$$D_1(A) \leftrightarrow D_2(f(A)) \tag{13}$$

an additional element can be:

"+" indicating a positive relationship, which means that with the increasing value of the first term, the value of the second term is increasing, too

"-" indicating a negative relationship, which means that with the increasing value of the first term, the value of the second term is decreasing

```
castingDefect(casting) ↔ castingDesign(casting)
```

This entry indicates that the occurrence of defect in casting affects the casting design (i.e. someone who is responsible for the casting performance has in mind not only the shape of the casting but also the need to design it in a way such as to reduce to minimum the least likelihood of failure). The other way round it works in a way such that the casting design affects the defect formation.

Mutual implication:

$$D_1(A)=R_1 \Leftrightarrow D_2(f(A))=R_2 \tag{14}$$

that on the example can be represented as follows:

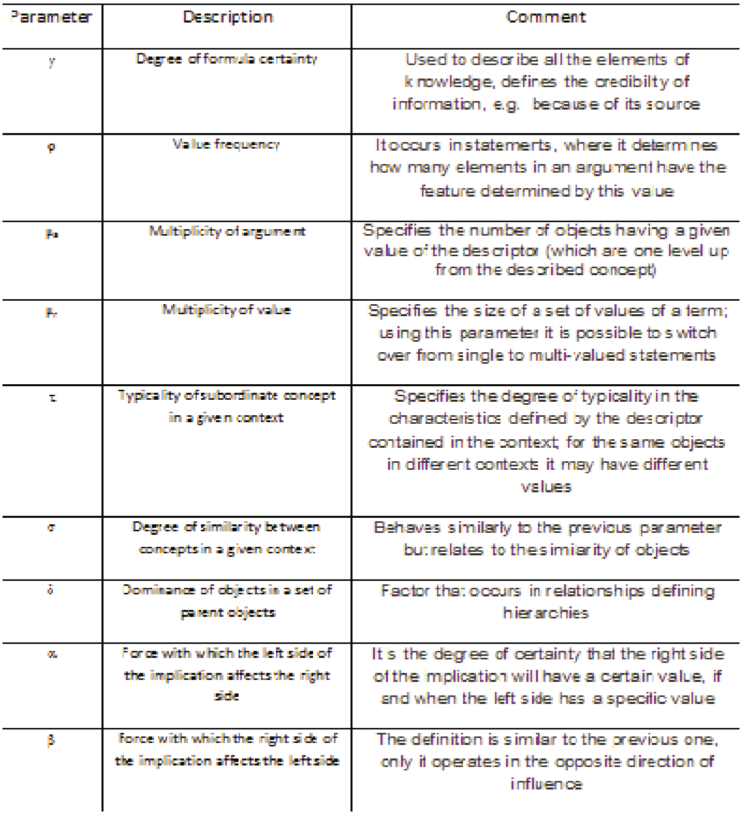
```
castingKnockingout(casting)=correct ⇔ mechanicalDa-
mage(casting)=absent
```

The above given example of mutual implication can be interpreted in a way such that if the casting is knocked out properly, the casting defect designated as a mechanical damage shall not occur. The other way round, this entry can be interpreted that if there has been no mechanical damage (in the sense of casting defect), the casting knocking out technique was correct.

4.1 Parameters Determining the Uncertainty of Information

A very important advantage of the logic of plausible reasoning is the ability to introduce parameters which enable representing the uncertainty of information. Such parameters are shown in Table 2, while Table 3 shows which relationships can be determined with these parameters.

Table 2. List of parameters of the knowledge uncertainty used in LPR



Parameter	Description	Comment
γ	Degree of formula certainty	Used to describe all the elements of knowledge, defines the credibilty of information, e.g. because of its source
φ	Value frequency	It occurs in statements, where it determines how many elements in an argument have the feature determined by this value
μ_a	Multiplicity of argument	Specifies the number of objects having a given value of the descriptor (which are one level up from the described concept)
μ_v	Multiplicity of value	Specifies the size of a set of values of a term; using this parameter it is possible to switch over from single to multi-valued statements
τ	Typicality of subordinate concept in a given context	Specifies the degree of typicality in the characteristics defined by the descriptor contained in the context; for the same objects in different contexts it may have different values
σ	Degree of similarity between concepts in a given context	Behaves similarly to the previous parameter but relates to the similarity of objects
δ	Dominance of objects in a set of parent objects	Factor that occurs in relationships defining hierarchies
α	Force with which the left side of the implication affects the right side	It's the degree of certainty that the right side of the implication will have a certain value, if and when the left side has a specific value
β	force with which the right side of the implication affects the left side	The definition is similar to the previous one, only it operates in the opposite direction of influence

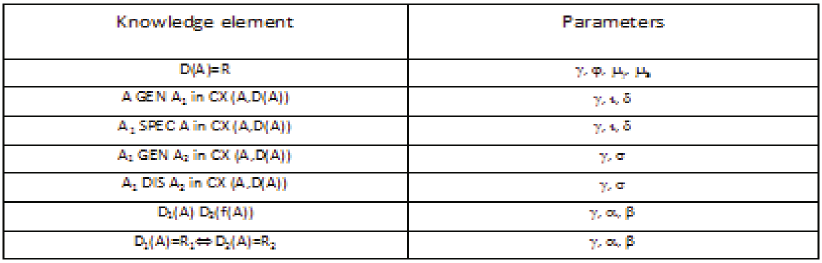
The above mentioned parameters are important in the process of inference using LPR, as illustrated by the following examples:

If we want to say that often during trimming (cutting of flashes) the castings breaks, we can write:

```
defectCause(coldCrack)=(damageDuringTrimming)γ=large
φ=frequent
```

parameter φ in this case determines with what frequency in the statement cause of cold crack defect may appear the value damage during trimming, while parameter γ defines the certainty of this formula.

Table 3. The way of assigning parameters to relationships



Knowledge element	Parameters
D(A)=R	γ, φ, μ_γ, μ_t
A GEN A_1 in CX (A,D(A))	γ, τ, δ
A_1 SPEC A in CX (A,D(A))	γ, τ, δ
A_1 GEN A_2 in CX (A,D(A))	γ, σ
A_1 DIS A_2 in CX (A,D(A))	γ, σ
D_1(A) D_2(f(A))	γ, α, β
$D_1(A)=R_1 \Leftrightarrow D_2(A)=R_2$	γ, α, β

5 Inference Based on Attribute Tables Using LPR

Slightly different mode of inference that can be used with attribute tables will be inference based on the logic of plausible reasoning. Here, the first step is to define the terms.

Writing the definition of the term is done by inserting

```
defectName (Fold)
```

In this notation *defectName* is a non-terminal concept, quite often on the top of the hierarchy, while *Fold* is the term value.

In this way, one should define all the objects that are in the attribute table, that is, the names of defects, location, distribution, shape, and place in the technological process. The technique by which this is done is to create a column header from the attribute table, followed by the next element in the table. For the purpose of the logic of plausible reasoning it has been assumed that the terms shall be recorded as separate operations, or entities. This means that even if in the attribute table the term is defined as a kind of damage, the scratches, wrinkles and erosion scabs will be assigned as a designation to one single defect, but because each of these designations appears as a separate designation single or grouped in a different way together with other defects, so the terms entered will be treated as separate entities which next, with the help of a hierarchy, will be assigned to individual defects. So, the task of the hierarchy is to define the relationships that prevail between the data in an attribute table. Below a few examples of the hierarchy that was used in the inference process to make the knowledge base complete are given.

To make inference with the use of the logic of plausible reasoning possible, it is necessary to introduce to the knowledge base all hierarchical dependencies, resulting directly from the structure of the attribute table.The next step is to define statements. Statement is the term to which a certain value has been assigned. Using statements, we introduce to the knowledge base a value that concepts may have, based on data from the attribute table.

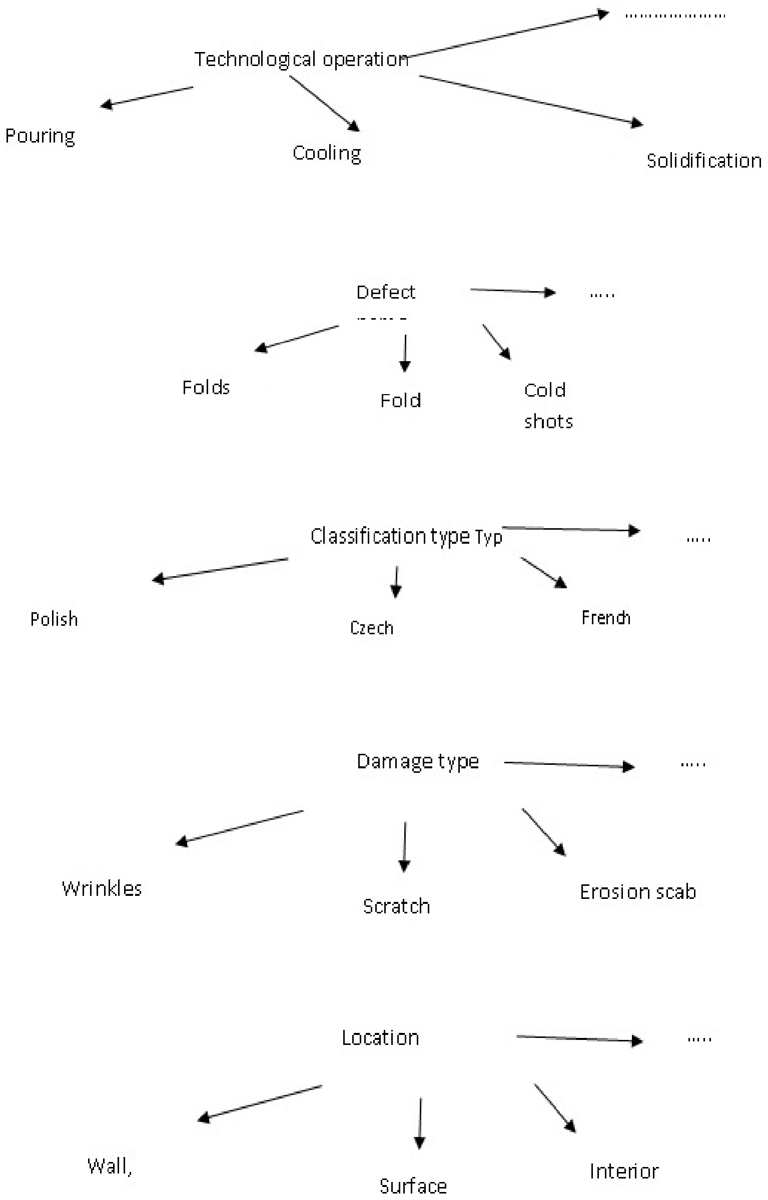


Fig. 3. Hierarchical dependencies

For example, if we have to introduce the information that folds as a type of damage are defined as scratches, wrinkles, and erosion scabs in the description of defects, this statement will be introduced by writing down the following notation:

```
typeDefect (fold)={scratch, wrinkles, erosion scab,..}
```

On the basis of defined concepts, a software was developed in the LPR. To the database, the above Despite the fact that, as regards the damage type, these two defects are determined in the attribute table by the same parameters, it is not possible to state if they cover entirely the same area. One of the defects folds is from the Czech classification, while another fold comes from the Polish classification. Although the definitions of the damage type and shape are the same, the other definitions differ, which can suggest that the case may not entirely refer to the same defect, and this fact should be allowed for in the system of inference.

Yet, to prevent introducing into the system the terms on the damage type separately for the defect fold and for the defect folds, and for other similar cases, a notation of similarity has been added. The similarity may occur as a similarity of arguments or of values. In this case we are dealing with the similarity of arguments.
mentioned notation has been introduced as

```
V (typeDamage, fold, scratch)
V (typeDamage, fold, wrinkles), etc.
```

The next step is to write down similar parameters listed in the attribute table to identify various defects.

For example, in the case of defect fold and defect folds that occur in the Polish and Czech classification, the determined distribution of attributes in the table for the location and shape of damage is the same.

So, it means that the following statements will be introduced:

```
typeDamage(fold)=(scratch, wrinkles, erosion scab)
typeDamage(folds)=(scratch, wrinkles, erosion scab)

typeDamage (fold) = (scratch, wrinkles, erosion scab)
fold SIM folds in CX (defect, defect name (defect))
defect name (defect) damage type (defect)
fold SPEC defect
folds SPEC defects
typeDamage (folds) = (scratch, wrinkles, erosion scab)
```

In the database, such a statement is introduced by means of the term

Similarity

```
S (fold, folds, typeDamage)
```

Relationships

```
E(fold, nameDefect, fold, typeDefect)
E(folds, nameDefect, folds, typeDefect)
```

However, a restriction should be put here. A notation like this can be entered into the system if and only if for at least two different defects the values of one of the conditional attributes are equal.

The second example of the practical use of this type of notation occurs when a system reasoning is based on user responses, or on other external factors and the data are incomplete. The problem is that while preparing a system we never know the type of the data that will be supplied from the outside.

If this is the case, then it is recommended to look for the internal links in the system, to enable deducing some of the parameters based on similarities. Besides, it is also recommended to save all the rules that can somehow affect the creation of other parameters, e.g. having only the location and shape of the defect, its name should be given in approximation.

Here it is also important that the diagnosis or the outcome of reasoning with a small number of the available attributes was defined as a result with the low parameter g, which is the formula confidence index.

It is important to remember in the process of reasoning about the parameters of uncertainty. In the example, the following parameters have been used: y1φ, …….

The parameters are responsible for the formula degree of certainty. The parameter ma is responsible for the multiplicity of argument, s for the degree of similarity between concepts in a given context, while a is the force with which the left side of the implication affects the right side. At present, the introduction of these parameters is done manually for each statement entered into the system, which is a great impediment. Therefore the intention is to develop an automatic input of these parameters, as they affect not only the reasoning process, but also an outcome of this process.

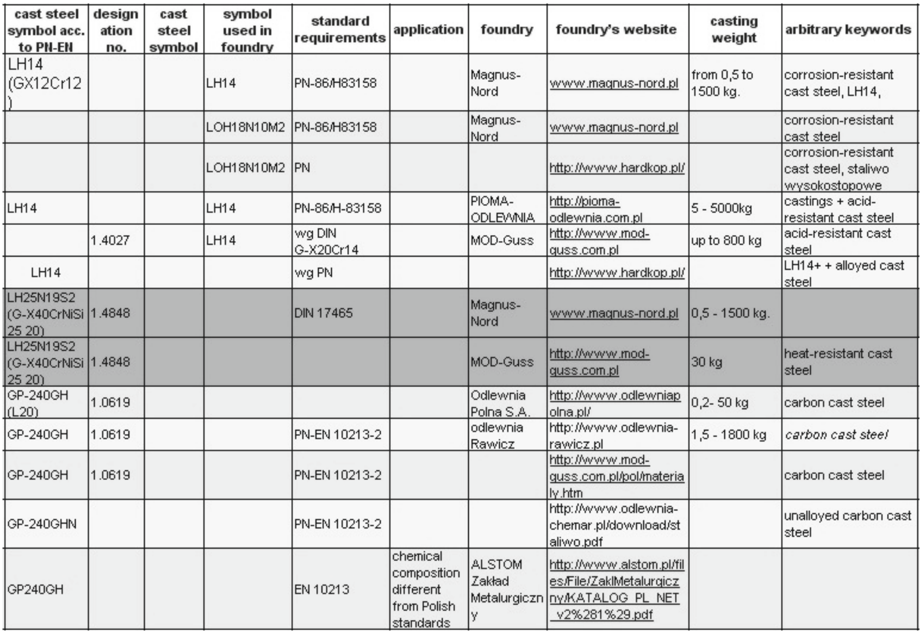
The value of these parameters determines whether the result of inference is true. The mechanism works on the principle that the lower is the value of the parameter g, the lower is the value of the result of inference, meaning that it is less probable. This allows using the LPR in reasoning, e.g. in expert systems creating a ranking of the results, which will enable arranging them in certain order, e.g. by relevance of the diagnoses, which is highly recommended in the case under consideration (e.g. the system in which the reasoning will be conducted will have as a resultant task specifying the names of foundries which have the technical potential sufficient to produce the material searched for). Owing to the applied parameters of uncertainty, the obtained result will have the form of a list of companies where on the top of the list will be placed the foundry plants for which the inference has the highest values of the parameters).

6 Knowledge Integration for Rough Logic-Based Reasoning

The problems of knowledge integration have long been the subject of ongoing work carried out by the Foundry Research Institute, Cracow, jointly with a team from the Faculty of Metals Engineering and Industrial Computer Science, AGH University of Science and Technology, Cracow [9, 10].

Various algorithms of knowledge integration were developed using a variety of knowledge representation formalisms. Today, the most popular technology satisfying the functions of knowledge integration includes various ontologies and the Semantic Web, but it does not change the fact that the relational databases remain the technique most commonly used in industrial practice for the data storage. On the one hand, to thus stored data the users get access most frequently, while - on the other - the databases are the easiest and simplest tool for quick data mining in a given field of knowledge. Therefore, the most effective, in terms of the duration of the process of knowledge acquisition, would be creating the knowledge bases from the ready databases. Studies are continued to create a coherent ontological model for the area which is metals processing, including also the industrial databases.

One of the stages in this iterative process is accurate modelling of the cases of the use of an integrated knowledge management system. A contribution to this model can be the possibility of using attribute tables for reasoning and classification. The process of classification is performed using a RoughCast engine, based on the generated



cast steel symbol acc. to PN-EN	designation no.	cast steel symbol	symbol used in foundry	standard requirements	application	foundry	foundry's website	casting weight	arbitrary keywords
LH14 (GX12Cr12)			LH14	PN-86/H83158		Magnus-Nord	www.magnus-nord.pl	from 0,5 to 1500 kg.	corrosion-resistant cast steel, LH14,
			LOH18N10M2	PN-86/H83158		Magnus-Nord	www.magnus-nord.pl		corrosion-resistant cast steel
			LOH18N10M2	PN			http://www.hardkop.pl/		corrosion-resistant cast steel, staliwo wysokostopowe
LH14			LH14	PN-86/H-83158		PIOMA-ODLEWNIA	http://pioma-odlewnia.com.pl	5 - 5000kg	castings + acid-resistant cast steel
	1.4027		LH14	wg DIN G-X20Cr14		MOD-Guss	http://www.mod-guss.com.pl	up to 800 kg	acid-resistant cast steel
LH14				wg PN			http://www.hardkop.pl/		LH14+ + alloyed cast steel
LH25N19S2 (G-X40CrNiSi 25 20)	1.4848			DIN 17465		Magnus-Nord	www.magnus-nord.pl	0,5 - 1500 kg.	
LH25N19S2 (G-X40CrNiSi 25 20)	1.4848					MOD-Guss	http://www.mod-guss.com.pl	30 kg	heat-resistant cast steel
GP-240GH (L20)	1.0619					Odlewnia Polna S.A.	http://www.odlewniapolna.pl/	0,2- 50 kg	carbon cast steel
GP-240GH	1.0619			PN-EN 10213-2		odlewnia Rawicz	http://www.odlewnia-rawicz.pl	1,5 - 1800 kg	*carbon cast steel*
GP-240GH	1.0619			PN-EN 10213-2			http://www.mod-guss.com.pl/pol/materialy.htm		carbon cast steel
GP-240GHN				PN-EN 10213-2			http://www.odlewnia-chemar.pl/download/staliwo.pdf		unalloyed carbon cast steel
GP240GH				EN 10213	chemical composition different from Polish standards	ALSTOM Zakład Metalurgiczny	http://www.alstom.pl/files/File/ZaklMetalurgiczny/KATALOG_PL_NET_v2%281%29.pdf		

Fig. 4. Fragment of cast steel manufacturers database

attribute table. The database from which the array is generated does not necessarily have to be dedicated to the system. This gives the possibility of using nearly any industrial database. The only requirement is to select from among the attributes present in the base the sets of conditional and decision attributes. If there are such sets, we can generate the attribute table using an appropriate query.

An example might be a database of manufacturers of different cast steel grades (Fig.4).

Using such a database, the user can get answer to the question which foundries produce the cast steel of the required mechanical properties, chemical composition or casting characteristics.

The decision attributes will be here the parameters that describe the manufacturer (the name of foundry) as well as a specific grade of material (the symbol of the alloy), while the conditional attributes will be user requirements concerning the cast steel properties. Using thus prepared attribute table, one can easily perform the reasoning.

7 Summary

The proposed by the authors procedure to create attribute tables and, basing on these tables, conduct the process of reasoning using the logic of plausible reasoning and the rough set theory enables a significant reduction in time necessary to build the models of reasoning. This method of preparation of the decision-making tables gives considerable labour savings for both the knowledge engineer and the expert who is exempt from manual data preparation.

Thus, the expert contribution has been limited to finding out in the database the conditional and decision attributes – other steps of the process can be performed by the system administrator. This solution allows a new use of the existing databases in reasoning about quite different problems, and thus - the knowledge reintegration. Reusing of knowledge is one of the most important demands of the Semantic Web, meeting of which should increase the usefulness of industrial systems.

References

1. Pawlak, Z.: Rough sets. Int. J. of Inf. and Comp. Sci. 11(341) (1982)
2. Kluska-Nawarecka, S., Wilk-Kołodziejczyk, D., Górny, Z.: Attribute-based knowledge representation in the process of defect diagnosis. Archives of Metallurgy and Materials 55(3) (2010)
3. Wilk-Kołodziejczyk, D.: The structure of algorithms and knowledge modules for the diagnosis of defects in metal objects, Doctor's Thesis, AGH, Kraków (2009) (in Polish)
4. Kluska-Nawarecka, S., Wilk-Kołodziejczyk, D., Dobrowolski, G., Nawarecki, E.: Structuralization of knowledge about casting defects diagnosis based on rough set theory. Computer Methods In Materials Science 9(2) (2009)
5. Regulski, K.: Improvement of the production processes of cast-steel castings by organizing the information flow and integration of knowledge, Doctor's Thesis, AGH, Kraków (2011)
6. Szydłowska, E.: Attribute selection algorithms for data mining. In: XIII PLOUG Conference, Kościelisko (2007) (in Polish)

7. Walewska, E.: Application of rough set theory in diagnosis of casting defects, MSc. Thesis, WEAIIE AGH, Kraków (2010) (in Polish)
8. Ligęza, A., Szpyrka, M., Klimek, R., Szmuc, T.: Verification of selected qualitative properties of array systems with the knowledge base. In: Bubnicki, Z., Grzech, A. (eds.) Knowledge Engineering and Expert Systems, pp. s.103–s.110. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław (2000)
9. Kluska-Nawarecka, S., Górny, Z., Pysz, S., Regulski, K.: An accessible through network, adapted to new technologies, expert support system for foundry processes, operating in the field of diagnosis and decision-making. In: Sobczak, J. (ed.) Innovations in Foundry, pt. 3, pp. s.249–s.261. Instytut Odlewnictwa, Kraków (2009) (in Polish)
10. Dobrowolski, G., Marcjan, R., Nawarecki, E., Kluska-Nawarecka, S., Dziadus, J.: Development of INFOCAST: Information system for foundry industry. TASK Quarterly 7(2), 283–289 (2003)
11. Collins, A.: Fragments of a theory of human plausible reasoning. In: Waltz, W.D. (ed.) Theoretical Issues in Natural Language Processing II, pp. ss.194–ss.201. Universyty of Illinoi (1978)
12. Collins, A., Michalski, R.S.: The logik of plausible reasoning: A core theory. Cognitive Science 13, 1–49 (1989)
13. Śnieżyński, B.: Zastosowanie logiki wiarygodnego rozumowania w systemach diagnostycznych diagnostyce, Rozprawa doktorska, WEAIE AGH (2003)
14. Słowiński, R., Greco, S., Matarazzo, B.: Rough set based decision support. In: Burke, E., Kendall, G. (eds.) Introductory Tutorials on Optimization, Search and Decision Support Methodologies, ch. 16. Kluwer Academic Publishers, Boston (2004)
15. Rutkowski, L.: Metody reprezentacji wiedzy z wykorzystaniem zbiorów przybliżonych. Metody i techniki sztuczej inteligencji, Wyd, pp. 20–50. Naukowe PWN, Warszawa (2009)
16. Kuncheva, L.I.: Fuzzy rough sets: Application to feature selection. Fuzzy Sets and Systems 51(2), 147–153 (1992)
17. Radzikowska, A.M., Kerreb, E.E.: A comparative study of fuzzy rough sets. Fuzzy Sets and Systems 126(2), 137–155 (2002)

A Clickstream Based Web Page Importance Metric for Customized Search Engines

Fatemeh Ahmadi-Abkenari and Ali Selamat

Knowledge Economy Research Alliance & Faculty of Computing,
Universiti Teknologi Malaysia,
81310 UTM Skudai, Johor, Malaysia
pkhoshnoud@yahoo.com, aselamat@utm.my

Abstract. The immense growing dimension of the World Wide Web causes considerable obstacles for search engine applications. Since the performance of any Web search engine regarding the degree of result set's accuracy and the portion of the presence of authoritative Web pages in result set is highly dependent to the applied Web page importance metric, therefore any enhancement to the existing metrics or designing novel related algorithms could guarantee better outcomes of search engine applications. Regarding the fact that employing the existing link dependent Web page importance metrics in search engines is not an absolute solution because of their accurateness dependency to the downloaded portion of the Web and their incapability in covering authoritative dark Web pages, therefore proposing and discussing on link independent approaches could be a solution to the mentioned barriers. This paper reviews our clickstream based Web page importance metric of *LogRank* that is independent of the link structure of the Web graph for extracting the best result set from the specific Web domain boundary and for importance estimation of a whole Web domain. Moreover, our Web page classification approach in order to be used in Web site importance calculation will be reviewed.

Keywords: Clickstream analysis, Search engines, Semantic Web, Web data management, Web page Importance metrics.

1 Introduction

The dramatic expanded dimension of the World Wide Web poses considerable challenges for search engines such as not only yielding accurate and up-to-date results for users' information demands, but also precise ranking of result set. A centralized single-process unfocused search engine traverses the Web graph and fetches any URLs from the initial or seed URLs and keeps them in a priority based queue called crawl frontier. Then according to a selection policy under utilizing a version of best-first algorithm selects the first most important URLs in an iterated manner for further processing. A search engine with parallel crawler agents on the other hand is a multi-processes search engine in which upon partitioning the Web into different segments,

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 21–41, 2013.

each parallel agent is responsible for crawling one of the Web fractions [14]. Moreover there is another class of search engines as focused ones that limit their function upon either semantic Web zone or a specific Web domain [13]. In search engines with topical focused notion, the relevant pages to predefined topic taxonomy are selectively sought out the and irrelevant Web regions are avoided as an effort to eliminate the irrelevant items in search results and maintaining a reasonable dimensions of the index while search engines that limit their search boundary to a specific Web domain seek the same notion and acts as a kind of customized search engine.

The bottleneck in the performance of any search engine is applying a robust and effective Web page importance metric. The existing importance metrics relies on the link structure in Web graph or the context similarity between each issued query and the indexed Web pages. These two groups of Web page ranking algorithms deal with some obstacles including their precision drawbacks and a bunch of spam activities. The focus of this paper is reviewing our previously proposed clickstream based Web page importance metric. The literatures on clickstream analysis are mostly targeted toward employing the analysis approaches on the hugely useful clickstream data set for e-commerce objectives. Although we employ the same analysis methods for analyzing the clickstream data set from these literatures, but our target angle of identifying the importance of Web pages needs special care that this paper will present them in detail. Since accessing clickstream data set from thousands of servers worldwide is not a possible task now, so the clickstream-based metric works as a Web page importance metric for a focused search engine with a limited Web boundary to a specific Web zone.

In this paper, we review our previously proposed clickstream based metric of *LogRank* for Web page importance calculation and for Web site importance estimation. Also one of the advantages of this metric regarding coverage of dark Web page will be discussed. According to this metric, the link structure of the Web graph is no longer used as a source to compute the rank of each Web page. As a result the problematic issues like the lack of crawler knowledge of the precise number of casted vote for each page, dependency to algorithms to cope with very large sparse matrixes (link matrix) and convergence criteria for an iterative-based formula will be removed.

In continue we first review the related works on Web page importance metrics and Web page classification approaches and then the *LogRank* metric will be reviewed.

2 Related Works

In this section, first, the existing Web page importance metrics, their notion of Web page importance, their drawbacks and modified versions of each of them will be reviewed. Then coverage of dark Web pages by different metrics will be discussed through defining the dark Web pages first. At the next step, Web page classification approaches will be reviewed since the *LogRank* approach for estimating the importance of a Web site in our approach needs classification of Web pages in first place.

2.1 Web Page Importance Metrics

In order to rank Web pages, the ranking module in search engines employs (a) Web page importance metric(s) in a single or combination modes. The set of these metrics could be organized in two main categories of text-based and link-based metrics that the former group includes Boolean or vector space based measures while the latter group includes the dominant metrics such as *HITS* and *PageRank*. In continue we shortly review the main characteristics, obstacles and also variations to each of the mentioned metrics.

2.1.1 Text Based Web Page Importance Metrics

Text-based metrics measures the textual similarity between the user issued query and each indexed Web page through finding out the degree of relevancy according to the Boolean or vector space models through different similarity measures like *Cosine*, *Best Match* (*BM*), *Pivoted Normalization Weighting* (*PNW*), … [24], [32]. The mere text-based approaches can be easily influenced by spam activities in a way that the stakeholder could put many popular but irrelevant terms in important code sensitive places like title, *Meta* and <*h*> *HTML* tags and make them invisible via coloring schemes to mislead the text-based engines. Besides, the computation of relevancy in some cases relies on the factor of *Inverse Document Frequency* (*IDF*) which describes the number of times the word appears in the entire collection. Therefore, since the terms need to carry an absolute measure across the whole collection and indexing the whole Web is still an open issue, precise calculation of *IDF* is impossible. Moreover, due to the noisy environment of the Web and the hurdles to realize semantically related terms, utilizing the mere text dependent approaches is definitely challenging. Although *Latent Semantic Indexing* (*LSI*) and the usage of *Singular Value Decomposition* (*SVD*) was proposed in the literature to discover the documents containing the semantically related terms to the query [17], but employing *SVD* within the Web environment is not a promising solution regarding its huge time complexity. As a result, text-based approaches usually applied in ranking modules in combination with other importance metrics.

2.1.2 *Hyper Text Induced Topic Search* Metric (*HITS*)

Hypertext Induced Topic Search (*HITS*) metric views the notion of Web page importance in its hub and authority scores in which the authority score of a node in Web graph is computed based on the hub scores of its parent nodes and the hub score of a node is calculated by the authority scores of its children nodes as depicted in Equation (1) and (2) in which *a(i)* is the authority score of page *i* and *h(i)* is the hub score of page *i* while *E* is the set of edges in the Web graph [23];

$$a(i) = \sum_{(j,i)\in E} h(j) \tag{1}$$

$$h(i) = \sum_{(i,j)\in E} a(j) \tag{2}$$

HITS metric has some drawbacks including the issue of topic drift, its failure in detecting mutually reinforcing relationship between hosts, its shortcoming to differentiate between the automatically generated links from the citation-based links within the Web environment, no anti-spamming feature and also its time complexity [6]. Due to the fact that all Web pages, to which a hub page points to, are not around the same topic, the problem of topic drift is formed. The second problem occurs when a set of documents in one host points to one document on another host. As a result, the hub score of pages on the first host and the authority score of the page on second host will be increased. But this kind of citation cannot be regarded as coming from different sources. Besides, Web authoring tools generate some links automatically that these links cannot be regarded as citation based links. Furthermore, this metric has no anti-spamming feature since it is easy to put out-links to authoritative pages to affect the hub score of the source page. Moreover, since the hub and authority measures is computed at query time, application of this metric in today's search engine architecture in not feasible. Although the literature includes some modifications to *HITS* algorithm such as the research on detecting micro hubs [10], neglecting links with the same root, putting weights to links based on some text analysis approach or using a combination of anchor text with this metric [6], [11], [12], there is no evidence of a satisfactory success of these attempts.

2.1.3 PageRank and Variations

Google search engine starts its life via employing *PageRank* in combination with context considering schemes like the usage of anchor text and font information, word proximity and other standard information retrieval measures. Although there are some aspects of *PageRank* definition either through using eigenvector concept in linear algebra or a definition based on Morkov chain, but simply *PageRank* metric is a modification to *Backlink* count metric. *Backlink* count metric simply counted the number of in-links to a page while *PageRank* calculates the weighted incoming links according to Equation (3) in which B_v is the set of all pages that have a link to page v, N_u is the number of out-links from page u and d is a damping factor which represents the probability of visiting the next page randomly. So, the *PageRank* of each page v for the $(i+1)^{th}$ iteration is computed offline according to the *PageRank* of its incoming links for the i^{th} iteration. The iteration continues until the *PageRank* stabilized to within some threshold.

$$\forall v PageRank_{i+1}(v) = (1-d) + d\sum_{u \in B_v} PageRank_i(u) / N_u \tag{3}$$

One of the problems of classic *PageRank* is its iterative nature. That is after how many number of iterations, the result could be more trusted. The iteration for *PageRank* computation is necessary as a solution to catch 22 situation. Under this situation, one iteration to calculate the rank is not trustworthy for two nodes of a and b that both link to each other and consequently affects each other's *PageRank* [7], [14]. There are some modifications to classic *PageRank* algorithm in order to minimize the number of iterations through proposing several ways to analyze the convergence criteria of the algorithm that is beyond the scope of this paper.

The query independence nature of *PageRank* is another criticism on this metric. Although, at query time a context comparison is performed between the query and the offline ranked pages, some modification tries to empower the metric by generating a query-specific importance score or designing a more intelligent surfer. The former focuses on designing a set of *PageRank* vectors based on a set of representative topics offline in order to estimate the page importance with respect to a specific topic instead of having a mere link dependent single vector in classic *PageRank*. Under the latter model, the surfer jumps from page to page not completely blinded but depending on the context of the destination page through noticing the anchor text which both models make the *PageRank* context sensitive. Another enhancement to *PageRank* in this area categorizes Web pages according to their context and for avoiding overlap of pages among different categories, employs surfer history analysis by considering the content category of back-links for each page [21], [22], [27], [30].

Online Page Importance Calculation (*OPIC*) is another variation to *PageRank* that works online and dynamically adapts its importance estimation upon visiting more portion of Web graph. For the start point, Web pages are initialized by some *cash* distribution. During the crawling process, the page distributes its cash equally among out-links and this distribution will be recorded in the page *history*. The importance of the page is then calculated according to its *credit history*. This algorithm uses less CPU, memory and disk access and also less storage resources by eliminating the need to store the huge sparse link matrix [1]. *Online Topical Importance Estimation* (*OTIE*) is the advancement to *OPIC* in a way that does not circulate the page cash equally among out-links but disperses the cash in a way to support on-topic and to suppress off-topic pages [19].

Another criticism behind classic *PageRank* is that it is not an absolute solution to spam activities like those bogus Web pages that all points to a single target page or a cluster of Web pages that repeatedly link to each other to obtain more than deserve rank position. Also, densely linked pages all located in a single server are nepotistic links because they increase the *PageRank* but hardly indicate authority. Two-party and multi-party nepotism are possible and this nepotism problem mostly occurs under commercial manipulation. *TrustRank* is a semi-automated *PageRank*-based metric proposed as a spam combating technique. Under this metric, a small set of reputable Web pages is selected as seed URLs based on human judgment. Then in a *PageRank* trend, trust measure is propagated onward by linking to other pages by considering incoming links as a vote casting from the source node to the destination one. Also the negative trust equal to inverse *PageRank* could be propagated backwards for spam pages by considering a function of outgoing links. Then both measures will be considered to rank Web pages. Trust score is attenuated as the document distance to seed set increases. The main two idea behind *TrustRank* are: First, reputable pages rarely link to spam pages, but spam pages often link to reputable pages in order to increase hub score. Secondly, the care with which people add links to a Web page is often inversely proportional to the number of links on the page [20].

Another barrier of *PageRank* is that it favors older pages that over the time of existence on the Web they accumulate in-links. Hence the authoritative fresh Web content is disregarded under the *PageRank* perspective. The *TimedPageRank* algorithm adds the temporal dimension to the *PageRank* as an attempt to pay a heed to the newly uploaded high quality pages into the search result by considering a function of time *f(t)* ($0 \leq f(t) \leq 1$) in lieu of the damping factor *d*. The notion of *TimedPageRank* is that a Web surfer at a page *i* has two options: First randomly choosing an outgoing link with the probability of $f(t_i)$ and second jumping to a random page without following a link with the probability of $1\text{-}f(t_i)$. For a completely new page within a Web site, an average of the *TimedPageRank* of other pages in the Web site is used [35].

All things considered, link dependent metrics suffer from a precision-related phenomenon. That is, the rank computation for each page relies on the subset of Web graph that has already been crawled. As a matter of fact, no crawler could claim to make an index near half of the whole Web. As a result, the calculated rank of each page is less than the real value resulted from the lack of crawler's knowledge to all incoming links to a page. In other words, upon downloading more portion of Web, the computed importance of a page will be affected and so the order of pages according to their importance. Moreover, the noisy links such as advertisement related links are not among the links with citation objectives. So these types of links could mislead the link dependent search engines.

There are two other old metrics as *Location* metric and *Forward- link* count. Under the former, the home pages, the URLs with less slashes or URLs with *.com* extension will be ranked higher. While under the latter, the emanated links from a page is checked with the notion that a page with a high forward link score is a hub page. This metric suffers from the spam defect in a way that a Web master could simply put forward links to many destinations to affect its own rank [15].

2.2 Dark Web Page Coverage

Search engines face the barrier of covering a group of problematic Web pages as dark pages in result set. These pages also called Web pages under the surface of the Web. There are different definitions for dark pages. One definition refers to dark side of the Web as a part of the Web that search engines cannot traverse and index. Form type pages, scripted contents, Web sites that require registration, textual content embedded in multimedia and unlinked or weakly linked content are categorized as dark pages in the first definition [8], [28], [29]; There are modifications to *PageRank* in order to enhance this metric to include the pages in dark net by starting from the form pages as an entry point to one region of dark net pages [25].

Another definition of dark pages refers to those parts of the Web that users never or rarely visit in a specific duration of observation. The latter definition emphasizes on the non-visit status of Web pages regardless of the content authority, link structure or the type of the page with above mentioned categorization. The pages with non visit status could be investigated from analyzing the log data of each Web domain through having knowledge of the Web site topology. Although dark pages under the second definition could be identified through server log data but covering the authoritative dark pages in search result set is another issue that the clickstream-based metric proposes a solution for that in order to cover those dark pages that have not appear in log data in a specific observation period regardless of why they are dark or what type of dark Web pages they are. The only important matter for clickstream-based metric to cover dark Web pages is their content authority to the user issued query.

Link-based metrics of *PageRank*, *TimedPageRank*, *OPIC*, *OTIE* and *TrustRank* do not cover the pages in dark net especially those dark Web pages with weak link structure. *HITS* metric because of its dependency to the link graph of the Web cannot cover the mentioned group of dark pages as well. *Forward link* count as a simpler version of hub-based *HITS*, behaves the same as *HITS*. All context similarity measures are able to cover the authoritative pages in dark net because of their mere dependency to the context. However, they suffer from great number of spamming activities against the context based searching systems. So these metrics are employed in Web searching system in combination to other metrics. While *Location* metric favors the pages of the first level in Web site graph, the coverage capability of it towards dark group of pages is negative since mostly this group of pages belongs to lower level in Web site graph.

2.3 Web Page Classification

In order to calculate the importance of the whole Web site based on the importance of its pages, the first step is to consider an approach to categorize Web pages inside a Web domain. To this end, in this section different Web page classification approaches are reviewed shortly. The research on Web page classification mostly focuses on grouping Web pages into semantically different classes in order to facilitating the result yielding for search engines. These categorizations perform manual selections, automatically employing clustering approaches or usage of *Meta* data [5], [26], [34]. Accordingly, Web pages are categorized into large number of classes and the result mostly applied to feed Web directories in order to list these semantic zones from general topics to the detailed titles.

Another approach classifies Web pages based on their hyperlink information through extracting *Meta* data [12]. There is also different perspective for Web page categorization in which all Web pages classified into a few distinct numbers of classes according to the types of information they include. One approach in this area, classifies Web pages into five classes of head pages, content pages, navigation pages, look up pages and personal pages [16]. This categorization has conceptual overlap among classes of pages. For example, a head page in many cases functions as a navigation page and behaves in a directory trend. Moreover, this categorization is old since the existence of look up pages and navigation pages alone are not very widespread nowadays according to the dynamic nature of the Web. In this paper, this classification approach is modified by the intention of making a more applicable categorization of Web pages for Web site importance estimation.

3 *LogRank* Computation Synopsis

In our companion papers we proposed the architectures for a clickstream-based focused trend parallel search engine [2]. Since the architecture of the clickstream based search engine has been discussed thoroughly in that paper, here we focus on the Web page importance calculation by the correspondent section in the architecture. Within the search engine, the distiller section is responsible to compute the importance of each URL in crawl frontier based on clickstream data in offline mode. Then the classifier section computes the text relevancy of each URL to the issued query based on the *Best Match* text similarity measure.

Clickstream dataset or server log file is a valuable source of information in which every single recorded entry corresponds to one *HTTP* request of Web resources from Web site visitors [18]. While the application of clickstream data analysis is roughly ignored for discovering the importance of Web resources, analysis on this hugely useful dataset were targeted toward fetching practical rules for e-commerce objectives like designing cross marketing strategies, evaluating the effectiveness of promotional activities and the similar intensions in the literature so far.

The clickstream data format according to *W3C* [33], includes the fields of IP address of the clients, date and time of each request, the requested resource, status of each request, *HTTP* mode used for the request, size of the requested Web object, the user agent (the name and version of client browser and operating system), client-side cookies, referring Web resource, and needed parameters to invoke a Web application in each entry. From this range of fields, a clickstream-based search engine in our approach needs the five fields of client IP address, date and time of each request, requested Web resource, referrer and user agent fields.

In order to use clickstream data for Web page importance calculation, the backbone of clickstream data preprocessing, user identification approaches, sessionization and path completion methods will be used from the research on clickstream analysis in the literature [18] but we alter some of them according to the objective of employing clickstream data in search engine framework. In our approach, Web page importance computation based on clickstream data is done in the following 5 phases of data cleaning, user identification, sessionization, path completion and Web page log ranking calculation as follows [3].

The first phase of data cleaning includes removing unnecessary fields from clickstream data set and so it hugely depends on the nature of the target analysis. Each entry corresponds to one *HTTP* request but does not correspond to one Web page request. In other words, a request for a Web page that consist images, scripts, cascading style sheets, sound files, etc produces multiple entries in log file which are not needed in our approach. Since the specific domain of the experiment is the Web site of *UTM University* (*Universiti Teknologi Malaysia*) so figure 1 illustrates a portion of the Web site topology which is used as an example throughout the rest of this paper. Table 1 describes each node and provides their uniform resource location. Table 2 depicts the log file entries corresponds to this portion in which we changed the IP address to an imaginary one. After the completion of phase 1 (data preprocessing), the entries of 3, 4, 11 and 12 will be removed.

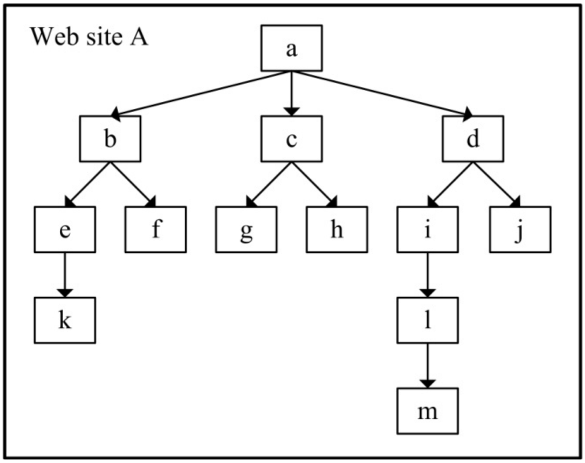


Fig. 1. A portion of *UTM* Web site topology

The analysis on clickstream data does not require knowing every user accessed the Web site. The aim of user identification approaches is to distinguish among different user sessions that include visiting the pages of the Web site. In order to differentiate among different user sessions, considering the IP address field alone is not sufficient since multiple users may use the same machine or one user may surf the Web from different terminals. So the user identification approaches try to find the different users as close as the real number of users.

For computing the Web page importance, in our approach it is important to identify the number of time a particular Web page is accessed even by one user session or by different user sessions. To this end, the output of user identification phase does not satisfy the mentioned state. Therefore, we need to break a visit performed by one user to multiple visits if he comes back to the Web site by a considerable interval. The sessionization approach is a solution to this need since the output of this phase is a divided navigation path by one particular user into multiple sessions if the time between requests exceeds a certain defined threshold of Δ [9, 16]. So after the completion of sessionization phase, the visit of first user is divided to two user sessions of u_1 and u_2 as shown in table 3.

Since the second access to a Web page is not recorded in log file according to some client or proxy-side caching, the log file should be completed before making any decision on the importance of Web page. The process is called path completion in which the missing references are added to each user session via having the knowledge of the Web site topology. Therefore if a new page could not be accessed through the visited last page, the user may move backward by hitting the browser back button upon which accessed the cached version of a page. Since this second visit to a Web page is a visit itself, so it should be taken into account as an access to that Web page. Table 3 shows the user sessions after the path completion process in which the second access to pages *e* and *b* are added to the first user session and a second access to page *a* is added to the last user session.

Table 1. Portion of UTM Web pages, their Descriptions and Uniform Resource Locations

Page	**Description**	**URL**
a	UTM home page	http://www.utm.my/
b	Faculty of Computer Science and Information Systems (FSKSM)	http://webs.cs.utm.my/
c	School of Graduate studies (SPS)	http://www.sps.utm.my/spshome/
d	Faculty of Civil Engineering (FKA)	http://www.civil.utm.my/
e	Department of Computer Systems and Communications	http://csc.fsksm.utm.my/v1/
f	Research Groups (Under FSKSM)	http://webs.cs.utm.my/resgroup.html
g	Current Students (Child node of SPS)	http://sps.utm.my/sps/index.php?option=com_content&view=article&id=48&Itemid=58
h	Prospective Students (Child node of SPS)	http://sps.utm.my/sps/index.php?option=com_content&view=article&id=49&Itemid=66
i	Departments and Units	http://www.civil.utm.my/content.php?id=6&cid=6&lang=1
j	Current Students (Child node of *d*)	http://www.civil.utm.my/content.php?id=64&cid=5&lang=1
k	Contact Us (Child node of *e*)	http://csc.fsksm.utm.my/v1/contact-us.html
l	Department of Environmental Engineering	http://www.civil.utm.my/content.php?id=80&cid=80&lang=1
m	One of professors' personal page	http://www.civil.utm.my/staff.php?staff=97

Since in our approach we are working on the duration of visits per each page, there is no access to the time spent in missing references. So assigning a default value of η=25 seconds to each missing reference will be used throughout this paper. Since the added missed reference(s) affect(s) the page-stay of the previous page in each path within a user session, we describe the way to calculate the page-stay of a page after which the missed reference(s) added to the navigation path through Equation (4) as follows [3];

$$D_{w_i}^{u_j} = TF_{w_l}^{u_j} - TF_{w_i}^{u_j} - n \times \eta \tag{4}$$

In Equation (4), w_i and w_l are two Web pages in the navigation path inside the same user session of u_j between them the missed reference(s) have been added after path completion. *TF* refers to the value of date/time field in the log file. So to calculate the page-stay of a Web page after which the missed reference(s) have been added as shown in table 3 (for example page *b* in u_4), we should go to the date/time filed of the Web page after the missed reference(s) in navigation path in the same user session (page *c* in u_4) and then subtract the entrance time to w_i and the number of missed reference(s) (n) multiplied by η seconds from the entrance time to w_l.

Table 4 describes the applied details of each of four phases discussed above [3].

Table 2. A portion of *UTM* server log corresponds to the Web site topology in figure 1

ID	**IP Address**	**Date/Time**	**Resource**	**Referrer**	**Agent**
1	1.2.3.4	[15/Jul/2010: 10:08:05]	a.html	-	Mosilla3.06(WinXP;2002;SV1)
2	1.2.3.4	[15/Jul/2010: 10:08:35]	b.html	a.html	Mosilla3.06(WinXP;2002;SV1)
3	1.2.3.4	[15/Jul/2010: 10:08:38]	header.gif	b.html	Mosilla3.06(WinXP;2002;SV1)
4	1.2.3.4	[15/Jul/2010: 10:08:45]	Staff1.jpeg	b.html	Mosilla3.06(WinXP;2002;SV1)
5	1.2.3.4	[15/Jul/2010: 10:08:55]	e.html	b.html	Mosilla3.06(WinXP;2002;SV1)
6	1.2.3.4	[15/Jul/2010: 10:09:07]	i.html	-	Mosilla3.06(WinXP;2002;SV1)
7	1.2.3.4	[15/Jul/2010: 10:09:24]	k.html	e.html	Mosilla3.06(WinXP;2002;SV1)
8	1.2.3.4	[15/Jul/2010: 10:09:30]	a.html	-	Mosilla3.04(WinNT;5.1;SV1)
9	1.2.3.4	[15/Jul/2010: 10:09:54]	b.html	a.html	Mosilla3.04(WinNT;5.1;SV1)
10	1.2.3.4	[15/Jul/2010: 10:09:55]	l.html	i.html	Mosilla3.06(WinXP;2002;SV1)
11	1.2.3.4	[15/Jul/2010: 10:09:59]	Staff2.gif	l.html	Mosilla3.06(WinXP;2002;SV1)
12	1.2.3.4	[15/Jul/2010: 10:10:22]	Style1.css	l.html	Mosilla3.06(WinXP;2002;SV1)
13	1.2.3.4	[15/Jul/2010: 10:10:32]	c.html	a.html	Mosilla3.04(WinNT;5.1;SV1)
14	1.2.3.4	[15/Jul/2010: 10:10:37]	f.html	b.html	Mosilla3.06(WinXP;2002;SV1)
15	1.2.3.4	[15/Jul/2010: 10:12:15]	g.html	c.html	Mosilla3.04(WinNT;5.1;SV1)
16	1.2.3.4	[15/Jul/2010: 12:32:20]	a.html	-	Mosilla3.06(WinXP;2002;SV1)
17	1.2.3.4	[15/Jul/2010: 12:33:42]	d.html	a.html	Mosilla3.06(WinXP;2002;SV1)
18	1.2.3.4	[15/Jul/2010: 12:34:43]	j.html	d.html	Mosilla3.06(WinXP;2002;SV1)

Actually, the span of time the visitor spent on each Web page equals to the difference between the date/time fields of an entry in each user session to the next entry in the same user session. In other words, the time field of the first entry is the entrance time to the page while the value of time field in second entry in the same user session is the exit time of the page. This duration is referred as a page-stay in this work. The problematic issue with page-stays is with the last page on each user session or the exit page for which there is no recorded value to the exit time. The literature includes either using a mean value of all page-stays in that user session or the mean of page-stays along the whole log file or using a default value for this kind of page-stays as 25 seconds. In this work, the last method with the mentioned value of μ=25 seconds is employed.

Table 3. The identified users sessions and the navigation path after completion of each phase

User Sessions	**Navigation Path**
Number of user sessions after *User Identification* phase	
User Session 1 (u_1)	$a \rightarrow b \rightarrow e \rightarrow k \rightarrow f \rightarrow a \rightarrow d \rightarrow j$
User Session 2 (u_2)	$i \rightarrow l$
User Session 3 (u_3)	$a \rightarrow b \rightarrow c \rightarrow g$
Number of user sessions after *Sessionization* phase	
User Session 1 ($u_{1\text{-}1}$)	$a \rightarrow b \rightarrow e \rightarrow k \rightarrow f$
User Session 2 ($u_{1\text{-}2}$)	$a \rightarrow d \rightarrow j$
User Session 3 (u_2)	$i \rightarrow l$
User Session 4 (u_3)	$a \rightarrow b \rightarrow c \rightarrow g$
User sessions 's content after *Path Completion* phase	
User Session 1 ($u_{1\text{-}1}$)	$a \rightarrow b \rightarrow e \rightarrow k \rightarrow e \rightarrow b \rightarrow f$
User Session 2 ($u_{1\text{-}2}$)	$a \rightarrow d \rightarrow j$
User Session 3 (u_2)	$i \rightarrow l$
User Session 4 (u_3)	$a \rightarrow b \rightarrow a \rightarrow c \rightarrow g$

Table 4. Summary of phases in *LogRank* computation

Phase No.	***Phase***	***Description***
1	Data cleaning	entries with the requested resource field equal to any embedded object rather that Web pages like resources with *.gif*, *.jpeg*, *.png*, *.ico*, *.css*, *.cgi*, *.js*, and other related extensions are removed. Also, entries that include an access to *PDF* files, any administrator related activity, *mailto* and access to Web pages from the domain outer space are removed.
2	User identification	A combination of IP address and user agent field is used to distinguish among different user sessions. At the next step, the number of distinct users may raise if each issued request could not be reached from already visited pages.
3	Sessionization	25.5 minutes is the timeout for Δ (according to empirical data in the literature) but in many commercial products, a threshold of 30 minutes is used. In this research, we also consider Δ=30 minutes.
4	Path completion	Assigning a default value of η=25 seconds to each missing reference in Equation (4).

The proposed notion of Web page importance of this research is that a page is more important if more time has been spend on it within an observation period of *T* by different distinct visitors. So according to this notion, the rank of each Web page is defined as the total page-stay durations from different user sessions per each single page multiplied by the number of distinct user sessions that contains (a) visit(s) to that page. The number of user sessions is taken into account for closing a spamming window upon which the page is visited by few numbers of visitors (mostly stakeholders). By considering the number of distinct user sessions, the more different user sessions visit the page at different time, the more important the pages are. Therefore, if $|d|$ is the number of days, k a counter for days in T, $W= \{w_{1,} w_{2,} \ldots, w_i, \ldots, w_n\}$ is the set of all pages in a Web site, $U = \{u_1, u_2, \ldots, u_j, \ldots, u_n\}$ is the set of identified user sessions, $|U|$ is the total number of distinct user sessions, u_{wi} is the user session that contain the page w_i in its navigation path, L is the length of each user session (the number of pages in the navigation path of the session) and $D_{w_i u_j}$ is the duration of page-stay of page w_i in user session of u_j, the *LogRank* (*LR*) of each Web page is as shown in Equation (5) [2], [3];

$$LR_{w_i} = \sum_{k=1}^{|d|} \left[|u_{w_i}| \times \sum_{j=1}^{|U|} \sum_{l=1}^{L} D_{w_i u_j} \right]_k \tag{5}$$

Since *LogRank* is a topic free metric, the result of this metric is combined with the context similarity measure of *BM25* (mostly known as *Okapi*) in order to make the search result set more enhanced. *BM25* approach measures the context relevancy of issued queries to each Web page according to Equation (6). In Equation (6), t_i is a term and f_{ij} is the number of occurrences of the term t_i in Web page w_j. f_{iq} is the number of occurrences of the term t_i in query q, N is the total number of documents in the collection, df_i is the number of documents that contain term $t_{i,}$, dl_j is the length of the document w_j and *avdl* is the average document length of the collection. k_1 as a parameter is between 1.0 and 2.0, b is a parameter set to 0.75 in *BM25* and k_2 is a parameter between 1 and 1000 [24], [31], [32].

$$BM_{25}(w_i, q) = \sum_{t_i \in q, w_i} \ln \frac{N - df_i + 0.5}{df_i + 0.5} \times \frac{(k_1 + 1) f_{ij}}{k_1 (1 - b + b \frac{dl_j}{avdl}) + f_{ij}} \times \frac{(k_2 + 1) f_{iq}}{k_2 + f_{iq}} \tag{6}$$

First the context similarity engine starts to find the similarity of each term to the Web pages in the index. Most part of this checking is designed offline in order to make the whole searching process faster. Then upon receiving the issued queries, the similar items (Web pages) in context similarity report are ranked based on the *LogRank* approach. The final result is ranked based on five *Ordering Rules* as described below in table 5 [2];

Table 5. Five *Ordering Rules* for ranking Web pages

Rule No.	Rule Description
R1	Existence of the similar Web pages to the issued query (according to *BM25* report) is searched in the most recent observed period of server log file. Upon the existence of (a) records per these pages, they are added to the answer set.
R2	Existence of the similar Web pages to the issued query is searched in the previous version of observed period of server log file. Upon the existence of (a) records per these pages, they are added to the answer set and their importance value will be damped.
R3	If similar Web pages to the issued query have not been happened in two sequential versions of processed log files, so the *LogRank* of these pages is zero. Regarding the fact that the parent node of a Web page, mostly includes the anchor text and clues to the desired child node, the parent node is searched in processed log file.
R4	If similar Web pages to the issued query have not been happened in two sequential versions of processed log files, so the *LogRank* of these pages is zero. Under this rule, the sibling nodes are searched in processed log file.
R5	Under this rule, some relevant pages to the issued query (according to *BM* similarity measure report) placed at the end of the answer set despite their zero *LogRank.* As a result considering a probability factor of $0 < P < 1$ which represents the user's click on this group of pages, they are given a chance to be visited and make a record in future logs and comes to the surface.

4 *LogRank* of Web Sites

This section discusses the employment of *LogRank* Web page importance metric in order to compute the importance of a whole Web site. Web surfers' notion toward finding the needed information on Web is spending the least time and finding the best information. Therefore, if the calculated importance value of each Web site is announced to the surfers via an item in add-on or auxiliary toolbars, users will be able to decide on staying on the domain or leaving it soon [4]. To this end, the clickstream approach considers the statistics of the total time that have been spent on the Web domain and regarding different weights of different pages in the Web site. Since not all Web pages in a Web site are from the same importance degree, the Web site should be ranked based on considering the different weights that its Web pages give to the site. As illustrated in table 6, under the proposed approach for Web page categorization, Web pages is classified into four different categories of H_1, H_2, B and F type pages. Due to the high importance degree of the Web site's home page and the considerable difference between its importance value and other Web pages, a different class for this type of pages is considered. The rest of Web pages categorized in three classes of *Head*, *Body* and *Foot* type pages. *Head* pages are important head oriented pages rather than the home page itself. *Body* pages are content-based pages while *Foot* class consists of lower importance pages like login, personal, form and contact us pages [4].

Table 6. Web page categorization approach [4]

Class of Web pages	**Description**
Head page (H_1 type)	Home page of the Web site
Head oriented pages (H_2 type)	The first pages in the main categories of a multiple department site
Body pages (*B* type)	Content pages rather than head and foot pages
Foot pages (*F* type)	Login, Contact us, Personal and Form pages

The *LogRank* of a Web site according to the inclusion of described classes of Web pages is as shown in Equation (7). In Equation (7), *IV* is the *LogRank*-based *Importance Value* calculated based on sessions $*$ seconds, $|I|$ is the number of H_2 type pages in the Web site of *W*, $|J|$ is the number of *B* type pages and $|K|$ is the number of *F* type pages. The importance calculation of a Web site as illustrated in Equation (7) empowers from the parameters of h_1 and h_2 for H_1 and H_2 types respectively while damps by the *b* and *f* parameters for the types of *B* and *F*. The parameters of h_1, h_2, *b* and *f* are called as *Type Factor* variable from now on [4]

$$IV_W = h_1 \times IV_{w_{h_1}} + h_2 \times \sum_{i=1}^{|I|} IV_{w_i} + b \times \sum_{j=1}^{|J|} IV_{w_j} + f \times \sum_{k=1}^{|K|} IV_{w_k} \qquad (7)$$

One of the reasons to propose a clickstream-based approach for Web site ranking is applying the Web site importance value in calculating the initial importance value for newly uploaded Web pages or infant Web pages [4]. The notion behind this approach is that a reputable Web site could be able to influence the rank of its Web pages as early as their upload time in order to produce an empowered search answer set that includes the authoritative newly uploaded Web pages. To this end, the importance value of an infant page is computed through considering the Web site importance value and the category the infant page belongs to. The detail of this process discussed in our companion paper [4].

5 Experimental Results

The accessed server log file of *Universiti Teknologi Malaysia* is from 15th of July 2010 to 24th of September 2010. The second dataset used in this research is crawled from *Universiti Teknologi Malaysia‘s* Web site in order to make a repository of Web pages. Each Web page is saved as a text file in the central repository of the architecture. The number of pages in the constructed repository of the experiment is 1721 pages out of *Universiti Teknologi Malaysia*‘s Web pages.

The searching system makes the answer set through the function of *Result Coordinator* (*RC*) section inside the architecture that has been discussed thoroughly in our companion paper [2]. The answer set is made through having the report based on context similarity of the issued query and the indexed pages and the two sequential processed log file and upon considering the five discussed *Ordering Rules*.

Table 7 illustrates the result of the search for the sample phrase of "*Software Engineering*". This result is based on the *Universiti Teknologi Malaysia*'s Web site content dated from 1st of March 2011 to end of the March 2011 as the indexing time. According to redesign of some part of the site after the index time, the search results may vary for the search performed after that time in some cases. Table 7 includes the answer set based on the Web pages' file name in the *Index*, URLs of the answers, the *Ordering Rules* upon which each the items placed in answer set, *LogRank* of each page, and the calculated *Importance Value* (*IV*). In response to the query of "*Software Engineering*", there are 12 responses by the clickstream-based searching system. The first item follows the first *Ordering Rule* so the documents appear in *BM* similarity measure and in most recent processed log. The second, third and forth items are the parent nodes of the nodes with similarity measure to the query while the fifth are the sibling node of such a Web page. The rest of the items in answer set are those authoritative similar pages to user query that have been labeled dark in the observation period and have a place in answer set. These items have zero *LogRank*, so following the fifth *Ordering Rule*, their *Importance Value* is only based on the similarity measure to the query calculated by *BM25* algorithm.

Figure 2 illustrates a comparison among three classes of metrics as mere context-based (red bars), weighted in-links (yellow bars, all mentioned link-based metrics except *HITS*), and *LogRank* as the metric of this work in blue bars. In figure 2, *X*-Axis shows different subjects of the experiment and *Y*-axis shows the coverage of authoritative dark pages (pages with no visit status or zero *LogRank*) in answer set around the queries in the represented topics. Although mere context-based approach cover all authoritative dark pages regardless of any link structure or any other factors rather than the context itself but Web searching system don't employ these metrics alone due to the great number of spam activity associated with these metrics. The coverage of dark pages of the weighted in-links group of metric is low in different subjects and never goes higher than 38 percent. While the *LogRank* metric performs well in covering dark pages and reaches 91.5 percent in the subject of "*Institutes*", 90.5 percent for the subject of "*Departments*", 80.5 percent for the subject of "*Research Alliances*", 75.5 percent for the subject of "*Services*" and 71 percent for other subjects.

Table 8 summarizes the importance value for the page of H_1 type, the sum of importance values for the pages of H_2, B and F types respectively and the calculated importance value for the Web site based on Equation (7).

Table 7. The search result for the sample query of "*Software Engineering*" by the clickstream-based searching system

Search Answer Set (SAS)-File Name	***Search Answer Set (SAS)-URL***	***Rule No.***	***Log Rank (LR)***	***Importance Value (IV)***
Research Management Centre (RMC)-Centre of Excellence (CENTERPIS)	*http://www.utm.my/faculties/index.php?option=com_content&task=view&id=48&Itemid=87*	*R1*	364	484
Faculty of Computer Science and Information Systems (FSKSM)-Home	*http://webs.cs.utm.my/*	*R3*	34	3054122
International Business School (IBS)-Home	*http://www.ibs.utm.my/*	*R3*	84	112864
Advanced Informatics School (AIS)-Home	*http://www.ic.utm.my/utmais/*	*R3*	220	4925
Faculty of Computer Science and Information Systems (FSKSM)-Publication	*http://webs.cs.utm.my/publication.html*	*R4*	642	25
Faculty of Computer Science-Event	*http://se.fsksm.utm.my/news_events.html*	*R5*	0	6.8886
Faculty of Computer Science-Master of Science	*http://webs.cs.utm.my/mscse.html*	*R5*	0	6.7431
Advanced Informatics School (AIS)-Master Software	*http://www.ic.utm.my/utmais/academics/master-programme/master-of-software-engineering/*	*R5*	0	6.1069
Advanced Informatics School (AIS)-History	*http://www.ic.utm.my/utmais/about/*	*R5*	0	5.8285
International Business School (IBS)-MBA healthcare	*http://www.ibs.utm.my/programmes/mba-healthcare-management.html*	*R5*	0	5.5793
Faculty of Computer Science(FSKSM)-Sitemap	*http://webs.cs.utm.my/sitemap.html*	*R5*	0	5.484
Faculty of Computer Science(FSKSM)-Research	*http://webs.cs.utm.my/resgroup.html*	*R5*	0	5.3071

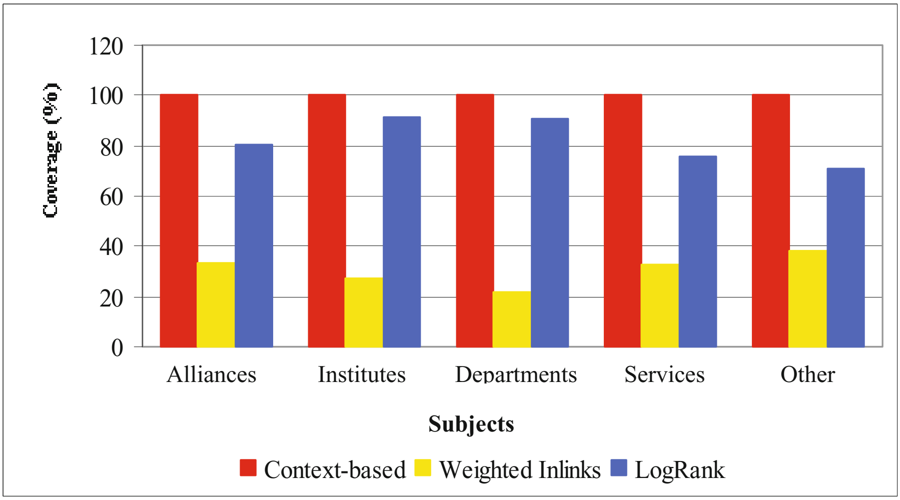


Fig. 2. Average coverage of authoritative dark pages in answer set in different classes of importance metrics

Table 8. *LogRank*-based importance value of different page categories

Classes	**Importance Value** (Second* Sessions)	**No. of Class Members**
H_1 type	$IV_{h_1} = 16324151214$	1
H_2 type	$\sum_{i=1}^{\lvert I \rvert} IV_{w_i} = 1161782402$	50
B type	$\sum_{j=1}^{\lvert J \rvert} IV_{w_j} = 207810164$	259
F type	$\sum_{k=1}^{\lvert K \rvert} IV_{w_k} = 61691689$	308
	$IV_W = 32827138307.3$	618

To calculate the exact importance of the Web site, h_1 parameter is set to 1.9, h_2 is set to 1.4, b is set to 0.8 and f is set to 0.3. The value for h_2 calculated based on one tenth of the ratio of the importance value of the home page to the sum of importance values of H_2 type pages. In the same way b and f values calculated based on one hundredth and one thousandth of the ratio of the importance value of the home page to the sum of importance values of B type and F type pages respectively.

The contribution of each class members in the total number of pages appeared in log file are 50, 259 and 308 for *I*, *J* and *K* out of $\left|N_L^{T_i}\right| = $ =618 Web pages respectively [4].

6 Conclusion

In this paper we reviewed our previously proposed Web page importance metric of *LogRank* that works based on analysis on server level clickstream data set. Upon employing this metric, the importance of each page is precise based on the observation period of log data and independent from the downloaded portion of the Web. In our approach, we will go beyond noticing the page importance in its connection pattern. Instead, the page credit computation is performed according to an algorithm which worked based on a textual structured log file in lieu of working with sparse matrixes of high dimensions that the latter results in an algorithm with reasonable time complexity. Also, the *LogRank*-based result set is an enhanced result set because of covering authoritative dark pages in the specific Web domain. Furthermore, this paper reviewed the *LogRank*-based approach for importance computation of the whole Web site with first reviewing our previously discussed Web page classification approach to be used in Web site importance estimation.

Since our research employs an improved version of a Web page classification approach in order to compute the rank of Web site and for rank initialization of infant Web pages, so our future work includes working on a more detail-oriented classification with more number of groups. Moreover, considering a minimum and a maximum threshold for page-stay durations is another open issue for future research.

Acknowledgment. The authors wish to thank Ministry of Higher Education Malaysia (MOHE) under Fundamental Research Funding Scheme (FRGS) (Grant no: 4F087) and Universiti Teknologi Malaysia for the funding of related research.

References

1. Abiteboul, S., Preda, M., Cobena, G.: Adaptive On-line Page Importance Computation. In: Proceeding of 12th International Conference on World Wide Web, pp. 280–290. ACM (2003); 1-58113-680-3/03/0005
2. Ahmadi-Abkenari, F., Selamat, A.: A Clickstream-Based Focused Trend Parallel Web Crawler. International Journal of Information Sciences 184, 266–281 (2012)
3. Ahmadi-Abkenari, F., Selamat, A.: LogRank: A Clickstream-based Web Page Importance Metric for Web Crawlers. JDCTA: International Journal of Digital Content Technology and its Applications 6(1), 200–207 (2012)
4. Ahmadi-Abkenari, F., Selamat, A.: Application of the Clickstream-Based Web Page Importance Metric in Web Site Ranking and Rank Initialization of Infant Web Pages. IJACT: International Journal of Advancements in Computing Technology 4(1), 351–358 (2012)
5. Attardi, G., Gulli, A., Sebastiani, F.: Automatic Web Page Categorization by link and conext analysis. In: Proceedings of THAI 1999, First European Symposium on Telematics, Hypermedia and Atificial Intelligence, Italy, pp. 105–119 (1999)

6. Bharat, K., Henzinger, M.R.: Improved Algorithms for Topic Distillation in a Hyperlinked Environment. In: Proceedings of the 21st ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 104–111 (1998)
7. Brin, S., Page, L.: The Anatomy of a Large-Scale Hypertextual Web Search Engine. Computer Networks 30(1-7), 107–117 (1998)
8. Barbosa, L., Freire, J.: An Adaptive Crawler for Locating Hidden Web Entry Points. In: Proceedings of the 16th International Conference on World Wide Web (2007)
9. Catledge, L., Pitkow, J.: Characterizing Browsing Behaviors on the World Wide Web. Computer Networks and ISDN Systems 27(6) (1995)
10. Chackrabarti, S.: Integrating Document Object Model with Hyperlinks for Enhanced Topic Distillation and Information Extraction. In: Proceedings of the 13th International World Wide Web Conference (WWW 2001), pp. 211–220 (2001)
11. Chackrabarti, S., Dom, B., Gibson, D., Kleinberg, J., Kumar, R., Raghavan, P., Rajagopalan, S., Tomkins, A.: Mining the Link Structure of the World Wide Web. IEEE Computer 32(8), 60–67 (1999)
12. Chackrabarti, S., Dom, B., Raghavan, P., Rajagopalan, S., Gibson, D., Kleinberg, J.: Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text. In: Proceedings of the 7th International World Wide Web Conference, WWW 7 (1998)
13. Chakrabarti, S., Van den Berg, M., Dom, B.: Focused Crawling: A New Approach to Topic Specific Web Resource Discovery. Computer Networks 31(11-16), 1623–1640 (1999)
14. Cho, J., Garcia-Molina, H.: Parallel Crawlers. In: Proceedings of 11th International Conference on World Wide Web. ACM Press (2002)
15. Cho, J., Garcia-Molina, H., Page, L.: Efficient Crawling through URL Ordering. In: Proceedings of 7th International Conference on World Wide Web, Brisbane, Australia (1998)
16. Cooley, R., Mobasher, B., Srivastava, J.: Data Preparation for Mining World Wide Web Browsing Patterns. Knowledge and Information Systems 1(1), 5–32 (1999)
17. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Arsman, R.: Indexing by Latent Semantic Analysis. Journal of American Society for Information Science 41, 391–407 (1990)
18. Giudici, P.: Applied Data Mining. In: Web Clickstream Analysis, ch. 8, pp. 229–253. Wiley Press (2003) ISBN: 0-470-84678-X
19. Guan, Z., Wang, C., Chen, C., Bu, J., Wang, J.: Guide Focused Crawlers Efficiently and Effectively Using On-line Topical Importance Estimation. In: Proceedings of the International Conference of SIGIR 2008. ACM (2008) 978-1-60558-164-4/08/07
20. Gyongyi, Z., Garcia-Molina, H., Pederson, J.: Combating Web Spam with TrustRank. In: Proceedings of 30th VLDB Conference, Toronto, Canada (2004)
21. Haveliwala, T.H.: Topic Sensitive PageRank. In: Proceedings of the WWW 2002, Hawaii, USA. ACM (2002), 1-58113-449-5/02/0005
22. Haveliwala, T.H.: Efficient Computation of PageRank. Technical Report, Stanford University, Stanford, CA, USA (1999)
23. Kleinberg, J.: Authoritative Sources in a Hyperlinked Environment. Journal of the ACM 46(5), 604–632 (1999)
24. Liu, B.: Web Data Mining. In: Information Retrieval and Web Search, ch. 6, pp. 183–215. Springer Press (2007) ISBN: 978-3-540-37881-5
25. Madhavan, J., Ko, D., Kot, L., Ganapathy, V., Rasmussen, A., Halevy, A.: Google's Deep Web Crawl. In: Proceedings of VLDB 2008, Auckland, New Zealand (2008)

26. Mangai, A., Kumar, S.: A Novel Approach for Web Page Classification Using Optimum Features. International Journal of Computer Science and Network Security (IJCSNS) 11(5) (2011)
27. Narayan, B.L., Murthy, C.A., Pal, S.K.: Topic Continuity for Web Document Categorization and Ranking. In: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, Halifax, Canada, pp. 310–315 (2004)
28. Ntoulas, A., Zerfos, P., Cho, J.: Downloading Texual Hidden Web Content Through Keyword Queries. In: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries (2005)
29. Raghavan, S., Garcia-Molina, H.: Crawling The Hidden Web. In: Proceedings of the 27th VLDB Conference, Roma, Italy (2001)
30. Richardson, M., Domingos, P.: The Intelligent Surfer: Probabilistic Combination of Link and Content Information in Page Rank, vol. 14. MIT Press, MA (2002)
31. Robertson, S.E., Walker, S., Beaulieu, M.: Okapi at TREC-7: Automatic Ad Hoc, Filtering, VLC and Filtering Tracks. In: Proceedings of the 7th Text Retrieval Conference (TREC-7), pp. 253–264 (1999)
32. Singhal, A.: Modern Information Retrieval: A Brief Overview. IEEE Data Engineering Bulletin 24(4), 35–43 (2001)
33. W3C, The common log file format (1995), `http://www.w3.org/Daemon/User/Config/Logging.html` (retrieved November 2010)
34. Ypma, A., Heskes, T.: Categorization of Web Pages and User Clustering with Mixtures of Hidden Markov Models. In: Zaïane, O.R., Srivastava, J., Spiliopoulou, M., Masand, B. (eds.) WebKDD 2003. LNCS (LNAI), vol. 2703, pp. 35–49. Springer, Heidelberg (2003)
35. Yu, P.S., Li, X., Liu, B.: Adding the Temporal Dimension to Search- A Case Study in Publication Search. In: Proceedings of Web Intelligence (WI 2005), pp. 543–549 (2005)

Opinion Analysis of Texts Extracted from the Social Web Contributions

Kristína Machová

Dept. of Cybernetics and Artificial Intelligence, Technical University, Letná 9,
042 00, Košice, Slovakia
kristina.machova@tuke.sk

Abstract. The paper focuses on the automatic opinion analysis related to web discussions. First, the paper introduces an approach to the extraction of texts from web forum discussion contributions and their commentaries as well as filtering these texts from irrelevant and junk information. Second, it describes variety of approaches to the sentiment analysis problem. It introduces a method for solving basic problems of opinion analysis (determination of word subjectivity, polarity as well as intensity of this polarity). The method solves the reversion of polarity by negation as well as determination of polarity intensity of word combinations. A dynamic coefficient for the word combinations processing is introduced and an implementation of the method is presented. In addition, the paper describes test results of the presented implementation and discussion of these results as well.

Keywords: Opinion analysis, opinion classification, social web, sentiment analysis, web discussions, dynamic coefficient.

1 Introduction

The Internet is a considerable source of various kinds of information and also a source of opinions and rankings. The opinions and rankings are very important for decision processes. These opinions and rankings can be obtained not only from prestigious agencies but also from common web users and their contributions to various social networks and discussion forums. They can contain a great number of rich information about opinions and attitudes of web users on some subject. These opinions and attitudes have to be extracted, processed and summarized into information, which is useful for some kind of decision. Because of great amount of these data, a manual processing of the data is usually not possible and, therefore, the automatic data extraction and subsequent automatic opinion analysis are desirable.

Opinion analysis represents a domain, which is a firm part of the field of social web analysis. The social web can be considered as an upgrade of the classic web. The classic web is the word-wide network of information sources, which are connected to each other. This network is under steady transition from the classic web through social and semantic web to the meta-web, what is illustrated in Fig. 1. The classic web can be illustrated with an idea of a world-wide billboard - anybody can publish some

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 42–68, 2013.

information piece or make it accessible for public inspection on the billboard (anybody who has necessary skills in web page creation - but considerably greater amount of web users have abilities only for reading this published information). On the other hand, the social web or web 2.0 reinforces social interactions among users and provides an opportunity for great majority of web users to contribute to web content. It can be said, that it increases the number of web content providers.

The same classical web has been also divergently developed into the semantic web by reinforcing knowledge connections. The semantic web [1] tries to achieve higher precision of web search than the search based on keywords and tries to obtain complex answers instead of only web pages, where the answer must be located and extracted as an additional step. It desires to adapt web pages structure to automatic processing by the introduction of explicit meta-data. It enables computers to understand data i.e. web pages. This principle is inevitable for new knowledge deduction. Knowledge can be represented by an ontology, which is a dictionary of concepts represented as tags of meta-languages. Thus, the main semantic technologies are: ontology (OWL), meta-languages (RDF, XML), logics and intelligent semantic agents. More information can be found on web pages of the Word Wide Web Consortium (W3C).

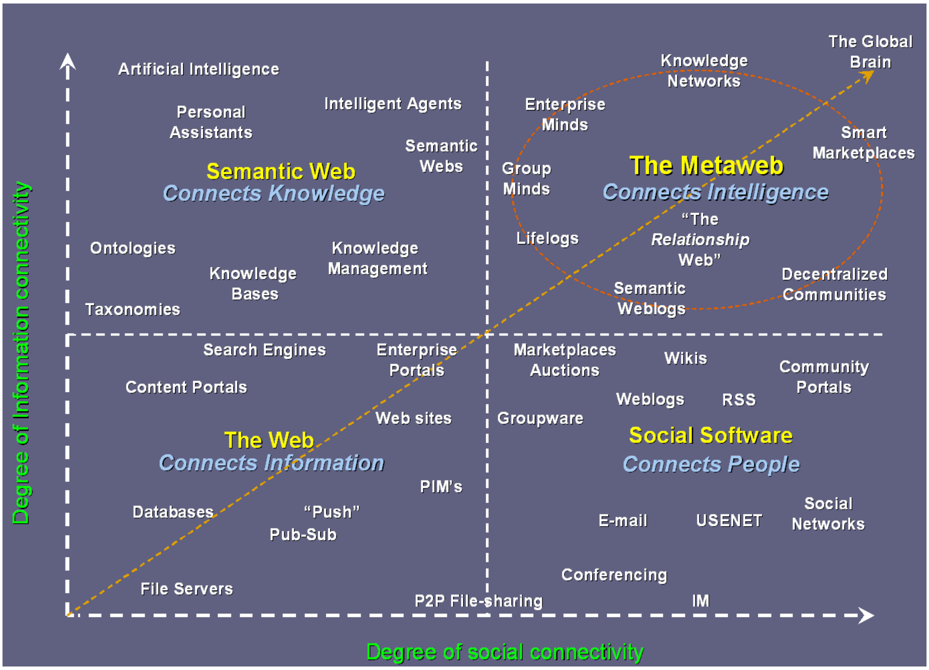


Fig. 1. Web and its trends for the future – source: www.mindingtheplanet.net

There is some idea about the final convergence both branches (the semantic and social web) into a meta-web in the future. This meta-web should be the result of the emergence of the classical, semantic and social web. It should have intelligence similar to human intelligence and it will enable intelligent connections. The social web is allocated on web pages provided by web applications, which enable creation of virtual social communities. This virtual social world supports reciprocal interactions and communications among web users. Within the social web there is the great number of social platforms with various purposes, for example: verbalization, sharing, networking and playing. Social networks tend also to facilitate communication among users as well as to support and maintain connections. In this way, social networks represent rich sources of information. According to [3], the social web represents web services, which enable users to:

- create public profile as a form of individual presentation (it can contain information representing user individuality as favorite citation, phrase…),
- create and edit a list of users from his/her social net,
- search lists of friends of other users,
- *share opinions, attitudes and comments,*
- share videos, photos, and other documents.

Social interactions among users are enabled by communication within social networks, by the possibility to contribute to web discussions, and so on. Specifically, discussion forums are large-scale data bases of opinions, attitudes and feelings of web users, who use the web for communication. Unlike classic data bases, they do not contain data in a structured form. For this reason, they need special methods for processing. One of such special methods is also opinion analysis. The main objective of opinion analysis is to summarize attitude of particular contributors to some particular theme. This theme can be, for example, an evaluation of some product, political situation, person (e.g. active in politics), event or company.

Opinion analysis or opinion classification can be used in those fields where the aggregation of a large amount of opinions into integrated information is needed. The input to opinion classification can be represented by a large amount of discussion contributions (e.g. content of a discussion forum) and the output of the classification is summarising information, for example “Users are satisfied with this product” or “People perceive this reform negatively”. From the point of view of a consumer, two kinds of information are important for decision making about purchase of a product. First, it is information about price and properties of the product, which usually are available on web pages of a producer or a seller. Second, it is information about satisfaction of other consumers with the product. The opinion classification can offer this information to prospective consumer. From the point of view of a producer, information about satisfaction and needs of consumers is also very important. The classic way of obtaining this information is performing market research. The market research carried out by telephone or by questionnaires is usually rather expensive and time consuming. The promptness of such information elicitation is a matter of principle. User contribution analysis provided by a system utilising opinion classification can offer the information about clients’ satisfaction more quickly.

2 Data Extraction from the Social Web

Data extraction is the process of data selection from larger amount of data, which can be weakly structured or even unstructured. It has to be provided in a way which ensures acquiring data without noise. Data extraction from web pages [27] is a transformation of web data into relevant data, for example addresses, telephone numbers, e-mails, product costs, product descriptions, contributions, opinions, etc. According to the type of the extracted data, the extraction can be divided into pattern extraction (e-mail address, hypertext links, date) and data mining. The data mining requires an analysis of the tree structure of an HTML document. There are more approaches to web data extraction possible:

- *manual methods* – based on human activity, time consuming but highly precise,
- *controlled machine learning methods* – based on artificial intelligence methods, human is necessary for the evaluation of the extracted data,
- *automatic techniques* – without human intervention but less reliable.

The process of web data extraction has to solve some problems as:

- low quality of sources which are not fully compliant to W3C standards,
- quite fast changes of web sources,
- heterogeneity and complexity of data types,
- inconsistent semantics and inconsistent structure of objects.

Many web sources and web presentations or simply web pages have the form of HTML documents. HTML tags do not contain any information about semantics of data, but only information about their form. Thus, it is a little bit complicated to obtain relevant data from HTML documents. The structure of these documents was oriented on their visualization in a web browser. This visualization has to be understandable for a human. The structure of an HTML document can be represented by an acyclic graph – a tree, which is illustrated in Fig. 2.

An automatic data extraction requires identification of certain level of dipping into the HTML tree. One of approaches to such data extraction from trees is the method based on partial tree alignment. *Partial tree alignment technique* [27] is based on the presumption that web page contains more structured notes about objects. These data are stored in databases and they are visualized on the web page using a fixed template. The problem of data extraction is focused on segmentation of these data records and extraction of particular data items.

The problem under focus is the extraction of target data. Our target data are texts of discussion contributions. Sources of discussion contributions can be web discussion forums and web pages, which enable adding opinions in the form of commentaries. Discussion contributions can be contained in various parts of web pages and their structure in the HTML code can have various forms.

2.1 Opinion Extraction from Discussion Forums

Within discussion forums, contributions occupy almost the whole web page. The structure of a discussion forum is often created with the aid of HTML tags as TABLE or DIV. In the case of table usage, particular discussion contributions are represented by rows of this table. The rows contain various additive data about users of the forums. These data represent noise for us and so they have to be identified and removed. Such additive data can be also date of contribution creation. On the other hand, date can make the process of data extraction a bit easier. Fig. 3 illustrates various positions of discussion contributions within discussion forum pages (taken from source: http://www.politicalforum.com/).

When a discussion forum as a source of discussion contributions is considered, some target regions from the web page have to be extracted. These regions have to contain pure text of contributions and they are marked by the red line in Fig.3.

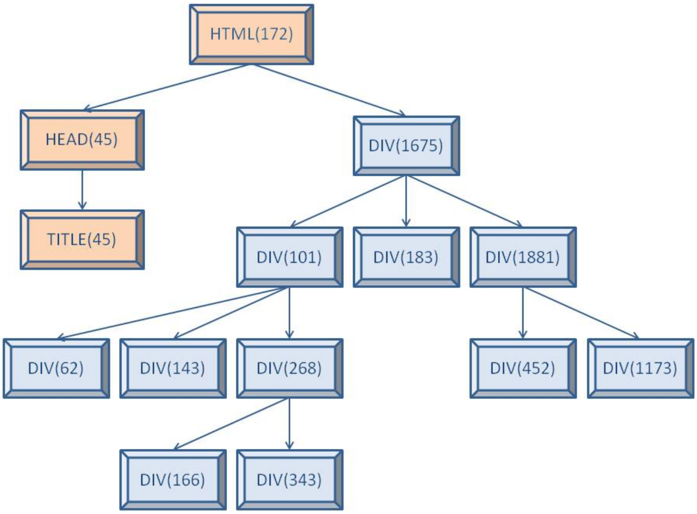


Fig. 2. An example of HTML document tree representation

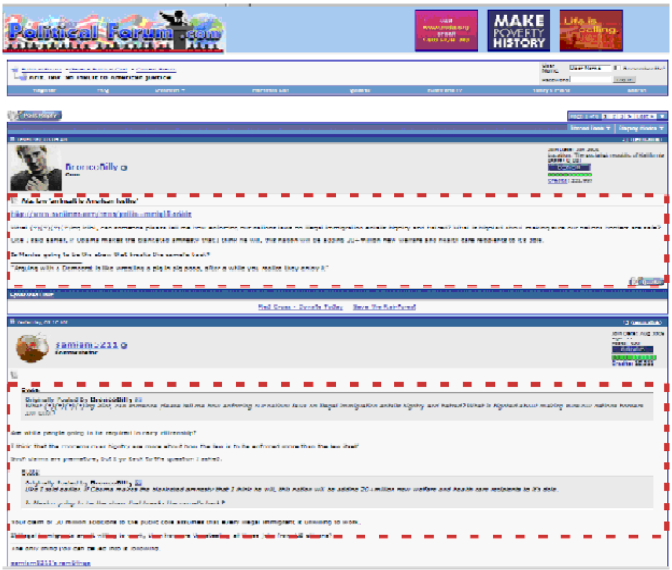


Fig. 3. An example of an discussion forum as the source of contributions

2.2 Opinion Extraction from Web Pages with Commentaries

Similarly, when a web page enabling to add user comments is considered as a source of discussion contributions, some regions from such web page have to be extracted. These regions contain pure text of commentaries and they are marked by the red line in Fig. 4. The text, which is commented, is marked by the green line in the same figure. The commentaries must be distinguished from the commented text.

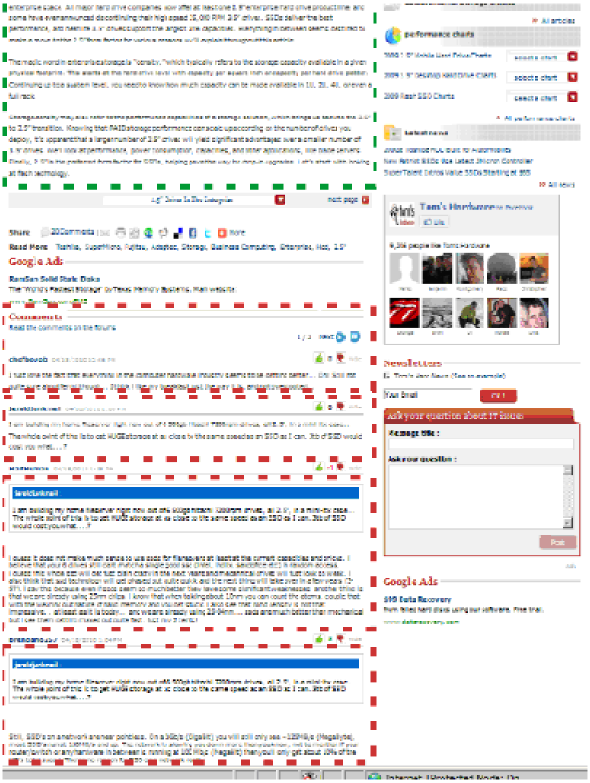


Fig. 4. An example of a web page with commentaries as the source of contributions

This commented text can be an introductory information text (for example product reviews or some news) which is followed by a discussion. Such web pages with commentaries can be pages of product sellers, information channels, etc. In the case of such pages, there is a need to describe exactly the region of web discussion. This description should be set apart from the introductory text. The introductory text is usually represented by a sequence of text blocks (<p> … </p>). They create continual region, which we must to avoid. Such pages contain more noise in the form of advertisements added into text or into menu with hyperlinks.

Users can create web pages content and in this way they can change the structure of the HTML code for example using DOM (Document Object Model). DOM enables users to go through the structure of an HTML code represented as the tree, to expand

and to search for given nodes of this tree. The contributions or comments, we are interesting in, are located in this tree.

The question is: How to determine the level in the tree, we are interested in because contributions we search for are located on this level? For example, the target contributions are marked by the red line in Fig.4. Unfortunately, these target regions can contain some noise as user names, dates, hypertext links, etc. One way how to clean blocks from the noise is to delete all double texts, which are in each block. We must also select such nodes in the tree, which do not contain code blocks created by introductory region without target data, because these regions must be different from the introductory commented text (marked by the green line in Fig.4).

2.3 Design of Extraction Algorithm

On the basis of knowledge about the problem of contributions extraction (see previous sections), the following facts about target region can be postulated:

- It should contain greater amount of text.
- It should contain lower number of hypertext links.
- It has to contain dots, question and exclamation marks (text divided into several sentences).
- It should contain lower number of elements of the type <P>.

These declarations enable to define coefficient "*k*", which represents properties of the target region within the formula (1).

$$. \qquad k = \frac{Z * I * B * T}{L + P} \tag{1}$$

The parameters of the formula (1) are defined in the following way:

Z … is the number of all symbols in the text of contribution
I … is the number of punctuation marks in the text
B … is the number of elements of the type <BR> in the text
T … is the number of TIMESTAMPS in the text
L … is the number of links (hypertext) in the text
P … is the number of tags <P> in the text

Moreover, it holds that: *I=0* → *I=1*, *B=0* → *B=1*, *T=0* → *T=1*, *L=0* → *L=1*.

The motivation and rationale for selecting these specific parameters is following:

Z … we suppose that our target region is such part of web page presenting a web discussion, which contains greater amount of text – enough text for expressing relevant opinion

I … we suppose that valuable opinion should be split into more sentences and so we want to avoid one sentence contributions (each punctuation mark must follow a word not number or other punctuation mark)

B … we suppose that more structured text is written by higher authority

T … TIMESTAMPS are closely connected with discussion contribution, because time of each contribution insertion is recorded

L … we supposed that the text containing more links (hypertext) does not contain original opinion of a contributor and so we put this parameter into the denominator of the formula (1) to avoid mediated opinions

P …we suppose that text containing more paragraphs does not represent the target region and so we put it into the denominator of the formula (1)

All nodes, on the particular level of the tree, are evaluated by the coefficient "*k*". Only the node with the highest value of this coefficient is expanded as it can be seen in Fig. 5.

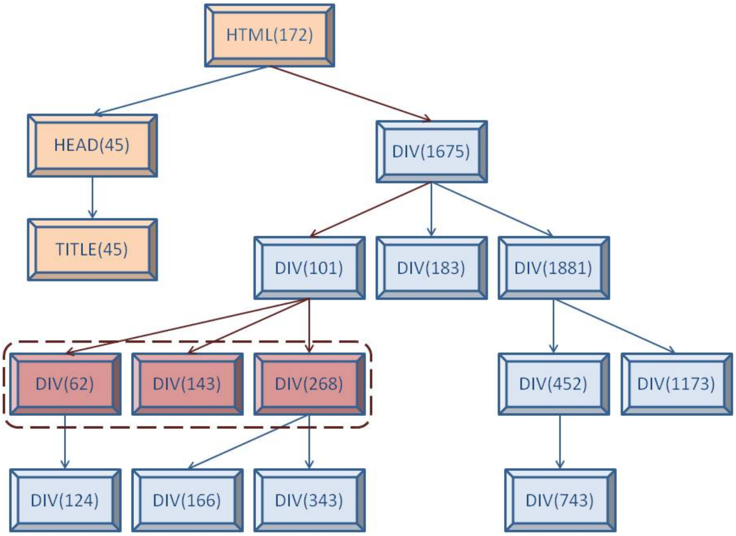


Fig. 5. An example of the usage of the coefficient k for node expansion (only nodes marked by red line are expanded)

This approach helps to avoid regions as hypertext menus, which contain lower amount of text (lower *Z* and *I*) but higher number of hypertext links (*L*). The parameters *T* and *B* prefer discussion contribution regions. The parameter *P* helps to discriminate introductory commented text, which usually contains more <P> tags. The target discussion contributions contain usually only one <P> tag and more <BR> tags (parameter *B*). It holds, that parameters *Z, I, B* and *T* increase the probability, that the target regions were found – comments or contributions. On the other hand, parameters *L* and *P* decrease this probability.

The process of extraction is divided into two steps:

- retrieval of the blocks with contribution texts,
- filtering these blocks from noise.

This algorithm [20] was designed for the process of extraction of contribution or commentary texts. A general description of this recursive algorithm is as follows:

```
Evaluate all incoming nodes by "k" coefficient
If    number of nodes = 1
Then  retrieve node descendants
      and go at the beginning
Else  If (max/sum) > 0.5
      Then  retrieve max node descendants
            and go at the beginning
      Else retrieve blocks by descendant wrapper
Clean retrieve blocks from noise
```

The expression (max/sum) represents ratio of coefficient k of the node to the sum of coefficients k of all nodes. This expression is used for the selection of the best node on the same level. This best or max node is expanded.

The above presented extraction algorithm selects blocks containing discussion contributions. These blocks can contain noise. This noise can be represented by: names of contributors, dates, advertisements, hyperlinks and so on. Complete noise deletion is complicated and it may not give valuable results. One way for cleaning textual blocks from noise is the elimination of all doubly generated texts. Another universal possibility is to find only such nodes in HTML document tree representation, which contain text of some contribution with maximal value of the coefficient *k* within the block (web page) containing the discussion contributions. Coefficient "*k*" (1) is designed in the way, which ensures the target region finding.

The extraction of discussion contribution or commentary texts is only the first step which should be followed by opinion and sentiment analysis of these contributions and commentaries.

2.4 Testing of the Extraction Algorithm Implementation

The extraction algorithm was tested on contributions from web discussions and commentaries on introductory text within several various domains (see Table 1). Within the testing of the extraction algorithm, the extraction was considered as appropriate, when the method had selected a whole block of contribution or commentary text with noise, which, in relation to this contribution text, was for example author, datum of adding of the contribution, and so on. For identifying weak places in the algorithm functioning, the number of contribution extracted from given web page was gradually decreased. The testing results in the form of extraction recall value are presented in the Table 1. These tests can be considered as successful. Within the domain 3, 4 and 5 a little bit weaker results were achieved. The reason can be caused by setting the minimal value of coefficient *k=20* within these experiments. This setting evoked problems in extraction of the contributions with small number of words. The resulting averaged recall of these tests on testing domains was from the interval <0.722, 1>.

Table 1. The results of the extraction algorithm testing

Experi-ment	URL	Contributions number	Extractions number	Recall
1.	**amazon.com**	10	10	1.000
		9	9	1.000
		6	6	1.000
		4	4	1.000
2.	**imdb.com**	162	158	0.975
		8	8	1.000
		7	7	1.000
		5	5	1.000
		4	4	1.000
3.	**tomsguide.com**	31	23	0.742
		20	18	0.900
		20	20	1.000
		18	13	0.722
		6	6	1.000
4.	**tomshardware.com**	20	18	0.900
		20	20	1.000
		20	20	1.000
		21	17	0.810
		18	13	0.722
5.	**cnn.com**	981	924	0.942
		382	345	0.903
		26	20	0.769
		21	17	0.810
		18	17	0.944
6.	**debatepolitics.com**	10	10	1.000
7.	**thought.com**	10	10	1.000
8.	**forum.khurram.ca**	15	15	1.000
9.	**cinemablend.com/reviews**	20	18	0.900
10.	**forum.notebookreview.com**	10	10	1.000

The cleaning of retrieved texts was completely successful within some domains. It was found, that data extraction is sensitive on parameters settings. After setting suitable parameters of the designed algorithm, it was possible to extract all discussion contributions within the selected domains.

3 Opinion Analysis

Some information, which is formulated in a selected language, can be a fact or an opinion [15]. The fact is objective information about an entity, event or its properties. On the other hand, the opinion usually reflects a subjective approach, which describes human feelings and evaluates the entities, events and properties. An extensive research has been carried out in the field of knowledge processing and discovery (data mining, text classification) but substantially less work has been done in the field of opinion mining. And just the opinion is the key element that affects human decisions and consequently their behaviour. There are some works [14], [28] and [23], which tried to apply automatic opinion analysis on the social web. These works were mainly focused on the product and service benchmarking, market intelligence, advertisement placement and opinion retrieval. Thus, the opinion analysis is a promising research field.

3.1 Various Approaches to the Problem of Opinion Analysis

According to [24], opinion analysis can be used in the following areas:

- *Sentiment and subjectivity classification* is a domain where the problem of sentiment and opinion analysis is considered as a classification problem. This problem is based on classification to positive or negative opinion class. There are two different approaches to opinion classification. First, it is the classification of the whole block of text (for example document, blog, etc.) to positive or negative class. Second, it is the classification of individual sentences to subjective and objective classes.
- *Feature based sentiment analysis* searches for an entity composed of given certain parts, properties and then determines its polarity. The entity may be a product, a service, a person, an organization, an event, etc. The fundamental difference to the previous method is the possibility to identify some object and its parameters and to determine the polarity of a selected parameter or property of the object (e.g., mobile and its display).
- *Sentiment analysis based on comparison.* The entities are compared and analyzed in terms of relationships between them. These relationships can be *non-equal gradable* (an entity is greater than another entity), *equality comparison* (an entity is as good as the second one), *superlative comparison* (an entity is better than anything else) and *non-gradable* (an entity is different from another entity). Methods that are used in this field can be found in [8].
- *Opinion search* can be based on keywords searching. For comparison, the classical search is required to search for result about some fact and in general it holds, that one fact is as good as the number of other facts about the same entity. Thus, a user can be satisfied with results obtained after the first search and she/he has not to repeat the routine. In opinion search, it is clear that one opinion is not always as good as a number of other opinions and, therefore, user is not satisfied with the first retrieval result - first opinion only [15].
- *Opinion spam search* is an area where text search is performed within discussion forums with the purpose of finding those opinions that are not expected and/or useful for the discussed issue (misguided opinions, irrelevant opinions, advertisements, questions and so on) [11].
- *Opinions utility determination* is the last research area, which corresponds to the usefulness and quality of opinions. Rated opinions can be sorted and the reader is able to obtain opinions of the highest quality [9].

In general, opinion analysis can be based on opinion mining or on dictionary approach. Our design of opinion analysis method is based on a dictionary approach and belongs to the first area – sentiment and subjectivity classification.

3.2 Basic Problems of Opinion Analysis

Three basic problems of opinion analysis are: *word subjectivity identification, word polarity (orientation) determination and determination of intensity of the polarity.* Opinion analysis focuses on those words, which are able to express *subjectivity* very well - mainly adjectives (e.g. 'perfect') and adverbs (e.g. 'beautifully') are considered. On the other hand, other word classes must be considered as well in order to achieve satisfactory precision, for example nouns (e.g. 'bomb') or verbs (e.g. 'devastate'). The words with subjectivity are important for opinion analysis; therefore they are identified and inserted into the vocabulary. Words with subjectivity are inserted into the constructed vocabulary together with their polarity.

The *polarity of words* forms a basis for the polarity determination of the whole discussion. There are three basic degrees of polarity being distinguished: positive (e.g. 'perfect', 'attract'), negative (e.g. 'junk', 'shocking', 'absurdity', 'destroyed') and neutral (e.g. 'averaged', 'effectively'). This scale can be refined to use more possible levels if needed. The determination of the polarity of words is connected with a problem of word polarity reversion – the reversion can be done by using negation, for example 'It was not very attractive film'. This problem serves as an argument for the extension of single words polarity determination to polarity determination of word combinations (considering whole sentences or parts of sentences).

The *intensity of word polarity* represents a measure of the ability of words to support the proof or disproof of a certain opinion. The polarity intensity of words can be determined according to a defined scale, which helps to classify words into more categories. Table 2 illustrates three such scales with different numbers of degrees.

The polarity intensity can be expressed both verbally as well as numerically. The numerical representation is more suitable for subsequent processing by computers. Discussion contributions very often contain some word combinations, which increase (decrease) the weak (strong) intensity of polarity of an original word, for example: 'surprisingly nice', 'high quality', 'markedly weaker' and 'extremely low-class'.

4 Dictionary Creation

In order to support the process of opinion analysis, it is necessary to create a dictionary. The opinion analysis systems commonly utilise large dictionaries, which are called seed-lists. For example WorldNet can be used as a basis for the creation of such seed-list dictionary. In accordance with [2], it is possible to derive tagsonomies from crowd. Similarly, we attempted to derive a dictionary directly from web discussions. This dictionary is specialized for a particular domain, the utilised web discussions focus on. Since it is possible to use it for classification of words into predefined categories, we denote it as a classification vocabulary.

Table 2. Scales using verbal or numerical representation of the intensity of word polarity

Number of Degrees	Scales of polarity intensity	
2	negative	positive
6	weak, gently, strong negative	weak, gently, strong positive
8	-1, -2, -3, -4	1, 2, 3, 4

There are some other available lexicons like SentiWordNet [21] and SenticNet [4]. We did not use these lexicons for our classification dictionary creation, because we focused on creation not so large and complex dictionary. We tended to generate smaller but more precise one. This is the reason that we decided to create this dictionary directly from analysed texts taken from web discussions. Such vocabulary better covers live language of the web discussion contributors and usually is smaller but more precise. For example, the dictionary like SenticNet, which focuses on recognition of various types of emotions seems to be less successful in classification of opinions from technical domains, where mobile telephones, notebooks or cameras are reviewed. The precision of opinion classification can be significantly increased using some words, which are specific for this (technical) domain and which are not used in other domains (e.g. film reviews) at all. What is more, we were not focused on recognition of so wide scale of emotions as SenticNet. Our interest was to recognize if the opinion was positive or negative and in which degree. Another difference between our approach and approach presented in [4] which started sentiment analysis only from the paragraph level is that, we started sentiment classification from lexical unit level. The last difference is the fact, that we did not use semantic technologies (ontology). But we intend to enrich our approach with semantic dimension in the future.

4.1 Classification Dictionary Creation Method

As stated before, our dictionary was derived directly from web discussions. Such dictionary is specialized for a particular domain. It can be created from more than one web discussion concerning the same domain. Many of web discussion respondents do use language far from being perfect. Therefore, our classification dictionary has to be able to adapt to colloquial language of users of the Internet including slang, absence of diacritical marks, and frequent mistakes.

Our application of opinion classification has two interfaces: for a user and for an administrator. Only the administrator can record, edit and delete entries in the classification dictionary and create dictionaries of the categories. The dictionary is filled by selected words of four word classes (adjectives, adverbs and some nouns and verbs) together with their category (1, 2, 8 and 9). The dictionary contains also intensifiers (4) and negatives (3). This dictionary can be modified during the process of opinion classification by administrator. The application enables the administrator to select some words from currently analyzed text, which were not found in the dictionary (unknown for dictionary) and to add them to this dictionary. The dictionary can be incrementally extended during the analysing of new web discussions from the given domain.

Only the administrator can enter menu items “dictionary” and “category dictionary” after log in. Within this possibility, administrator can add new words to the dictionary, can use filter to search for words with sentiment and can edit any word in the dictionary. In the case, when administrator selects menu item “analyzing” and the analysed text contains words with sentiment, which are not in the dictionary yet, the new window pop-up. This window enables administrator to add this word into the suitable category of the dictionary.

Classification vocabulary can be created manually or automatically. Automatic generation of the vocabulary can be based on the method of keywords generation presented in [18]. We intend to develop a novel automatic generation of classification dictionaries. The step of dictionary generation would be a part of the process of opinion classification. The dictionary will be created within the stage of pre-processing of web discussions, which are to be analyzed. This pre-processing can be provided by known methods of text processing using suitable weight techniques and some statistical methods for selection of characteristic words from the given domain, which are able to express sentiment and opinion. After selecting a hyperlink of a given web discussion, all texts from this discussion will be extracted. Consequently, a dictionary will be automatically generated from characteristic words from these texts for the purpose of the opinion classification within the specified domain.

Our application of opinion classification can be applied for different languages. The language of the opinion analysis is given by language used by the classification dictionary. The language of the classification dictionary is given by language used by the web discussions, because the dictionary is loaded by some words from the live discussions. So, the system is able to adapt to different languages. It is also able to adapt to users, because it transfers words from living language of contributors into the dictionary.

On the other hand, our application does not support usage of many languages simultaneously. The values of selected specific factors (*Z, I, B, T, L and P*) are not affected by the selected language. These factors (as they are defined in the section 2.3) do not aim to reflect the grammar specifics of some given language. They are designed to reflect structure of web discussions and to identify the target region – contribution containing valuable opinion - using the structural and statistical information.

5 Design of Opinion Classification Method

The design of an opinion classification method is based on dictionary approach. It has to consider all steps of the classification process and provide them in the right and logical sequence. The method we have designed solves the following problems:

- Basic problems of opinion analysis
- Word polarity reversion by negation
- Determination of the intensity of polarity

- Establishment of a dynamic coefficient
- Polarity determination of word combinations

Our access takes into account not only polarity of single words but also the intensity of polarity of word combinations including negation. Our method analyzes texts of discussion contributions from a certain domain and for this domain a classification vocabulary is generated from the given texts. The quality of the vocabulary and its cardinality play the key role in the process of opinion classification.

The method transforms textual content of a discussion contribution into an array of words. Each word with subjectivity is assigned a numerical value (numerical code) as it is illustrated in Fig. 6. This value represents the category of word polarity to which the given word belongs (see Table 3). Particular sentences are identified. First non zero value of word category starts the creation of word combination procedure. The length of a certain combination is limited by a coefficient *K*. Each combination of words is also assigned a numerical value which represents a polarity degree from the <-3, 3> interval. The polarity degrees of all word combinations within the given text form the polarity of this text as a whole. Subsequently, the polarity of the whole discussion can be calculated from the polarities of all contributions (texts).

The whole contribution is considered to be positive/negative when it contains more positive/negative word combinations. Similarly, the whole discussion is considered to be positive/negative when it contains more positive/negative word contributions.

The analyzed contribution can contain only one sentence, but more often the contribution consist of more sentences. The processing of each sentence can be divided into processing of one or more lexical units. Each lexical unit is represented by one word combination. The results of each word combination analysis are summarized and combined into useful information containing opinion of the whole contribution. Consequently, the opinion of the whole discussion is determined. The presented algorithm of the opinion analysis does not use any method for context recognition. We suppose that the subject of the discussion is known and it is given by domain of the web discussion. Our design is based on the presumption, that all sentences are related to the same subject.

The neutral contribution (discussion) contains the same number of positive and negative word combinations (contributions). This approach to neutrality determination is rather strict. A more benevolent approach uses the following rule for neutrality detection (see formula (2)).

$$\text{IF } |\text{Number\_pozit} - \text{Number\_negat}| \leq \text{H THEN neutrality.} \tag{2}$$

Where threshold *H* represents range of neutrality, which can be changed by setting another value of the *H* parameter ($H \geq 1$ and it is an integer). Strict approach to neutrality with $H=0$ is more suitable for very short contributions, because such short contributions can contain only one sentence and only one positive or negative word. Wider neutrality range could absorb this word and subsequently the system of opinion classification can evaluate it as a neutral contribution. The wider neutrality range is more suitable for longer contributions processing.

5.1 Basic Problems Solution

In our approach, words with subjectivity are selected and the category value from a given scale (the scale from 0 to 9 is used) is assigned to each of these words, what is illustrated in Fig. 6. The words with positive polarity are classified to categories 1 or 8 (see Table 3). Similarly, the words representing negative polarity are classified to categories 2 or 9 and words with neutral polarity to 0 category.

Table 3. Categories of words polarity

weak positive and strong positive	1 and 8
weak negative and strong negative	2 and 9
neutral	0
negation – polarity reversion	3
increasing of polarity intensity	4

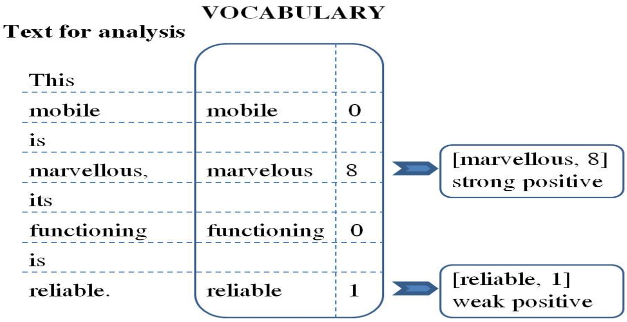


Fig. 6. Polarity determination of words from the sentence 'This mobile is marvellous, its functioning is reliable'

To illustrate usage of these categories, Fig. 6 illustrates categorization of words into polarity categories based on the example 'This mobile is marvellous and its functioning is reliable'. This word classification system also solves determination of the intensity of the polarity, because values 1 and 2 represent weak polarity in contrast to values 8 and 9, which represent strong polarity (being positive or negative). Thus, the designed method uses a five degree scale of the intensity of polarity determination (including neutral).

There is one more addition in our design for determining the intensity of polarity. It is the category 4 used for each word, which increases the intensity of polarity of another word in the same word combination (e.g. 'high quality').

To summarise, the used polarity categories are introduced in Table 3. All words with subjectivity are expected to be inserted into the classification vocabulary together with their category codes.

5.2 Word Polarity Reversion by Negation

The reversion of word polarity caused by the usage of negation enables to reflect actual meaning and therefore to increase precision of opinion classification. The words, which represent negation (e.g. 'none', 'no') belong to the category 3. This category can be used only in the combination with another category (1, 2, 8 or 9). It changes positive polarity into negative polarity and vice versa within the same degree of intensity (weak or strong) as it can be seen in Table 4.

Table 4. Word polarity reversion by negation

3 + 1	3 + 8	3 + 2	3 + 9
negation + weak positive = *weak negative*	negation + strong positive = *strong negative*	negation + weak negative = *weak positive*	negation + strong negative = *strong positive*

The polarity reversion is a rather complicated issue due to the fact, that the structure of various sentences is not homogenous. For example, the sentence 'This mobile isn't reliable' can be represented by the code 0031 (the code of a sentence is created by replacing each word with the number indicating its category). Another sentence 'It isn't, according to my opinion, reliable mobile' has the same meaning but different code 03000010. The aim of our technique is to recognise various codes 0031 and 03000010 as opinions with the same polarity. Thus, there is a need of some dynamic coefficient, which enables to estimate an appropriate length of those word combinations, which will be processed together as one lexical unit. In general, it enables to process one sentence as two different combinations – lexical units.

5.3 Determination of the Intensity of Polarity

Words, which increase the intensity of polarity, have no polarity and their influence on polarity of a lexical unit can be evaluated only within a combination with the given lexical unit. These words belong to the category 4. Table 5 presents two different examples of such combinations.

Both these combinations contain a word increasing the intensity of polarity. The word combinations are represented with codes 00041 and 04002. Words from the category 4 are usually adverbs (e.g. 'very', 'really', 'totally'). Processing of the words enabling to increase the intensity of word polarity needs to use the dynamic coefficient in a similar manner as the negation processing.

5.4 Dynamic Coefficient Determination

The designed method of opinion classification has an ambition to manage the variability of sentence structures using the dynamic coefficient *DC*. The value of this parameter is being dynamically changed during processing of different lexical units. The dynamic coefficient adapts itself to the code length of a lexical unit (sequence of words) under investigation. The value *DC* represents the number of words, which are included into the same word combination (beginning from the first non-zero word code in the sequence of words). In the case, when the value is higher than the number of words in the sentence, this value is dynamically decreased in order to ensure, that the combination contains only words from the investigated sentence, not from the beginning of the following sentence. A word combination can be shortened also in some other cases. For example, let us take the case *DC=4* while the combination 3011 is being processed. In this case, two disjunctive combinations are created 301 (*DC=3*) and 1 (*DC=1*). Table 6 illustrates the principle of using the dynamical coefficient.

Table 5. Analysis of lexical units with word increasing polarity intensity

This	**mobile**	**is**	**totally**	**conforming.**
0-neutral	0-neutral	0-neutral	**4 + intensity**	1-weak positive
It	**really**	**drives**	**me**	**mad**
0-neutral	**4 + intensity**	0-neutral	0-neutral	2-weak negative

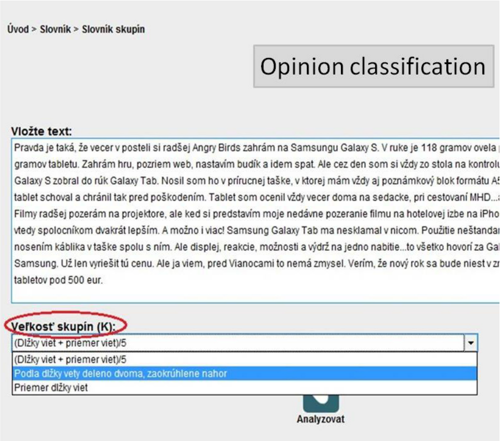
Table 6. Principle of employing the dynamical coefficient *DC* (Words processed within one combination are given in bold)

DC	Never	buy	this	nice	mobile.
1	**3**	0	0	**1**	0
2	**3**	**0**	0	**1**	**0**
4	**3**	**0**	**0**	**1**	0

On the other hand, the value can be increased in some cases. As we can see in Table 6, value *DC=1* is not appropriate for processing of the sentence ‘Never buy this nice mobile!’, because negation ‘never’ would be in a combination different from the combination comprising the word ‘nice’, to which the negation is related. Setting *DC=1* represents processing of words in isolation from each other. The alternative *DC=2* allows processing of neighbouring words as combinations, but it does not prevent the isolation of negation from relating word either. This sentence can be satisfactorily processed only when the coefficient has value *DC≥4*.

The length of a processed word combination is represented by a dynamic coefficient. The value of the dynamic coefficient DC is being dynamically changed during processing of different lexical units. The value of this dynamic coefficient is determined in an automatic way. The dynamic coefficient adapts itself to the length of sentences (sequence of words) under investigation. The value DC represents the number of words, which are included into the same word combination.

Three ways of the dynamic coefficient determination (must be done before the process of opinion classification starts) was proposed. These three ways are implemented and used in the modification of the opinion classification application, which is illustrated in Fig.7.


Úvod > Slovník > Slovník skupin

Opinion classification

Vložte text:

Pravda je taká, že vecer v posteli si radšej Angry Birds zahrám na Samsungu Galaxy S. V ruke je 118 gramov ovela
gramov tabletu. Zahrám hru, pozriem web, nastavím budík a idem spat. Ale cez den som si vždy zo stola na kontrol
Galaxy S zobral do rúk Galaxy Tab. Nosil som ho v prírucnej taške, v ktorej mám vždy aj poznámkový blok formátu A
tablet schoval a chránil tak pred poškodením. Tablet som ocenil vždy vecer doma na sedacke, pri cestovaní MHD...
Filmy radšej pozerám na projektore, ale ked si predstavím moje nedávne pozeranie filmu na hotelovej izbe na iPho
vtedy spolocníkom dvakrát lepším. A možno i viac! Samsung Galaxy Tab ma nesklamal v nicom. Použitie neštanda
nosením káblika v taške spolu s ním. Ale displej, reakcie, možnosti a výdrž na jedno nabitie...to všetko hovorí za Ga
Samsung. Už len vyriešit tú cenu. Ale ja viem, pred Vianocami to nemá zmysel. Verím, že nový rok sa bude niest v z
tabletov pod 500 eur.

Veľkosť skupín (K):

(Dĺžky viet + priemer viet)/5

(Dĺžky viet + priemer viet)/5
Podla dĺžky vety deleno dvoma, zaokrúhlene nahor
Priemer dĺžky viet

Fig. 7. Three ways of the dynamic coefficient counting

The first used method of dynamic coefficient setting calculates the *average length of all sentences* of the discussion contribution text, which is analyzed. Thus, each contribution text from a live discussion can have different value of the dynamic coefficient. The value of the dynamic coefficient is used for processing of all sentences of the given contribution. So, each sentence is processed using the same value of the dynamic coefficient, although real lengths of these sentences are different.

The second used way of dynamic coefficient setting is based on calculation of the *half length of each sentence* of the discussion contribution text. If needed, the determined value is rounded up. Each analyzed sentence is processed using different value of the dynamic coefficient.

The last used method is the *hybrid approach***,** which determines the value of dynamic coefficient as an average of two values obtained from two previous ways of dynamic coefficient setting: *Average length of all sentences* and *Half length of each sentence*.

5.5 Polarity Determination of Word Combinations

Generation of suitable word combinations using the dynamic coefficient *K* is the key factor of effective opinion classification. These combinations are sets words (their cardinality differs according to changing value of *DC*), to which a polarity degree, representing the polarity of the word combination as a whole, is assigned. This polarity degree is an integer from the set {-3, -2, -1, 1, 2, 3}. For example, the polarity degree 2 in the second column of the Table 7 can be interpreted as strong positive polarity (SP) or weak positive polarity modified by intensity (WP + I). This intensity is introduced into the given combination by another word, which can precede or follow the word with weak positive polarity. Table 7 illustrates examples of most often used word combinations for *DC* from 2 to 4 together with their interpretation and resulting polarity degree.

According to the second column of the Table 7, the polarity degree 2 (with its interpretation SP or WP + I) for *DC=4* represents two basic alternatives. The first possible alternative is represented by a strong positive word (8), which is complemented by neutral words (8000). The second possibility is a weak positive word (1) followed (within the same combination) by word increasing polarity intensity (4) and they are complemented by two neutral words in order to form a combination of the given length (4100). These words having non-zero code can be differently ordered within the given word combination (e.g. 4010, 4001).

Table 7. Polarity degree determination of words combinations with various code lengths (SP+I is Strong Positive + Intensity, SP or WP+I represents Strong Positive or Weak Positive + Intensity and WP is Weak Positive. Similarly, it holds for negative polarity)

Interpre-tation	SP + I	SP or WP + I	WP	WN	SN or WN + I	SN + I
DC = 2	48	80, 41	10, 32, 23	20, 31, 13	90, 42	49
DC = 3	480,408	800, 410, 401	100, 320, 230, 302, 203	200, 310, 130, 301, 103	900, 420, 402	490, 409
DC = 4	4800, 4080, 4008	8000, 4100, 4010, 4001	1000, 3200,2300, 3020,2030, 3002,2003	2000, 3100,1300, 3010,1030, 3001,1003	9000, 4200, 4020, 4002	4900, 4090, 4009
Polarity	**3**	**2**	**1**	**-1**	**-2**	**-3**

Table 7 is not presented in its entirety. It only illustrates the most often employed combinations. For example, the second column can be completed with other combinations, for example a weak positive word can be followed by a word increasing polarity intensity (1400, 1040 and 1004).

5.6 Implementation of the Opinion Classification Method

The presented design of the method of opinion classification has been implemented as well. The implementation within OCS (Opinion Classification System) was used to experiment with the designed method. The OCS is a server application with two interfaces – one interface for “guest” users and another one for “admin” users.

This is why the system architecture consists of two modules: guest module and admin module. The guest module includes the following competencies:

- initialization of opinion classification of a selected text,
- setting value of the dynamic coefficient.

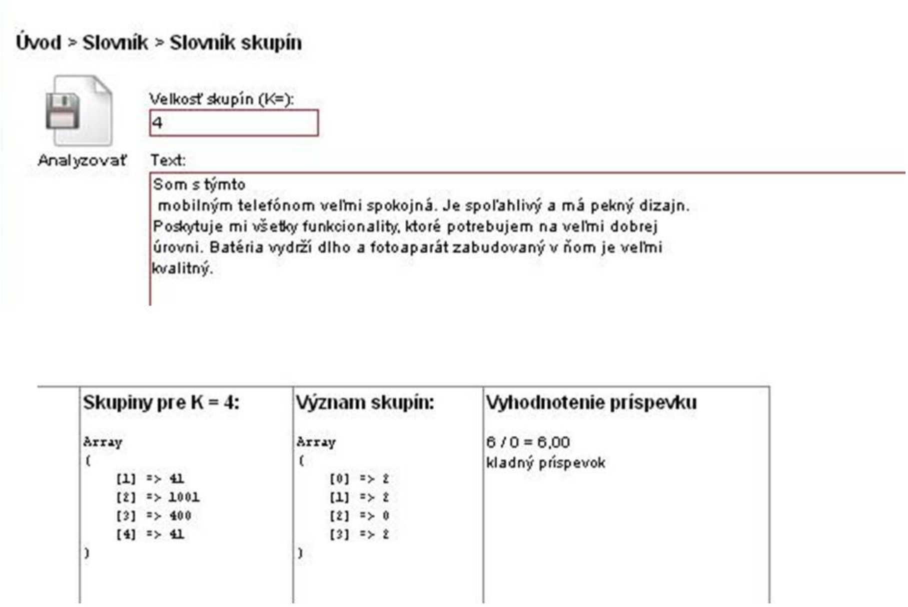
Within the modified version of the system OCS, the guest module has another competency – selecting one of three automatic ways of dynamic coefficient setting.

The admin module includes the following competencies:

- initialization of opinion classification of a selected text
- setting value of the dynamic coefficient.
- selecting one of three automatic ways of dynamic coefficient setting
- creation of the classification vocabulary

- editing of the classification vocabulary. When the OCS system detects a new word within the processed text, it offers administrator the possibility to insert this new word into the classification vocabulary. The administrator can decide whether to insert this unknown word into the vocabulary or not.

This implementation has been realized in the programming language PHP. More information about this implementation can be found in [19]. Fig. 8 presents the result of an analysis of a more complicated contribution, consisting of more than one sentence. This contribution is classified into positive opinion with final degree of positivity equal to 6.



Velkosť skupín (K=): 4

Text:

Som s týmto
 mobilným telefónom veľmi spokojná. Je spoľahlivý a má pekný dizajn.
Poskytuje mi všetky funkcionality, ktoré potrebujem na veľmi dobrej
úrovni. Batéria vydrží dlho a fotoaparát zabudovaný v ňom je veľmi
kvalitný.

Skupiny pre K = 4:	Význam skupín:	Vyhodnotenie príspevku
Array ([1] => 41 [2] => 1001 [3] => 400 [4] => 41)	Array ([0] => 2 [1] => 2 [2] => 0 [3] => 2)	6 / 0 = 6,00 kladný príspevok

Fig. 8. The example of a result of the OCS text analysis

The implementation was tested on the set of discussion contributions from the portal http://www.mobilmania.sk. This portal focuses on mobile telephones evaluation. Our tests were focused on the discussion thread related to reviews of the mobile telephone LGKU990. The set of contributions used for testing purposes contained 1558 words and 236 lexical units (combinations). The structure of the classification vocabulary was the following: 27 positive words, 27 negative words, 10 negations and 11 words, which increased the intensity of polarity. The evaluation was based on the comparison of results achieved by the OCS system and results obtained from an expert. The expert provided logical analysis of contributions taking into account the structure and meaning of particular sentences. The resulting precision of the implementation OCS according to introduced tests was 78,2%, which is arithmetical average of precision of OCS on positive contributions (86,2%) and on negative contributions (69,2%), what can be seen in Table 8.

We can see in the table, that the OCS implementation classified some neutral or even negative (positive) contribution to the positive (negative) opinion category. There are 4 mistakes in the classification of 29 contributions as positive opinions. For example, the sentence ‘Also my old Sony Ericsson makes *better* photos’ was classified to positive opinion category because of the positive word ‘better’ and lack of ability of OCS to identify hidden irony of this sentence.

Table 8. Results of experiments with the implementation OCS

	OCS result	Expert result	Precision
positive	29	25	0,862
negative	26	18	0,692

The opinion classification is sometimes very complicated not only due to the irony. Complications can arise from indirectly expressed opinion as well. For example, let us consider the sentence ‘I would not buy other brand’. It contains only neutral words and negation without positive or negative word, which this negation is related to. Therefore, the OCS classified this sentence to the neutral opinion class.

The modification of the opinion classification application with automatic determination of dynamic coefficient, which is illustrated in Fig.7, was tested on discussions from the page http://recenzie.sme.sk. These reviews were classified by an expert into two categories with positive and negative opinion. Consequently, they were classified by our application and results of this classification were compared with results obtained from the expert. We worked with a sample of 50 reviews. The number of really positive reviews was the same as the number of negative reviews (25).

The first method used for the dynamic coefficient setting was *“average length of all sentences”*. It achieved precision 0.76 (76%) for positive reviews and 0.84 (84%) for negative reviews.

The second used way of the dynamic coefficient setting *“half length of each sentence”* achieved precision 0.8 (80%) for positive reviews and 0.88 (88%) for negative reviews.

The last used possibility for the dynamic coefficient setting *“hybrid approach”* achieved precision 0.8 (80%) for positive reviews and 0.84 (84%) for negative reviews.

6 Related Works

Sometimes, the introduced opinion analysis is denoted as opinion mining, because it focuses on the extraction of positive or negative attitude of a participant to commented objects with the aid of mining techniques applied to text documents. Opinion mining can be extended from the level of whole texts perception to the level of extraction of properties of those objects which match users’ interests [7]. Parallel approach to opinion mining is sentiment analysis [22]. Deeper view on sentiment analysis, which is presented in [13], focuses on feature selection. Different access to web discussion processing is represented by the estimation of authority degree of some information sources, for example of actors contributing to discussion forums or social nets. An important technique for authoritative actors searching is visualization approach, which is introduced in [10]. Some effort was spent on semantically enriching algorithms for analysis of web discussion contributions by authors of [16]. Also dedicated information can be used as an interface to newsgroup discussions [17].

Nowadays, opinion analysis has become an important part of social networks analysis. Existing opinion analysis systems use large vocabularies for opinion classification into positive or negative answer categories. Such approach was used in [5]. Authors studied accuracy of the opinion analysis of Spanish documents originated in the field of economic. This approach uses a regression model for classification into negative or positive opinions. Authors studied how quality depends on the granularity of opinions and rules, which were used in the regression model. Another study [12] was focused on the possibility of using lesser granularity without any significant precision decrease. The results of this study show no substantial difference between one and two parameter regression models as well as no statistically significant difference between models with different granularity. Thus, for example, simpler models can be used with the used sentiment scale reduced to five degrees only.

The approach, presented in this paper, uses a scale with five degrees for opinion classification as well, but it differs from the previous approaches in vocabulary cardinality. Our work focuses on creating vocabularies with strong orientation on the discussion domain, not so large but created directly from live discussions. We do not use regression models. First, words from discussions are classified into predefined categories and after that, this classification is transformed into another one enabling classification of the whole contribution into one of five degrees (strong negative, negative, neutral, positive and strong positive).

There are some approaches similar to our approach to opinion classification [25], [26] and [6]. The originality of our approach in comparison with these approaches is mainly in the different technique of negation and intensification processing using the dynamic coefficient.

The most similar to our approach is Taboada at al: "Lexicon-Based Methods for Sentiment Analysis" in [25]. They also use a dictionary of words annotated with their orientation (polarity) as we are. Their approach splits this dictionary into more sub dictionaries according to word classes (adjectives, nouns, verbs and adverbs), and these dictionaries are checked for consistency and reliability. On the other hand, we use only one dictionary with all word classes and this dictionary can be created directly from the web discussion, which is analyzed in the phase of pre-processing, which increases the precision of opinion analysis. According to [25], more words in the dictionary can lead to noise increase and subsequently to precision decrease. In [25], intensification is provided by increasing (respectively decreasing) the semantic intensity of neighbouring lexical items using a special dictionary of intensifiers. Our approach is more general and independent on specialized sub-dictionary. An intensifier and the related word need not to be neighbours. They can take any position within one lexical unit, while their distance is limited by the given dynamic coefficient. The intensifier can be before or after the related word. Within our approach, not only intensification, but also negation processing is different, based on processing various combinations of words (lexical units) defined with the aid of dynamic coefficient and sequential interpretation into six categories (-3, -2, -1, 1, 2, 3), three for positive final polarity and three for negative final polarity (strong positive + intensifier, strong positive or gentle positive + intensifier, gentle positive, gentle negative, strong negative or gentle negative + intensifier, strong negative + intensifier). Our modified application achieves better results (higher precision). On the other hand, our tests were not so complex as in [25].

Another approach presented in Thelwall at al: "Sentiment strength detection in short informal text" [26] is focused on the SentiStrength detection algorithm, which solves some problems connected with sentiment analysis (generation of the sentiment strength list, optimisation of the sentiment word strengths, allocation of the miss words, spelling correction, creation of the booster word list, the negating word list and emoticon list, repeated letters processing and ignoring of a negative emotion in questions). Their approach is based on using machine learning techniques but our approach is lexicon based and is more universal. They devote more effort to correction of non standard spelling in informal texts. Our approach is not aiming to correct spelling of contributions since the dictionary can easily accommodate misspelled words as well. Algorithm described in [26] was tested on data from MySpace. We have provided our tests on data from narrow domains of newspaper contributions (http://recenzie.sme.sk/) and discussion contributions (http://www.mobilmania.sk) and therefore our results were a little bit better.

Paper "Learning with compositional semantics as structural inference for substantial sentiment analysis" [6] written by Choi and Cardie is focused on sentiment analysis similarly as our approach. It presents a novel learning based approach that incorporates inference rules inspired by compositional semantics into the learning procedure. Our approach differs from their work because our method is dictionary based and not machine learning oriented and so we use simple bag-of-word approach. But we also integrate compositional semantics using dynamic coefficient K. Our method incorporates surrounding of processed word up to distance K (maximally K neighboring words from the given word). Design of the method in [6] represents meaning of a composed expression as a function of the meanings of its parts within the compositional semantics. In our approach, these parts are lexical units, lengths and number of which are defined by the dynamic coefficient. The approach presented in [6] processes negations and intensification separately. This processing is made over the whole text with the aid of "voting-based inference". In our approach, the negation and intensifications are processed by the same mechanism of using dynamic coefficient. This processing is made separately in each lexical unit, not as the majority vote. Our approach does not use any methods of natural language processing and so it seems to be simple. It is based on some statistical principles and so it better processes longer texts, which contain more than one sentence. The semantics of such texts is partially incorporated using the mentioned dynamic coefficient. This method is universal enough for application in any case, but short texts (one sentence texts) can be analysed with lower precision.

7 Conclusions

The automatic opinion classification definitely belongs to up-to-day research agenda. There is a great potential of using the opinion classification within web discussion portals as a service not only for ordinary users (consumers) but for business-entities or organizations (Internet marketing) as well. The application of opinion classification can offer help supporting common users in decision making. Similarly, it can offer some services to business-entities and organizations (e.g. political parties, subjects of civil services, printed and electronic media, marketing agencies, etc.), for example the prediction of the development of society feelings or measuring degree of freedom of media. From this point of view, it is very promising research field.

The paper is an extended version of the paper from the ICCCI 2011 conference (Opinion Analysis from the Social Web Contributions). The modification of the method of opinion analysis presented here includes three various ways of automatic dynamic coefficient determination, which are described within section 5.4 “Dynamic Coefficient determination”. There are many other extensions of the presented paper in comparison with paper from the ICCCI 2011, mainly the new section 2 “Data extraction from the Social Web”.

The problem of extracting texts of contributions or comments from the social web was also described in this paper. The extraction algorithm for target data – contribution text particularly from discussion forums and web pages with commentaries - was introduced. The implementation of this algorithm was tested. The resulting averaged recall of these tests on various testing domains was from the interval <0.722, 1>. The cleaning of retrieved texts was completely successful within some domains. It was found, that data extraction is sensitive on parameters settings.

The paper describes also our design of an opinion classification algorithm. The implementation of this algorithm has achieved average precision 0.782 (78,2%). It can be perceived as a relatively good result considering the beginning stage of development. During next research stage, this implementation should be improved in order to perform deeper analysis of the given text and to provide more precise opinion classification. Higher precision can be achieved by means of irony and ambiguity detection. Also, it would be appropriate to test the improved implementation within the more extensive testing environment setting.

The paper has introduced three various ways, how to determine the dynamic coefficient. These approaches have been implemented and tested. The average precision obtained for positive and negative reviews was 80% for the method of Average length of all sentence, 84% for the method of Half length of each sentence and 82% for the Hybrid method.

For the purposes of some existing applications it is not needed to achieve the 100 percentage precision. An important factor is also the language which is used to present the classified opinions. Although languages have some similar features, the complexity of the task is evaluated on the basis of the emergent expression of the given particular language.

Research in the field of opinion classification has big importance for the future. A successful application of opinion classification can be very helpful in the process of decision making. This process can help both customers with decision about purchase as well as producers with information acquisition for new strategies creation.

Acknowledgements. The work presented in this paper was supported by the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic within the 1/1147/12 project " Methods for analysis of collaborative processes mediated by information systems".

References

1. Antoniu, G., Van Harmelen, F.: A Semantic Web Primer. In: Massachusetts Institute of Technology, USA, p. 238 (2004) ISBN 0-262-01210-3
2. Barla, M., Bieliková, M.: On Deriving Tagsonomies: Keyword Relations Coming from Crowd. In: Nguyen, N.T., Kowalczyk, R., Chen, S.-M. (eds.) ICCCI 2009. LNCS, vol. 5796, pp. 309–320. Springer, Heidelberg (2009)
3. Boyd, D., Ellison, N.B.: Social network sites: Definition, history and scholarship. Journal of Computer-Mediated Communication 13(1) (2007) ISSN 1083-6101, `http://jeme.indiana.edu/vol13/issue1/boyd.ellison.html` (accessed on October 27, 2011)
4. Cambria, E., Speer, R., Havasi, K., Hussain, A.: SentiNet: A Publicly Available Semantic Resource for Opinion Mining. Commonsense Knowledge. In: Proc. of the AAAI Fall Symposium
5. Catena, A., Alexandrov, M., Ponomareva, N.: Opinion Analysis of Publications on Economics with a Limited Vocabulary of Sentiments. International Journal on Social Media - MMM: Monitoring, Measurement, and Mining 1(1), 20–31 (2010)
6. Choi, Y., Cardie, C.: Learning with Compositional Semantics as Structural Inference for Subsentential Sentiment Analysis. In: Proc. of the EMNLP 2008, Conference on Empirical Methods in Natural Language Processing, pp. 793–801 (2008)
7. Ding, X., Liu, B., YuA, P.: Holistic Lexicon-Based Approach to Opinion Mining. In: Proc. of the Int. Conf. on Web Search and Web Data Mining, WSDM 2008, New York, NY, USA, pp. 231–240 (2008)
8. Ganapathibhotla, G., Liu, B.: Identifying Preferred Entities in Comparative Sentences. In: Proceedings of the International Conference on Computational Linguistics, COLING (2008)
9. Ghose, A., Ipeirotis, P.G.: Designing Novel Review Ranking Systems: Predicting Usefulness and Impact of Reviews. In: Proceedings of the International Conference on Electronic Commerce, ICEC (2007)
10. Heer, J., Boyd, D.: Vizster: Visualizing Online Social Networks. In: Proceedings of the IEEE Symposium on Information Visualization, INFOVIS 2005, Washington, USA, pp. 5–13 (2005)
11. Jindal, N., Liu, B.: Opinion Spam and Analysis. In: Proceedings of the Conference on Web Search and Web Data Mining (WSDM), pp. 219–230 (2008)
12. Kaurova, O., Alexandrov, M., Ponomareva, N.: The Study of Sentiment Word Granularity for Opinion Analysis (a Comparison with Maite Taboada Works). International Journal on Social Media - MMM: Monitoring, Measurement, and Mining 1(1), 45–57 (2010)
13. Koncz, P., Paralič, J.: An Approach to Feature Selection for Sentiment Analysis. In: Proc. of the INES 2011 - 15th International Conference on Intelligent Engineering Systems, Poprad, pp. 357–362 (2011) ISBN 978-142448956-5

14. Liu, B.: Sentiment Analysis & Opinion Mining, http://www.cs.uic.edu/~liub/FBS/Sentiment-Analysis-tutorial-AAAI-2011.pdf
15. Liu, B.: Sentiment Analysis and Subjectivity, http://www.cs.uic.edu/~liub/FBS/NLP-handbook-sentiment-analysis.pdf
16. Lukáč, G., Butka, P., Mach, M.: Semantically-enhanced Extension of the Discussion Analysis Algorithm in SAKE. In: 6th International Symposium on Applied Machine Intelligence and Informatics, SAMI 2008, Herľany, Slovakia, pp. 241–246 (January 2008)
17. Mach, M., Lukáč, G.: A Dedicated Information Collection as an Interface to Newsgroup Discussions. In: 18th International Conference on Information and Intelligent Systems, IIS 2007, Varazdin, Croatia, September 12-14, pp. 163–169 (2007) ISBN 978-953-6071-30-2
18. Machová, K., Bednár, P., Mach, M.: Various Approaches to Web Information Processing. Computing and Informatics 26(3), 301–327 (2007) ISSN 1335-9150
19. Machová, K., Krajč, M.: Opinion Classification in Threaded Discussions on the Web. In: Proc. of the 10th Annual International Conference Znalosti 2011, Stará Lesná, pp. 136–147. FEI Technická univerzita Ostrava, Czech Republic (2011) (in press)
20. Machová, K., Penzéš, T.: Extraction of Web Discussion Texts for Opinion Analysis. In: IEEE 10th Jubilee International Symposium on Applied Machine Intelligence and Informatics, Herľany, January 26-28, pp. 31–35. Óbuda University, Budapest (2012) ISBN 978-1-4577-0195-5
21. Ohama, B., Tierney, B.: Opinion mining with SentiWordNet, pp. 1–21. IGI Global (2011)
22. Pang, B., Lee, L.: Opinion Mining and Sentiment Analysis. Foundation and Trends in Information Retrieval 2(1-2), 1–135 (2008)
23. Russell, M.: With sentiment analysis, context always matters, http://radar.oreilly.com/2011/03/sentiment-analysis-context.html
24. Szabó, P., Machová, K.: Various Approaches to the Opinion Classification Problems Solving. In: IEEE 10th Jubilee International Symposium on Applied Machine Intelligence and Informatics, 26, Herľany, January 26-28, pp. 59–62. Óbuda University, Budapest (2012) (in press) ISBN 978-1-4577-0195-5
25. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-Based Methods for Sentiment Analysis. Computational Linguistics 37(2), 267–307 (2011)
26. Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., Kappas, A.: Sentiment Strength Detection in Short Informal Text. Journal of the American Society for Information Science and Technology 61(12), 2544–2558 (2010)
27. Yanhong, Z., Bing, L.: Web data extraction based on partial tree Alignment. In: Proc. of the 14th International World Wide Web Conference, WWW 2005, Chiba, Japan, May 10-14, pp. 76–85 (2005)
28. Yared, P.: Why Sentiment Analysis is the Future of ad Optimization, http://venturebeat.com/2011/03/20/why-sentiment-analysis-is-the-future-of-ad-optimization/

Time and Personality Based Behaviors under Cognitive Approach to Control the Negotiation Process with Incomplete Information

Amine Chohra, Arash Bahrammirzaee, and Kurosh Madani

Images, Signals, and Intelligent Systems Laboratory (LISSI / EA 3956),
Paris-East University (UPEC), Senart Institute of Technology,
Avenue Pierre Point, 77127 Lieusaint, France
{chohra,bahrammirzaee,madani}@u-pec.fr

Abstract. Finding the adequate negotiation strategy with incomplete information, even in one to one negotiation, is a complex problem. Inspired from research works aiming to analyze human behavior and those on social negotiation psychology, the integration of personality aspects, with the essential time parameter, is becoming necessary. For this purpose, first, one to one bargaining process, in which a buyer agent and a seller agent negotiate over single issue (price), is developed, where the basic behaviors based on time and personality aspects (conciliatory, neutral, and aggressive) are suggested. Second, a cognitive approach, based on the five-factor model in personality, is suggested to control the resulting time-personality behaviors with incomplete information. In fact, the five factors are the extraversion, the agreeableness, the conscientiousness, the neuroticism, and the openness to experience. Afterwards, experimental environments and measures, allowing a set of experiments are detailed. Results, concerning time-personality behaviors, demonstrate that more increasing conciliatory aspects lead to increased agreement point (price) and decreased agreement time, and more increasing aggressive aspects lead to decreased agreement point and increased agreement time. Finally, from a study case, of three different personalities corresponding to three different cognitive orientations, experimental results illustrate the promising way of the suggested cognitive approach in the control of the time-personality behaviors.

Keywords: Decision-making, incomplete information, negotiation, five-factor model in personality.

1 Introduction

This paper deals with social and cognitive negotiation behaviors for autonomous agents with incomplete information in order to find the adequate negotiation strategy in one to one negotiation which is a complex problem. Inspired from research works aiming to analyze human behavior and those on social negotiation psychology, the integration of psychological aspects of the agent personality, with the essential time parameter, is becoming necessary.

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 69–88, 2013.

The aim of this paper, in a first part, is to analyze the psychological personality impacts (effects) on the negotiation particularly with regard to agreement point and agreement time. In effect, an important aspect of the analysis of the suggested negotiation model is to assess the variation consequences of different psychological agent characters (conciliatory, neutral, and aggressive) on the decisions that agents make.

In a second part, the aim of this paper is to suggest a cognitive approach, based on the five-factor model in personality, where the negotiation cognition is considered as mental orientation of the negotiator towards different cognitive orientations: Win-Lose orientation, Win-Win orientation, Lose-Win orientation, or No-Orientation.

Thus, in this paper, after related works presented in Section 2, a one to one bargaining process, in which a buyer agent and a seller agent negotiate over a single issue (price), is suggested in Section 3, where the negotiation behaviors are based on the time and personality aspects. Then, a cognitive approach based on the five-factor model in personality is suggested in Section 4 in order to control the resulting time-personality behaviors. Afterwards, experimental environments and measures, allowing a set of experiments, are detailed in Section 5. In the first part of Section 6, experimental results of time-personality dependent behaviors are analyzed with regard to time dependent behaviors for different time deadlines. In the second part, the experimental results of the cognitive approach are given and analyzed from a study case of three different personalities corresponding to different cognitive orientations.

2 Related Works

Interesting surveys on negotiation models in the Artificial Intelligence field are given in [1], [2], [3]. Elsewhere, Lomuscio *et al.* [4] identified the main parameters (cardinality of the negotiation, agent characteristics, environments and goods characteristics, event parameters, information parameters, and allocation parameters) on which any automated negotiation depends and provided a classification scheme for negotiation models. Instead of focusing on analyzing the strategy equilibrium and historical information as in game theory, Artificial Intelligence researchers are interested in designing adaptive negotiation agents, with incomplete information, to environment changes [5]. Agents have incomplete and uncertain information about each other, and each agent's information (e.g., deadline, utility function, strategy, …) is its private knowledge.

An important research work has been developed by Faratin *et al.* [6] which devised a negotiation model that defines a range of strategies and behaviors for generating proposals based on time, resource, and behaviors of negotiators. By another way, in the research works developed aiming to analyze and describe human behavior in [7], twelve categories representing three major behavior parts have been defined: positive socio-emotional part, a neutral task part, and negative socio-emotional part. In another side, in research works on the social negotiation psychology of Rubin and Brown developed in [8], the interpersonal orientation of a person has an influence on his negotiating behavior. It is predominantly concerned with the degree of a person's responsiveness. Responsive people are more co-operative and therefore expect positive results. Personality type should therefore be determined first to obtain the best results in negotiation. Thus, negotiation behaviors, in which characters such as conciliatory, neutral, or aggressive define a ‘psychological’ personality aspect of a negotiator, play an important role in real negotiation.

Negotiations have received wide attention from distributed Artificial Intelligence community [9] and in general, any negotiation settings will have four different components [10]:

1) a negotiation set, space of possible proposals that agents can make ;

2) a protocol, legal proposals that agents can make ;

3) a collection of strategies, one for each agent, which determines what proposals agents will make ;

4) an agreement rule that determines reach agreement stopping negotiation.

Negotiation usually proceeds in a series of rounds, with every agent making a proposal at every round. A particular source of complexity is the agent number involved, and way in which these agents interact [10]: one to one, many to one (can be treated as concurrent one to one negotiations), and many to many (hard to handle).

By another way, the effects of personality factors in negotiation have been widely investigated by different researchers [8], [11], [12], [13]. After decades, there is a global consensus on five-factor model in personality to be the most comprehensive, empirical, data-driven research findings in personality psychology [11].

First articulated in the pioneering studies of Fiske (1949) [14], Tupes and Christal (1961) [15], and Norman (1963) [16], the five-factor model has become an increasingly influential framework during the last decades for organizing and understanding the universe of personality traits. In fact, this model is composed of five factors corresponding to five broad dimensions of the personality which are used to describe human personality. These factors were gradually discovered over three or four decades of research, and defined by several independent sets of researchers. These factors are: Neuroticism, Extraversion, Openness to experience, Agreeableness, and Conscientiousness.

3 One to One Negotiation

In this Section, one to one bargaining process shown in Fig. 1, in which buyer and seller agents negotiate, over a single issue (price), is developed.

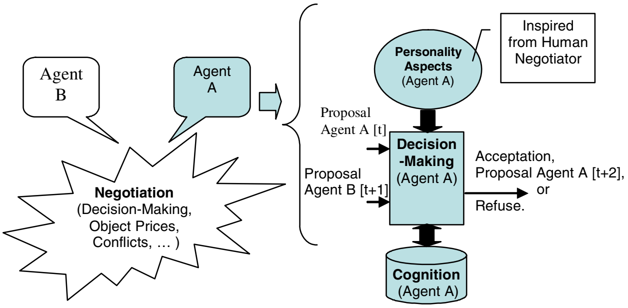

Agent B
Agent A
Negotiation (Decision-Making, Object Prices, Conflicts, ...)
Personality Aspects (Agent A)
Inspired from Human Negotiator
Proposal Agent A [t]
Proposal Agent B [t+1]
Decision -Making (Agent A)
Acceptation, Proposal Agent A [t+2], or Refuse.
Cognition (Agent A)

Fig. 1. Bilateral negotiation implying an autonomous agent

3.1 Negotiation Set

A negotiation set is the space of possible proposals that agents can make. The negotiation set (objects): the range of issues over which an agreement must be reached. Let i represents the negotiating agents, in bargaining bilateral negotiation $i \in \{buyer(b), seller(s)\}$, and j the issues under negotiation, in single issue negotiation j = *price*. The value for issue *price* acceptable by each agent i is $x^i \in [min^i, max^i]$.

3.2 Negotiation Protocol

A protocol is the legal proposals that agents can make. The process of negotiation can be represented by rounds, where each round consists of an offer from agent b (buyer) at time t_1 and a counter-offer from an agent s (seller) at time t_2. Then, a negotiation consists in a sequence of rounds: round1 (t_1, t_2), round2 (t_3, t_4), … Thus, for a negotiation between agents b and s, and if agent b starts first, then it should offer in times (t_1, t_3, t_5, …, t^b_{max}), and agent s provides counter-offers in (t_2, t_4, t_6, …, t^s_{max}), where t^b_{max} and t^s_{max} denote negotiation deadline for agents b and s, respectively.

Note that the three different deadline cases are allowed:

1) $t^b_{max} > t^s_{max}$, where considered deadline is $T_{max} = t^s_{max}$;
2) $t^b_{max} = t^s_{max}$, where considered deadline is $T_{max} = t^b_{max} = t^s_{max}$;
3) $t^b_{max} < t^s_{max}$, where considered deadline is $T_{max} = t^b_{max}$.

For agent b, the proposal to offer or accept is within interval [min^b, max^b], where max^b is the buyer reservation price in negotiation thread, and min^b is the lower bound of a valid offer. Similarly, for agent s, the proposal to offer or accept is within interval [min^s, max^s], where min^s is the seller reservation price and max^s is the upper bound of a valid offer. Initially a negotiator offers most favorable value for himself: agent b starts with min^b and agent s starts with max^s. If proposal is not accepted, a negotiator concedes with time proceeding and moves toward other end of the interval.

3.3 Negotiation Behaviors

The paces of concession depend on the negotiation behaviors of agent b and agent s which are characterized by negotiation decision functions. For negotiation strategies, time t is one of predominant factors used to decide which value to offer next.

Time Dependent Behaviors: Time dependent functions are used as negotiation decision functions varying the acceptance value (price) for the offer depending on the remaining negotiation time (an important requirement in negotiation) [6], i.e., depending on t and t^b_{max} for agent b and depending on t and t^s_{max} for agent s. Thus, proposal $x^b[t]$ to be offered by agent b and the one $x^s[t]$ to be offered by agent s at time t, with $0 <= t <= t^i_{max}$ belonging to [0, T - 1], are as follows. The proposal $x^s[t]$ to be offered by agent s at time t, with $0 <= t <= t^s_{max}$ belonging to [0, T - 1], is defined by Eq. (1).

$$x^s[t] = min^s + (1 - \alpha^s(t))\,(max^s - min^s), \tag{1}$$

where $\alpha^s(t)$ are time-dependent functions ensuring that: $0 <= \alpha^s(t) <= 1$,

$\alpha^s(0) = K^s$ (positive constant) and $\alpha^s(\, t^s_{max}\,) = 1$.

Such $\alpha^s(t)$ functions can be defined in a wide range according to the way in which $\alpha^s(t)$ is computed (the way they model the concession), e.g., polynomial in Eq. (2).

$$\alpha^s(t) = K^s + (1 - K^s)\left(\frac{min(t, t^s_{max})}{t^s_{max}}\right)^{\frac{1}{\beta}}. \tag{2}$$

Indeed, the constant β > 0 determines the concession pace along time, or convexity degree of the offer curve as a function of the time. By varying β a wide range of negotiation behaviors can be characterized: Boulware (B) with β < 1 and Conceder (C) with β > 1 [3], and the particular case of Linear (L) with β = 1.

Social and Cognitive Behaviors: The proposal $x^b[t]$ to be offered by agent b at time t, with $0<= t <= t^b_{max}$ belonging to [0, T - 1], is defined using behaviors based on time and personality aspects of a negotiator agent detailed in Sect. 3. 4.

3.4 Negotiation Strategies

A collection of strategies, one for each agent, which the role is to determine what proposals agents will make (which behavior should be used at any one instant).

Time Dependent: During a negotiation *thread* (the sequence of rounds with offers and counter-offers in a two-party negotiation), a negotiation strategy based on time dependent behaviors defined in [5] consists to define the way in which such behaviors are used. In this paper, each strategy uses individually the behaviors Boulware (B), Linear (L), or Conceder (C) during a negotiation thread.

Parameter β ranges [17], [18] are defined as: $\beta1 \in [20.00, 40.00]$ for Conceder (C), $\beta2 = 1.00$ for Linear (L), $\beta3 \in [0.01, 0.20]$ for Boulware (B). Then, constants K^i are chosen as small positive $K^i = 0.1$, for s, in order to not constrain the behavior of each time dependent function.

Time-Personality Dependent: These strategical behaviors integrate time and personality aspects and it is expected from such strategy the following hypothesis:

Hypothesis. The suggested strategy is expected to integrate time and personality aspects such that more increasing Conciliatory character leads to increasing agreement point and decreasing agreement time; and more increasing Aggressive character leads to decreasing agreement point and increasing agreement time.

Thus, such strategy is detailed from buyer point of view, where seller offers first.

Step 1 (Computing First Offers). The agent proposal is obtained in Eq. (3) from Conciliatory (Con) part Eq. (4), Neutral (Neu) part Eq. (5), and Aggressive (Agg) part Eq. (6).

$$x_j^b[t] = (W_{Con}^b[t] * xCon_j^b[t]) + (W_{Neu}^b[t] * xNeu_j^b[t]) + (W_{Agg}^b[t] * xAgg_j^b[t]) \,, \tag{3}$$

where the sum of weights is $W_{Con}^b[t] + W_{Neu}^b[t] + W_{Agg}^b[t] = 1$,

$$xCon_j^b[t] = \min{}_j^b + \alpha Con_j^b[t](\max{}_j^b - \min{}_j^b) \,, \tag{4}$$

$$\text{where } \alpha Con_j^b[t] = k_j^b + (1 - k_j^b)\left(\frac{\min(t, t_{\max}^b)}{t_{\max}^b}\right)^{\frac{1}{\beta 1}} ,$$

$$xNeu_j^b[t] = \min{}_j^b + \alpha Neu_j^b[t](\max{}_j^b - \min{}_j^b) \,, \tag{5}$$

$$\text{where } \alpha Neu_j^b[t] = k_j^b + (1 - k_j^b)\left(\frac{\min(t, t_{\max}^b)}{t_{\max}^b}\right)^{\frac{1}{\beta 2}} ,$$

$$xAgg_j^b[t] = \min{}_j^b + \alpha Agg_j^b[t](\max{}_j^b - \min{}_j^b) \,, \tag{6}$$

$$\text{where } \alpha Agg_j^b[t] = k_j^b + (1 - k_j^b)\left(\frac{\min(t, t_{\max}^b)}{t_{\max}^b}\right)^{\frac{1}{\beta 3}} .$$

Step 2 (Predicting β of the Seller). b predicts βs in Eq. (7) of s by Eq. (1) and (2).

Step 3 (Character to Change from Predicted β). According to the result of the predicted *β*s, the buyer changes the corresponding character such as in Eq. (8).

$$\beta s = \frac{Ln \frac{t-1}{t_{\max}^b}}{Ln \frac{\alpha_j^s[t] - k_j^b}{1 - k_j^b}} \quad \text{where } \alpha_j^s(t_i) = \frac{x_j^s[t-1] - \min_j^b}{x_j^s[0] - \min_j^b} \,. \tag{7}$$

If *βs > 1* Then Character Con Changes:

$$NewxCon_j^b[t] = \min{}_j^b + New\alpha Con_j^b[t](\max{}_j^b - \min{}_j^b) \,, \tag{8}$$

$$New\alpha Con_j^b[t] = k_j^b + (1 - k_j^b)\left(\frac{\min(t, t_{\max}^b)}{t_{\max}^b}\right)^{\frac{1}{\beta s}} .$$

If *βs = 1* Then Character Neu Changes:

$$NewxNeu_j^b[t] = \min_j^b + New\alpha Neu_j^b[t](\max_j^b - \min_j^b),$$

$$New\alpha Neu_j^b[t] = k_j^b + (1 - k_j^b)\left(\frac{\min(t, t_{\max}^b)}{t_{\max}^b}\right)^{\frac{1}{\beta s}}.$$

If *βs < 1* Then Character Agg Changes:

$$NewxAgg_j^b[t] = \min_j^b + New\alpha Agg_j^b[t](\max_j^b - \min_j^b),$$

$$New\alpha Agg_j^b[t] = k_j^b + (1 - k_j^b)\left(\frac{\min(t, t_{\max}^b)}{t_{\max}^b}\right)^{\frac{1}{\beta s}}.$$

Step 4 (Computing DeltaCharacter). According to the result of the character to change, the buyer computes the DeltaCharacter DC^b *as follows*:

If Character To Change is Con: $DC^b = \dfrac{NewxCon_j^b[t] - xCon_j^b[t]}{xCon_j^b[t]}$. (9)

If Character To Change is Neu: $DC^b = \dfrac{NewxNeu_j^b[t] - xNeu_j^b[t]}{xNeu_j^b[t]}$.

If Character To Change is Agg: $DC^b = \dfrac{NewxAgg_j^b[t] - xAgg_j^b[t]}{xAgg_j^b[t]}$.

Step 5 (Weight Updating). According character to change, buyer updates:

If Character To Change is Con: $W_{Con}^b[t] = W_{Con}^b[t-1] + DC^b[t]$, (10) $W_{Neu}^b[t] = W_{Neu}^b[t-1] - 0.3 * DC^b[t]$, $W_{Agg}^b[t] = W_{Agg}^b[t-1] - 0.7 * DC^b[t]$.

If Character To Change is Neu: $W_{Con}^b[t] = W_{Con}^b[t-1] - 0.5 * DC^b[t]$, $W_{Neu}^b[t] = W_{Neu}^b[t-1] + DC^b[t]$, $W_{Agg}^b[t] = W_{Agg}^b[t-1] - 0.5 * DC^b[t]$.

If Character To Change is Agg: $W_{Con}^b[t] = W_{Con}^b[t-1] - 0.7 * DC^b[t]$, $W_{Neu}^b[t] = W_{Neu}^b[t-1] - 0.3 * DC^b[t]$, $W_{Agg}^b[t] = W_{Agg}^b[t-1] + DC^b[t]$.

Step 6 (Computing the Proposal). According the character to change, b updates:

$$x_j^b[t] = (W_{Con}^b[t] * xCon_j^b[t]) + (W_{Neu}^b[t] * xNeu_j^b[t]) + (W_{Agg}^b[t] * xAgg_j^b[t]). \quad (11)$$

3.5 Agreement Rule

An agreement rule determines the reach agreements stopping negotiation. Agent b accepts an offer $x^s[t]$ from agent s at time t if it is not worse than the offer he would submit in next step, i.e., only if the relation given in Eq. (12) is satisfied. Similarly, s accepts an offer $x^b[t]$ from b at time t only if the relation given in Eq. (12) is satisfied.

$$\begin{cases} x^b(t+1) >= x^s(t) \\ t <= T_{max} \end{cases}, \begin{cases} x^s(t+1) <= x^b(t) \\ t <= T_{max} \end{cases}. \qquad (12)$$

4 Cognitive Approach Based on the Five-Factor Model in Personality

In this Section, the negotiation cognition is considered as a mental orientation of the negotiator towards Win-Lose orientation, Win-Win orientation, Lose-Win orientation, or No-Orientation. In the first orientation, agent has strong desire to win even with cost of opponent agent, while in the second orientation, the agent is trying to increase and maximize mutual utilities. In Lose-Win orientation, the agent sacrifies his own utility for some reasons like reputation, seeking trust, generosity, or to save time or resources. Such negotiation cognition can be deduced from the personality factors, using the five-factor model, of the negotiator in order to control the negotiation behaviors. In addition, it has been proven in [13] that negotiator cognitions mediate the effects of personality on negotiation behaviors and also the negotiation behaviors mediate the effects of negotiator cognitions on negotiation outcomes.

In this work, the negotiation cognition is exploited to determine, in each orientation case, the adequate weights of the linear combination of the time-personality behaviors (Conciliatory, Neutral, and Aggressive) used by each agent in order to affect the final outcome of negotiation. Thus, in the cognitive approach scheme illustrated in Fig. 2, the five-factors in personality influence the negotiator cognition, and consequently this negotiation cognition influences the negotiation process and outcomes.

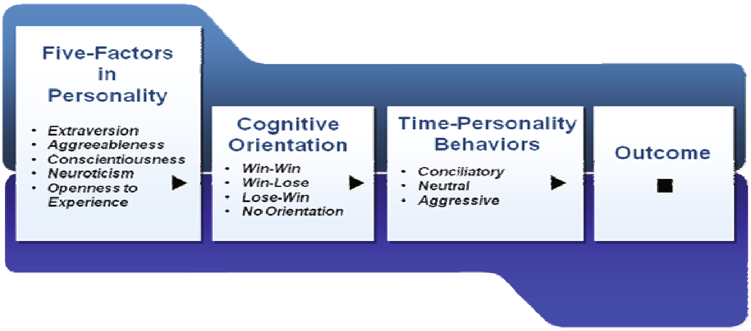


Fig. 2. Cognitive approach scheme

The five-factors in personality are individual characteristics: affective, experiential, and motivational, as well as interpersonal [12]:

- Extraversion: sociable, assertive, talkative, and active;
- Agreeableness: courteous, flexible, trusting, cooperative, and tolerant;
- Conscientiousness: careful, responsible, and organized;
- Neuroticism (emotional stability): anxious, depressed, worried, and insecure;
- Openness to experience: imaginative, curious, original, and broad-minded.

From the work developed in [13], it is concluded that Extraversion, Agreeableness, and Neuroticism are the three most important personality factors in the five-factor model that predict conflict styles.

The value of each personality factor (Agreeableness: ν^a_{Ag}, Extraversion: ν^a_{Ex}, Openness to experience: ν^a_{Oe}, Conscientiousness: ν^a_{Co}, and Neuroticism: ν^a_{Ne}), for agent a, is chosen from continuum 0 to 10. Then, based on their effects, the five personality factors are grouped, as shown in Fig. 3, in three different sets leading to different negotiation cognitions:

- s1: Extraversion and Agreeableness factors leading to Win-Win orientation or Lose-Win orientation,
- s2: Neuroticism factor leading to Win-Lose orientation,
- s3: Conscientiousness and Openness to experience factors leading to No-Orientation.

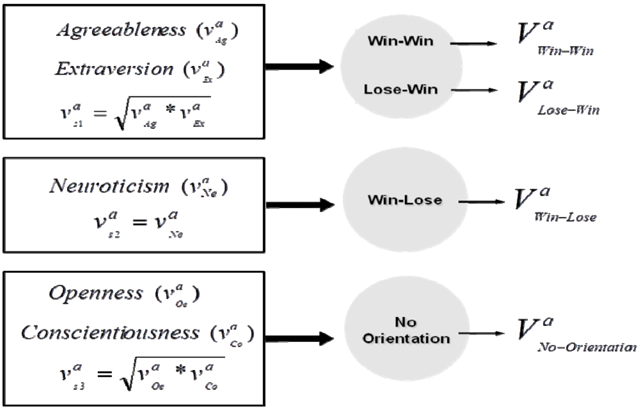


Fig. 3. Personality factors and cognitive orientations

For each set, the corresponding personality value, of a negotiator, is computed as follows:

$$\nu^a_{s1} = \sqrt{\nu^a_{Ag} * \nu^a_{Ex}} \,, \tag{13}$$

$$\nu^a_{s2} = \nu^a_{Ne} \,,$$

$$\nu^a_{s3} = \sqrt{\nu^a_{Oe} * \nu^a_{Co}} \,.$$

Then, using these values, a value is computed for each cognitive orientation as follows:

$$V^a_{Win-Win} = V^a_{Utility} * \frac{\nu^a_{s1}}{\sum_{i=1}^{i=3} \nu^a_{si}} \,, \tag{14}$$

$$V^a_{Lose-Win} = V^a_{Social} * V^a_{Personal} * \frac{\nu^a_{s1}}{\sum_{i=1}^{i=3} \nu^a_{si}} \,,$$

$$V^a_{Win\ -Lose} = \frac{\nu^a_{s2}}{\sum_{i=1}^{i=3} \nu^a_{si}} \,,$$

$$V^a_{No-Orientatio\ n} = \frac{\nu^a_{s3}}{\sum_{i=1}^{i=3} \nu^a_{si}} \,,$$

$$\text{with } V^a_{Utility} + V^a_{Social} + V^a_{Personal} = 1 \,,$$

where $V^a_{Utility}$ is the value attributed with regard to the intrinsic utility ; V^a_{Social}, is the value attributed with regard to social reasons (e.g., seeking for trust in market, seeking for mutual support, generosity or charity purposes); and $V^a_{Personal}$ is the value attributed with regard to personal reasons (e.g., saving negotiation time, saving negotiation resources).

Then, the maximal value (among the cognitive orientation values) will determine the corresponding cognitive orientation (e.g., if $V^a_{Win-Win}$, is the maximal value, then the cognitive orientation of the agent is Win-Win orientation).

In the next step, the updating of weights for the linear combination, see Eq. (11), of time-personality behaviors (Conciliatory, Neutral, and Aggressive) will be chosen based on the negotiation cognition of the agent. If the agent has Win-Win orientation, then the set of weights will be chosen in the way to maximize both utilities. If the negotiation cognition of agent is oriented towards Lose-Win orientation, then the set of weights will be chosen in the way to maximizes opponent's agent utility. If the negotiation cognition of agent is oriented towards Win-Lose, then the set of coefficients will be chosen in the way to maximize agent's own utility. Finally, if the negotiation cognition of agent is No-orientation, then no change will be suggested.

In this model, in each state different combination sets of updating weights will be tested using a simple positive reinforcing method and the utility of the mid-point defined as follows:

$$\mathrm{Um}_j^a = \frac{\max_j^a + \min_j^a}{2}. \tag{15}$$

In other words, in each round i and for each agent a, different combinations of $\mathrm{W}_{\mathrm{Con}}^a$, $\mathrm{W}_{\mathrm{Neu}}^a$, and $\mathrm{W}_{\mathrm{Agg}}^a$ will be randomly created Rn times, between 0 and 1 in such manner that their sum equal to 1, and chosen from the obtained sets in the following way for each orientation. During this process, a reward will be given to each updating weight, and finally the one with maximum amount of reward will be chosen as final updating weight.

For Win-Win Orientation:

$$\mathrm{Rw}_j^a(\mathrm{i})(\mathrm{y}) = \frac{1}{\left|\mathrm{U}_j^a(\mathrm{i}) - \mathrm{Um}_j^a\right|} \text{ with } \mathrm{y} \in \{1,\ldots,\mathrm{Rn}\}. \tag{16}$$

Note that in incomplete information state, when each agent doesn't know anything about the opponent's reservation interval, one way is then to use the utility of mid-point (Um_j^a) equal to 0.5, and therefore substitute the utility of mid-point of the buyer $\mathrm{Um}_j^b(\mathrm{i})$ by 0.5.

Because the aim is to have Win-Win negotiation, where the amount of utilities of both agents are maximized as much as possible, the above mentioned equation ensures that the set of updating weights which distributes closest utilities to that of mid-point will be chosen. To do so, the reward function gives more rewards to set of weights which produces the utilities closer to mid-point utility.

For Lose-Win Orientation:

$$Rw_j^a(i)(y) = Um_j^a - U_j^a(i) \text{ with } \mathrm{y} \in \{1,\ldots,\mathrm{Rn}\}. \tag{17}$$

Because the aim is to have Lose-Win negotiation, where the amount of utility of opponent agent has to be maximized as much as possible, the above mentioned equation ensures that the set of updating weights which distributes the maximum possible utility to opponent agent will be chosen. To do so, the reward function gives more rewards to the set of weights which produces less utility to the agent and consequently more utility to the opponent agent.

For Win-Lose Orientation:

$$Rw_j^a(i)(y) = U_j^a(i) - Um_j^a \text{ with } y \in \{1,...,Rn\}. \tag{18}$$

According to this function, more rewards will be attributed to more utility. In other words, if using first set of weights results to more utility than the other sets, then it will be chosen to be used.

5 Experiments: Environments and Measures

In this Section, experimental environments and measures are presented and a set of experiments, carried out for different deadlines of agents b and s, are detailed.

5.1 Experimental Environments

Environments are defined in bargaining bilateral negotiation between buyer(b) and seller(s), in single issue negotiation j = *price*. The experimental environment is defined by the following variables [t_{max}^{b} , t_{max}^{s} , T_{max}, K^b, K^s, min^b, max^b, min^s, max^s].

The negotiation interval (difference between minimum and maximum values of agents) for price is defined using: θ^i (length of the reservation interval for an agent i) and Φ (degree of intersection between the reservation intervals of the agents, ranging between 0 for full overlap and 0.99 for virtually no overlap). In the experimental environment: θ^i are randomly selected between the ranges [10, 30] for both agents, and Φ = 0. The negotiation intervals are then computed, setting min^b = 10, by:

$$min^b = 10,\ max^b = min^b + \theta^b,\ min^s = \theta^b \Phi + min^b, \text{ and } max^s = min^s + \theta^s. \tag{19}$$

The analysis and evaluation of negotiation behaviors and strategies developed in [19], indicated that negotiation deadlines significantly influence the negotiation performance. From this, the experimental environment is defined from random selection of the round number within [10, 50] which corresponds to a random selection of T_{max} within [20, 100]. Initiator of an offer is randomly chosen because the agent which opens the negotiation fairs better, irrespective of whether agent is b or s.

5.2 Experimental Measures

To produce statistically meaningful results the precise set of environments is sampled from parameters specified in Sect. 5.1 and environment number used is N = 200, in each experiment. This ensures that the probability of the sampled mean deviating by more than 0.01 from true mean is less than 0.05. In the following the used measures, in this work, are detailed.

Average Round Number (AR): rounds to reach an agreement (deal), lengthy negotiation incurs penalties for resource consumption, thus shrinking utilities obtained by negotiators indirectly [20]. Average round number AR is given in Eq. (20):

$$AR = \frac{\sum_{n=1}^{N} R_D[n]}{N_D} \tag{20}$$

, where R_D is the number of rounds, for each environment with deal, and N_D is the number of environments with deals.

Intrinsic Utility (U):

$$U_j^b(i) = \frac{\max_j^b - x_j^b(i)}{\max_j^b - \min_j^b} \text{ and } U_j^s(i) = \frac{x_j^s(i) - \min_j^s}{\max_j^s - \min_j^s} \tag{21}$$

Average Intrinsic Utility (AU):

$$AU_j^a(e) = \frac{\sum_{e=1}^{E_j} U_j^a(e)}{E_j}, \tag{22}$$

where E_j is the total number of environments with deals, and $U_j^a(i)$ the utility of each agent, for each environment with deal.

Utility Product (UP): once an agreement is achieved, the product, of the utilities obtained by both participants UP_j is computed. This measure indicates the joint outcome:

$$UP_j = U_j^b . U_j^s \text{ and } AUP_j = \frac{\sum_{e=1}^{E_j} \sqrt{UP_j(e)}}{E_j}. \tag{23}$$

Utility Difference (UD): once an agreement is achieved, the difference, of the utilities obtained by both participants UD_j is computed. This measure indicates the distance between both utilities:

$$UD_j = \left| U_j^b - U_j^s \right| \text{ and } AUD_j = \frac{\sum_{e=1}^{E_j} UD_j(e)}{E_j}. \tag{24}$$

Average Deal Number (AD): the average deal number (AD_j^a) is obtained as follows:

$$\mathrm{AD_j} = \frac{\mathrm{E_j}}{\mathrm{N_j}} \quad \text{with } 0 < \mathrm{AD_j} < 1 \tag{25}$$

where $\mathrm{E_j}$ is the number of environments with deals, and $\mathrm{N_j}$ is the total number of environments for issue j.

Average Performance (AP): the Average Performance ($\mathrm{AP_j^a}$) is an average evaluation measure implying the three experimental measures, i.e., the average intrinsic utility, the average time (round number), and the average deal number:

$$\mathrm{AP_j^a} = \frac{\mathrm{AU_j^a(e)} + (1 - \frac{\mathrm{At_j^a(e)}}{\mathrm{At_{max}^a(e)}}) + \mathrm{AD_j(e)}}{3} \quad \text{with } \mathrm{At_{max}^a(e)} = \frac{\sum_{e=1}^{E_j} \mathrm{t_{max}^a(e)}}{\mathrm{E_j}}, \tag{26}$$

where $\mathrm{E_j}$ is the number of environments with deals, and $\mathrm{At_{max}^a(e)}$ is average negotiation deadline for $\mathrm{E_j}$.

Final Performance (FP): the final performance measure ($\mathrm{FP_j^a}$) is an average evaluation measure implying the three experimental measures, i.e., the average performance, the average utility difference, and the average utility product:

$$\mathrm{FP_j^a} = \frac{\mathrm{AP_j^a} + (1 - \mathrm{AUD_j}) + \mathrm{AUP_j}}{3}. \tag{27}$$

6 Experimental Results

6.1 Time and Personality Experimental Results

In this Section, experimental results of the time-personality dependent behaviors are presented (varying curves), analyzed, and compared for different deadlines with regard to time dependent behaviors (constant curves) where both agents b and s use a Linear strategy.

The results presented in Fig. 4 and Fig. 5 concern the variation effects of the Conciliatory character of the buyer on negotiation behaviors. For both deadlines (short and long), results demonstrates that more Conciliatory character is increasing implies more agreement point is increasing while more agreement time is decreasing.

The results presented in Fig. 6 and Fig. 7 concern the variation effects of the Neutral character of the buyer (i.e., of the personality aspects) on the negotiation. For both deadlines (short and long), results demonstrates that more Neutral character is increasing implies more agreement point is decreasing while more agreement time is increasing (for long term deadline).

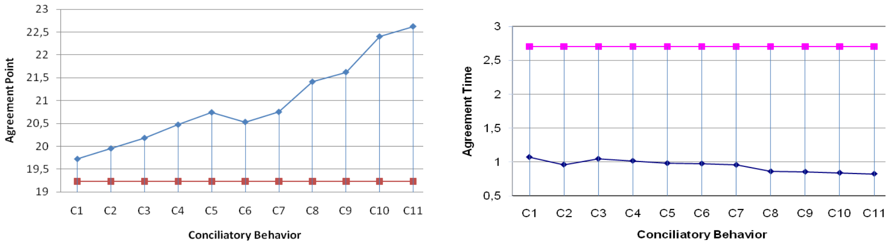


Fig. 4. Conciliatory behaviors (short term deadlines): agreement point and agreement time

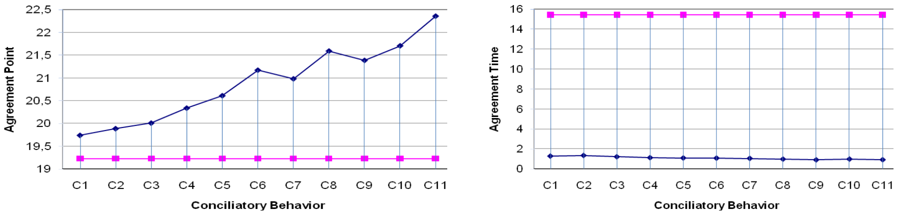


Fig. 5. Conciliatory behaviors (long term deadlines): agreement point and agreement time

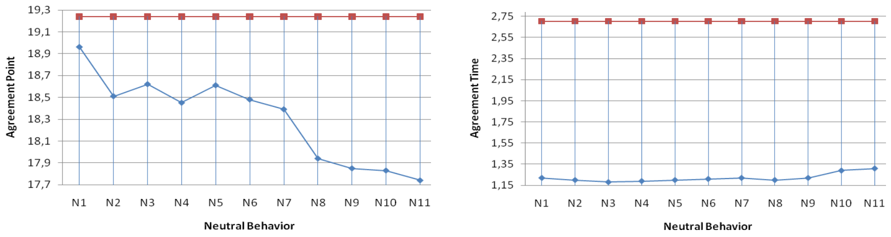


Fig. 6. Neutral behaviors (short term deadlines): agreement point and agreement time

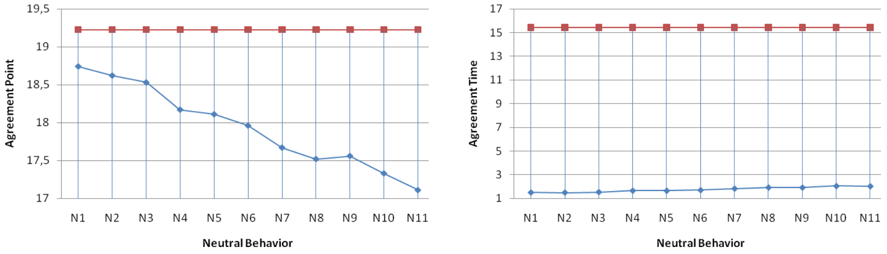


Fig. 7. Neutral behaviors (long term deadlines): agreement point and agreement time

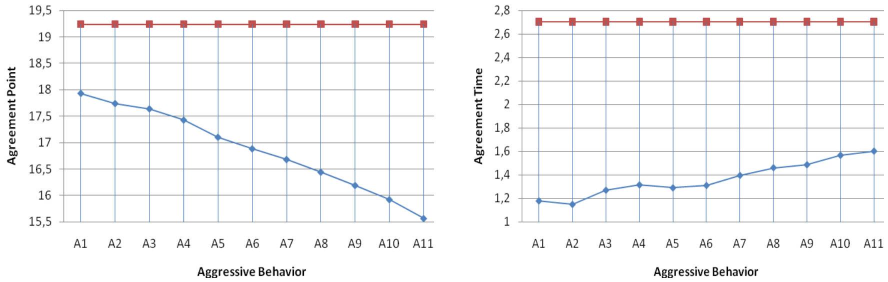


Fig. 8. Aggressive behaviors (short term deadlines): agreement point and agreement time

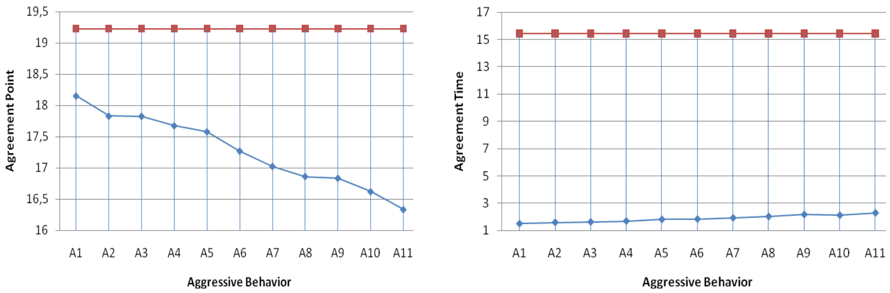


Fig. 9. Aggressive behaviors (long term deadlines): agreement point and agreement time

The results presented in Fig. 8 and Fig. 9 concern the variation effects of the Aggressive character of the buyer (i.e., of the personality aspects) on negotiation behaviors. For both deadlines (short and long terms), these results demonstrates that more Aggressive character is increasing implies more agreement point is decreasing while more agreement time is increasing.

6.2 Cognitive Approach Results

In order to validate both online and offline effects of personality on negotiation outcomes we investigate, in this section, the effect of personality factors on final negotiation outcomes. To do so, three different personality cases corresponding to three different negotiators, Personality 1 (P-1), Personality 2 (P-2), Personality 3 (P-3), are defined as shown in Table 1.

Table 1. Three personality cases.

Values **Cases**	v^{b}_{Ag}	v^{b}_{Ex}	v^{b}_{Oe}	v^{b}_{Co}	v^{b}_{Ne}	$V^{b}_{Utility}$	V^{b}_{Social}	$V^{b}_{Personal}$
P-1	2	2	3	5	8	0.50	0.25	0.25
P-2	9	7	3	5	2	0.75	0.15	0.10
P-3	9	7	3	5	2	0.25	0.30	0.45

According to the first personality case, the value of Neuroticism factor (8) is the predominant factor value (largely more than other ones). Therefore, the agent is expected to have Win-Lose orientation which leads to high intrinsic utility and average performance. Then:

Hypothesis 1: The buyer agent with first personality gets more intrinsic utility and average performance comparing to negotiators with other personality cases but gets minimum final performance.

According to the second personality case, the values of Agreeableness (9) and Extraversion (7) factors are the predominant factor values. In addition, this agent gives more value to his utility (0.75) comparing to social (0.15) and personal (0.10) reasons. Therefore, the agent is expected to have Win-Win orientation which leads to high utility product, less utility difference and therefore, high final performance. Then:

Hypothesis 2: The buyer agent with the second personality gets more utility product, less utility difference and therefore, more final performance comparing to negotiators with other personality cases.

According to the third personality case, the values of Agreeableness (9) and Extraversion (7) factors are the predominant factor values. In addition, this agent gives more value to his social (0.30) and personal (0.45) reasons, comparing to his utility (0.25). Therefore, the agent is expected to have Lose-Win orientation which leads to less intrinsic utility for buyer agent, but less time to reach an agreement. Then:

Hypothesis 3: The buyer agent with third personality gets minimum intrinsic utility but needs less time to agreement comparing to negotiators with other personality cases. In addition, seller gets maximum intrinsic utility in this case comparing to two other cases.

The results for all three sets of experiments are presented in Table 2 in terms of intrinsic utility, time to reach to agreement, utility product, utility difference, average performance, and final performance.

Table 2. Cognitive approach results for three personality cases

Measures	U_j^b	U_j^s	t_j	UP_j	UD_j	AP_j^b	FP_j^b
Cases							
P-1	0.4866	0.1800	24	0.0876	0.3066	0.5622	0.5171
P-2	0.3111	0.3555	20	0.1106	0.0443	0.5481	0.6121
P-3	0.2549	0.4117	19	0.1049	0.1567	0.5405	0.5692

The results presented in Table 2, show that the intrinsic utility of buyer agent in first personality case (0.4866) is, considerably, more than second and third cases (0.3111 and 0.2549). This high amount of intrinsic utility of first personality case, re-compensate the high amount of the time which he needs to reach to agreement (24) and, therefore, the average performance of first personality case (0.5622) is more than two other personality cases (0.5481 and 0.5405). However, he gets minimum final performance (0.5171) which is direct result of less utility product (0.0876) and more utility difference (0.3066) of first personality case, comparing to the other cases, supporting thus the Hypothesis 1.

According to Table 2, buyer agent with second personality case gets maximum utility product (0.1106), minimum utility difference (0.0443), and maximum final performance (0.6121), comparing to the other two cases, supporting thus the Hypothesis 2.

According to results presented in Table 2, the buyer agent with the third personality case has minimum intrinsic utility (0.2549) and average performance (0.5405), but needs less time to agree (19) comparing to the other personality cases. In addition, in this case, the intrinsic utility of seller (0.4117) is maximum comparing to the other cases, supporting thus the Hypothesis 3.

7 Conclusion

In this paper, first the time-personality dependent behaviors have been suggested for the negotiation process with *incomplete* information in one to one single issue (price) intending to find the adequate strategy. Results demonstrate, more increasing conciliatory aspects lead to increased agreement point (price) and decreased agreement time. On the other hand, more increasing aggressive aspects lead to decreased agreement point and increased agreement time.

Second, a cognitive approach is suggested, based on the five-factor model, where the negotiation cognition is considered as mental orientation of the negotiator towards different cognitive orientations: Win-Lose orientation, Win-Win orientation, Lose-Win orientation, or No-Orientation. From a study case, of three different personalities corresponding to three different cognitive orientations, experimental results illustrate the promising way (since the results illustrate a tendency towards the cognitive orientation deduced from the personality factors) of the suggested cognitive approach in the control of the time-personality behaviors. More, the important point in this suggested personality model is in its mediating effect on cognitive orientation. In other words, it affects the negotiation process and outcomes indirectly. Therefore, if, for example, the cognitive orientation of agent is Win-Win, it doesn't mean that the final output of negotiation will be also Win-Win. It means that based on his personality, the agent is more willing to have a tendency towards Win-Win results, since with incomplete information the final results will be attained also by the negotiation's environment and opponent's agent strategy.

Of course, such cognitive approach stills have challenging open questions with regard to the used personality model (five-factor model) which is not defined and stated completely according to current psychology science advances, and with regard to its modeling from the psychology science to the computer science.

Another important and necessary step is to integrate fuzzy reasoning to the suggested time-personality dependent behaviors [21]. Afterwards, learning from interaction, which is fundamental from embodied cognitive science and understanding natural intelligence perspectives [22], [23] for understanding human behaviors and developing new solution concepts [24], will be necessary in negotiation.

References

1. Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra, C., Wooldridge, M.: Automated negotiation: prospects, methods, and challenges. Int. J. of Group Decision and Negotiation 10(2), 199–215 (2001)
2. Gerding, E.H., van Bragt, D., Poutré, J.L.: Scientific Approaches and Techniques for Negotiation: A Game Theoretic and Artificial Intelligence Perspective. CWI, Technical Report, SEN-R0005 (2000)
3. Li, C., Giampapa, J., Sycara, K.: Bilateral negotiation decisions with uncertain dynamic outside options. IEEE Trans. on Systems, Man, and Cybernetics, Part C: Special Issue on Game-Theoretic Analysis and Stochastic Simulation of Negotiation Agents 36(1), 1–13 (2006)
4. Lomuscio, A.R., Wooldridge, M., Jennings, N.R.: A classification scheme for negotiation in electronic commerce. Int. J. of Group Decision. and Negotiation 12(1), 31–56 (2003)
5. Lin, R., Kraus, S., Wilkenfeld, J., Barry, J.: Negotiating with bounded rational agents in environments with incomplete information using an automated agent. Artificial Intelligence 172(6-7), 823–851 (2008)
6. Faratin, P., Sierra, C., Jennings, N.R.: Negotiation decision functions for autonomous agents. Int. J. of Robotics and Autonomous Systems 24(3-4), 159–182 (1998)
7. Bales, R.F.: Interaction Process Analysis: A Method for the Study of Small Groups. Addisson-Wesley, Cambridge (1950)
8. Rubin, J.Z., Brown, B.R.: The Social Psychology of Bargaining and Negotiation. Academic Press, New York (1975)
9. Rosenschein, J., Zlotkin, G.: Rules of Encounter. MIT Press, Cambridge (1994)
10. Wooldridge, M.: An Introduction to MultiAgent Systems. John Wiley & Sons, England (2002)
11. McAdams, D.P.: The five-factor model in personality: a critical appraisal. Journal of Personality 60(2), 328–361 (1992)
12. Barry, B., Friedman, R.A.: Bargainer characteristics in distributive and integrative negotiation. Journal of Personality and Social Psychology 74(2), 345–359 (1998)
13. Ma, Z.: All Negotiations are not Perceived Equal: The Impact of Culture and Personality on Cognitions, Behaviors, and Outcomes. PhD Report, Faculty of Management, McGill University, Montréal (2005)
14. Fiske, D.W.: Consistency of the factorial structures of personality ratings from different sources. Journal of Abnormal and Social Psychology 44, 329–344 (1949)
15. Tupes, E.C., Christal, R.E.: Recurrent Personality Factors on Trait Ratings. Technical Report Nos., pp. 61-67. Lackland TX: U. S. Air Force Aeronautical Systems Division (1961)
16. Norman, W.T.: Toward an adequate taxonomy of personality attributes: replicated factor structure in peer nomination personality ratings. Journal of Abnormal and Social Psychology 66, 574–583 (1963)
17. Pruitt, D.: Negotiation Behavior. Academic Press, London (1981)
18. Raiffa, H.: The Art and Science of Negotiation. Harvard University Press, Cambridge (1982)
19. Wang, K.-J., Chou, C.-H.: Evaluating NDF-based negotiation mechanism within an agent-based environment. Robotics and Autonomous Systems 43, 1–27 (2003)
20. Lee, C.-F., Chang, P.-L.: Evaluations of tactics for automated negotiations. Group Decision and Negotiation 17(6), 515–539 (2008)

21. Richter, J., Kowalczyk, R., Klusch, M.: Multistage fuzzy decision making in bilateral negotiation with finite termination times. In: Nicholson, A., Li, X. (eds.) AI 2009. LNCS (LNAI), vol. 5866, pp. 21–30. Springer, Heidelberg (2009)
22. Pfeifer, R., Scheier, C.: Understanding Intelligence. MIT Press, Cambridge (1999)
23. Chohra, A.: Embodied Cognitive Science, Intelligent Behavior Control, Machine Learning, Soft Computing, and FPGA Integration: Towards Fast, Cooperative and Adversarial Robot Team (RoboCup). Technical GMD Report, No. 136, Germany (June 2001) ISSN 1435-2702
24. Zeng, D., Sycara, K.: Benefits of learning in negotiation. In: Proc. of the 14th National Conference on Artificial Intelligence (AAAI 1997), Providence, RI, pp. 36–41 (July 1997)

Web Server Support for e-Customer Loyalty through QoS Differentiation

Grażyna Suchacka[1] and Leszek Borzemski[2]

[1] Institute of Mathematics and Informatics, Opole University,
Oleska 48, 45-052 Opole, Poland
[2] Institute of Informatics, Wrocław University of Technology,
Janiszewskiego 11/17, 50-370 Wrocław, Poland
gsuchacka@uni.opole.pl, leszek.borzemski@pwr.wroc.pl

Abstract. The paper deals with the problem of offering predictive service in e-commerce Web server systems under overload. Due to unpredictability of Web accesses, such systems often fail to effectively handle peak traffic, which results in long delays and incomplete transactions. As a consequence, online retailers miss an opportunity to attract new customers, retain the loyalty of regular customers, and increase profits. We propose a method for priority-based admission control and scheduling of requests at the Web server system in order to differentiate Quality of Service (QoS) with regard to user-perceived delays, i.e., Web page response times provided by the system (as opposed to HTTP request response times). To detect and cope with the system overload, a new kind of a load indicator is proposed, based on online measurements of page response times. Simulation results demonstrate that our solution is capable of providing key customers with limited delays while improving QoS for ordinary customers under heavy load.

Keywords: Web server, quality of service, QoS, scheduling, admission control, e-commerce, customer loyalty.

1 Introduction

As a result of the rapid development in computer networking technologies in recent years, support for distributed computer systems in different areas has become the focus of intensive research. Much work has addressed the issue of Quality of Service (QoS) in distributed real-time computer systems running in unpredictable environments; in particular, the problem of guaranteeing predictive delays in the face of highly variable number of tasks has been studied.

This problem is clearly visible in the case of World Wide Web (WWW), the biggest distributed computer system ever. WWW is a system of interlinked Web pages accessed via the Internet; it has to face a potentially unlimited population of users, who send their requests for Web content. The "best-effort" paradigm of request processing on the Internet does not allow for request differentiation nor quality guarantees. For this reason, the global network often fails to satisfy QoS requirements

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 89–107, 2013.

of many up-to-date Web-based applications. However, as more companies rely on the Internet to conduct their business, there is an increasing need to develop QoS-enabled Web services for a variety of e-business domains.

We address this issue in the context of Web server support for QoS on e-commerce Business-to-Consumer (B2C) Web sites. We consider an online store, where Internet users are customers accessing a Web site to search for information on products and to purchase them. The site consists of many pages, each of which may be assigned to one of the "typical" Web interactions at the B2C site. An example set of possible Web interactions includes entry to the home page of the site, browsing new products and bestsellers, searching for products according to specific keywords, adding selected items to a virtual shopping cart, user registration, placing an order, and making a payment. During a single visit to the site, a user accesses pages one after another, thus performing a sequence of Web interactions, which makes up a *user session*.

Interactions involved in making a purchase are usually performed at the end of a user session. Thus, to complete a purchase transaction, users should be provided with high QoS during their whole session. The main determinant for user-perceived QoS is time needed to display the requested Web page by the user's Internet browser. The Web is a kind of soft real-time system, for which timing constraints are not strictly specified; they are connected with users' tolerance of delays, which may differ depending on individual user's expectations, experience, and other factors. A page latency limit for a typical Internet user has been claimed to be determined by the "8-second rule" [1] – it means that if a user does not receive the requested content in eight seconds, they are likely to give up on the given Web page. For B2C Web sites the user page latency limit is even shorter, about 4 seconds [2]. Long delays are very damaging to e-business [2, 3]. They negatively influence customers' online purchase intentions and loyalty, weaken company image, and discredit Web site security in the eyes of customers. As a consequence, users are less likely to make a purchase during a visit at the site as well as to return to the site in the future. Poor QoS leads to aborted user sessions, incomplete transactions, and revenue losses.

Efficiency and scalability of a Web server are key factors for the success of a Web store. Although many components contribute to user-perceived QoS, including data transmission delays in local and wide area networks, end-to-end delays have been shown to be strongly dominated by Web server delays [4, 5]. This tendency is especially evident with the intensive use of dynamic Web content, e.g., on B2C Web sites. Moreover, unlike network nodes, Web servers are under control of commercial providers, and possible QoS solutions may be put into practice.

The majority of Web sites are capable of serving the incoming Web traffic timely most of the time. However, the bigger the number of concurrent user sessions at the site is, the longer delays perceived by users are. Unpredictability and variability of Web traffic makes it practically impossible to ensure a perfect service at all peak times. The well-known example is "Cyber Monday", the busiest online shopping day of the year in the USA and other countries, followed by several days of continued heavy online spending into the middle of December. In 2012 Cyber Monday in USA reached $1.46 billion in online spending, up 17% from the year before [6]. Many e-commerce sites experience outages on that day every year. Let us recall a spectacular Yahoo case from 2007, when its shopping checkout service fell under Cyber Monday traffic, preventing half of Yahoo's 40,000 online merchant customers from processing any transactions for more than eleven hours [7]. Such situation means undermined business partner relationships and huge revenue losses both for online retailers and service providers.

The presented background has motivated us to consider the problem of QoS from the perspective of an online retailer and to address the problem of a server-side support for e-customer loyalty through offering more predictive service. The remainder of the paper is organized as follows. Section 2 discusses motivation for our approach to QoS differentiation taking customer loyalty aspects into consideration. Section 3 outlines the way of request processing in popular multi-tiered Web server systems subject to overloads. Section 4 proposes a new method for request service control in such a system. Section 5 discusses key results of simulation experiments comparing the efficiency of our approach to FIFO (*First In First Out*) service. Section 6 overviews related work, and Section 7 concludes the paper.

2 Motivation for Our Approach

E-commerce environment is highly competitive as users can easily switch between different online stores. One of the effective ways to cope with this situation is building strong relationships with customers based on their long-term loyalty and a reliable company image. In the case of brick-and-mortar and electronic businesses, a predominant part of profit is achieved thanks to a small percentage of the most profitable customers [8]. It has been shown that adoption of a customer-oriented strategy and effective customer relationship management results in a bigger customer retention and loyalty [9].

Many companies nowadays use information technologies to realize a customer-focused strategy. In particular, customer relationship management (CRM) systems are used to acquire, analyze, and use the knowledge of customers. Customer knowledge is typically used in such areas as marketing, sales, customer service, and technical support. However, as the key factor affecting the customer satisfaction and loyalty in e-commerce environment is QoS with respect to page response times, the effort made by an online retailer to realize a CRM strategy may be easily thwarted by FIFO scheduling at the Web server system hosting the B2C Web site.

The Web server is able to automatically acquire and manage customer knowledge as users proceed with their sessions at the site. In [10], we proposed extensions of a Web server system with the ability to identify the most valuable customers based on their purchase histories and to offer differentiated QoS to them under heavy load. That solution has been aimed at maximizing current revenue with customer values being an additional scheduling criterion. In this paper, we explore request scheduling according to customer values as the main criterion. Furthermore, we recognize the need to win over new customers and to serve unknown users – potential buyers – as efficiently as possible. We propose a method for priority-based request service control in a Web server system organized as a multi-tiered application. The method is called ECLO (*E-Customer Loyalty-Oriented admission control and scheduling*). It applies admission control and scheduling of requests taking into consideration properties of user sessions the requests belong to. The primary goal is to offer premium service, with regard to page response times, to the most valuable and loyal customers. The subsidiary goal is to aim at acceptable page delays for users without prior purchases, especially at the beginning of their sessions. To meet these requirements, soft page due times are introduced and considered in combination with current lengths of user sessions. Control decisions on request admission control and scheduling at the system bottleneck are taken based on the customer class and the fact whether page due time has been exceeded or not.

Additionally, we propose monitoring page response times offered by the system, and using these values to construct the system load indicator. We use this indicator as a control variable for admission control decisions as an alternative to commonly used system-level load indicators, such as CPU utilization or length of the queue to a bottleneck resource.

3 Request Service in Multi-tiered Web Server Systems

Accessing Web content by Internet users is realized according to the idea of client-server network processing. The communication between the client and the server proceeds according to HyperText Transfer Protocol (HTTP), the application-level protocol layered on top of TCP/IP protocol suite. The client, which is user's Internet browser software, sends HTTP requests to the Web server through the Internet. The server uses its resources to generate responses to incoming requests and to send the responses back to the client.

When a user issues a single Web page request, their browser generates and sends to the server a sequence of multiple HTTP requests. The first request in the sequence is for an HTML document, containing a Web page description, and the following requests are for objects embedded in the page (such as image or video files). After receiving all of the page's objects from the server, the browser completes the requested page and displays it to the user in a browser window. HTTP-level Web traffic is known to be highly unpredictable, variable ("bursty"), and self-similar [11, 12]. These characteristics negatively affect the efficiency of Web servers, leading to their temporary overloads, dropped or timed-out requests, and aborted user sessions (with all the undesirable consequences for e-business discussed in Sec. 1).

Web servers process many HTTP requests concurrently by multiple processes or threads. They typically handle a queue of incoming requests according to FIFO policy without any facilities for overload protection or QoS differentiation. Highly accessed e-commerce sites are typically organized as locally distributed systems comprised of many various components: a Web switch (called a Web dispatcher or a load balancer), Web servers, cache and image servers, application servers, database servers, and others (Fig. 1). All these elements are interconnected with high-speed Ethernet links and make up a multi-tiered architecture, in which request processing is realized at three logical software layers: Web server, application server, and database server (Fig. 2). The first layer, composed of one or more Web servers, sometimes aided by cache and image servers, is in charge of serving static Web content only.

Application and database servers are back-end servers responsible for generating dynamic personalized content through online computations and database accesses.

Web sites may be globally distributed systems with geographically distributed nodes. Furthermore, they may contain external links to other Web servers on the Internet. In this paper, we consider the case when the whole Web site content is hosted in one geographical location.

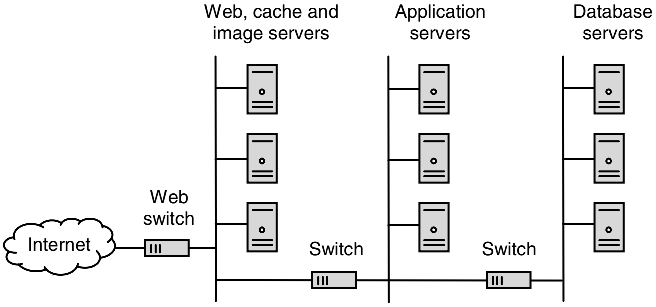


Fig. 1. A typical configuration of a locally distributed Web server system for a highly accessed B2C Web site

We use the phrase "Web server system" to represent the whole multi-tiered server node. We distinguish two coarse-grained components in such a system: a *front-end subsystem* including Web, image, and cache servers, and a *back-end subsystem* including application and database servers (Fig. 2).

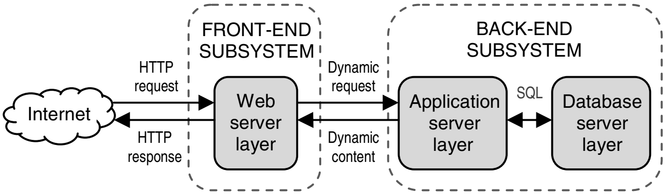


Fig. 2. A typical multi-tiered architecture of a B2C Web site

Two kinds of requests are considered in the system: HTTP requests and dynamic requests. HTTP requests arrive at the front-end subsystem from Web clients located on the Internet. If such a request is for a static Web object (e.g., image file), it is served solely by the front-end subsystem, in which the requested file is fetched from a Web server disk or cache and sent to the client. Otherwise, if an HTTP request is for a dynamic Web content (e.g., when bestsellers' data must be read from database), the front-end subsystem submits a dynamic request to the back-end subsystem. The content generated dynamically in the back-end is then passed to the front-end, which sends the completed HTTP response to the client.

Depending on the incoming Web traffic, different resources may become a bottleneck in such a system. When the traffic is dominated by static Web objects, the bottleneck resource is typically Web server CPU, disk, or network interface. However, in the case of e-commerce Web server systems, which are subject mostly to dynamic database-driven workload, the bottleneck lies typically in the back-end [13, 14, 15, 16]. These results justify exploring methods for scheduling of dynamic requests at the input of the back-end subsystem to alleviate negative results of system overload.

4 An Approach towards Support for E-Customer Loyalty through QoS Differentiation at the Web Server System

4.1 Acquisition and Management of Customer Knowledge

We classify each user interacting with a B2C Web site to one of two customer classes: a key customer (*KC*) class or an ordinary customer (*OC*) class. A key customer is a user who has bought something from the store in the past – for such a user the online retailer has some knowledge concerning their purchase history. All other customers are ordinary customers.

We assume that every user starts a session as an ordinary customer and may be identified as a key customer after logging on. We advocate identification of key customers by their logins instead of other possible ways, e.g., the ones based on clients' IP addresses or persistent HTTP cookies. An IP address contained in each HTTP request allows the site to identify a client at the very beginning of the session but it identifies a computer instead of a user, who may use different client machines to access the site during multiple visits. Moreover, client IP addresses are often masked due to firewalls or proxies, where one proxy server may mask even thousands of clients. Much more precise client identifier is a persistent HTTP cookie, which is a piece of information kept on the client and sent by a browser to a server during each visit to a given site. However, a user may remove existing cookies or forbid their storing on disk, so an application of persistent cookies is limited in practice. Unlike these two approaches, making use of customer logins allows the site to unambiguously identify users and clearly distinguish human users from robots, such as agents analyzing the workload, cataloguing the Web, or looking for product information, e.g., for price comparison services.

To acquire knowledge of key customers in e-commerce environment, we apply a method based on a recency-frequency-monetary (RFM) analysis as it has been originally proposed in [10]. RFM analysis is based on the customers' behavioural data observed throughout the entire life of their purchase online transactions. Recency (R) means the time interval from the customer's last purchase until now, frequency (F) means the total number of customer's purchase visits at the site, and the monetary value (M) means the total amount of money spent by the customer. RFM analysis consists of performing a customer database segmentation to assign recency, frequency and monetary codes to the customers. A single customer value *V* may be derived from a weighted sum of these codes:

$$V = w_R R + w_F F + w_M M \ , \tag{1}$$

where R, F, M are codes of the customer's behavioural variables computed as a result of a database segmentation, i.e., recency, frequency and monetary, respectively, and w_R, w_F, w_M are weights assigned to the corresponding behavioural variables according to the business strategy of the company. For instance, if R, F, $M \in \{1, \ldots, 5\}$, and the weights w_R, w_F, w_M are all equal to three, the possible customer values will be integers ranging from nine to 45. A detailed description of key customer database segmentation according to RFM analysis in included in [21].

Customer values may be stored in a dedicated key customer database on the Web server and read at each customer's logging into the site. Each customer's data should be updated during their subsequent purchase. Furthermore, RFM analysis of database should be performed periodically to reflect the actual customer knowledge.

4.2 Request Service Control in a Web Server System

We consider a Web server system consistent with the model presented in Fig. 2. Operation of the system is analyzed in a given observation window at discrete moments in time, determined by requests' arrivals and completions. The main symbols used in the problem description are summarized in Tab. 1.

Let n (n = 1, 2, …) represent the moment of a new HTTP request arrival at the front-end subsystem, and n' (n' = 1, 2, …) be the moment of a new dynamic request arrival at the back-end subsystem. Furthermore, let m (m = 1, 2, …) be the moment of HTTP request's rejection or successful completion in the system when some output values are observed. A diagram of request service control is shown in Fig. 3.

Control decisions concern admission control of HTTP requests arriving at successive moments n at the front-end subsystem as well as scheduling of dynamic requests arriving at successive moments n' at the back-end subsystem. Control decisions are determined by taking user sessions' progress into consideration as described below. Moments of request arrivals are not known in advance so the control tact has a variable length.

Each HTTP request is classified with regard to the pth page in user session s. The ith HTTP request in the pth page in session s is denoted by $x_{s,p,i}(n)$, where n is the moment of this request's arrival at the front-end subsystem. Thus, i = 1 means the first HTTP request in the page, i.e., a hit for an HTML document, whereas i > 1 means a hit for a Web object embedded in the page (e.g., a gif or jpg file). Enforcement of admission control for HTTP requests means that under the system overload some HTTP requests will be rejected at the system input, and they will not be processed by the system (Fig. 3).

Since each dynamic request is always generated within the process of serving an HTTP request, it may be considered to be a subtask of the corresponding HTTP request. Thus, a dynamic request for the corresponding HTTP request $x_{s,p,i}(n)$ is denoted by $x'_{s,p,i}(n')$, where $n' = n + l$ (l = 1, 2, …). Dynamic requests are placed in an unlimited queue, denoted by Q, in front of the back-end subsystem where they wait for access to the subsystem. Scheduling according to ECLO policy is realized in queue Q.

Table 1. List of the main symbols used in the paper

Symbol	Explanation
KC	key customer class
k	the number of recently completed OC page response times used to compute $L(n)$ (k is a parameter of ECLO algorithm)
$L(n)$	system load indicator at the nth moment [ms]
$l_{\mathrm{s}}(n)$	length of session s (i.e., the number of pages visited in session s) at the nth moment
m	moment of an HTTP request's rejection or completion
n	moment of a new HTTP request arrival at the front-end subsystem, $n = 1, 2, \ldots$
n'	moment of a new dynamic request arrival at the back-end subsystem, $n' = 1, 2, \ldots$
OC	ordinary customer class
$O_{p,s}$	a set of Web objects making up the pth page in session s
$P_{\mathrm{s}}(n)$	priority of session s at the nth moment
$p_{s,p,i}(n')$	a position in queue Q determined for request $x'_{s,p,i}(n')$ at the n'th moment
Q	a queue of dynamic requests in front of the back-end subsystem
$t_{s,p,i}(m)$	request response time for the ith HTTP request in the pth page in session s, completed at the mth moment
$t_{s,p}(m)$	page response time for the pth page in session s (computed after the page completion)
$\hat{t}_{s,p}(n)$	current page response time for the pth page in session s at the nth moment
T_{AC}	threshold for the system load above which admission control is enforced (a parameter of ECLO algorithm given in ms)
T_{due}	soft due time specified for OC pages (a parameter of ECLO algorithm given in ms)
T_{long}	threshold for the session length, which differentiates between short-lasting and long-lasting sessions (a parameter of ECLO algorithm given as the number of pages)
T_{user}	a user page latency limit, i.e. maximum page response time users are likely to tolerate [ms]
$x_{s,p,i}(n)$	the ith HTTP request in the pth page in session s, arrived at the front-end subsystem at the nth moment
$x'_{s,p,i}(n')$	a dynamic request issued to the back-end subsystem within the service of a corresponding HTTP request $x_{s,p,i}(n)$ at the n'th moment, $n' = n + l$ ($l = 1, 2, \ldots$)
$v_{\mathrm{s}}(n)$	value of the customer in session s at the nth moment

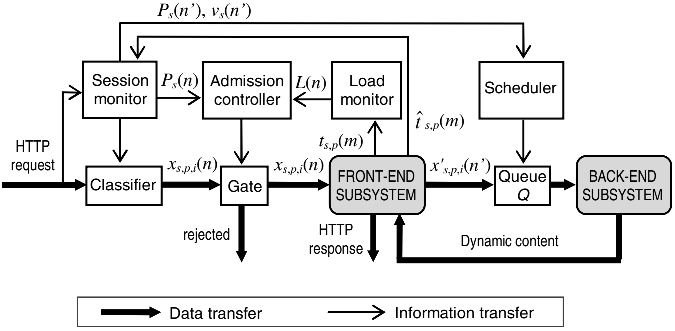


Fig. 3. Diagram of request service control in a Web server system

The accepted HTTP requests may experience different response times at the system. *Request response time* is defined as time needed by the system to complete a single HTTP request, i.e., the interval from the moment of request's arrival at the classifier till the moment when the last byte of the HTTP response has been sent. Request response time includes two main components:

- the first component is a service demand of the request (including times of request's residence at system's resources), which is load-independent;
- the second component is request waiting time (including times in system's internal queues and/or in queue Q), which heavily depends on the current system load.

Request response time for the *i*th HTTP request in the *p*th page in session s is denoted by $t_{s,p,i}(m)$, where m is the moment of request completion.

When all requests composing a Web page have been completed at the system, page response time is computed. *Page response time* is defined as the time needed by the system to complete a whole Web page, i.e., all HTTP requests for that page. Page response time for the *p*th page in session s, denoted by $t_{s,p}(m)$, is computed according to the formula:

$$t_{s,p}(m) = \sum_{x_{s,p,i}(n) \in O_{p,s}} t_{s,p,i}(n + l_i)\,, \tag{2}$$

where $x_{s,p,i}(n)$ is the *i*th HTTP request in the *p*th page of session s, arrived at the *n*th moment, $t_{s,p,i}(n+l_i)$ is request response time computed at the moment $n+l_i$ ($l_i = 1, 2, \ldots$) for request $x_{s,p,i}(n)$, and $O_{p,s}$ is a set of Web objects making up the *p*th page in session s. Page response time is computed only for successfully completed pages.

We assume that a user requests Web pages in the session one after another: when they receive the requested page, it takes them some time to browse it and issue the next page request. As many users interact with a B2C site simultaneously, at any moment n there are multiple user sessions at the Web server system. Depending on the system load, each session may be successfully completed or aborted.

Session s is considered to be *successfully completed* at the *m*th moment if an HTTP request $x_{s,p,i}(n)$ completed at the *m*th moment has been the last request of the last page of session s, and page response time $t_{s,p,i}(m)$ has not exceeded user page latency limit T_{user}. Such a situation means that all pages of session s have been completed before T_{user} since session s had not been aborted earlier. If a user gives up the interaction for reasons other than long response time, the session is considered to be successfully completed as well.

Session s is considered to be *aborted* at the *m*th moment in two cases. The first case is when an HTTP request $x_{s,p,i}(n)$ has been rejected at the *m*th moment due to admission control ($n = m$ in this case). Such a situation means that page p could not have been completed. The second case is when an HTTP request $x_{s,p,i}(n)$ has been successfully completed at the *m*th moment but page response time $t_{s,p}(m)$ exceeded user page latency limit T_{user} (in such a situation we assume that the user gave up the interaction because they had to wait too long).

When a Web server system operates according to FIFO policy, all user sessions have the same chance of being aborted under overload. To overcome this drawback, we propose differentiating between user sessions based on their progress and the customer value. At the *n*th moment each session is characterized with two attributes:

1) The customer value $v_s(n)$: for a key customer this value (computed using RFM analysis) is read from the customer database at logging on, and for an ordinary customer it is equal to zero.
2) The session length $l_s(n) = 1, 2, \ldots$, means the number of Web pages visited in the session so far, including the current page.

In addition, for each session, current page response time is monitored. *Current page response time* for the *p*th page in session *s* at the *n*th moment is denoted by $\hat{t}_{s,p}(n)$, and it is updated after completion of each HTTP request belonging to the *p*th page in session *s*.

In order to support differentiated QoS in the Web server system, we propose introducing dynamic *session priorities*, which are updated at arrivals of HTTP requests based on the session attributes and the current page response time for the session. Priority-based approach has proven to be successful in guaranteeing differentiated levels of service [17, 18, 21].

We propose four priority levels ranging from 1 to 4, where 1 means the highest priority. A priority of session *s* at the *n*th moment, $P_s(n)$, is determined according to the following formula:

$$P_s(n) = \begin{cases} 1 & \text{for } v_s(n) > 0, \\ 2 & \text{for } (v_s(n) = 0) \text{ and } (l_s(n) \le T_{long}), \\ 3 & \text{for } (v_s(n) = 0) \text{ and } (l_s(n) > T_{long}) \text{ and } (\hat{t}_{s,p}(n) > T_{due}), \\ 4 & \text{otherwise,} \end{cases} \tag{3}$$

where $v_s(n)$ is the value of customer conducting session *s*, $l_s(n)$ is the length of session *s*, $\hat{t}_{s,p}(n)$ is current page response time for the *p*th page in session *s*, and T_{due} and T_{long} are two parameters of the algorithm: T_{due} is a soft due time specified for *OC* pages and T_{long} is a threshold for the session length, which differentiates between short-lasting and long-lasting sessions.

The rationale for such policy is as follows.

- Priority 1, the highest priority, is assigned to all sessions in which customers are characterized with non-zero values, i.e., to all *KC* sessions. Such sessions are considered the most important with regard to long-term e-business profitability so they should get the best possible service.
- Priority 2 is assigned to all short-lasting *OC* sessions in order to give a chance of successful interaction to all users at the beginning of their sessions and to allow key customers to log into the site (notice that all users are ordinary customers at the beginning of their sessions).
- When the *OC* session length is bigger than T_{long}, it is considered long-lasting and receives priority 3 or 4 depending on the current page response time: “overdue” sessions will be scheduled before non-overdue ones.

4.3 ECLO Algorithm

Session attributes and priorities are used in a heuristic admission control and scheduling algorithm called ECLO (*E-Customer Loyalty-Oriented admission control and scheduling*). System load is constantly monitored, and when it exceeds admission control threshold T_{AC}, admission control is enforced. We introduce a system load indicator at the nth moment, $L(n)$, which is directly related to QoS offered by the system. $L(n)$ is computed as a simple moving average of k recently observed page response times for ordinary customers. Such metric is easy to implement in a distributed Web server system as it does not require knowledge of low-level server resources, which may be difficult to obtain in practice [19]. The details of ECLO algorithm are presented in Fig. 4.

Algorithm 1. ECLO (*E-Customer Loyalty-Oriented admission control and scheduling*)

```
if a new HTTP request $x_{s,p,i}(n)$
  if the request is for an HTML document  /* a new Web page */
    update attributes of session $s$: $v_s(n)$ , $l_s(n)$
    update priority of session $s$: $P_s(n)$ according  to (3)
    /* admission control */
    if $L(n) > T_{AC}$ and $P_s(n) = 4$
      reject the request
    else
      accept the request
    end if
  else /* the request is for an embedded object */
    if $P_s(n) = 4$ and $\hat{t}_{s,p}(n) > T_{due}$
      $P_s(n) := 3$
    end if
    accept the request
  end if
else if a new dynamic request $x'_{s,p,i}(n')$
  /* scheduling */
  put the request into queue $Q$ at the position $p_{s,p,i}(n')$ determined according  to (8)
```

Fig. 4. Pseudocode of ECLO algorithm

Scheduling of dynamic requests in queue Q makes it possible to change the order of request execution in the back-end subsystem. Between four priorities a strict priority scheduling is applied, which means that all higher-priority requests are queued before the lower-priority ones. Requests belonging to key customer sessions, i.e., to sessions with priority 1, are ordered decreasingly according to customer values. Requests belonging to ordinary customer sessions, i.e., to sessions with priority 2, 3, and 4 are queued within the corresponding priority according to FIFO order.

At arrival of a new dynamic request at the back-end subsystem, a position in the queue for the request is determined in the following way. Let a_z denote a request a belonging to session z waiting in queue Q. We define the following set and subsets of requests waiting in the queue at the moment n' when a new request $x'_{s,p,i}(n')$ arrives:

$$Q_1(n') = \{a_z \in Q : (P_z(n') = 1) \wedge (v_z(n') > v_s(n'))\}\,, \tag{4}$$

$$Q_2(n') = \{a_z \in Q : P_z(n') \in \{1,2\}\}\,, \tag{5}$$

$$Q_3(n') = \{a_z \in Q : P_z(n') \in \{1,2,3\}\}\,, \tag{6}$$

$$Q_4(n') = \{a_z \in Q\}\,, \tag{7}$$

When a new request $x'_{s,p,i}(n')$ arrives, its position in the queue, $p_{s,p,i}(n')$, is determined according to the formula:

$$p_{s,p,i}\,(n') = \left|Q_k(n')\right| + 1 \text{ for } P_s(n) = k, \tag{8}$$

where $k \in \{1, 2, 3, 4\}$ and $|\bullet|$ means the cardinality of the corresponding set of requests. Requests waiting in queue Q are executed according to their positions.

5 Simulation Results

We have applied a simulation-based approach to evaluate the efficiency of the proposed algorithm. Based on the literature study, we worked out a simulation model of a B2C Web server system, including two main components. The first component is a session-based HTTP-level workload model. The second component is a queuing network model of a multi-tiered Web server system, in which the front-end subsystem is modeled as a single Web server with one CPU, one disk and cache, and the back-end subsystem is modeled as a single resource with one queue.

We implemented the model in a discrete event simulator using C++ and CSIM19 package [20]. The simulation model and tool have been discussed in [21] in detail.

Using the simulator, the study of Web server system performance under FIFO and ECLO algorithms was performed for the same input workload and system parameters in both cases. The workload contained 10% of *KC* sessions. The user page latency limit, T_{user}, was equal to 8000 ms. Other parameter values were the following: T_{long} = 2 pages, T_{AC} = 10,000 ms, T_{due} = 4000 ms, k = 20 observations. *KC* values ranged from nine to 45. Each single simulation experiment was run for a constant session arrival rate (i.e., for the fixed number of sessions initiated per minute), which ranged from 20 to 260 sessions per minute with a step of 20. During each experiment, system performance was monitored in a 3-hour observation window following a 10-hour preliminary phase of the simulated system operation.

Let us analyze page response times offered by the system to key and ordinary customers. Fig. 5 shows the 90-percentile of KC and *OC* page response time as a function of the session arrival rate. As it can be seen, for FIFO scheduling all customers equally suffer from extremely long delays under heavy load. Above the session arrival rate of about 160 sessions per minute, the 90-percentile of page response time exceeds the 8-second threshold of a user tolerance both for ordinary and key customers. At the maximum load level, this metric amounts nearly to 13 s. On the other hand, ECLO was able to offer differentiated QoS and process 90% of key customer Web pages within 0.6 s regardless of load intensity. Improvements for *OC* page response times were much smaller but clearly visible under overload. As it can be seen in Fig. 6, presenting the 90-percentile of page response time for all customers, results for our approach are below the assumed threshold T_{AC} = 10 s throughout the whole load range as opposed to FIFO.

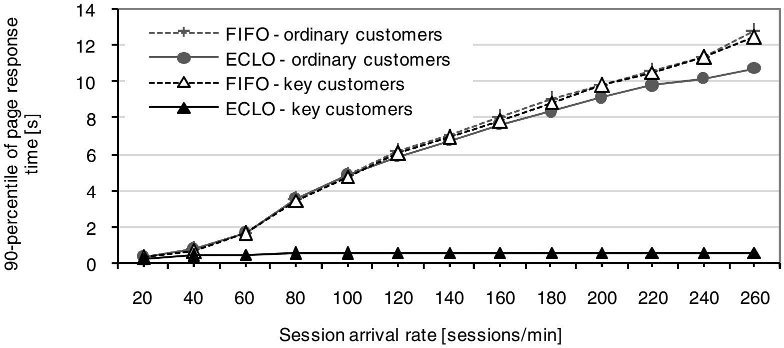


Fig. 5. 90-percentile of page response time for key and ordinary customers (ECLO vs. FIFO)

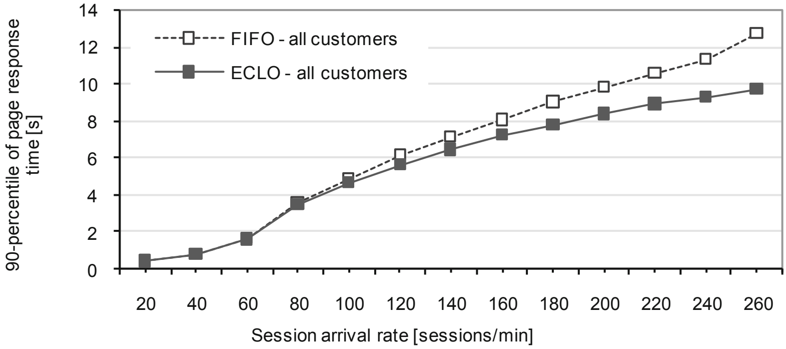


Fig. 6. 90-percentile of page response time for all customers (ECLO vs. FIFO)

Fig. 7 shows the percentage of successfully completed user sessions for FIFO and ECLO. Results for FIFO may be surprising because noticeably more key customers have been successfully served than the ordinary ones. The reason lies in the characteristics of the sessions for both types of customers [21]. The average length of *KC* sessions is much smaller than that of *OC* sessions, and a FIFO Web server is known to favor shorter sessions under overload. Besides, ordinary customers perform more Web interactions connected with searching for products, which are very time-consuming. Nevertheless, FIFO has not been able to provide key customers with acceptable QoS whereas ECLO has managed to successfully serve almost all key customers regardless of load intensity. The percentage of successfully completed *OC* sessions was a bit lower than for the FIFO case; however, taking into consideration benefits in page response times, especially for key customers, such result is justified. As it can be seen in Fig. 8, the aggregated results are better for ECLO than for FIFO.

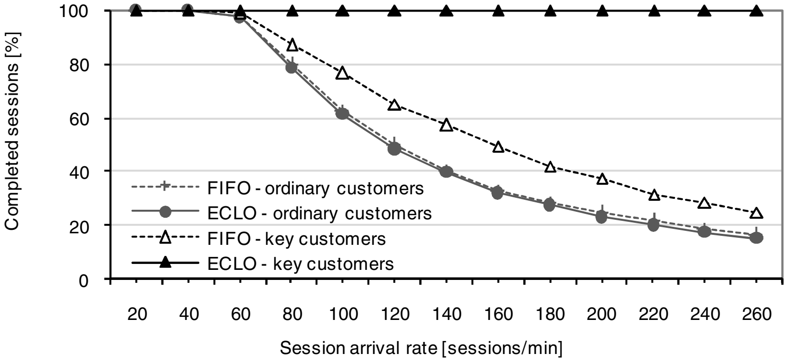


Fig. 7. Percentage of successfully completed key customer and ordinary customer sessions (ECLO vs. FIFO)

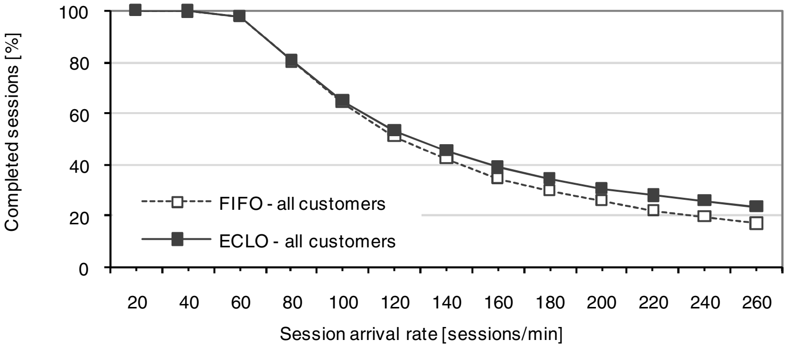


Fig. 8. Percentage of successfully completed sessions for all customers (ECLO vs. FIFO)

Fig. 9 presents the 90-percentile of page response time for various non-zero customer values (i.e. for *KC* class) for ECLO algorithm. Trend lines in the figure indicate that the system tended to offer higher page response times to sessions with lower ranks. One can observe that the higher the system load, the bigger QoS differentiation with regard to customer values.

To sum up this section, simulation experiments run for realistic bursty workload typical of B2C Web sites have shown that, with FIFO scheduling applied at the back-end subsystem, all customers equally suffer from long delays and aborted sessions under the system overload. Our approach has turned out to be successful in offering differentiated QoS for users characterized by different customer values. In particular, it was able to ensure limited page delays and almost hundred-percent completion of

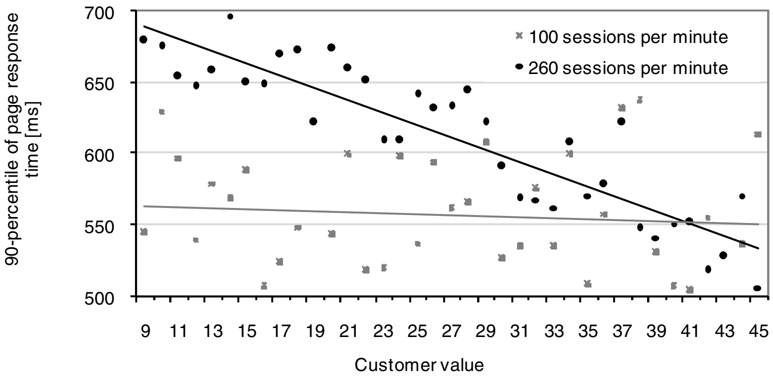


Fig. 9. 90-percentile of page response time for different key customer values (ECLO)

key customer sessions. QoS improvements with regard to page response times have also been visible for ordinary customers.

6 Related Work

Over the last years, there has been a growing interest in applying QoS control in distributed real-time systems running in open environments. As far as Internet-based services are concerned, much research effort has been made on QoS at different levels of client-server communication with a special focus on network nodes and Internet servers. For example, network QoS has been considered in the context of packet classification and scheduling aiming at differentiated queuing delays [22, 23].

A great advancement in improving user-perceived delays has been achieved by content caching, i.e., storing copies of frequently accessed data closer to users for future use. Different caching mechanisms have been proposed for Web clients, Web servers, and proxy servers deployed over the network near large communities of users. In particular, approaches for caching dynamic and personalized data for e-commerce applications have been proposed [24, 25]. The idea of content replication present in caching has been further broadened to other techniques, such as mirroring, Contents Delivery Networks, and distributed Web systems, for which various load balancing and request dispatching algorithms have been proposed [26, 27].

There has been a lot of research on Web server architectures being able to prevent server overload and support predictive service. In particular, EDF scheduling has proven to be successful in guaranteeing limited request response times [28, 29] and Web page response times [30]. There have also been attempts to combine network and Web server QoS to provide end-to-end delay guarantees for HTTP requests [31] and Web pages [32]. However, none of these solutions considered B2C applications, especially in the context of differentiated customer values.

First studies on QoS-enabled Web servers for B2C applications have indicated differences between the Web server throughput measured in completed HTTP requests and in completed sessions [33, 34]. It was shown that in the latter case, the server throughput radically decreases with the increase in the server load, even though it seems to be good in the former case. As a consequence, admission control algorithms respecting user session integrity have been proposed [16, 19, 34]. Further studies have proposed QoS approaches taking into account a specificity of a user session at a B2C Web site. They considered using different kinds of session-related information, such as session states corresponding to different Web interactions [17, 18, 21, 35, 36, 37, 38], probabilities of transitions between these states [18, 35], financial value of products in customers' shopping carts and the session length [17, 21], as well as purchase history-based customer values [10, 21]. There have also been studies on non-transparent solutions, e.g., an admission control and scheduling algorithm with load forecasting based on a discount-charge model [39].

A Web server architecture combining session-based admission control and session state-wise request queuing has been proposed in [36]. Simulation results have shown that requests in the final session state receive low delays, which enables users to complete purchase transactions. However, requests in the initial session state receive unacceptable delays and consequently many sessions are aborted by users at early stages. The state-wise request queuing is also a basis of LIFO-Pri algorithm, which combines LIFO and priority-based scheduling [18]. Experimental results have shown that this approach significantly improves the overall Web server throughput and increases the completion rate of "revenue-generating" requests at the cost of starvation of many "browsing" requests.

A two-dimensional service differentiation scheme for an e-commerce Web server has been proposed in [37]. A scheduling algorithm aimed at differentiated request delays in two dimensions: session states and customer classes, depending on a user profile. Simulations showed the efficacy of the approach in terms of the service slowdown. However, the problem of service differentiation was considered on the assumption that the resource demand of the workload was within the server capacity.

Some approaches have assessed transition probabilities between the session states for different user profiles based on data registered in Web server logs. In [35], such probabilities have been used to predict future aggregated load at the server in order to prevent the e-commerce Web server overload and to ensure shorter timeouts for sessions having high probability of being ended with a purchase. In [38], authors predict a future session structure by comparing requests seen in a user session so far with the aggregated information about recent customers' behavior, and use this prediction to ensure the bigger number of buying sessions at the e-commerce site.

In [17], a family of priority-based resource management policies for a B2C Web server has been discussed. In particular, a method for dynamic changes of the session priority depending on the session state, the session length, and the contents of a shopping cart has been proposed. A priority-based scheduling algorithm was proposed for the server CPU and disk. Simulation experiments have shown a significant improvement of the Web server performance in terms of business-oriented metrics, such as revenue throughput. However, these improvements occurred at the cost of deterioration of the mean page response time. The authors have not explicitly addressed the Web server overload.

An interesting solution from the business point of view is an admission control algorithm proposed in [16]. It takes a decision on a request admission or rejection based on information on a Buy-to-Visit (B2V) ratio correlated with the sender's IP address, thus taking the aspect of previous customers' purchases into account. Big advantages of this approach are its simplicity and low computational complexity. However, many users without prior purchases may be rejected at the very beginning of their sessions without a chance of even entering the site. Also, the system has no knowledge on the accepted sessions' progress, all of which are equally treated at the system.

An approach to distinguishing key customers of an online store and differentiating between them based on different aspects of their purchase histories has been proposed in [10]. The main goal of admission control and scheduling algorithm was to maximize the current revenue of an online retailer while providing key customers with high QoS as an additional service criterion. Further study on this approach, including various scheduling policies, has been discussed in [40]. On the contrary, in this paper we focused on the ability of the Web server system to offer the premium service to key customers with respect to their customer values and to offer the best possible service to other customers, who are potential buyers in a B2C scenario.

7 Concluding Remarks

Our work discussed in this paper complements and extends previous studies by considering the server-side support for customers' loyalty aiming at limited page delays. The proposed algorithm provides key customers with limited delays, improves delays faced by ordinary customers, and reduces the number of aborted sessions under heavy load.

Another contribution of the paper is proposal of a new load indicator for a multi-tiered Web server system. The indicator is directly related to user-perceived delays, i.e., to page response times offered by the system as opposed to system-level load indicators connected with HTTP response times or the system low-level resource usage. Simulation results have shown that such indicator can be successful in system load control. Our future work will explore other ways of load indicator construction, e.g., using a median instead of a mean value as well as other methods for limiting delays in a B2C Web server system.

References

1. Silverpop: 8 Seconds to Capture Attention. Technical report (2007)
2. Retail Web Site Performance: Consumer Reaction to a Poor Online Shopping Experience. JupiterResearch and Akamai Report (2006)
3. Bhatti, N., Bouch, A., Kuchinsky, A.: Integrating User-Perceived Quality into Web Server Design. Computer Networks 33(1-6), 1–16 (2000)
4. Harchol-Balter, M., Schroeder, B., Agrawal, M., Bansal, N.: Size-Based Scheduling to Improve Web Performance. ACM Trans. Comp. Syst. 21(2), 207–233 (2003)

5. Cardellini, V., Casalicchio, E., Colajanni, M., Yu, P.S.: The State of the Art in Locally Distributed Web-Server Systems. ACM Computing Surveys 34(2), 263–311 (2002)
6. comScore: Cyber Monday Spending Soars to $1.46 Billion, Ranking as Heaviest U.S. Online Spending Day in History (2012), http://www.comscore.com/Insights/Press_Releases/2012/11/Cyber_Monday_Spending_Soars_to_1.46_Billion
7. Goldman, J.: Yahoo's Cyber Morning After: A Case of Merchant's Wrath. CNBC (2007), http://www.cnbc.com/id/21991959
8. Koch, R.: The 80/20 Principle: The Secret of Achieving More with Less. Doubleday, New York (2008)
9. Kim, S.-Y., Jung, T.-S., Suh, E.-H., Hwang, H.-S.: Customer Segmentation and Strategy Development Based on Customer Lifetime Value: A Case Study. Expert Syst. Appl. 31(1), 101–107 (2006)
10. Borzemski, L., Suchacka, G.: Discovering and Usage of Customer Knowledge in QoS Mechanism for B2C Web Server Systems. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010, Part II. LNCS, vol. 6277, pp. 505–514. Springer, Heidelberg (2010)
11. Crovella, M., Bestavros, A.: Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes. IEEE/ACM Trans. Netw. 5(6), 835–846 (1997)
12. Kant, K., Venkatachalam, M.: Transactional Characterization of Front-End E-Commerce Traffic. IEEE Globecom 3, 252–2527 (2002)
13. Elnikety, S., Nahum, E., Tracey, J., Zwaenepoel, W.: A Method for Transparent Admission Control and Request Scheduling in E-Commerce Web Sites. In: WWW, New York, pp. 276–286 (2004)
14. Zhang, Q., Riska, A., Gui, J., Smirni, E.: Bottlenecks and Their Performance Implications in E Commerce Systems. In: Chi, C.-H., van Steen, M., Wills, C. (eds.) WCW 2004. LNCS, vol. 3293, pp. 273–282. Springer, Heidelberg (2004)
15. Datla, V., Goševa-Popstojanova, K.: Measurement-Based Performance Analysis of E-Commerce Applications with Web Services Components. In: IEEE ICEBE, pp. 305–314 (2005)
16. Yue, C., Wang, H.: Profit-Aware Overload Protection in E-Commerce Web Sites. J. Netw. Comput. Appl. 32(2), 347–356 (2009)
17. Menascé, D.A., Almeida, V.A.F., Fonseca, R., Mendes, M.A.: Business-Oriented Resource Management Policies for E-Commerce Servers. Perform. Eval. 42(2-3), 223–239 (2000)
18. Singhmar, N., Mathur, V., Apte, V., Manjunath, D.: A Combined LIFO-Priority Scheme for Overload Control of E-commerce Web Servers. In: IISW 2004 (2004)
19. Kihl, M., Widell, N.: Admission Control Schemes Guaranteeing Customer QoS in Commercial Web Sites. IFIP Net-Con 235, 305–316 (2002)
20. CSIM19, Development Toolkit for Simulation and Modeling, http://www.mesquite.com
21. Borzemski, L., Suchacka, G.: Business-Oriented Admission Control and Request Scheduling for e-Commerce Web Sites. Cybernet. Syst. 41(8), 592–609 (2010)
22. Dovrolis, C., Stiliadis, D., Ramanathan, P.: Proportional Differentiated Services: Delay Differentiation and Packet Scheduling. IEEE/ACM Trans. Netw. 10(1), 12–26 (2002)
23. Świątek, P., Grzech, A., Rygielski, P.: Adaptive Packet Scheduling for Requests Delay Guarantees in Packet-Switched Computer Communication Network. Systems Science 36(1), 7–12 (2010)
24. Liu, F., Makaroff, D., Elnaffar, S.: Classifying E-Commerce Workloads under Dynamic Caching. In: IEEE SMC, vol. 3, pp. 2819–2824 (2005)

25. Soundararajan, G., Amza, C.: Using Semantic Information to Improve Transparent Query Caching for Dynamic Content Web Sites. In: IEEE DEEC, pp. 132–138 (2005)
26. Borzemski, L., Zatwarnicki, K., Zatwarnicka, A.: Adaptive and Intelligent Request Distribution for Content Delivery Networks. Cybernet. Syst. 38(8), 837–857 (2007)
27. Soundararajan, G., Manassiev, K., Chen, J., Goel, A., Amza, C.: Feedback-Based Scheduling for Back-End Databases in Shared Dynamic Content Server Clusters. In: 2nd IEEE ICAC, pp. 348–349 (2005)
28. Kanodia, V., Knightly, E.W.: Ensuring Latency Targets in Multiclass Web Servers. IEEE Trans. Parallel Distrib. Syst. 14(1), 84–93 (2003)
29. Quan, Z., Chung, J.-M.: Statistical Admission Control for Real-Time Services under Earliest Deadline First Scheduling. Computer Networks 48(2), 137–154 (2005)
30. Zatwarnicki, K.: Providing Web Service of Established Quality with the Use of HTTP Requests Scheduling Methods. In: Jędrzejowicz, P., Nguyen, N.T., Howlet, R.J., Jain, L.C. (eds.) KES-AMSTA 2010, Part I. LNCS, vol. 6070, pp. 142–151. Springer, Heidelberg (2010)
31. Lin, W., Liu, Z., Xia, C.H., Zhang, L.: Optimal Capacity Allocation for Web Systems with End-to-End Delay Guarantees. Perform. Evaluation 62(1-4), 400–416 (2005)
32. Wei, J., Xu, C.-Z.: eQoS: Provisioning of Client-Perceived End-to-End QoS Guarantees in Web Servers. IEEE Trans. Comput. 55(12), 1543–1556 (2006)
33. Bhatti, N., Friedrich, R.: Web Server Support for Tiered Services. IEEE Network 13(5), 64–71 (1999)
34. Cherkasova, L., Phaal, P.: Session Based Admission Control: A Mechanism for Peak Load Management of Commercial Web Sites. IEEE Trans. Comput. 51(6), 669–685 (2002)
35. Chen, H., Mohapatra, P.: Overload Control in QoS-Aware Web Servers. Computer Networks 42(1), 119–133 (2003)
36. Carlström, J., Rom, R.: Application-Aware Admission Control and Scheduling in Web Servers. In: IEEE INFOCOM, vol. 2, pp. 506–515 (2002)
37. Zhou, X., Wei, J., Xu, C.-Z.: Resource Allocation for Session-Based Two-Dimensional Service Differentiation on E-Commerce Servers. IEEE Trans. Parallel Distrib. Syst. 17(8), 838–850 (2006)
38. Totok, A., Karamcheti, V.: RDRP: Reward-Driven Request Prioritization for E-Commerce Web Sites. Electron. Commerce Res. Appl. 9, 549–561 (2010)
39. Shaaban, Y.A., Hillston, J.: Cost-Based Admission Control for Internet Commerce QoS Enhancement. Electron. Commerce Res. Appl. 8(3), 142–159 (2009)
40. Suchacka, G., Borzemski, L.: A Research Study on Business-Oriented Quality-Driven Request Service in a B2C Web Site. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part II. LNCS (LNAI), vol. 6923, pp. 425–434. Springer, Heidelberg (2011)

Applying IPC-Based Clustering and Link Analysis to Patent Analysis on Thin-Film Solar Cell

Tzu-Fu Chiu

Department of Industrial Management and Enterprise Information
Aletheia University, Taiwan
chiu@mail.au.edu.tw

Abstract. Patent analysis has been recognized as an important task at the government and company levels. Patent data contain plentiful technical information, which is worthwhile to be used in patent analysis in order to find out the technical categories and the technological trend. Due to the complex nature of patent data, two data mining methods: IPC-based clustering and link analysis, are used to figure out the possible technological trend on thin-film solar cell. IPC-based clustering, a proposed clustering method for exploiting the professional knowledge of the patent office examiners, will be utilized to generate the IPC-based clusters via the IPC and Abstract fields; while the link analysis will be adopted to draw a link diagram via the Abstract, Issue Date, and Assignee Country fields. During experiment, the major technical categories will be identified using IPC-based clustering, and the technological trend will be recognized through the link diagram. Finally, the major technical categories and technological trend will be provided to the managers and stakeholders for assisting their decision making.

Keywords: IPC-based clustering, link analysis, patent analysis, thin-film solar cell.

1 Introduction

Solar cell (especially thin-film solar cell) is a key technology option to realize the shift to a decarbonized energy supply and tends to offer a reduction of prices, rather than an increase in the future [1]. In addition, up to 80% of the technological information disclosed in patents is never published in any other form [2]. Meanwhile, patent analysis has been recognized as an important task at the government and company levels. Through appropriate analysis, technological details and relations, business trends, novel industrial solutions, or investment policy making can be achieved [3]. Due to the textual characters of patent data (in Abstract, Claim, and Description fields), a clustering method, IPC-based clustering is proposed to manipulate the homogeneity and heterogeneity of patents on the Abstract field so as to increase the cohesiveness (similarity) within a cluster and the dispersion (dissimilarity) among clusters. In patents, the IPC codes are provided by the examiners of patent office and contain the professional knowledge of the examiners

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 108–122, 2013.

[4]. It would be reasonable for a research to base on the term vectors of Abstract and the IPC codes to classify the patents into a certain number of clusters for facilitating patent analysis. Afterward link analysis is employed to find out the relations between year (/country) and cluster. Consequently, the technical categories will be obtained and the technological trend of thin-film solar cell will be recognized for assisting the decision making of managers and stakeholders.

2 Related Work

As this study is attempted to observe the technological trend for companies and stakeholders via patent data, a research framework is required and can be built via a utilization of IPC-based clustering (a modified clustering method) and an adoption of link analysis. In order to manipulate the homogeneity and heterogeneity of patent data, IPC-based clustering is proposed for dividing the patents into different clusters. Due to the collected data spreading over ten years (2000 to 2009) and in different countries, link analysis is employed to generate the linkages between year (/country) and clusters. Subsequently, the research framework will be applied to identify the technical categories and to recognize the technological trend on thin-film solar cell. Therefore, the related areas of this study would be patent analysis and technological trend, patent data and thin-film solar cell, IPC-based clustering, and link analysis, which will be described briefly in the following subsections.

2.1 Patent Analysis and Technological Trend

Patent analysis has been reviewed in the literature [5-8] and can be classified as: country level (policy making and international comparison), industry level (science and technology, knowledge spillovers, and competitive intelligence), organization level (technology licensing, corporate strategy, and business function), and technology level (technology development and product management) [8]. This study, attempting to explore the technological trend, is in the type of industry level (competitive intelligence).

Trend analysis is the practice of collecting information and attempting to spot a pattern, or trend, in the information [9]. Technology forecasting is to predict a moving trend of technological change. It also supports mining knowledge for technology marketing and reducing risk of R&D investment in company and government [10]. Moreover, technological trend investigation is useful for finding promising business fields in the future and for detecting the direction of competitive technical development, for examples: the trend of market entry, the trend of technological evolution, and the maturity of fields (matured, maturing, or undeveloped) [11]. Consequently, in order to draw the data mining techniques for observing the technological trend via patent data, a research framework will be designed by using IPC-based clustering and link analysis to perform the patent analysis on thin-film solar cell in this study.

2.2 Patent Data and Thin-Film Solar Cell

A patent document is similar to a general document, but includes rich and varied technical information as well as important research results [3]. Patent data, among the better structured and monitored data sources, is the official filings of inventions [12]. Patent documents can be gathered from a variety of sources, such as the United States Patent and Trademark Office [13], the European Patent Office [14], the Intellectual Property Office in Taiwan [15], and so on. A patent document includes numerous fields [13], such as: Patent Number, Title, Abstract, Issue Date, Application Date, Application Type, Assignee Name, Assignee Country, International Classification (IPC), Current US References, Claims, Description, etc.

Photovoltaics (PV) is the technology that generates direct current electrical power from semiconductors (or some other materials) when they are illuminated by photons [16]. Solar cell is the basic building block of solar photovoltaics and a sort of green energy. It can be mainly divided into two categories (according to the light absorbing material): crystalline silicon (in a wafer form) and thin films (of other materials) [17]. A thin-film solar cell (TFSC), also called a thin-film photovoltaic cell (TFPV), is made by depositing one or more thin layers (i.e., thin film) of photovoltaic material on a substrate [18]. The most common materials of TFSC are amorphous silicon and polycrystalline materials (such as: CdTe, CIS, and CIGS) [17]. In 2009, the photovoltaic industry production increased by more than 50% (yearly growth rates in average over the last decade: 40%) and reached a world-wide production volume of 11.5 GWp of photovoltaic modules, whereas the thin film segment grew faster than the overall PV market [1]. Therefore, thin film is the most potential segment with the highest production growth rate in the solar cell industry, and it would be appropriate for academic and practical researchers to contribute efforts to explore the technological trend of this segment.

2.3 IPC-Based Clustering

An IPC (International Patent Classification) is a classification derived from the International Patent Classification System (supported by WIPO) which provides a hierarchical system of symbols for the classification of patents according to the different areas of technology to which they pertain [19]. IPC classifies technological fields into five hierarchical levels: section, class, subclass, main group and sub-group, containing about 70,000 categories [20]. As stated by the Intellectual Property Office of UK [4], each patent document published will have at least one IPC code applied to it; and the EPO and other patent offices worldwide also use it to classify their own patent documents. The IPC codes of every patent are assigned by the examiners of the national patent office and contain the professional knowledge of the experienced examiners [21]. Therefore, it would be reasonable for a research to base on the IPC code and the term vectors of Abstract to cluster the patents into a number of categories. The IPC codes have been applied for assisting patent retrieval in some researches [21, 22].

As IPC codes of patent are provided by the examiners and contain professional knowledge, they are suitable to be exploited to direct the clustering process. So, a modified clustering method, IPC-based clustering, is proposed to include the IPC codes to enrich the clustering mechanism in this study. The idea of this method is also based on the author's previous studies [23, 24]. However, some differences between this study and the previous ones are: the IPC-based clustering method was modified and rewritten in more detail and more precisely; the research framework was reconstructed to be more appropriate for guiding the experiment; and the paper was reorganized so as to state the problem domain, the related work, the problem solving approach, and the experiment and explanation more clearly and completely.

The processes of the IPC-based clustering are IPC Group Centroid Generation, IPC-based Cluster Generation, Clustering Alternative Generation, and Optimal Alternative Selection, which are explained as follows:

(1) IPC Group Centroid Generation: The patents with the same IPC code will be put together to form an IPC code group (G_i), if a patent which has more than one IPC code will be assigned to multiple groups. Patents in the same IPC code group will then be applied to calculate a group centroid (called IPC group centroid) c_i using the term vector of the Abstract field (i.e., x_{ij}) via Equation (1) where G_i is the ith group.

$$c_i = \frac{1}{|G_i|} \sum\nolimits_{x \in G_i} x_{ij} \tag{1}$$

(2) IPC-Based Cluster Generation: According to the IPC group centroids and term vectors, the whole dataset of patents will be distributed into a certain number of clusters via the Euclidean distance measure as in Equation (2) [25] where x_{ij} is a term vector of patent in G_i.

$$d(x_{ij}, c_i) = \sqrt{(x_{ij} - c_i)^2} \tag{2}$$

A patent will be assigned to a specific IPC code cluster according to the shortest distance $d(x_{ij}, c_i)$ existing between that patent and the IPC code centroid c_i. The patents distributed to a code group form an IPC-based cluster.

(3) Clustering Alternative Generation: The first clustering alternative is made initially by including its composing IPC-based clusters of 4 clusters. The following alternatives are then made successively by 5 clusters up to a certain number (e.g., 31 in this study), which is determined based on the research requirements and the domain knowledge. Furthermore, for enhancing the accuracy of clustering, an adjusted method is suggested which retains the larger clusters (with more patents) from a potential alternative by setting the threshold of the number of comprising patents to a suitable value (e.g., 6 in this study) and then repeat the "IPC-based Cluster Generation" again to obtain an adjusted alternative. Subsequently, the original and adjusted alternatives will be used to form the overall clustering alternatives.

(4) Optimal Alternative Selection: Among the clustering alternatives, F score (in Equation (3)) is employed to evaluate the accuracy of the clustering results, where the Precision and Recall are in Equation (4) and (5) [26].

$$F = \frac{2}{(1/\mathrm{Re}call) + (1/\mathrm{Pr}ecision)} \tag{3}$$

$$\mathrm{Pr}ecision = \frac{|\{relevant \cap retrieved\}|}{|\{retreieved\}|} \tag{4}$$

$$\mathrm{Re}call = \frac{|\{relevant \cap retrieved\}|}{|\{relevant\}|} \tag{5}$$

An original or adjusted alternative with the higher F score will be selected as an optimal alternative of the clustering result.

2.4 Link Analysis

Link analysis is a collection of techniques that operate on data that can be represented as nodes and links [27]. A node represents an entity such as a person, a document, or a bank account. A link represents a relationship between two entities such as a parent/child relationship between two people, a reference relationship between two documents, or a transaction between two bank accounts. The focus of link analysis is to analyze the relationships between entities. The areas related to link analysis are: social network analysis, search engines, viral marketing, law enforcement, and fraud detection [27]. In search engines, the page rank of page A, $PR(A)$, can be calculated as in Equation (6), where T_j is a page pointing to A; $C(T_j)$ is the number of going out links from page T; and d is a minimum value assigned to any page [28, 29]. In social network analysis, the degree centrality of a node can be measured as in Equation (7), where $a(P_i, P_k) = 1$ if and only if P_i and P_k are connected by a link (0 otherwise) and n is the number of all nodes [30]. Additionally, in data mining, the relationship strength (i.e., the similarity between nodes) can be measured by Jaccard coefficient as in Equation (8), where r_i is the ith record of a data set [31, 32].

$$PR(A) = d + (1-d) * \sum_{j=1}^{n} (PR(T_j) / C(T_j)) \tag{6}$$

$$C_D(P_k) = \sum_{i=1}^{n} a(P_i, P_k) / (n-1) \tag{7}$$

$$Ja(r_i, r_j) = \frac{Freq(r_i \cap r_j)}{Freq(r_i \cup r_j)} \tag{8}$$

In this study, link analysis will be employed to generate the linkages between the year (/country) and the IPC-based cluster for the technological trend observation.

3 A Research Framework for Technological Trend Observation

A research framework for the technological trend observation, based on IPC-based clustering and link analysis, has been constructed as shown in Fig. 1. It consists of

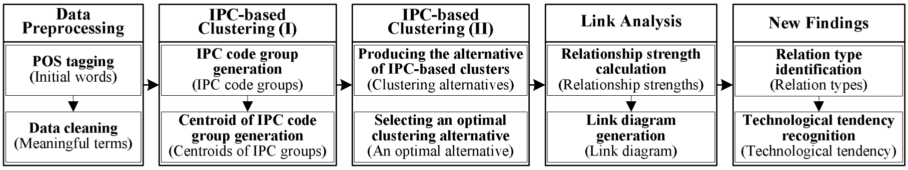

Data Preprocessing
POS tagging
(Initial words)
Data cleaning
(Meaningful terms)
IPC-based Clustering (I)
IPC code group generation
(IPC code groups)
Centroid of IPC code group generation
(Centroids of IPC groups)
IPC-based Clustering (II)
Producing the alternative of IPC-based clusters
(Clustering alternatives)
Selecting an optimal clustering alternative
(An optimal alternative)
Link Analysis
Relationship strength calculation
(Relationship strengths)
Link diagram generation
(Link diagram)
New Findings
Relation type identification
(Relation types)
Technological tendency recognition
(Technological tendency)

Fig. 1. A research framework for the technological trend observation

five phases: data preparation, IPC-based clustering (I), IPC-based clustering (II), link analysis, and new findings; and will be described in the following subsections.

3.1 Data Preprocessing

In first phase, the patent data of thin-film solar cell (during a certain period of time) will be downloaded from the USPTO [13]. For considering an essential part to represent a patent document, the Abstract, Issue Date, and Assignee Country fields are selected as the objects for this study. Afterward, two processes, POS tagging and data cleaning, will be executed to clean up the textual data of the abstract field.

(1) POS Tagging: An English POS tagger (i.e., a Part-Of-Speech tagger for English) from the Stanford Natural Language Processing Group [33] will be employed to perform word segmenting and labeling on the patents (i.e., the abstract field). Then, a list of proper morphological features of words needs to be decided for sifting out the initial words.

(2) Data Cleaning: Upon these initial words, files of n-grams, synonyms, and stop words will be built so as to combine relevant words into compound terms, to aggregate synonymous words, and to eliminate less meaningful words. Consequently, the meaningful terms will be obtained from this process.

3.2 IPC-Based Clustering (I)

Second phase is intended to describe the first two steps of the IPC-based clustering as stated in Subsection 2.3. The IPC code group generation is used to generate the IPC code groups according to the IPC field of patent data. The centroid of IPC code group generation is utilized to calculate the centroids of IPC code groups based on the patents in every group.

(1) IPC code group generation: A patent is distributed to an IPC code group if the patent contains that specific IPC code. Since a patent holds at least one to several IPC codes, a patent will be distributed to one or to several IPC code groups. For example, Patent 07605328 contains H01L031/00, B05D005/12, and H01L02/00 codes; and will be distributed to these three IPC code groups.

(2) Centroid of IPC code group generation: The comprising patents of an IPC code group are used to calculate the centroids for that IPC code group according to

Equation (1). For example, 48 patents of the first IPC code group (G_1) will be used to calculate its centroid (c_1).

3.3 IPC-Based Clustering (II)

Third phase is utilized to depict the other two steps of the IPC-based clustering. Producing the alternative of IPC-based clusters is applied to produce successively a series of clustering alternatives, each one consisting of a certain number of clusters. Selecting an optimal clustering alternative is to select an optimal alternative based on the F score measure for generating the technical categories.

(1) Producing the Alternative of IPC-Based Clusters: A clustering alternative is made by including its composing IPC-based clusters (e.g., 4 clusters in the first alternative; and 5 to 31 clusters in the following alternatives). An adjusted alternative is obtained by retaining the larger clusters (with more patents) from a potential alternative via setting the threshold of the number of comprising patents to a suitable value (e.g., 6 in this study) and then redistributing the patents again to the retained clusters. Both the original and adjusted alternatives form the clustering alternatives

(2) Selecting an Optimal Clustering Alternative: Among the clustering alternatives, *F* score (in Equation (3)) is employed to evaluate the accuracy of the clustering results. An original or adjusted alternative with the higher *F* score will be selected as the appropriate clustering result. In the selected alternative, every IPC-based cluster is regarded as a technical category and will be utilized for further analysis in the next phase.

3.4 Link Analysis

Third phase is designed to perform the link analysis for producing the relationship strengths and the link diagram so as to obtain the relations between years (/countries) and technical categories.

(1) Relationship Strength Calculation: In order to generate the summary table and link diagram, the relationship strength between nodes and the centrality of nodes (i.e., technical categories, years, and countries) need to be calculated via Equation (5) and Equation (4) respectively. The calculation result of relationship strength will be summarized in tables, so as to facilitate the identification of the linkages between year (/countries) and technical categories.

(2) Link Diagram Generation: Based on the summary tables and the node centrality, a link diagram will be drawn, so that the relations between year (/countries) and technical categories can be constructed through the threshold settings of relationship strength and node centrality. These relations will be utilized to identify the relation types between the year (/countries) and technical categories and then to explore the technological trend in the following phase.

3.5 New Findings

Last phase is intended to identify the relation types between technical categories and years (/countries) and to recognize the technological trend, based on the relationship strengths and the link diagram.

(1) Relation Type Identification: According to the relationship strengths and the link diagram, the relation types between the technical categories and years (/countries) will be identified. For the relations between categories and years, four relation types are likely found: a category existing in the full period of time (i.e., not less than 5) (type A1), existing in the first half (type A2), existing in the second half (type A3), and existing randomly in the period of time (type A4). For the relations between the categories and countries, three relation types are likely found: a category spreading in various countries (i.e., not less than 5) (type B1), spreading in the dominant countries (i.e., JP with 70 patents and US with 52 patents) (type B2), and spreading in the non-dominant countries (type B3).

(2) Technological Trend Recognition: In accordance with the relation types of technical category, the technological trend of thin-film solar cell will be recognized and then provided to the managers and stakeholders for assisting their decision making.

4 Experimental Results and Explanation

The experiment has been implemented according to the research framework. The experimental results will be explained in the following five subsections: result of data preprocessing, result of IPC code group and IPC group centroid, result of clustering alternatives and IPC-based clusters, result of link analysis, and result of new findings.

4.1 Result of Data Preprocessing

As the aim of this study is to explore the trends of thin-film solar cell, the patent documents are the target data for the experiment. Mainly, the Abstract, IPC, Issue Date, and Country fields were used in this study. The issued patents (160 records) during year 2000 to 2009 were collected from USPTO (USPTO, 2010), using key words: "'thin film' and ('solar cell' or 'solar cells' or 'photovoltaic cell' or 'photovoltaic cells' or 'PV cell' or 'PV cells')" on "title field or abstract field". Afterward, the POS tagger was triggered and the data cleaning process was executed to do the data preprocessing upon the Abstract data. Consequently, the Abstract data during year 2000 to 2009 were cleaned up and the meaningful terms were obtained.

4.2 Result of IPC Code Group and IPC Group Centroid

According to the IPC field, the number of IPC code groups (down to the fifth level) in 160 patents were 190, as many patents contained more than one IPC code, for example, Patent 06420643 even contained 14 codes. But there were up to 115 groups consisting of only one patent. If the threshold of the number of comprising patents was set to 5, there were 31 leading groups including the first group H01L031/18 (consisting of 48 patents), the second group H01L021/02 (27 patents), and till to the 31st group H01L031/0236 (5 patents), as in Fig. 2.

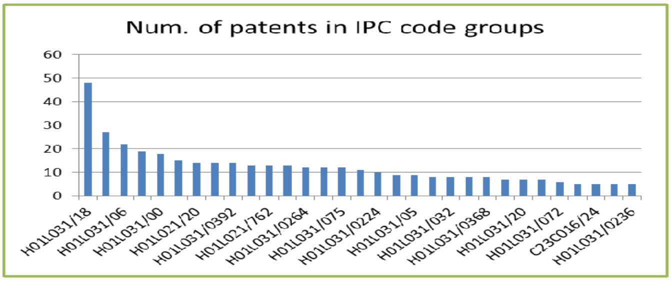


Fig. 2. The number of patents in IPC code groups

The patents contained in each IPC code group were used to generate the IPC group centroid for that group via Equation (1). These IPC group centroids would be utilized to produce the IPC-based clusters afterward.

4.3 Result of Clustering Alternatives and IPC-Based Clusters

The clustering alternatives contained the ones from including 4 leading clusters, to 5 leading clusters, …, till 31 leading clusters. The first alternative was constructed by distributing patents of the whole dataset into the 4 leading clusters (i.e., H01L031/18,

Table 1. A summary of clustering alternatives with their including clusters and accuracies

Alternative	Num. of clusters	Num. of patents in the IPC-based cluster	Accuracy
1	4	118, 11, 19, 12	0.4514
2	5	93, 11, 15, 10, 31	0.5109
3	6	87, 11, 15, 10, 29, 8	0.5213
4	7	84, 5, 15, 8, 29, 8, 11	0.5140
5	8	77, 5, 12, 8, 26, 8, 11, 13	0.5055
6	9	77, 5, 12, 1, 26, 8, 11, 13, 7	0.5052
7	10	73, 5, 12, 1, 24, 8, 11, 10, 7, 9	0.4988
8	15	55, 2, 2, 1, 24, 8, 0, 7, 10, 9, 10, 0, 10, 10, 12	0.5466
9	20	48, 2, 2, 0, 16, 8, 0, 5, 8, 5, 9, 0, 10, 10, 11, 8, 1, 6, 8, 3	0.5251
10	25	47, 2, 2, 0, 16, 7, 0, 5, 6, 5, 5, 0, 5, 10, 1, 8, 0, 6, 7, 3, 10, 2, 2, 5, 6	0.4992
11	31	42, 2, 0, 0, 16, 7, 0, 2, 6, 5, 5, 0, 4, 7, 1, 8, 0, 3, 6, 3, 10, 2, 0, 0, 6, 5, 4, 5, 5, 2, 4	0.5192
adjusted	9	70, 23, 15, 12, 10, 10, 7, 7, 6	0.5617

H01L021/02, H01L031/06, and H01L031/036) via Equation (2), using the IPC group centroids of 4 clusters and the term vectors of Abstract data. The number of patents distributed into 4 clusters was: 118 in H01L031/18, 11 in H01L021/02, 19 in H01L031/06, and 12 in H01L031/036. The accuracy (i.e., average *F* score) of this alternative was 0.4514. Each IPC group with its distributed patents was regarded as an IPC-based cluster. The other alternatives (from including 5 to 31 clusters) were then constructed successively. Some of the clustering alternatives with their including IPC-based clusters and accuracies were calculated and summarized as in Table 1.

According to Table 1, the accuracies of most alternatives varied from 0.50 to 0.53. The adjusted method was applied to pinpoint the leading clusters from the potential alternative 11 (with 31 clusters) by setting the threshold of the number of comprising patents to 6, so as to increase the accuracy of clustering to 0.5617. After the IPC-based Cluster Generation, the adjusted alternative, including 9 IPC-based clusters: H01L031/18, H01L031/00, H01L021/00, H01L021/20, H01L031/052, H01L031/048, H01L031/04, H01L031/0336, and H01L031/20, was the appropriate alternative as shown in Table 2 (with the containing patents and IPC code description). These nine IPC-based clusters were regarded as the major technical categories and used in the following link analysis.

Table 2. The appropriate alternative with including IPC-based clusters and IPC code description

IPC-based cluster	Num. of patents	IPC code description
H01L031/18	70	Processes or apparatus specially adapted for the manufacture or treatment of these devices or of parts thereof
H01L031/00	23	Semiconductor devices sensitive to infra-red radiation, light, electromagnetic radiation of shorter wavelength, or corpuscular radiation and specially adapted either for the conversion of the energy of such radiation into electrical energy or for the control of electrical energy by such radiation; Processes or apparatus specially adapted for the manufacture or treatment thereof or of parts thereof; Details thereof
H01L031/048	15	encapsulated or with housing
H01L031/052	12	with cooling, light-reflecting or light-concentrating means
H01L021/20	10	Deposition of semiconductor materials on a substrate, e.g. epitaxial growth
H01L031/0336	10	in different semiconductor regions, e.g. Cu_2X/CdX hetero-junctions, X being an element of the sixth group of the Periodic System
H01L021/00	7	Processes or apparatus specially adapted for the manufacture or treatment of semiconductor or solid state devices or of parts thereof
H01L031/04	7	adapted as conversion devices
H01L031/20	6	such devices or parts thereof comprising amorphous semiconductor material

4.4 Result of Link Analysis

Using link analysis, the relationship strengths between the major technical categories and years (/countries) were calculated and summarized in Table 3 and 4, where the items in italic face were the ones not less than the threshold setting: 0.05.

Table 3. The relationship strengths between major technical categories and years

Category	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
H01L031/18	***0.0588***	***0.1325***	***0.0824***	***0.1646***	***0.0741***	0.0274	***0.0921***	***0.1579***	0.0411	***0.0506***
H01L031/00	0.0238	0.0444	***0.0976***	0.0465	***0.0526***	0	0.0286	***0.1081***	***0.0741***	***0.1613***
H01L031/048	***0.0606***	***0.1143***	***0.1563***	0.0278	***0.0667***	0	0.0370	0	0	0
H01L031/052	0	***0.1250***	***0.0968***	***0.0625***	0.0357	0	0.0417	0.0345	0	0
H01L021/20	***0.1538***	0.0303	0.0323	***0.0667***	0.0385	***0.0714***	0	0	0	0
H01L031/0336	***0.2000***	0	***0.0667***	0	0.0385	***0.0714***	0	0	0	0.0455
H01L021/00	0	0	0	0	0.0435	***0.0909***	***0.0526***	0.0417	***0.0833***	***0.1111***
H01L031/04	0.0385	0	0	0.0357	***0.1429***	0	***0.0526***	0	0	***0.0526***
H01L031/20	***0.0833***	***0.0714***	0	0.0370	0	0	***0.0556***	0	0	0

Table 4. The relationship strengths between major technical categories and countries

Category	JP	US	DE	NL	FR	KR	AU	CA	BE	IT	CH	TH	FI	TW
H01L031/18	***0.2500***	***0.2323***	***0.1467***	0.0411	0.0139	*0*	*0*	0.0286	0.0141	*0*	*0*	0.0143	*0*	*0*
H01L031/00	***0.0690***	***0.2500***	0.0263	*0*	*0*	*0*	*0*	*0*	*0*	*0*	0.0435	*0*	*0*	*0*
H01L031/048	***0.1333***	0.0308	*0*	***0.0500***	***0.0588***	*0*	*0*	*0*	*0*	*0*	*0*	*0*	***0.0667***	*0*
H01L031/052	***0.0513***	0.0492	***0.0769***	*0*	*0*	*0*	***0.1667***	*0*	***0.0769***	*0*	*0*	*0*	*0*	*0*
H01L021/20	***0.1111***	0.0333	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*
H01L031/0336	***0.0667***	0.0333	0.0400	***0.0667***	***0.0833***	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*
H01L021/00	0.0132	***0.0727***	*0*	*0*	*0*	***0.1250***	*0*	*0*	*0*	***0.1429***	*0*	*0*	*0*	*0*
H01L031/04	***0.0694***	*0*	*0*	*0*	*0*	***0.1250***	*0*	*0*	*0*	*0*	*0*	*0*	*0*	***0.1429***
H01L031/20	0.0411	0.0175	0.0476	***0.0909***	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*

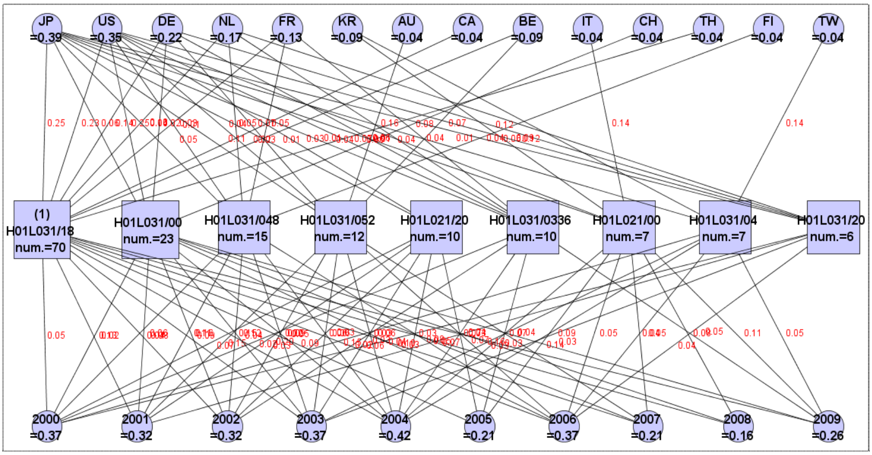


Fig. 3. A link diagram for 9 major technical categories (H01L031/18 to H01L031/20)

Based on Table 3 and 4, a link diagram for 9 major technical categories was drawn via the relationship strengths (not less than the threshold 0.05) between the categories and the years (/countries) and the centralities of the year and country nodes in order to demonstrate the relations between the categories and the years (/countries), which was shown in Fig. 3. In the diagram, the digits under the year and country nodes were the centralities (e.g., year 2000: 0.37; JP: 0.39); the digits on the link lines were the relationship strengths (e.g., between H01L031/18 and 2000: 0.05; between H01L031/18 and JP: 0.25).

4.5 Result of New Findings

The link diagram (i.e., Fig. 3) would be utilized to identify the relation types. Afterward, relation types would be applied to explore the technological trend.

(1) Relation Type Identification: According to the link diagram (Fig. 3) and the summarized table (Table 3), the relation type A1 (between technical categories and years) were: H01L031/18 and H01L031/00. The relation type A2 were: H01L031/048, H01L031/052, H01L021/20, and H01L031/0336. The relation type A3 were: H01L021/00 and H01L031/04. The relation type A4 was: H01L031/20. In addition, according to the link diagram (Fig. 3) and the summarized table (Table 4), the relation type B1 (between technical categories and countries) were: H01L031/18, H01L031/048, H01L031/052, and H01L031/0336. The relation type B2 were: H01L031/00, H01L021/20, and H01L031/20. The relation type B3 were: H01L021/00 and H01L031/04. Subsequently, the relation types between major technical categories and years as well as between major technical categories and countries were summarized below in Table 5 and then used to recognize the technological trend.

Table 5. A summary of major technical categories and relation types

Category	Focused year	Type	Related country	Type
H01L031/18	2000, 2001, 2002, 2003, 2004, 2006, 2007, 2009	A1	JP, US, DE, NL, FR,CA, BE, TH	B1
H01L031/00	2002, 2004, 2007, 2008, 2009	A1	JP, US, DE, CH	B2
H01L031/048	2000, 2001, 2002, 2004	A2	JP, US, NL, FR, FI	B1
H01L031/052	2001, 2002, 2003	A2	JP, US, DE, AU, BE	B1
H01L021/20	2000, 2003, 2005	A2	JP, US	B2
H01L031/0336	2000, 2002, 2005	A2	JP, US, DE, NL, FR	B1
H01L021/00	2005, 2006, 2008, 2009	A3	JP, US, KR, IT	B3
H01L031/04	2004, 2006, 2009	A3	JP, KR, TW	B3
H01L031/20	2000, 2001, 2006	A4	JP, US, DE, NL	B2

(2) Technological Trend Observation: According to the link diagram (Fig. 3) and the above summarized table (Table 5), the technological trend of thin-film solar cell could be observed. As then major technical categories were: H01L031/18, H01L031/00, H01L021/00, H01L021/20, H01L031/052, H01L031/048, H01L031/04, H01L031/0336, and H01L031/20, the technological trend of each technical category would be observed and described as follows.

(a) H01L031/18 category: Referring to Table 5, this technical category was continuously developing in the full period of time from 2000 to 2009 (relation type A1) and widely spreading in eight countries (relation type B1). It seemed to be an essential category of the industry, as it is related to the “manufacturing processes or devices”.

(b) H01L031/00 category: From the above Table 5, this category existed in the whole period of time from 2002 to 2009 (type A1) and spread in the dominant countries (type B2). It seemed that the category was growing constantly and participated by the technologically advanced countries. It is related to "semiconductor devices sensitive to infra-red radiation" and "the conversion of the energy".

(c) H01L031/048 category: According to Table 5, this category existed in the first half of the period of time (type A2) and spread in the various countries (type B1). It was likely that the category had been active during 2000 to 2004 and was out of the technical mainstream afterward. It was emphasized by several countries. It is about the "encapsulated or with housing".

(d) H01L031/052 category: Referring to Table 5, this category existed in the first half of the period of time (type A2) and spread in the various countries (type B1). It seemed that the category had been popular during 2001 to 2003 and declined gradually. It is relating to the "cooling, light-reflecting or light-concentrating means".

(e) H01L021/20 category: From the above Table 5, this category existed in the first half of the period of time (type A2) and spread in the dominant countries (type B2). It was likely that the category had been common during 2000 to 2005 and became minor afterward. It was focused mainly by the dominant countries JP (Japan) and US (United States). It is regarding the "deposition of semiconductor materials on a substrate".

(f) H01L031/0336 category: According to Table 5, this category existed in the first half of the period of time (type A2) and spread in the various countries (type B1). It was plausible that the category had been active during 2000 to 2005 and became unimportant eventually. It was stressed by several countries. It is concerning the "different semiconductor regions, e.g. Cu2X/CdX hetero-junctions".

(g) H01L021/00 category: Referring to Table 5, this category existed in the second half of the period of time (type A3) and spread in the non-dominant countries (type B3). It seemed that the category became popular lately from 2005 to 2009 and was contributed by the non-dominant countries as KR (Korea) and IT (Italy). It is relating to the manufacturing processes or devices of semiconductor.

(h) H01L031/04 category: From the above Table 5, this category existed in the second half of the period of time (type A3) and spread in the non-dominant countries (type B3). It was likely that the category gained emphasis slowly from 2004 to 2009 and was participated by the non-dominant countries like KR (Korea) and TW (Taiwan). It is concerning the "adapted as conversion devices".

(i) H01L031/20 category: According to Table 5, this category existed randomly in the period of time (type A4) and spread in the dominant countries (type B2). It seemed that the category was not in the technical mainstream and supported randomly the dominant countries as JP, US, DE (Germany) and NL (Netherlands). It is regarding the "devices comprising amorphous semiconductor material".

In addition, the significant H01L031/18 category possesses 70 patents (about 44%), which reflects that a big portion of patents put efforts in the manufacturing process and device aspects. The dominant countries JP and US possess 70 and 52 patents (about 44% and 32%) respectively, which shows that these two countries held the powerful innovative ability and resources in this industry and can affect the technological trend strongly.

5 Conclusions

The research framework (based on IPC-based clustering and link analysis) for observing the technological trend on thin-film solar cell has been formed. The experiment was performed and the experimental results were obtained. The major technical categories were: H01L031/18, H01L031/00, H01L021/00, H01L021/20, H01L031/052, H01L031/048, H01L031/04, H01L031/0336, and H01L031/20. The technological trend was as follows. The specific categories which existed in the full period of time and developed continuously were: H01L031/18 and H01L031/00 categories. The specific categories which existed in the first half of the period of time and became active earlier were: H01L021/00, H01L021/20, H01L031/052, and H01L031/048 categories. The specific categories which existed in the second half of the period of time and became common lately were: H01L031/04 and H01L031/0336 categories. The dominant countries which possessed the powerful innovative ability and resources in the thin-film solar cell industry were Japan and United States. The above experimental results and findings would be helpful to the managers and stakeholders for their decision making on R&D aspects.

In the future work, the other aspects of company information (e.g., the public announcement, open product information, and financial reports) can be included so as to enhance the validity of research result. Additionally, the patent database can be expanded from USPTO to WIPO or TIPO in order to perform the technological trend observation on thin-film solar cell widely.

Acknowledgments. This research was supported by the National Science Council of the Republic of China under the Grants NSC 99-2410-H-156-014.

References

1. Jager-Waldau, A.: PV Status Report 2010: Research, Solar Cell Production and Market Implementation of Photovoltaics, JRC Scientific and Technical Reports (2010)
2. Blackman, M.: Provision of Patent Information: A National Patent Office Perspective. World Patent Information 17(2), 115–123 (1995)
3. Tseng, Y., Lin, C., Lin, Y.: Text Mining Techniques for Patent Analysis. Information Processing and Management 43, 1216–1247 (2007)
4. Intellectual Property Office, Patent classifications (March 15, 2011), http://www.ipo.gov.uk/pro-types/pro-patent/p-class.htm
5. Basberg, B.L.: Patents and the Measurement of Technological Change: A Survey of the Literature. Research Policy 16, 131–141 (1987)
6. Ashton, W.B., Sen, R.K.: Using Patent Information in Technology Business Planning - I. Research Technology Management 31(6), 42–46 (1988)
7. Breitzman, A.F., Mogee, M.E.: The Many Applications of Patent Analysis. Journal of Information Science 28(3), 187–205 (2002)
8. Lai, K.K., Lin, M.L., Chang, S.M.: Research Trends on Patent Analysis: An Analysis of the Research Published in Library's Electronic Database. The Journal of American Academy of Business 8(2), 248–253 (2006)
9. Wikipedia, Trend analysis (March 16, 2012), http://en.wikipedia.org/wiki/Trend_analysis
10. Jun, S.: A Forecasting Model for Technological Trend using Unsupervised Learning. In: Kim, T.-h., Adeli, H., Cuzzocrea, A., Arslan, T., Zhang, Y., Ma, J., Chung, K.-i., Mariyam, S., Song, X. (eds.) DTA / BSBT 2011. CCIS, vol. 258, pp. 51–60. Springer, Heidelberg (2011)

11. Willfort, Technological Trend Investigation (March 16, 2012), http://www.willfort.com/english2/index.html
12. Russell, S.: Technology Forecasting. In: Narayanan, V.K., O'Connor (eds.) Encyclopedia of Technology and Innovation Management, pp. 37–45. John Wiley & Sons (2010)
13. USPTO (2010) USPTO: the United States Patent and Trademark Office (July 14, 2010), http://www.uspto.gov/
14. EPO (2010) EPO: the European Patent Office (July 14, 2010), http://www.epo.org/
15. TIPO (2010) TIPO: the Intellectual Property Office (July 14, 2010), http://www.tipo.gov.tw/
16. Luque, A., Hegedus, S.: Handbook of Photovoltaic Science and Engineering. John Wiley and Sons (2003)
17. Solarbuzz, Solar Cell Technologies (October 20, 2010), http://www.solarbuzz.com/technologies.htm
18. Wikipedia, Thin film solar cell (October 20, 2010), http://en.wikipedia.org/wiki/Thin_film_solar_cell
19. WIPO, Preface to the International Patent Classification (IPC) (October 30, 2010), http://www.wipo.int/classifications/ipc/en/general/preface.html
20. Sakata, J., Suzuki, K., Hosoya, J.: The Analysis of Research and Development Efficiency in Japanese Companies in the Field of Fuel Cells using Patent Data. R&D Management 39(3), 291–304 (2009)
21. Kang, I.S., Na, S.H., Kim, J., Lee, J.H.: Cluster-based Patent Retrieval. Information Processing & Management 43(5), 1173–1182 (2007)
22. Chen, Y.L., Chiu, Y.T.: An IPC-based Vector Space Model for Patent Retrieval. Information Processing & Management 47(3), 309–322 (2011)
23. Chiu, T.F., Hong, C.F., Chiu, Y.T.: To Propose Strategic Suggestions for Companies via IPC Classification and Association Analysis. In: Nguyen, N.T., Kim, C.-G., Janiak, A. (eds.) ACIIDS 2011, Part I. LNCS, vol. 6591, pp. 218–227. Springer, Heidelberg (2011a)
24. Chiu, T.-F., Hong, C.-F., Chiu, Y.-T.: Using IPC-based Clustering and Link Analysis to Observe the Technological Directions. In: Katarzyniak, R., Chiu, T.-F., Hong, C.-F., Nguyen, N.T., et al. (eds.) Semantic Methods for Knowledge Management and Communication. SCI, vol. 381, pp. 183–197. Springer, Heidelberg (2011b)
25. Feldman, R., Sanger, J.: The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press (2007)
26. Hotho, A., Nürnberger, A., Paaß, G.: A Brief Survey of Text Mining. LDV Forum - GLDV Journal for Language Technology and Computational Linguistics 20(1), 19–62 (2005)
27. Donoho, S.: Link Analysis. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, 2nd edn., pp. 355–368. Springer (2010)
28. Brin, S., Page, L.: The Anatomy of a Large-Scale Hypertextual Web Search Engine. Computer Networks and ISDN Systems 30, 107–117 (1998)
29. Weiss, S.M., Indurkhya, N., Zhang, T.: Fundamentals of Predictive Text Mining. Springer (2010)
30. Freeman, L.C.: Centrality in Social Networks: Conceptual Clarification. Social Networks 1, 215–239 (1979)
31. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Pearson Addison-Wesley (2006)
32. Wikipedia, Jaccard index (March 5, 2012), http://en.wikipedia.org/wiki/Jaccard_index
33. Stanford Natural Language Processing Group, Stanford Log-linear Part-Of-Speech Tagger (October 15, 2009), http://nlp.stanford.edu/software/tagger.shtml

Multi-agent Virtual Machine Management Using the Lightweight Coordination Calculus

Paul Anderson, Shahriar Bijani, and Herry Herry

School of Informatics, University of Edinburgh,
10 Crichton Street, Edinburgh, EH8 9AB, UK
{dcspaul,s.bijani}@ed.ac.uk,
h.herry@sms.ed.ac.uk

Abstract. LCC is a Lightweight Coordination Calculus which can be used to provide an executable, declarative specification of an agent interaction model. In this paper, we describe an LCC-based system for specifying the migration behaviour of virtual machines within, and between datacentres. We present some example models, showing how they can be used to implement different policies for the machine allocation and migration. We then show how LCC models can be used to manage the workflows that involve creation and deletion of virtual machines when migrating services between different datacentres.

Keywords: autonomic computing, multi-agent systems, virtual machines, OpenKnowledge, Lightweight Coordination Calculus.

1 Introduction

Virtualisation technology has recently transformed the availability and management of compute resources. Each *physical machine* (PM) in a datacentre is capable of hosting several *virtual machines* (VMs). From the user's point of view, a virtual machine is functionally equivalent to a dedicated physical machine; however, new VMs can be provisioned and decommissioned rapidly without changes to the hardware. VMs can also be *migrated* between physical machines without noticeable interruption to the running applications. This allows dynamic load balancing of the datacentre, and high availability through the migration of VMs off failed machines. The resulting *virtual infrastructure* provides the basis for *cloud computing*.

Managing the placement and migration of VMs in a datacentre is a significant challenge; existing commercial tools are typically based on a central management service which collates performance information from all of the VMs. If the current allocation is unsatisfactory (according to some policies), then the management service will compute a new VM allocation and direct agents on the physical machines to perform the necessary migrations.

As the size and complexity of datacentres increases, this centralised management model appears less attractive; even with a high-availability management

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 123–142, 2013.

service, there is possibility of failure and loading problems. If we would like to extend the domain of the virtual infrastructure to encompass multiple datacentres, managed by different providers, then the central model is no longer appropriate; in this federated "cloud" scenario, there may no longer be a single organisation with ultimate authority over all of the infrastructure.

This motivates us to propose a less centralised solution where agents located on the physical machines negotiate to transfer VMs between themselves, without reference to any centralised authority. This seems particularly appropriate for many situations where a globally optimal solution is not necessary or feasible; for example, if a machine is overloaded, it is often sufficient to find some other machine which will take some of the load. Likewise, an underloaded machine simply needs to take on additional VMs to improve its utilisation; there is no need for any global knowledge or central control.

However, moving virtual machines *between* datacentres is also more difficult: in general it is not possible to perform a live migration, and a new virtual machine must be started in the target datacentre, and the services transferred, before stopping the original virtual machine. The new machine will also have a different IP address, and possibly other differences, which mean that the migration may not be transparent to clients of the service. In this case, the clients will need to be notified about the change, and a comparatively complex workflow may be needed to avoid any break in the service. Once again, there may be no obvious central authority to sequence this workflow, and this motivates an agent-based approach to the workflow execution.

In this paper, we present a solution to the above problem where agents follow *interaction models* (IMs) described in the *lightweight coordination calculus* (LCC). The agents use the OpenKnowledge framework to locate appropriate interaction models and to identify suitable peers. These interaction models specify the agent behaviour, and allow them to make autonomous decisions; for example, the choice of VM to accept could be based on local capabilities, the properties of the VM being offered, the financial relationship with the donor, etc. Once a transfer has been agreed, the participating machines will execute interaction models which implement a workflow to effect the transfer. This may be a simple live migration which is transparent to any clients of the service, or it may be a more complex workflow which involves notifying clients, and stopping and starting virtual machines in different datacentres.

One important consequence of this approach is that we can very easily change the global policy of an entire infrastructure by introducing new interaction models. For example, a particular model may encourage the physical machines to distribute the load evenly among themselves; this makes a lightly-loaded infrastructure very agile and able to accept new VMs very quickly. Alternately, a different interaction model may encourage the machines to prefer a full, or empty, loading as opposed to a partial one. Some of the machines would then be able to dispose of all their VMs, allowing them to be turned off and hence saving power.

Section 2 provides some background on LCC and the OpenKnowledge framework, and section 3 presents LCC interaction models for various scenarios involving virtual machine allocation. These interaction models, together with a live prototype which implements them on a real datacentre, are described in more detail in [1]. Section 4 presents new work which describes an extension of the interaction models to manage the workflows which are necessary to deploy the allocations when services are migrated between datacentres and live migration is not possible. Section 5 covers some of the consequences and issues raised by this approach, and section 6 provides a brief discussion of some related work on VM management, including the state-of-the-art in commercial tools, as well as more experimental, agent-based approaches.

2 LCC and OpenKnowledge

A computational agent - such as one responsible for one of our physical machines - must be capable of acting autonomously, but it will also need to communicate with other agents in order to achieve its goals. In a multi-agent system (MAS), the agents often observe conventions which allow them to co-operate. These are analogous to the *social norms* in human interactions, and may be more or less formal – an oft-cited example is the rules which govern the bidding process in an auction. In our application, agents must be able to compare the respective resource utilisation of their hosts, and reach an agreement about the transfer of a virtual machine. Typically, the social norms in a MAS will be defined using an explicit protocol. The *lightweight coordination calculus* (LCC) is a declarative, executable specification language for such a protocol.

2.1 LCC

LCC [3] is based on a process algebra which supports formal verification of the interaction models. In contrast with traditional specifications for *electronic institutions*, there is no requirement to predefine a “global” script which all agents follow - the protocols can be exchanged and evolved dynamically during the conversation. LCC is used to specify “if” and “when” agents communicate; it does not define how the communication takes place[1], and it does not define how the agents rationalise internally. There are several different implementations of the LCC specification, including OpenKnowledge (see below), Li[2], UnrealLCC[3] and Okeilidh[4].

[1] The inter-agent communication mechanism is defined by the implementation.

[2] `http://sourceforge.net/projects/lij`

[3] `http://sourceforge.net/projects/unreallcc`

[4] `http://groups.inf.ed.ac.uk/OK/drupal/okeilidh`

There is insufficient space here to describe the LCC language in detail; the OpenKnowledge website contains a good introduction[5], and there are also some video tutorials[6]. The following brief summary should be sufficient to follow the annotated example presented in the next section:

Each IM includes one or more clauses, each of which defines a *role*. Each role definition specifies all of the information needed to perform that role. The definition of a role starts with: `a(`*roleName, PeerID).* The principal operators are outgoing message (`=>`), incoming message (`<=`), conditional (`<-`), sequence (`then`) and committed choice (`or`). Constants start with lower case characters and variables (which are local to a clause) start with upper case characters. LCC terms are similar to Prolog terms, including support for list expressions. Matching of input/output messages is achieved by structure matching, as in Prolog.

The right-hand side of a conditional statement is a *constraint*. Constraints provide the interface between the IM and the internal state of the agent. These would typically be implemented as a Java *component* which may be private to the peer, or a shared component registered with a discovery service. One advantage of the separation of interaction models from the constraints is that the interaction models can easily be shared.

2.2 OpenKnowledge

OpenKnowledge (OK[7])[4,5] provides an implementation of LCC, together with some additional functionality, including a distributed *discovery service* (2.2) and an *ontology matching service* (2.2). Having decided to participate in a particular interaction, peers register their desired roles with the discovery service. This identifies a suitable set of peers to fulfil each role in the interaction. The peers are then notified and the interaction proceeds without further involvement of the discovery service[8].

The Discovery Service: In addition to locating peers with matching roles, the OK discovery service provides facilities for discovering and distributing both interaction models and components (OKCs). This means that a physical machine (in our application) need only register its willingness to participate, and the behaviour will then be defined by the IMs and OKCs which are retrieved from the discovery service. Each peer has a choice of interaction models to suit various different scenarios, but once it has subscribed to an IM, all of the peers in that interaction will be following the same "script".

[5] `http://groups.inf.ed.ac.uk/OK/index.php?page=tutorial.txt`

[6] `http://stadium.open.ac.uk/stadia/preview.php?whichevent=984&s=29`

[7] `http://groups.inf.ed.ac.uk/OK/`

[8] In practice, the OK implementation elects a random peer to be a coordinator for the interaction, and the coordinator executes the IM, only making calls to other peers when it is necessary to evaluate a constraint. However, this is largely an optimisation decision and different implementations take different approaches.

The OK implementation is a scalable, open, efficient and robust service based on top of the FreePastry DHT implementation. This relies on keyword matching and is based-on a completely decentralised storing mechanism that requires O(log(N)) messages to store and search for N peers (see [6] for a discussion of the implementation). In a large scale evaluation, the OK discovery service significantly outperformed the two reference approaches [7].

Ontology Matching: A major strength of the OK system is that there is no need for a *global* agreement on interaction protocols. Any group of peers can subscribe to an IM which may be publicly available, or shared between a restricted group using some private mechanism. Likewise, there is no need for a global agreement on vocabulary for the OKCs or roles – there only needs to be agreement between those peers participating in a particular interaction, and only on those terms which appear in that interaction. Rather than an a-priori semantic agreement amongst component designers (which does not scale), the OpenKnowledge implementation provides dynamic ontology coordination at runtime. This uses various different mechanisms such as structural semantic matching and statistical analysis.

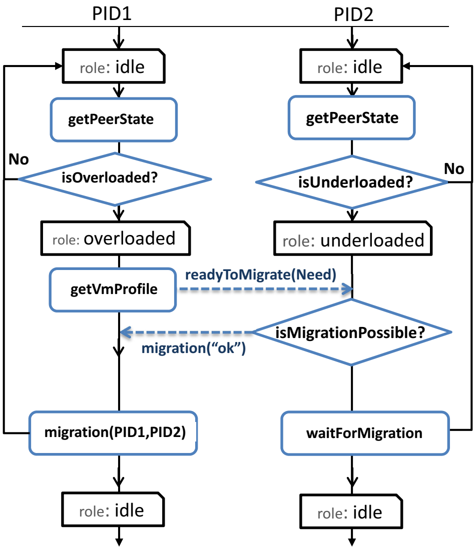


Fig. 1. The interaction diagram of a live migration: overloaded peer PID1 and underloaded peer PID2 interact to balance their loads

3 Interaction Models for VM Allocation

In this section we describe interaction models for managing the negotiation and transfer of virtual machines between two physical machines in the same datacentre (i.e. where live migration is possible). In the first instance we implement a simple policy which aims to migrate VMs from busy peers to underloaded peers in order to balance the load of each peer.

There are three states: *idle, overloaded* and *underloaded.* The *idle* state is the initial and the goal state, in which the peer is balanced. Each peer is assumed to be balanced at the beginning of the interaction. It may then change state based on its load, or other factors[9]. Once a peer becomes unbalanced, it advertises its status to the discovery service where it will be matched with *potential* candidates for a transfer. The peer then negotiates with these candidates to find one which is prepared to participate in the transfer. The conditions for acceptance of the transfer, and the complexity of the negotiation are completely determined by the interaction models of the individual peers – these may depend on, for example, security policies or cost considerations as well as the capabilities of the physical machine (processing power, network bandwidth, memory, etc.).

Figure 1 shows the interaction diagram of a very simply implementation[10], and figure 2 shows the corresponding LCC code. After an exchange of VMs, both agents revert to the "idle" role. If they are balanced, no further action takes place. Otherwise the unbalanced peers query the discovery service again for more potential exchange partners.

The feasibility of this model for a real live system has been validated using a prototype implementation based on a real physical cluster. This is described in more detail in [1]. In addition, we used a simple simulator to investigate the behaviour of more complex models with a larger number of machines and more controlled loading. Figure 3 shows the results of this interaction model applied to 50 simulated virtual machines running on 15 physical machines. In this example, physical machines offload VMs if they have a load greater than 120% of the average, and they accept VMs if they have a load less than 80%. Initially, the VMs are allocated randomly and the resulting load is uneven. The system stabilises after a time with all the physical machines except one within the desired range (the load on the remaining machine cannot be reduced because all of the machines have a load greater than 80%). Further results and details of the simulator are available in [8].

It may be the case that we would prefer to have the minimum number of active peers, each using almost all of their resources (e.g. to minimise the cost). A major advantage of the proposed approach is that such changes to the overall policy can be easily implemented by deploying a new LCC specification which implements a different interaction model. An implementation of this alternative

[9] For example, a peer which needs to be taken down for maintenance needs simply declare itself to be "overloaded" in order to dispose of all its virtual machines.

[10] Single-corner rectangles, diamonds and dashed-arrows represent agent roles, constraints, and message passing between agents, respectively.

```
// Definition of the "idle" role. Here, "idle" means the "balanced" state
a(idle, PeerID) ::
   // the constraint to check the state of the peer
   null <- getPeerState(Status) then
   //select the next state based on the peer's status
   (
      // if the peer is overloaded, change its role to "overloaded" and pass the status
      a(overloaded(Status), PeerID)<- isOverLoaded() then
   ) or (
      // if the peer is underloaded, change its role to "underloaded"
      a(underloaded(Status), PeerID) <- isUnderLoaded() then
   ) or
   // otherwise, remain in the idle role (recursion)
   a(idle, PeerID)

// Definition of the "overloaded" role. "Need" is the amount of resources required
a(overloaded(Need), PID1) ::
   // send the "readyToMigrate(Need)" message to an underloaded peer
   readyToMigrate(Need) => a(underloaded, PID2) then
   // wait to receive "migration(ok)" from the underloaded peer
   migration("ok") <= a(underloaded, PID2) then
   // live migration: send VMs from this peer to the underloaded peer
   null <- migration(PID1, PID2) then
   // change the peer's role to "idle"
   a(idle, PID1)

// Definition of the "underloaded" role: "Capacity" is the amount of free resources
a(underloaded(Capacity), PID2) ::
   // receive the "readyToMigrate(Need)" message from an overloaded peer
   readyToMigrate(Need)<= a(overloaded, PID1) then
   // send back the "migration(ok)" message, if the migration is possible, e.g.
   // free "Capacity" of this peer > "Need" of the overloader peer
   migration("ok") => a(overloaded, PID1) <-
        isMigrationPossible(Capacity, Need) then
   null <- waitForMigration() then
   // change the peer's role to "idle"
   a(idle, PID1)
```

Fig. 2. An LCC implementation of the interaction in figure 1

policy requires only a very small change to the LCC code, and is shown in full in [1][11].

In general, the LCC will provide a clear description of the interaction, and the constraints (usually implemented in Java) will be used to implement the interface to the machine (hypervisor) itself – for example detecting the load status. The policy itself may be defined either in the lcc, or within the constraints, or in a combination of both. Since the Java components, and the LCC interaction

[11] All of the LCC code for the models described in this paper is also available from `http://homepages.inf.ed.ac.uk/dcspaul/publications/ijicic.lcc`

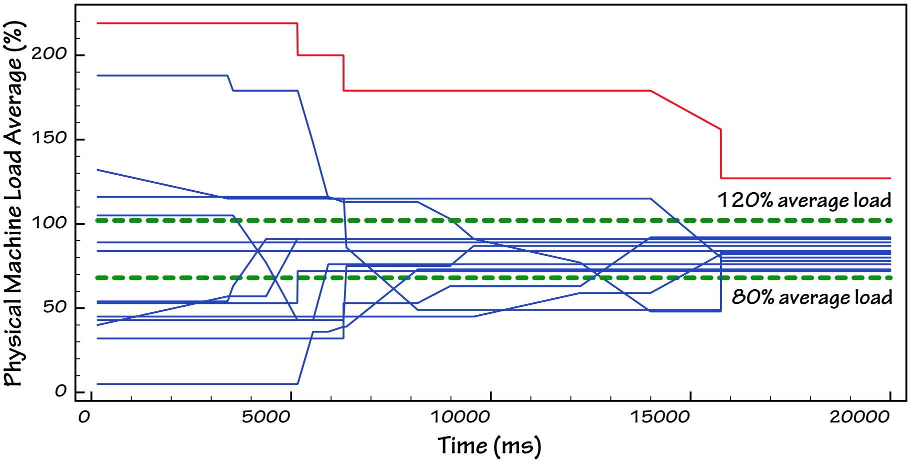


Fig. 3. A simulation showing the load on 15 physical machines as they interact to balance a load of 50 virtual machines

models can both be retrieved from the discovery service, the choice here depends on which is most appropriate in each case.

4 Managing the Workflow

Having negotiated to transfer a VM *within* the same datacentre , live migration is transparent and effected with a simple instruction to the hypervisor – the VMs and their clients need not be aware that the transfer has occurred. However, if the transfer is to occur *between* datacentres, then an *offline-migration* will be required and there will be a number of changes which are not transparent to any clients of the service – in particular, the IP address of the service will usually change. To maintain service during such a transfer requires a careful sequence of operations. Traditionally, this would be orchestrated by a centralised workflow engine (see section 6). However, in a distributed, federated environment this approach to the workflow suffers from exactly the same problems as a centralised approach to the allocation. In this section, we describe a typical pattern which occurs during service transfer, and we show how LCC interaction models can be used to sequence the necessary workflow without the need for a central controller.

In a very typical situation for a "cloud" environment, a particular service may need to be migrated from one datacentre to another – perhaps this is necessary because the internal capacity has been reached, or because of failures, or for contractual reasons. If the service has active clients, then the new copy of the service must be started, and all of the clients then transferred, before the original service can be stopped. This sequence is illustrated in figures 4 to 7.

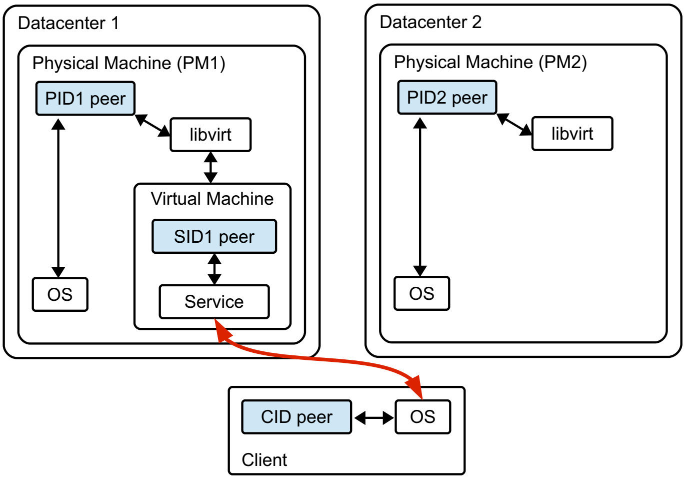


Fig. 4. The initial state of the system: A client is using a service which is running in datacentre 1

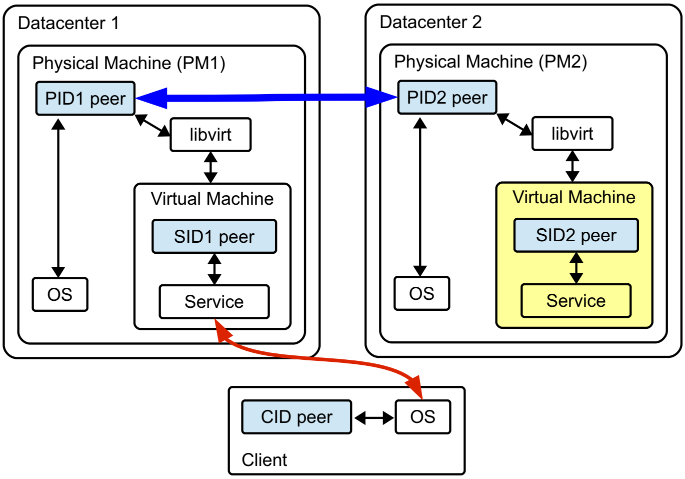


Fig. 5. The first stage of the workflow: A transfer of the service to datacentre 2 has been agreed, and a new virtual machine is started in the target datacentre to host the service

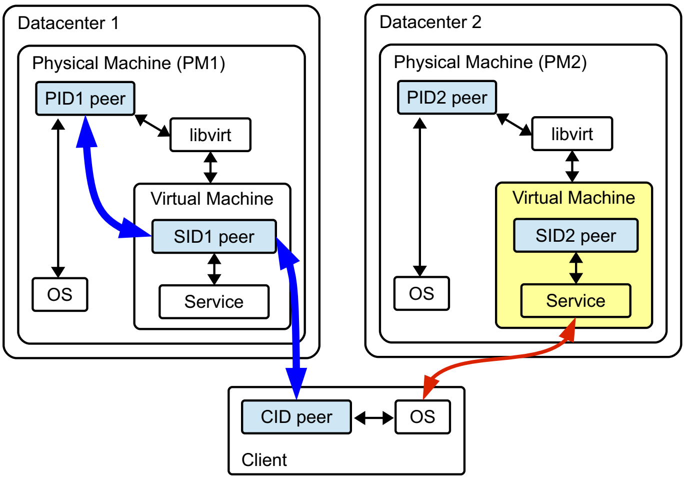


Fig. 6. The second stage of the workflow: The client is notified about the imminent removal of the original service and it locates and reattaches to the new service

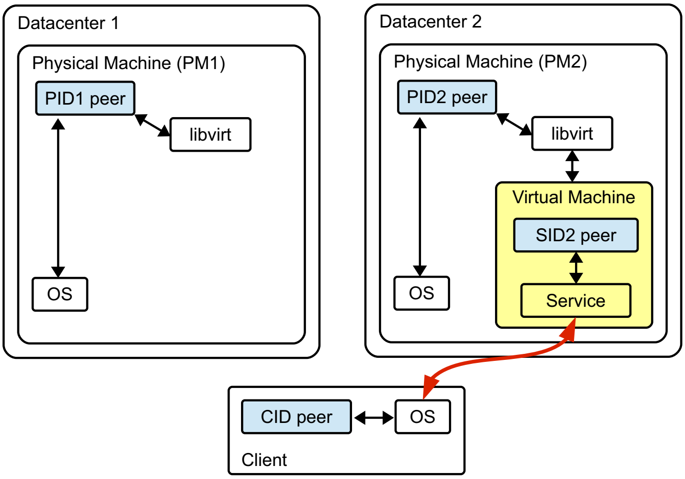


Fig. 7. When the client has left the original service, the service shuts down and the virtual machine is deleted

This workflow can be implemented in a fully distributed way using LCC. The workflow is separated into a number of roles which are assigned to associated agents, and the interaction between these agents will automatically execute the workflow. Figure 8 shows the corresponding interaction diagram[12].

This interaction operates as follows:

1. **Initial State** (figure 4)
   - The initial state includes a client which is using a service provided by a virtual server in datacentre 1. The client is managed by an agent (CID) which is capable of redirecting the reference from one service to another.
   - The service itself is managed by an agent (SID1), and is running on a virtual machine (VM1). This is hosted on a physical machine managed by agent (PID1).
   - A similar agent (PID2) is managing another physical machine (PM2) in datacentre 2.
   - The agents for all of the physical machines start in the "initial" role.
2. **Stage 1** (figure 5)
   - Now assume that PM2 has spare capacity and it therefore moves from the "initial" role into the state "canAcceptLoad". As in the example from the previous section, this is now registered with the discovery service as available to negotiate the acceptance of additional VMs.
   - If PM1 now needs to be removed from service for some reason, PID1 must migrate all of its virtual machines to another physical machine. It therefore moves into the "emigrant" role and is matched with PID2 by the discovery service.
   - Once the service transfer has been agreed, PID2 starts a new virtual machine (VM2) to host the new service instance, and informs PID1 when the service is available.
3. **Stage 2** (figure 6)
   - Before deleting the VM from PM1, agent PID1 must contact all of the clients of the service and inform them that the service is shutting down, and that they should redirect.
   - The client agents will attempt to locate a replacement server using the discovery service, and will reattach to the newly started service on VM2 (or possibly, some other alternative service).
4. **Final State** (figure 7)
   - Once all of the clients have redirected away from the original service, the VM on PM1 can be deleted.
   - PM1 is now free from virtual machines and able to shut down.

[12] The full LCC code is available at
`http://homepages.inf.ed.ac.uk/dcspaul/publications/ijicic.lcc`

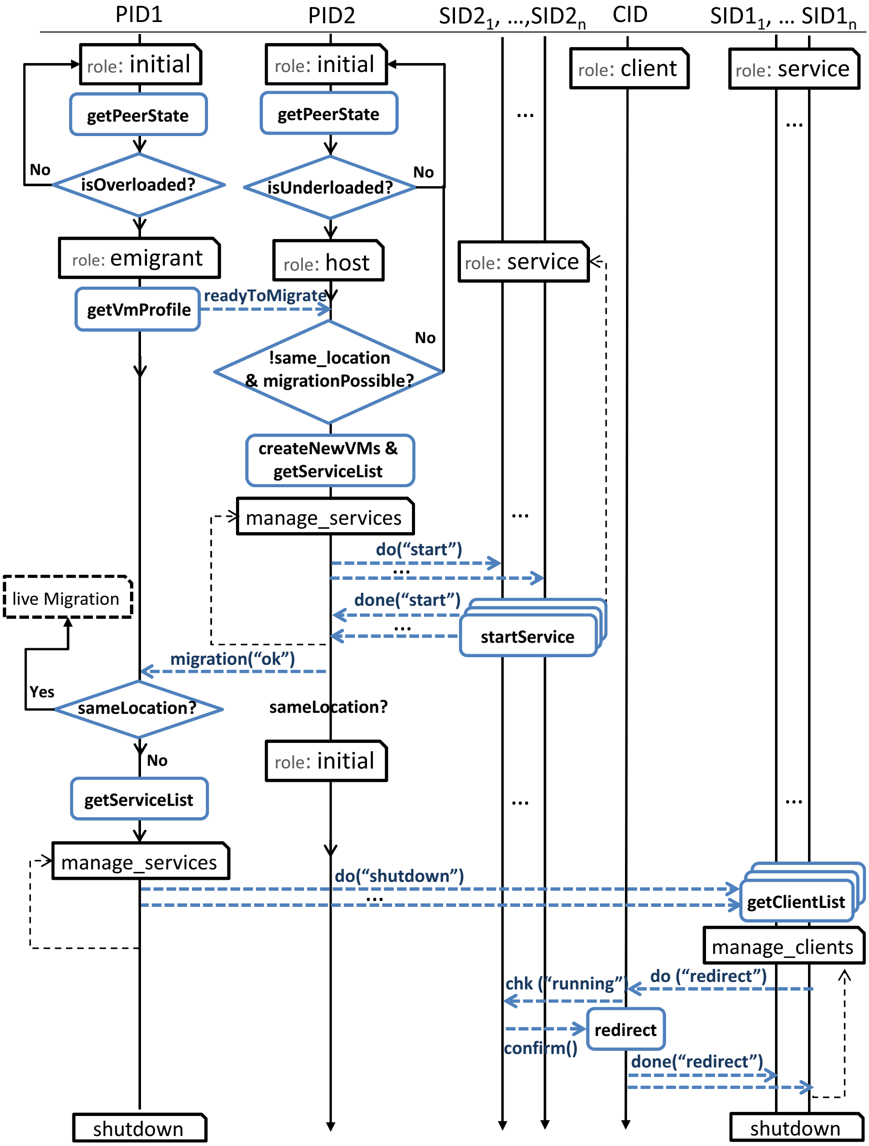


Fig. 8. An interaction diagram for an LCC interaction model of offline migration

5 Discussion and Evaluation

5.1 Centralised vs Distributed Approaches

The examples provided above have been designed to explore an extreme version of the distributed approach to the VM migration problem – individual peers have no overall knowledge of the system, and interactions are *choreographed* by a small number of peers interacting among themselves. This is a deliberate contrast to the conventional approach where a single controller with global knowledge *orchestrates* the entire interaction. In practice however, there is a continuous spectrum between these approaches - even the fully centralised tools devolve some details of the migration process to protocols which operate directly between the peers. And our distributed version relies on a discovery service which could be viewed as a type of centralised service.

The relative advantages and disadvantages of these approaches will vary, depending on the desired policies. For example, attempting to balance the load exactly across a whole datacentre clearly requires knowledge and comparison of the load on every machine. In this case, the distributed solution has no benefits and the extra overhead means that it will not perform as well as a simple centralised service. If however, we have a very large number of potential machines, and we only need (for example) to negotiate with *any* machine having a particular property, then the distributed solution excels by avoiding the (performance and reliability) bottleneck of a central controller.

However, one of the key strengths of the proposed approach is the flexibility with which the entire policy can be changed – it is easy to imagine a policy in which one machine assumes the role of “controller” and proceeds to orchestrate the remaining machines (or some subset of them) in a conventional way, thus emulating a centralised solution. So, functionally, the centralised approach is subsumed under our more general approach. Trecarichi et al. [9] showed that the OK system can support both centralised and decentralised architectures in this way. Their experimental results in an emergency response application, demonstrate similar outcomes and comparable performance under the ideal assumptions for both cases.

For any given situation, an interaction model can be chosen which operates at a specific point on the centralised/decentralised spectrum to suit the current requirements – for example, we may choose more or less complex negotiations which yield a more or less efficient solution. These interaction models may be even be run simultaneously (between disjoint sets of peers). In practice, we might expect to see a hierarchical model evolve, based on geographical and organisational boundaries – perhaps with tightly coupled protocols achieving high efficiency among the local machines, and more flexible protocols negotiating remote transfers based on more complex factors such as latency and cost.

In a real, production system there would also be a range of non-functional requirements to consider. Security, for example, may be considered simpler to guarantee in the centralised case where there is a single point of authority. Alternatively, the distributed model with restricted capabilities for individual agents may have security advantages (see [10] for a discussion of security attacks and proposed solutions in an LCC-based system). Similarly, a centralised system presents a single point of failure which would seem to be inherently less reliable than a more distributed system. However, we have not explored these issues in detail – they will depend heavily on the details of the implementation, and it would not be meaningful to compare our prototype to a highly-engineered production system. There are also many different approaches which could be taken to the implementation of an LCC-based system – OpenKnowledge and Okeilidh for example use completely different discovery services (Pastry vs OKBook), take different approaches to the coordination of the interactions (elected coordinator vs fully distributed messaging), and use different underlying protocols.

5.2 Policies and Complexity

One non-functional requirement which merits further discussion is the ease with which the interaction models can be created, understood, tested, and validated. LCC specifications have the advantage of a small syntax which is both lightweight and simple – this is easily understood both for designing new policies or modifying existing ones. However, operators of large data centres are unlikely to cede control of their resources to systems which may exhibit unexpected emergent behaviour, or unpredictable policy conflicts. Such problems are clearly possible, but they are mitigated by several factors:

- Only the "owner" of a machine may control which interaction models that peer is permitted to subscribe to.
- Within any particular interaction, all of the participating peers are following the same interaction model. This means that all of the participants will share a common goal, and a consistent policy.
- These interaction models can be model-checked to verify properties of their behaviour (see, for example [11] where the interaction models are translated into the μ-calculus for verification).
- The LCC language makes the interaction model very explicit, and supports tools for analysis and visualisation of the interactions. This is preferable to having interactions embedded implicitly in the implementation code.

Within an individual interaction, conflicts are unlikely to occur – all of the peers will be following the same (potentially formally verified) interaction model and they will have voluntarily subscribed to this model trusting both the IM author, and the other peers. Of course, it is possible to envisage several problem scenarios: certain peers may not follow their claimed role (due to error or malicious intent), peers may adopt a policy for a new interaction which negates the previous one, etc. However, these issues are no more problematic than in a conventional centralised solution, and indeed, the explicitness and isolation of the policies is likely to make such problems easier to detect and rectify.

5.3 Peer and IM Discovery

The discovery service is clearly a critical component of the proposed solution. In very simple scenarios, such as that described in section 3, it appears to be almost equivalent to a central controller, in that all of the participating peers perform most of their communication directly with this service. However:

- The discovery service is only required to match the initial subscriptions to the interaction roles. As the interactions become more complex, the proportion of interaction with the discovery service diminishes.
- The service is extremely lightweight and efficient, and can be easily replicated (the state is very simple).
- There are many different technologies which could be used to implement the service with varying characteristics - OpenKnowlegde and Okeilidh, for example, use completely different approaches.
- In a large, practical implementation, it is likely that the discovery service would be hierarchical. Local matches would be found quickly, and more remote matches would be forwarded to additional hubs. This appears to match the natural desire to solve negotiations locally and quickly when possible.

On a local scale, and with a comparatively low traffic rate, it it is possible to envisage a broadcast-based solution with no central service at all, but the discovery service approach is consistent with most agent-based systems which require some mechanism for participating peers to locate one another before the interactions can take place.

5.4 Federation

Apart from the issues of performance and scale, federation has been one of the main motivating factors for our approach – different organisations may have different services to offer, different requirements, and different restrictions on the information that they are willing to share and the peers with whom they are willing to interact.

A key feature of the OpenKnowledge approach is that there is no requirement for global agreement, either on protocols or ontologies – a group of peers can participate in an interaction simply by agreeing on the terms (constraints) used in that IM, and following the protocol that it provides. If a peer does not understand the terminology used by a particular IM, or is not willing to share the information that it requires, then it cannot participate in that particular interaction - but it is free to participate in other interactions, or even propose its own alternative model in which others can be invited to participate.

In particular, status or monitoring information is never shared or synchronised explicitly between the peers. A particular interaction may require knowledge of some specific parameter (say the network bandwidth available to the VM) in which case the interaction model will specify that participants must implement a constraint to determine this value, and be willing to share it.

Within one organisation, it is likely that the interaction models and OKCs will be curated centrally. In this case they will have been designed to interoperate, and the vocabulary used will be consistent. In a federated environment, IMs and corresponding constraints may be proposed by multiple organisations and it is necessary to understand how these relate, which are equivalent, and how we may map between them. For example, if two different interaction models both require us to provide the network bandwidth, do they use the same units? As we have already stressed, there is no requirement for a global agreement on such matters, but the OpenKnowledge ontology matching service proposes a solution to this problem by aggregating a number of techniques for ontological matching.

One other issue for large, federated systems is the potential performance degradation due to the larger number of potential participants, and the increased latency of interactions. This is an issue for both the discovery service, and the execution of the interactions themselves. In practice however, we would expect to see different kinds of interactions between local peers and remote ones – for example, we might expect a good deal of activity between local machines as they negotiate an efficient placement. But at some point, the local cluster may become overloaded and there may be an inter-site negotiation to transfer a block of machines into the cloud. This leads quite naturally to a hierarchical organisation of discovery services, and interaction models, suited to the locality of the communications.

The example in section 4 shows a basic workflow for this "cloud bursting" scenario. In practice, such an application is likely to require a more complex model: there may be significant dependencies between the virtual machines and the services running on them – for example, it may be necessary to make changes in firewall configurations. This clearly increases the complexity of the interaction models, although the basic principles remain unchanged. We are also presuming that it is possible to run agents on the physical machines within the data centre. This is clearly not the case for current commercial services such as EC2, for example. However, we could envisage running proxy servers representing such a service and managing the associated resources.

5.5 Configuration Patterns

In creating interaction models for various scenarios, it has become clear that there are some common interaction patterns. Perhaps the most obvious of these is the client-server pattern described in section 4 – there are many cases where a "service" of some sort needs to be moved, and this requires corresponding modifications to the client. Another pattern occurs when there is contention over some resource, and some further action is required to free up the necessary resource: for example, assume that we require a physical host for a big virtual machine which needs the full resources of one PM. If all of the available PMs are running small VMs, we may need to move one of the small VMs to create space for the large one.

Such patterns can be viewed in the same way as software design patterns and used to aid and clarify the manual construction of interaction models. But it may also be possible to incorporate these into the tooling, by providing users with a higher level view of the system and allowing them to specify and compose such patterns explicitly. We are currently investigating the use of automated planning techniques [12,13] to compose such (parameterised) patterns automatically for managing complex workflows .

6 Related Work

There is a considerable amount of existing work on load balancing of virtual infrastructures. This usually involves a central service which collects monitoring data from the physical and virtual machines, computes any necessary re-allocation, and orchestrates the appropriate migrations. Analysis of "hotspots"[14] or SLA violations[15] is necessary to plan a new allocation, but despite some success with statistical machine learning[16,17], effective prediction of *future* performance seems unrealistic in many cases. This type of centralised control limits the degree to which it is possible to exploit the resources of a more federated service[18,19]. Managing the interactions of imperative control algorithms in a centralised system is also a problem[20].

VMWare is a popular provider of commercial management infrastructure for virtual datacentres. The VMWare *vSphere Distributed Resource Scheduler* (DRS) product allows the user to specify rules and policies to prioritise how resources are allocated to virtual machines. DRS[13]"continuously monitors utilisation across resource pools and intelligently aligns resources with business needs" . *vSphere Distributed Power Management* (DPM) allows workloads to be consolidated onto fewer servers so that the rest can be powered-down to reduce power consumption. Citrix Essentials[14] and Virtual Iron "Live capacity" [15] are other commercial products offering similar functionality, and LBVM[16] is an open-source product based on Red Hat Cluster Suite. However, all of these products use a centralised management model.

Likewise, tools for managing the workflow of configuration changes are also standard practice, but based on a centralised execution model; IBM Tivoli Provisioning Manager[17] (TPM) is a common commercial solution with a workflow executed from a central control server. ControlTier[18], is a popular alternative which orchestrates the execution of the workflow by sending a secure shell remote command to the target node.

The term *autonomic computing*[21] was popularised by IBM in 2001 to describe computing systems which are *self-configuring*, *self-healing*, *self-optimising*, and

[13] http://www.vmware.com/pdf/vmware_drs_wp.pdf

[14] https://h20392.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=HPE4XSE

[15] http://www.storageengineers.com/pdf_virtualiron/Evaluation_Guide_0107.pdf

[16] http://lbvm.sourceforge.net/

[17] http://www.ibm.com/software/tivoli/products/prov-mgr/

[18] http://controltier.org/

self-protecting (*self-**). Kephart and Walsh[22] noted that agent-based technologies are a natural fit for implementing this type of system, and this has led to the development of market-based resource management systems such as [23]. Several people have applied these techniques to virtual machine management: Xing[24] describes a system where "each virtual machine can make its own decision when and where to live migrate itself between the physical nodes" - for example, two VMs may notice that the applications running on them are communicating frequently, and the VMs may decide that they should attempt to migrate so that they are physically closer. Spata and Rinaudo[25] describe a FIPA-compliant system with very similar objectives to our own which is intended to load-balance VMs across a cluster. However, we are not aware of any other systems which are driven directly from a declarative specification of the interaction model.

7 Conclusions and Future Work

We have demonstrated that an agent-based approach using LCC interaction models is a viable technique for negotiating both virtual machine placement and execution of the associated workflows. This provides a framework for supporting arbitrary interaction models which are capable of implementing a wide range of policies and approaches, suitable for different situations. The interaction models clearly expose the protocols which can be easily verified, shared, composed and modified. A particular strength of this approach is the lack of any requirement for prior agreement on protocols or ontologies, which makes it a particularly effective solution in federated environments.

We are currently investigating more complex workflows, and particularly the automatic generation of interaction models using automated planning techniques.

Acknowledgments. This research has been partly supported by a grant from HP Labs Innovation Research Program Award.

References

1. Anderson, P., Bijani, S., Vichos, A.: Multi-agent negotiation of virtual machine migration using the lightweight coordination calculus. In: Proceedings of the 6th International KES Conference on Agents and Multi-agent Systems – Technologies and Applications (2012)
2. Walton, C., Robertson, D.: Flexible multi-agent protocols. Technical report, University of Edinburgh (2002)
3. Robertson, D.: A lightweight coordination calculus for agent systems. In: Leite, J., Omicini, A., Torroni, P., Yolum, p. (eds.) DALT 2004. LNCS (LNAI), vol. 3476, pp. 183–197. Springer, Heidelberg (2005)
4. Pinninck, A.P.D., Kotoulas, S., Siebes, R.: The OpenKnowledge kernel. In: Proceedings of the IX CESSE Conference (2007)

5. Siebes, R., Dupplaw, D., Kotoulas, S., Perreau de Pinninck, A., van Harmelen, F., Robertson, D.: The OpenKnowledge System: an interaction-centered approach to knowledge sharing. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part I. LNCS, vol. 4803, pp. 381–390. Springer, Heidelberg (2007)
6. Kotoulas, S., Siebes, R.: Adaptive routing in structured peer-to-peer overlays. In: 3rd Intl. IEEE workshop on Collaborative Service-oriented P2P Information Systems (COPS workshop at WETICE 2007), Paris, France. IEEE Computer Society Press, Los Alamitos (2007)
7. Anadiotis, G., Kotoulas, S., Lausen, H., Siebes, R.: Massively scalable web service discovery. In: International Conference on Advanced Information Networking and Applications, AINA 2009, pp. 394–402. IEEE (2009)
8. Li, J.: Agent-based management of virtual machines for cloud infrastructure. Master's thesis, School of Informatics, University of Edinburgh (2011)
9. Trecarichi, G., Rizzi, V., Vaccari, L., Marchese, M., Besana, P.: Openknowledge at work: exploring centralized and decentralized information gathering in emergency contexts (2009)
10. Bijani, S., Robertson, D.: A review of attacks and security approaches in open multi-agent systems. In: Artificial Intelligence Review. Springer (2012)
11. Osman, N., Robertson, D., Walton, C.D.: Dynamic model checking for multi-agent systems. In: Baldoni, M., Endriss, U. (eds.) DALT 2006. LNCS (LNAI), vol. 4327, pp. 43–60. Springer, Heidelberg (2006)
12. Herry, H., Anderson, P., Wickler, G.: Automated planning for configuration changes. In: Proceedings of the 2011 LISA Conference. Usenix Association (2011)
13. Herry, H., Anderson, P.: Planning with global constraints for computing infrastructure reconfiguration. In: The AAAI 2012 Workshop on Problem Solving Using Classical Planners, CP4PS 2012 (2012)
14. Wood, T., Shenoy, P., Venkataramani, A., Yousif, M.: Black-box and gray-box strategies for virtual machine migration. In: Proceedings of the 4th Usenix Symposium on Networked Systems Design and Implementation, Usenix (April 2007)
15. Bobroff, N., Kochut, A., Beaty, K.: Dynamic placement of virtual machines for managing SLA violations. In: 10th IFIP/IEEE International Symposium on Integrated Network Management, IM 2007, Yearly 21-25, pp. 119–128 (2007)
16. Bodk, P., Griffith, R., Sutton, C., Fox, A., Jordan, M., Patterson, D.: Statistical machine learning makes automatic control practical for internet datacenters. In: Proceedings of Workshop on Hot Topics in Cloud Computing, HotCloud (2009)
17. Liu, X.: Prediction of resource requirements for cloud computing. Master's thesis, School of informatics, University of Edinburgh (2010)
18. Ruth, P., Rhee, J., Xu, D., Kennell, R., Goasguen, S.: Autonomic live adaptation of virtual computational environments in a multi-domain infrastructure. In: IEEE International Conference on Autonomic Computing, ICAC 2006, pp. 5–14 (June 2006)
19. Grit, L., Irwin, D., Aydan, C.J.: Virtual machine hosting for networked clusters: Building the foundations for "autonomic" orchestration. In: Virtualization Technology in Distributed Computing, VTDC 2006, p. 7 (November 2006)
20. Schmid, M., Marinescu, D., Kroeger, R.: A Framework for Autonomic Performance Management of Virtual Machine-Based Services. In: Proceedings of the 15th Annual Workshop of the HP Software University Association (June 2008)
21. Murch, R.: Autonomic Computing, 1st edn. IBM Press (2004)

22. Kephart, J., Walsh, W.: An artificial intelligence perspective on autonomic computing policies. In: Proceedings of the Fifth IEEE International Workshop on Policies for Distributed Systems and Networks, POLICY 2004, pp. 3–12 (June 2004)
23. Schnizler, B., Neumann, D., Veit, D., Reinicke, M., Streitberger, W., Eymann, T., Freitag, F., Chao, I., Chacin, P.: Catnets deliverable 1.1: Theoretical and computational basis. Technical report, CatNet Project (2005)
24. Xing, L.: A self-management approach to service optimization and system integrity through multi-agent systems. Master's thesis, University of Oslo, Department of Informatics (May 2008)
25. Rinaudo, M.O.S.S.: Virtual machine migration through an intelligent mobile agents system for a cloud grid. Journal of Convergence Information Technology 6 (June 2011)

Modelling Evacuation at Crisis Situations by Petri Net-Based Supervision

František Čapkovič*

Institute of Informatics, Slovak Academy of Sciences
Dúbravská cesta 9, 845 07 Bratislava, Slovak Republic
Frantisek.Capkovic@savba.sk
http://www.ui.sav.sk/home/capkovic/capkhome.htm

Abstract. Place/transition Petri Nets (P/T PN) are utilized here to: (i) model modules of the endangered area (EA) having some attributes of simple agents; (ii) coerce the autonomous modules to a cooperation by means of supervision; (iii) model workflow of the evacuation process from EA. The approach is applied to analyzing the possibilities of evacuation EA being a part of a building and to finding the suitable free and safety escape routes. A supervisor is synthesized in order to force modules (agents) into a cooperation. The supervisor represents a desired goal of the cooperation and ensures its achievement. Thus, it makes the cooperation of modules possible. Its synthesis is based on the supervision methods known in DES (discrete-event systems) control theory. A simple case study of the building evacuation illustrates the proposed approach. The P/T PN based modelling the EA and the P/T PN-based evacuation workflow are proposed there. The differences between them as well as the possibilities of their mutual complementing are pointed out.

Keywords: Agent, cooperation, crisis situation, discrete-event systems, evacuation, modularity, place/transition Petri nets, supervision, Petri net-based workflow.

1 Introduction

Any crisis usually demands swift and effective decision-making. In general, regardless of the domain, crises usually share [16] four main features: uncertainty, rapid onset, imminent or realized severe losses, and a lack of controllability. It is difficult to know how to proceed in order to manage the crisis. The goal of the response is, first of all, to prevent or reduce negative consequences. Although the events of the crisis cannot be completely controlled, they can be partially influenced. To manage a crisis situation, it is necessary to perform a set of actions taken to exert control over the events of the crisis to minimize losses. These facts should be multiplied especially in cases when the human life is in a danger. In such a case the evacuation of people from EA has the highest priority. In recent

[*] Partially supported by the Slovak Grant Agency for Science (VEGA) under grants # 2/0075/09 and # 2/0039/13.

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 143–170, 2013.

years, there is an increasing number of communication systems and intelligent tools [10] supporting first responders and evacuees in the tasks of communication, decision-making, information exchange and coordination in emergency and crisis evacuation. To manage the evacuation process from EA in a crisis situation (particularly in case of a panic) flexible strategies for safety solving the problem are required. Especially, it is necessary to find safety and free escape routes [14]. Of course, immediate information from the real system (the real scenario) is valuable on this way. A network of distributed sensors yields necessary information.

However, a model representing the layout (schema) of EA and main aspects of the evacuation dynamics has also an importance. By means of such a model different partial problems can be analyzed off-line in the process of simulation. The main aim of this paper is to point out that the P/T PN-based model of EA as well as the P/T PN-based evacuation workflow are useful on that way. It will be shown that a suitable supervisor synthesized also by means of P/T PN will guarantee the successful evacuation. The P/T PN-based EA model yields a possibility to describe and analyze the sequences of steps at the escape from the EA in a simulation process. The model describing EA structure can be built from elementary modules. These modules can be named agents (e.g. in the sense of [24]), at least agents without own goal (passive agents). Actually, we have not in mind any standardized kind of agents, but only autonomous modules equipped by sensors and able to cooperate when an authority (the supervisor) asks this. Thus, by the term agent a material entity (elementary module) able to cooperate (at least with a supervisor) is deemed here. The enforced cooperation of such agents favourably affects the evacuation dynamics. In the case study introduced below, such agents are represented by doorways equipped by sensors. Because such system has the character of DES, it can be modelled by P/T PN. There exist methods in DES control theory how to synthesize the supervisors [11, 13, 12]. Supervisors will ensure the intended agent cooperation.

P/T PN-based model of EA helps us to find the evacuation workflow. Namely, it yields all feasible paths from a given initial state (the building occupancy before the evacuation of EA) to the desired terminal state (the empty EA). Then, the 'flow' of people during the evacuation process can be directed through the most suitable path(s). A workflow in general can be defined as a consecutive flow, i.e. a sequence of steps where any step immediately follows the precedent one and ends just before starting the subsequent step. Just P/T PN are very suitable to model such a procedure, even in analytical terms. Thus, the P/T PN model of the workflow affords the most suitable (from the safety point of view) escape route(s) from EA and makes possible to analyze them in the process of simulation.

It is necessary to say that the approach proposed here is off-line and no on-line approach is studied. On the other hand, the models can be useful at prevention crisis situations in some areas (e.g. buildings) already at their design. Namely, as to the evacuation (especially finding the escape paths) the models allow to analyze the areas in advance, i.e. when accidents in future are only expected. As to modelling the workflow, there exist approaches using PN - see e.g. [17–20, 23, 22, 8]. In this paper, we will use P/T PN for modelling EA structure and evacuation dynamics as well as for the supervisor synthesis. The P/T PN-based modelling the evacuation workflow will be utilized too. Moreover, extending the workflow modelling by adding the supervision is presented and illustrated. The P/T PN graphical tool will be used for testing the properties and dynamical behaviour of the particular agents. However, because existing P/T PN graphical tools do not allow the supervisor synthesis, computing by MATLAB will be performed on that way.

This paper represents the extending of the paper [7] presented on the conference KES AMSTA 2012.

2 Problem Statement and Preliminaries

To gather knowledge to be used at agent-based modelling EA and at solving the evacuation problem, let us introduce concisely necessary accomplishments concerning the P/T PN, the modular structure, the cooperation of modules (agents) by supervision, the supervisor synthesis and the PN-based workflow. On that way, especially the earlier author's results concerning the agents cooperation and negotiation [1–7] can be utilized as well as the theory of supervision [11–13].

2.1 Place/Transition Petri Nets

The place/transition Petri nets (P/T PN) [15] are used here in the process of modelling the agents, at the agent cooperation and at the process of supervision. As to the structure P/T PN are bipartite directed graphs

$$< P, T, F, G > ; \quad P \cap T = \emptyset, \quad F \cap G = \emptyset \tag{1}$$

with P, T, F, G being, respectively, the set of places, the set of transitions, the set of directed arcs from places to transitions and the set of directed arcs from transitions to places. Moreover, P/T PN have their dynamics

$$< X, U, \delta, \mathbf{x}_0 > ; \quad X \cap U = \emptyset, \quad \delta : X \times U \rightarrow X \tag{2}$$

with X, U, δ, $\mathbf{x}_0$ being, respectively, the set of states (marking the places), the set of discrete events (states of transitions), the transition function and the initial state vector. The formal expression of the transition function δ in (2) can be rewritten into the form of the linear discrete system as follows

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B}.\mathbf{u}_k \ , \quad k = 0, ..., N \tag{3}$$

$$\mathbf{B} = \mathbf{G}^T - \mathbf{F} \tag{4}$$

$$\mathbf{F}.\mathbf{u}_k \leq \mathbf{x}_k \tag{5}$$

where k is the discrete step of the dynamics development; $\mathbf{x}_k = (\sigma^k_{p_1}, ..., \sigma^k_{p_n})^T$ is the n-dimensional state vector; $\sigma^k_{p_i} \in \{0, 1, ..., c_{p_i}\}$, $i = 1, ..., n$ express the states of atomic activities by 0 (passivity) or by $0 < \sigma_{p_i} \leq c_{p_i}$ (activity); c_{p_i} is the capacity of p_i; $\mathbf{u}_k = (\gamma^k_{t_1}, ..., \gamma^k_{t_m})^T$ is the m-dimensional control vector; its components $\gamma^k_{t_j} \in \{0, 1\}$, $j = 1, ..., m$ represent occurring of elementary discrete events (e.g. starting or ending the activities, failures, etc.) by 1 (presence of the corresponding discrete event) or by 0 (absence of the event); $\mathbf{B}$, $\mathbf{F}$, $\mathbf{G}$ are incidence matrices of integers; $\mathbf{F} = \{f_{ij}\}_{n \times m}$, $f_{ij} \in \{0, M_{f_{ij}}\}$, expresses the causal relations among the states (as causes) and the discrete events occurring during the DES (discrete-event systems) operation (as consequences) by 0 (nonexistence of the relation) or by $M_{f_{ij}} > 0$ (existence and multiplicity of the relation); $\mathbf{G} = \{g_{ij}\}_{m \times n}$, $g_{ij} \in \{0, M_{g_{ij}}\}$, expresses analogically the causal relations among the discrete events (as causes) and the DES states (as consequences); $\mathbf{B}$ is given according to (4); $(.)^T$ symbolizes the matrix or vector transposition. From (3)-(5) it is clear that the structure and the step-by-step dynamics development are expressed here in a uniform way. Just such an exact mathematical expression of P/T PN, in contrast to high-level PN, yields the possibility to deal with the PN models in analytical terms.

2.2 Modular Structure of P/T PN-Based Models

Having the P/T PN models of particular modules (agents), we can think about building a suitable global structure (an aggregate) from such modules. In case of N_A autonomous agents the structural matrices of the global model can have the following form

$$\mathbf{F} = \begin{pmatrix} \mathbf{F}_1 & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & | \, \mathbf{F}_{c_1} \\ \mathbf{0} & \mathbf{F}_2 & \dots & \mathbf{0} & \mathbf{0} & | \, \mathbf{F}_{c_2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & | \, \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{F}_{N_A-1} & \mathbf{0} & | \, \mathbf{F}_{c_{N_A-1}} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{F}_{N_A} & | \, \mathbf{F}_{c_{N_A}} \end{pmatrix} = \big(\text{blockdiag}(\mathbf{F}_i)_{i=1,N_A} | \, \mathbf{F}_c\big) \quad (6)$$

$$\mathbf{G} = \begin{pmatrix} \mathbf{G}_1 & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{G}_{N_A-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{G}_{N_A} \\ \text{----} & \text{----} & \text{----} & \text{----------} & \text{----} \\ \mathbf{G}_{c_1} & \mathbf{G}_{c_2} & \dots & \mathbf{G}_{c_{N_A-1}} & \mathbf{G}_{c_{N_A}} \end{pmatrix} = \begin{pmatrix} \text{blockdiag}(\mathbf{G}_i)_{i=1,N_A} \\ \text{--------------------} \\ \mathbf{G}_d \end{pmatrix} \quad (7)$$

where $\mathbf{F}_c = (\mathbf{F}^T_{c_1}, \mathbf{F}^T_{c_2}, ..., \mathbf{F}^T_{c_{N_A}})^T$, $\mathbf{G}_c = (\mathbf{G}_{c_1}, \mathbf{G}_{c_2}, ..., \mathbf{G}_{c_{N_A}})$. Here, $\mathbf{F}_i$, $\mathbf{G}_i$, $i = 1, ..., N_A$, represent the parameters of the PN-based model of the agent A_i. $\mathbf{F}_c$, $\mathbf{G}_c$ represent the structure of the interface between the cooperating agents. As we can see from the incidence matrices (6), (7), the interface is built by means of extending the number of PN transitions.

Analogically, the interface can be realized by means of the additional PN places as follows

$$\mathbf{F} = \left(\begin{array}{c} \text{blockdiag}(\mathbf{F}_i)_{i=1,N_A} \\ \text{--------------------} \\ \mathbf{F}_d \end{array} \right); \; \mathbf{G} = \left(\text{blockdiag}(\mathbf{G}_i)_{i=1,N_A} \mid \mathbf{G}_d \right) \tag{8}$$

where $\mathbf{F}_d = (\mathbf{F}_{d_1}, \mathbf{F}_{d_2}, ..., \mathbf{F}_{d_{N_A}})$; $\mathbf{G}_d = (\mathbf{G}_{d_1}^T, \mathbf{G}_{d_2}^T, ..., \mathbf{G}_{d_{N_A}}^T)^T$. $\mathbf{F}_i$, $\mathbf{G}_i$, $i = 1, ..., N_A$, represent the parameters of the PN-based model of the agent A_i, and $\mathbf{F}_d$, $\mathbf{G}_d$ represent the structure of the interface between the cooperating agents.

Finally, combining both previous models we can obtain the kernel of interface in the form of the PN subnet (another agent or even an agent system) containing additional places and additional transitions. Its structure is given by the matrix $\mathbf{F}_{d \leftrightarrow c}$ and the matrix $\mathbf{G}_{c \leftrightarrow d}$. The row and the column consisting of corresponding blocks model the interconnections of the kernel with the autonomous agents. Hence,

$$\mathbf{F} = \left(\begin{array}{c|c} \text{blockdiag}(\mathbf{F}_i)_{i=1,N_A} & \mathbf{F}_c \\ \text{------------------} & \text{--------} \\ \mathbf{F}_d & \mathbf{F}_{d \leftrightarrow c} \end{array} \right); \mathbf{G} = \left(\begin{array}{c|c} \text{blockdiag}(\mathbf{G}_i)_{i=1,N_A} & \mathbf{G}_d \\ \text{------------------} & \text{--------} \\ \mathbf{G}_c & \mathbf{G}_{c \leftrightarrow d} \end{array} \right) \tag{9}$$

where $\mathbf{F}$, $\mathbf{G}$ acquire a special structure. Each of them has the big diagonal block describing the structure of N_A autonomous agents and the specific part in the form of the letter L turned to the left over the vertical axe. $\mathbf{F}_{d \leftrightarrow c}$, $\mathbf{G}_{c \leftrightarrow d}$ are situated, respectively, on their diagonals just in the breakage of the turned L. However, because below we will synthesize only the supervisors in the form of the interface based on the P/T PN places, it is sufficient to consider the incidence matrices in the form (8) only. The additional places and their interconnections with the autonomous agents will be obtained by means of the supervision. The supervisor(s) will be synthesized by means of the methods described in the section 3.

2.3 P/T PN-Based Modelling the Workflow

Workflow is defined in [21] as consisting of three parts: (i) the process definition - it is a description of the process itself. It specifies which steps are required and in what order they should be executed. It is also known as the routing definition or procedure or workflow script; (ii) the resource classification - it is a classification of the resources to be used. It consists of resource (a participant, actor, user, agent). A resource can execute certain tasks for certain cases. It can be human and/or non-human (a technical device - e.g. like a sensor, a 'material agent', etc.). A resource class is a set of resources with similar characteristic(s) and specialities; (iii) the resource management rules - they express how to map work onto the resources.

Petri nets represent also a popular technique [17–20] for modelling workflows. It seems to be natural. Namely, when somebody models a workflow, he tends to draw nodes representing tasks or activities. Then, he draws arrows between the nodes representing sequencing of activities. The resulting diagrams look like PNs, and so PNs seem to be a natural technique for modelling workflows. The following arguments are often used [9] to support this: (i) Petri nets are also the graphical tool; (ii) they have a formal semantics; (iii) they can express most of the desirable routing constructs; (iv) there is an abundance of analysis techniques for proving properties about them; and finally (v) they are 'vendor-independent', i.e. sufficiently general to be independent on the particular applications. Although not all of these arguments [9] refer to the workflow modelling, it can be said that PNs are very useful at the workflow modelling. It will be clear from the case study introduced below, where also an original trial with supervised evacuation wokflow will be performed. Moreover, the PN-based workflow theory continually evolves - see e.g. [8, 22, 23] and especially [25].

In this paper the P/T PN model of the evacuation workflow is used. Moreover, its combination with the supervisor(s) will be analyzed and tested. The role of the supervisor(s) consists in forcing such a policy to the workflow primitives (primarily OR-split) given in the Fig. 1 in order to achieve the smallest number of states (and steps as well) of the evacuation process.

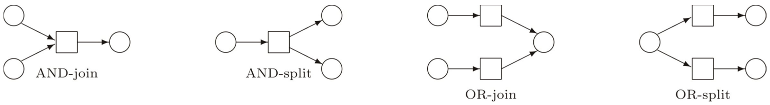


Fig. 1. The P/T PN-based workflow primitives

Concisely said, in our case the evacuation workflow is the P/T PN model of the evacuation process (even in analytical terms) and contains all information about what has to be done and in which order. The sequential routing describes sequentially executed tasks where one task is followed by the next task. The parallel routing describes a situations where two or more tasks are executed at the same time. We will use the parallel routing where the transitions are understood to be fired simultaneously. However, because usually there exist also OR-split primitives in the workflow the supervisor has to ensure desired priorities between the transitions in question.

3 Cooperation of Agents

In order to ensure the cooperation of the autonomous modules (agents) an additional entity has to be used. The entity has to represent the goal prescribed for the activity of the group of agents and to ensure its achievement. It means, that the entity forces the agents into the cooperation and controls (supervises) its attainment. Just the supervisor operates in the role of such an entity. The term supervisor is used in DES control theory. It epitomizes an analogy to the term feedback controller (regulator) used in classical control theory. Although it is impossible to synthesize the feedback controller for DES, it is possible to synthesize a supervisor in virtue of a prescribed set of conditions. Therefore, in order to encompass the agent cooperation process the supervisor has to be synthesized by virtue of prescribed obvious conditions describing the goal of the group of agents. Thus, the model of the EA evacuation process can be built up. Then, the model can be used in the simulation process to analyze and test the possibilities how to successfully and effectively carry the real EA evacuation out. Two methods of the supervisor synthesis are described here.

3.1 Supervision and the Supervisor Synthesis

The classical feedback control typical for continuous systems cannot be applied to DES. However, some kinds of supervision used in the DES control theory can be applied in order to affect the system behaviour. In P/T PN models of DES the supervision can be computed - i.e. the supervisors can be synthesized - in analytical terms. Here, two kinds of supervision are utilized, namely: (i) the supervision based on P-invariants of P/T PN utilizing only the state vector $\mathbf{x}$; (ii) the extended supervision based on the state vector $\mathbf{x}$, control vector $\mathbf{u}$ and Parikh's vector $\mathbf{v}$.

The former approach utilizes definition of the P-invariant [15] as the vector $\mathbf{w}$ satisfying the relation $\mathbf{w}^T.\mathbf{B} = \mathbf{0}$ or in case of more P-invariants as $\mathbf{W}^T.\mathbf{B} = \mathbf{0}$, where $\mathbf{W}$ is the matrix of invariants. This approach operates only with the state vector $\mathbf{x}$.

The latter approach is typical by usage of the Parikh's vector $\mathbf{v}$ defined by means of developing the system (3) as follows

$$\mathbf{x}_1 = \mathbf{x}_0 + \mathbf{B}.\mathbf{u}_0 \tag{10}$$

$$\mathbf{x}_2 = \mathbf{x}_1 + \mathbf{B}.\mathbf{u}_1 = \mathbf{x}_0 + \mathbf{B}.(\mathbf{u}_0 + \mathbf{u}_1) \tag{11}$$

$$\dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \tag{12}$$

$$\mathbf{x}_k = \mathbf{x}_0 + \mathbf{B}.(\mathbf{u}_0 + \mathbf{u}_1 + \cdots + \mathbf{u}_{k-1}) = \mathbf{x}_0 + \mathbf{B}.\mathbf{v} \tag{13}$$

Just the vector $\mathbf{v} = (\mathbf{u}_0 + \mathbf{u}_1 + \cdots + \mathbf{u}_{k-1})$ is the Parikh's vector. It gives us information about how many times the particular transitions are fired during the development of the system dynamics from the initial state $\mathbf{x}_0$ to the final state $\mathbf{x}_k$.

Supervisor Synthesis Based on P-invariants. As a matter of fact, P/T PN P-invariants (place invariants) in general are vectors $\mathbf{w}$ having the property that

$$\mathbf{w}^T.\mathbf{x}_k = \mathbf{w}^T.\mathbf{x}_0 \tag{14}$$

for each state vector $\mathbf{x}_k$ reachable from the initial state $\mathbf{x}_0$. Above introduced definition $\mathbf{w}^T.\mathbf{B} = \mathbf{0}$ [15] arises from (14) when it is modified by means of the P/T PN model (3) as follows. Namely, when (13) is multiplied by $\mathbf{w}^T$ from the left, it holds that

$$\mathbf{w}^T.\mathbf{x}_k = \mathbf{w}^T.\mathbf{x}_0 + \mathbf{w}^T.\mathbf{B}.\mathbf{v} \tag{15}$$

Because of (14) the relation $\mathbf{w}^T.\mathbf{B}.\mathbf{v} = \mathbf{0}$ has to be valid. However, because the Parikh's vector $\mathbf{v}$ depends on independent discrete events, we must not limit this vector anyway. Consequently, it has to hold that

$$\mathbf{w}^T.\mathbf{B} = \mathbf{0} \tag{16}$$

Therefore, just this relation is taken in [15] for the definition of the P-invariant of P/T PN. Consider now, that a set of constraints imposed on P/T PN has the form of n_s inequalities in the vector form as follows

$$\mathbf{L}_p.\mathbf{x} \leq \mathbf{b} \tag{17}$$

where $\mathbf{L}_p$ is $(n_s \times n)$-dimensional matrix of integer constants and $\mathbf{b}$ is the n_s-dimensional vector of integer constants representing a limit value for a linear combination of some selected entries of the state vector $\mathbf{x}$ described on the left side of the (17). The constraints represent the prescribed conditions, fulfilling of which will guarantee a desired behaviour of the system. Namely, a properly chosen set of constraints can affect the behaviour of the system by means of a corresponding supervisor. Note, that the inequalities (17) can be transformed into following equations

$$\mathbf{L}_p.\mathbf{x} + \mathbf{I}_s.\mathbf{x}_s = \mathbf{b} \tag{18}$$

by introducing the auxiliary variables (slacks) $\mathbf{x}_s$.

Then, the relation (16), defining the P-invariants, can be replaced by the following one

$$(\mathbf{L}_p \;\; \mathbf{I}_s).\begin{pmatrix} \mathbf{B} \\ \mathbf{B}_s \end{pmatrix} = \mathbf{0} \tag{19}$$

where $\mathbf{I}_s$ is $(n_s \times n_s)$-dimensional identity matrix and $\mathbf{B}_s$ is the structural matrix of the supervisor (till now unknown) to be found. Hence,

$$\mathbf{L}_p.\mathbf{B} + \mathbf{B}_s = \mathbf{0}; \quad \mathbf{B}_s = -\mathbf{L}_p.\mathbf{B} \tag{20}$$

where $\mathbf{B}_s = \mathbf{G}_s^T - \mathbf{F}_s$. With respect to (18)

$$(\mathbf{L}_p \;\; \mathbf{I}_s).\begin{pmatrix} \mathbf{x}_0 \\ {}^s\mathbf{x}_0 \end{pmatrix} = \mathbf{b}\ ; \quad {}^s\mathbf{x}_0 = \mathbf{b} - \mathbf{L}_p.\mathbf{x}_0 \tag{21}$$

where ${}^s\mathbf{x}_0$ is the initial state of the supervisor. In such a way the structure of the supervisor and its initial state were obtained. They make possible to build the augmented system (the original system supervised by the supervisor) as follows

$$\mathbf{x}_a = \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \end{pmatrix}; \;\; \mathbf{F}_a = \begin{pmatrix} \mathbf{F} \\ \mathbf{F}_s \end{pmatrix}; \;\; \mathbf{G}_a^T = \begin{pmatrix} \mathbf{G}^T \\ \mathbf{G}_s^T \end{pmatrix} \tag{22}$$

The matrices $\mathbf{F}_s$, $\mathbf{G}_s$ in (22) corresponds to the matrices $\mathbf{F}_d$, $\mathbf{G}_d$ indicated in (8).

Supervisor Synthesis Based on the Extended Method. Although the P-invariant based method for the supervisor synthesis is suitable for an enough wide class of applications, it can be extended. Namely, constraints can be imposed not only on the state vector $\mathbf{x}$ like above, but also on the control vector $\mathbf{u}$ and even on the Parikh's vector $\mathbf{v}$. Namely, a correlation among P/T PN transitions and a correlation among the entries of the Parikh's vector can also be introduced into the conditions for the supervisor synthesis. This fact extremely broadens the class of applications. Therefore, the general linear constraints are extended [13] into the form

$$\mathbf{L}_p.\mathbf{x} + \mathbf{L}_t.\mathbf{u} + \mathbf{L}_v.\mathbf{v} \le \mathbf{b} \tag{23}$$

where $\mathbf{L}_p$, $\mathbf{L}_t$, $\mathbf{L}_v$ are, respectively, $(n_s \times n)-$, $(n_s \times m)-$, $(n_s \times m)-$dimensional matrices. Moreover, such an approach makes the variability of constraints possible. The variability consists in the possibility to use: (i) only the individual constraints represented either by means of $\mathbf{L}_p$ or $\mathbf{L}_t$ or $\mathbf{L}_v$; (ii) the constraints represented by means of matrix pairs - either $\mathbf{L}_p$ and $\mathbf{L}_t$ or $\mathbf{L}_p$ and $\mathbf{L}_v$ or $\mathbf{L}_t$ and $\mathbf{L}_v$; (iii) all of the three matrices like in (23).

It was proved in [13] that when $\mathbf{b} - \mathbf{L}_p.\mathbf{x} \ge \mathbf{0}$ holds, the supervisor with the following structure and initial state

$$\mathbf{F}_s = \max(\mathbf{0},\, \mathbf{L}_p.\mathbf{B} + \mathbf{L}_v,\, \mathbf{L}_t) \tag{24}$$

$$\mathbf{G}_s^T = \max(\mathbf{0},\, \mathbf{L}_t - \max(\mathbf{0},\, \mathbf{L}_p.\mathbf{B} + \mathbf{L}_v)) - \min(\mathbf{0},\, \mathbf{L}_p.\mathbf{B} + \mathbf{L}_v) \tag{25}$$

$${}^s\mathbf{x}_0 = \mathbf{b} - \mathbf{L}_p.\mathbf{x}_0 - \mathbf{L}_v.\mathbf{v}_0 \tag{26}$$

guarantees that the constraints are verified for the states resulting from the initial state. Here, the max(.) is the maximum operator for matrices. However, the maximum is taken element by element. Namely, in general, for the matrices $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$ of the same dimensionality $(n \times m)$, the relation $\mathbf{Z} = \max(\mathbf{X},\, \mathbf{Y})$ holds, where maximum is performed element by element, i.e. $z_{ij} = \max(x_{ij},\, y_{ij})$, $i = 1, ..., n,\; j = 1, ..., m$. The minimum is computed analogically - i.e. the relation $\mathbf{Z} = \min(\mathbf{X},\, \mathbf{Y})$ holds, where minimum is performed element by element, i.e. $z_{ij} = \min(x_{ij},\, y_{ij})$, $i = 1, ..., n,\; j = 1, ..., m$.

4 Case Study - Modelling EA and Evacuation Workflow

Commonly used buildings in different daily activities (shopping centers, schools, dance halls, hotels, university colleges, etc.) that involve the meeting of a large number of people within a closed area are very sensitive as to crisis situations (e.g. at the occurrence of fire). Designers of such kinds of buildings usually attempt to maximize profits (the productivity of the available space) but at the same time they are oblivious of the safety devices. Therefore, it is necessary to consider a suitable planning for assuring people safety when an unusual situation (leading to unusual behavior of the crowd) occurs. The emergency evacuation due to the threat of fire is one of the most frequent causes of this kind of behavior. In such a situation, large number of people must be evacuated from a closed area with a relatively small number of fixed exits. It is very difficult task, especially in case when safety devices are not sufficient. A process of simulation can help us on that way, because it allows to specify different scenarios with people and environmental features. Having simulation results, then a more qualified evacuation can be performed. However, it depends on the technical equipment of buildings. For example, in case of a fire, not only the sensors indicating fire, but also a net of sensors indicating movement of people are needed in order to ensure the successful evacuation.

This case study is devoted only to modelling a segment of a common building, analysing the possibilities of the evacuation and off-line finding safety escape routes from EA in the process of simulation, not to on-line executing the real evacuation in the real time. To illustrate the application of the methods described above (in the sections 2 and 3) the following elementary steps will be shown in this case: (i) defining agents and creating their P/T PN-based models; (ii) creating the agent-based P/T PN model of the whole EA; (iii) synthesizing supervisors based on predefined constraints for the EA model; (iv) designing the P/T PN-based models of the evacuation workflow; (v) combining the P/T PN-based workflow with supervision and synthesizing the supervisors assuring the effective and safety escape routes.

As we can see in the EA scheme given in Fig. 2, it consists of two bands of rooms and the corridor between the bands. Such an architecture is typical e.g. for hotels, university colleges, etc. For simplicity, consider only two rooms in any band. Thus, there are four rooms R1 - R4 and the corridor R5 in EA. The corridor is accessible from any room. R2 and R4have, respectively, own emergency exits E2 and E3 while R5 has the main emergency exit E1. The escape routs depends on the doors, primarily on the sequence of their usage during the evacuation process. While doors D1, D3 and the emergency exits E1 - E3 are one-way (i.e. suitable only for the escape in the direction outside from the corresponding room), the doors D2-D4 are two-way (i.e. suitable for the escape in the direction outside from the room and contrariwise - e.g. in case when E1 does not operate or when it is crowded).

It is necessary to say, that in general, the EA may be not only a segment of a hotel floor but it also can be e.g. a flat, even, it can also be an arbitrary kind of EA with varied shapes of the rooms and various parts of the escape routs.

4.1 Modules (Agents) and Their P/T PN-Based Models

Let us consider doors equipped with sensors to be the modules. They can be understood to be autonomous 'material' agents without own goals, i.e. passive agents. They are equiped by sensors and they are principal parts of the EA as to the evacuation process. Namely, only these modules are flexible, because they can be either open or closed. This depends upon an actual situation in the evacuation process. The rest of buildings is usually fixed. Perhaps only windows represent an exception. Because other cognitive properties typical of standard agents are missing in case of our agents, it is not necessary to use any agent standards like e.g. the IEEE Computer Society standard FIPA or environments like e.g. the Java-based ABLE (Agent Building and Learning Environment) or

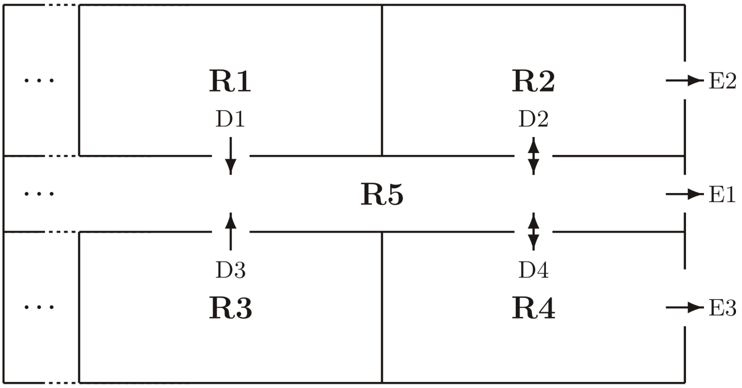


Fig. 2. The schema of the EA

frameworks like e.g. Java-based JADE (Java agent development framework), etc. Namely, the doors are not able to cooperate each other directly, but only by way of a coordinating entity. Such an entity is just the supervisor.

In order to model EA, it is necessary to distinguish two kinds of P/T PN-models of the doors [14], namely, the one-way door P/T PN-model and two-way door one. The differences between both kinds of the doors are evident not only from the schema of EA given in Fig. 2 but also from their PN-models given in Fig. 3. Nevertheless, in reality both kinds of the doors need not be mutually different from the physical point of view. Even, they can be the same. In a normal situation they are two-way, of course. However, it is clear that when a room with only one door is just evacuated, the door is used exclusively as one-way - the way is oriented outside from the room. It is nonsense to allow anybody to enter the room in such a situation. From the evacuation aspect it is ineligible.

The P/T PN model of the one-way door is given on the left in Fig. 3, while the P/T PN model of the two-way door is given on the right. In general, when windows will be used as a part of escape routes, a window can be understood to be the one-way door from a room to the space being outside the building. However, before using the windows as the part of the escape route, the safety of the escape has to be ensured by a fire-brigade - e.g. by a ladder or a working floor - in order to avoid another danger (the downfall from a height). As to the P/T PN places in the door models, the place p_1 models a room from which the door exits while the place p_4 models the room to which the door enters. The two-way doors can be entered and exited from both sides. The place p_2 represents the availability of the door. Of course, the door can be passed only in case when it is available. Finally, the place p_3 represents the process of passing the door. Firing the transitions t_1 - t_4 (if enabled) makes possible to perform the marking dynamics development. The incidence matrices of the P/T PN models of agents (according to (4)), i.e. $\mathbf{F}_1$, $\mathbf{G}_1^T$ for the one-way door and $\mathbf{F}_2$, $\mathbf{G}_2^T$ for the two-way door are the following

$$\mathbf{F}_1 = \begin{pmatrix} 1\,0 \\ 1\,0 \\ 0\,1 \\ 0\,0 \end{pmatrix} ; \ \mathbf{G}_1^T = \begin{pmatrix} 0\,0 \\ 0\,1 \\ 1\,0 \\ 0\,1 \end{pmatrix} ; \ \mathbf{F}_2 = \begin{pmatrix} 1\,0\,0\,0 \\ 1\,0\,0\,1 \\ 0\,1\,1\,0 \\ 0\,0\,0\,1 \end{pmatrix} ; \ \mathbf{G}_2^T = \begin{pmatrix} 0\,0\,1\,0 \\ 0\,1\,1\,0 \\ 1\,0\,0\,1 \\ 0\,1\,0\,0 \end{pmatrix}$$

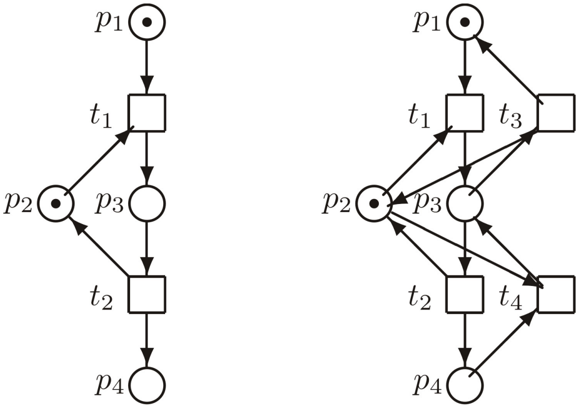
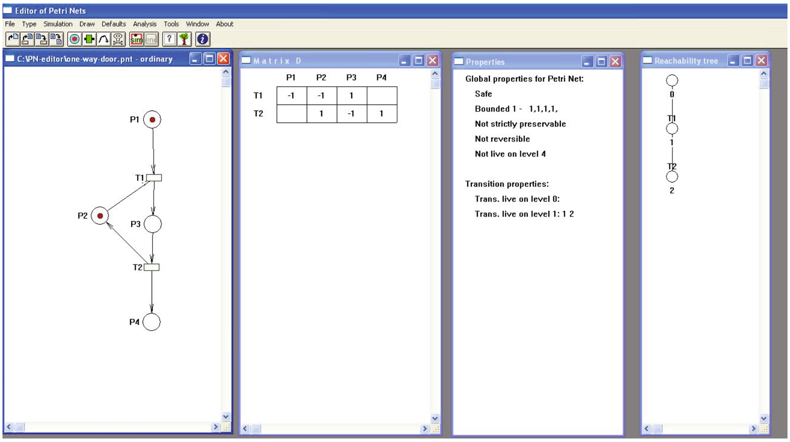


Fig. 3. The P/T PN-based models of the one-way door (on the left) and that of two-way door (on the right)

In other words, the meaning of the PN places is the following: p_1 - represents the evacuated room and the token inside expresses a state of the room - e.g. the presence of a person (or a group of persons) in this room; p_2 - models availability of the door - when there is a token inside it means that the door is available, in the opposite case the door is not available; p_3 - models passing the door - when it contains a token it means that a person just passes this door; p_4 - models an external room or the corridor. We can analyze the behaviour (marking dynamics) of both kinds of doors by means of the P/T PN graphical simulator. The analysis of the one-way door is given in Fig. 4 while the analysis of the two-way door is given in Fig. 5. In these figures the P/T PN-based models are on the left side, the structural matrices corresponding to $\mathbf{B}_i^T = (\mathbf{G}_i^T - \mathbf{F}_i)^T$, $i = 1, 2$, respectively, are next to them, the lists of the PN models properties are next to the matrices, and finally, the corresponding reachability trees (RT) of the P/T PN models are on the right side. The RT nodes represent feasible state vectors (reachable from the initial state $\mathbf{x}_0$), while RT edges contain information about transitions which have to be fired in order to evolve system dynamics from a state to its adjacent state(s). To distinguish repeated RT nodes from the original ones in the P/T PN graphical tool, the edges leading to the repeated nodes are displayed in the red color. However, the repeated RT nodes can be connected with their corresponding original ones. In such a way the RT turns to the reachability graph


Editor of Petri Nets
File Type Simulation Draw Defaults Analysis Tools Window About
C:\PN-editor\one-way-door.pnt - ordinary
P1
T1
P2
P3
T2
P4
Reachability tree
0
T1
1
T2
2

Matrix D

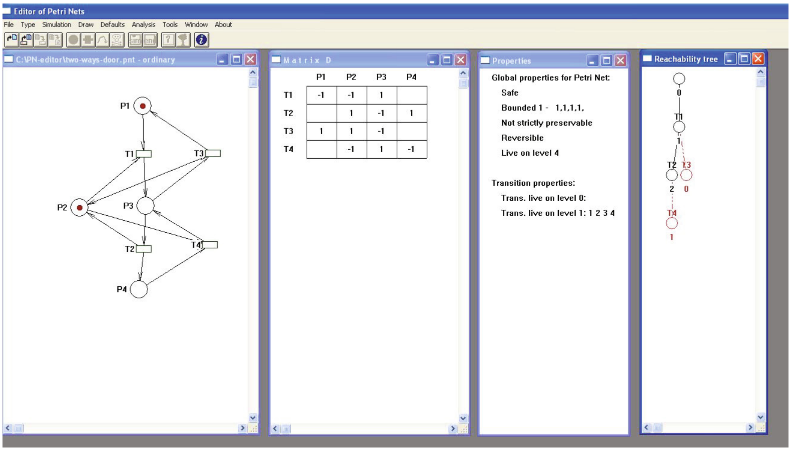
	P1	P2	P3	P4
T1	-1	-1	1	
T2		1	-1	1

Properties

Global properties for Petri Net:
Safe
Bounded 1 - 1,1,1,1,
Not strictly preservable
Not reversible
Not live on level 4

Transition properties:
Trans. live on level 0:
Trans. live on level 1: 1 2

Fig. 4. The analysis of the P/T PN-model of the one-way door by means of the graphical tool


Editor of Petri Nets
File Type Simulation Draw Defaults Analysis Tools Window About
C:\PN-editor\two-ways-door.pnt - ordinary
P1
T1
T3
P2
P3
T2
T4
P4
Reachability tree
0
T1
1
T2
T3
2
0
T4
1

Matrix D

	P1	P2	P3	P4
T1	-1	-1	1	
T2		1	-1	1
T3	1	1	-1	
T4		-1	1	-1

Properties

Global properties for Petri Net:
Safe
Bounded 1 - 1,1,1,1,
Not strictly preservable
Reversible
Live on level 4

Transition properties:
Trans. live on level 0:
Trans. live on level 1: 1 2 3 4

Fig. 5. The analysis of the P/T PN-model of the two-ways door by means of the graphical tool

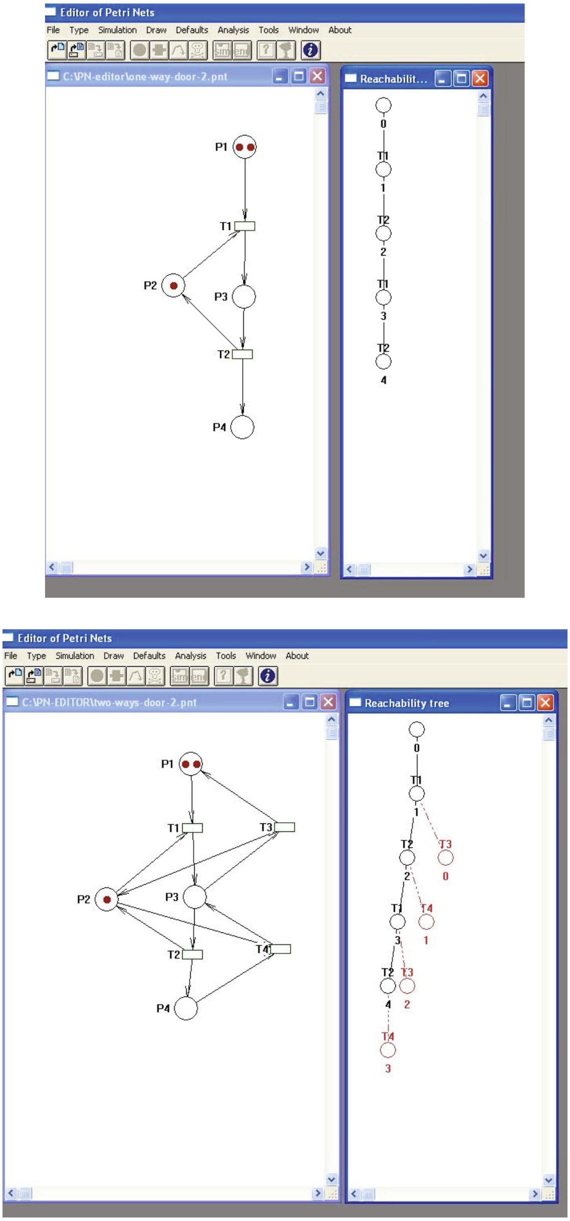

Editor of Petri Nets
File Type Simulation Draw Defaults Analysis Tools Window About
C:\PN-editor\one-way-door-2.pnt
Reachabilit...
P1
T1
P2
P3
T2
P4
0
T1
1
T2
2
T1
3
T2
4
Editor of Petri Nets
File Type Simulation Draw Defaults Analysis Tools Window About
C:\PN-EDITOR\two-ways-door-2.pnt
Reachability tree
P1
T1
T3
P2
P3
T2
T4
P4
0
T1
1
T2
T3
2
0
T1
T4
3
1
T2
T3
4
2
T4
3

Fig. 6. The one-way-door model with 2 tokens in p_1 and the corresponding reachability tree are on the left. The two-ways door model with 2 tokens in p_1 and the corresponding reachability tree are on the right.

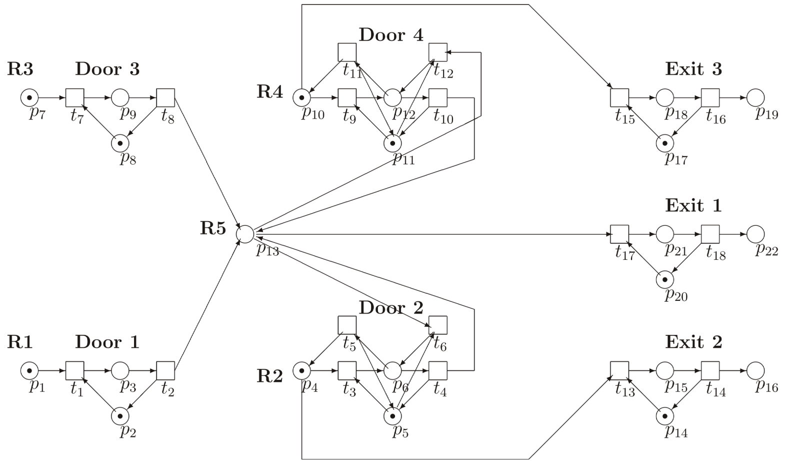


Fig. 7. The P/T PN-based model of the whole EA

(RG). When two tokens are in the place p_1, RT of the one-way door and two-ways door are given in Fig. 6. As we can see, RT expressing the state space (i.e. the space of feasible states $\mathbf{x}_k$, $k = 1, 2, ...,$ being the states reachable from the initial state $\mathbf{x}_0$), grows. The number of states depends on the number of tokens in p_1 - i.e. on the number of people (or groups of people) to be evacuated from a room represented in the P/T PN model by p_1.

Here, P/T PN model of the whole EA consists of modules being two kinds of the doors. However, in general, with a measure of abstraction, EA by itself can be understood to be a complex door. It can be interconnected with other complex doors by means of some paths limited by the building structure. In such a way the P/T PN models of the building segments can be linked into a global P/T PN model of the whole building.

4.2 P/T PN-Based Model of the Whole Endangered Area

Having particular modules (agents) being 'bricks', the model of EA can be created (built) by means of the modularity approach. Consider the P/T PN-based models of the doors given in Fig. 3 to be modules of the P/T PN-model of the whole EA. Of course, the EA model has to be designed with respect to the structure of the real EA given by the physical scheme in Fig. 2. We can see that the one-way doors from R1 and R3 enter the corridor as well as the two-way doors from R2 and R4. Simultaneously, the one-way doors (exits) E2 and E3 from the rooms R2 and R4, respectively, enter the free and safety area outside of the building. The one-way door (exit) E1 from the corridor R5 enter the free and safety area outside of the building too. Thus, we obtain the PN model given in Fig. 7. The initial state vector of such a model is as follows

$$\mathbf{x}_0 = (1\,1\,0\,1\,1\,0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,0\,0)^T \tag{27}$$

As to the structure, the P/T PN model of EA fully corresponds to the real scheme of EA. It is able to develop dynamics of the evacuation process with respect to the mathematical model (3) possessing the incidence matrices $\mathbf{F}$, $\mathbf{G}^T$ (see Appendix) as its parameters. The incidence matrices consist of the incidence matrices $\mathbf{F}_i$, $\mathbf{G}_i^T$ of corresponding P/T PN models of particular modules and their interconnections. Thus, the animation of the tokens (typical for PN graphical tools) can also be computed step-by-step in analytical terms e.g. by means of the simulation tool MATLAB. More details how to do can be found in [1], [2]. Of course, the P/T PN model of EA given in Fig. 7 can also be drawn by means of the graphical P/T PN simulator and the properties can be tested and/or the RT can be found. However, it is better to use the computational approach in MATLAB, because in larger PN models ergonomic problems may occur. Namely, a human operator can lost orientation in large P/T PN graphical models. Moreover, both the P/T PN model and its RT can be so large that it does not go in the screen of the graphical tool. By computing the adjacency matrix of RT in MATLAB we can analyze it and find [2] the path from a given arbitrary initial state to a desired terminal state. It is very useful at finding the escape routes from EA to a safety area situated outside EA. The adjacency matrix can have a large dimensionality. It means that there are many states in the evacuation process. Namely, the number of different states of such a system in our case is $N = 1849$. The reasons for this are that: (i) the level of abstraction of the P/T PN model of EA is too high - the model is very detailed; (ii) the movement of marks (marking dynamics) inside the EA model is completely free. It is not organized, i.e. no control interferences are performed. Such a model comprehends all possible state trajectories (paths) including all of escape routes from EA. Consequently, the situation cannot be managed without an entity organizing the escape. Therefore, the only passable way how to deal with such a problem is to supervise the process. Now, let us synthesize the supervisor(s).

4.3 Supervisor Synthesis

To guarantee prescribed properties of the P/T PN model of the evacuation process, the supervisor(s) can be synthesized. Two supervisors S1, S2 will be synthesized below. Namely, while the supervisor S1 simply expresses a monitor - the quantitative indicator of the modelled evacuation (it yields the number of evacuated people), the supervisor S2 concerns qualitative indicators - namely, its synthesis touches directly the escape routes.

Synthesizing the Supervisor S1. Taking into account the condition $p_1 + p_4 + p_7 + p_{10} + p_{13} \leq b$, where $b = 4$ means the global number of tokens (e.g. persons or groups of persons) in the rooms including the corridor represented by the PN places p_1, p_4, p_7, p_{10}, p_{13} we obtain

$$\mathbf{L}_p = (1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0);\ \mathbf{b} = (4)$$
$${}^1\mathbf{F}_s = (0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0)$$
$${}^1\mathbf{G}_s^T = (1\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 0);\ \mathbf{x}_{s0} = (0)$$

The number of states of the system supervised only by the S1 is $N = 1849$ too, of course. However, the supervisor checks the number of evacuated persons, what is very important. To ensure qualitative properties, another supervisor has to be synthesized.

Synthesizing the Supervisor S2. In general, it can happen that some doors can have a higher priority than others during the evacuation process. Namely, by means of handling the doors it is possible to flexibly change the escape routes. In our case, the evacuation of the rooms R2 and R4 should prefer the exits E2, E3, respectively, in order to relieve the exit E1 when it is too busy. Then, the exit E1 is able to manage better the evacuation of the rooms R1 and R3. At necessity to solve priority problems, usage of the Parikh's vector is very helpful. In the P/T PN model the conditions described verbally can be expressed in mathematical terms as follows $v_4 \leq v_{13}$; $v_{10} \leq v_{15}$; $v_6 \leq v_{17}$; $v_{12} \leq v_{17}$. Here, v_i are the entries of the Parikh's vector $\mathbf{v}$ and they concern the PN transitions t_i with the same indices, i.e. $i \in \{4, 13, 10, 15, 6, 17, 12\}$. The conditions in the form of the inequalities express the fact that the transitions on their left sides are used more rarely (less often) than the transitions on their right side. Consequently, we can form the condition

$$\mathbf{L}_v.\mathbf{v} \leq {}^2\mathbf{b} \tag{28}$$
$$\mathbf{L}_v = \begin{pmatrix} 0&0&0&1&0&0&0&0&0&0&0&0&-1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1&0&0&0&0&-1&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&-1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&-1&0 \end{pmatrix};\ {}^2\mathbf{b} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Thus, for the initial state vector of the augmented system with the supervisor S1 is

$$\mathbf{x}_{a0} = (\mathbf{x}_0^T\ \mathbf{x}_{s0}^T)^T = (1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ |\ 0)^T \tag{29}$$

and the parameters of the supervisor S2 are as follows

$${}^2\mathbf{F}_s = \begin{pmatrix} 000100000000000000 \\ 000000000100000000 \\ 000001000000000000 \\ 000000000001000000 \end{pmatrix}$$
$${}^2\mathbf{G}_s^T = \begin{pmatrix} 000000000000100000 \\ 000000000000001000 \\ 000000000000000010 \\ 000000000000000010 \end{pmatrix};\ {}^2\mathbf{x}_{s0} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

The supervisor S1 is given in Fig. 8 by the place p_{23} together with its interconnections (given by $\mathbf{F}_s$, $\mathbf{G}_s^T$) with other parts of the original P/T PN model of EA, while the supervisor S2 is creted by the places p_{24}, p_{25}, p_{26}, p_{27} (included into the dashed box) together with their interconnections (given by ${}^2\mathbf{F}_s$, ${}^2\mathbf{G}_s^T$) with other parts of the model. The supervisors S1 and S2 were synthesized in analytical terms. The number of states of the system supervised simultaneously by both the S1 and the S2 is $N = 608$, i.e. less than 1/3 from 1849. This documents that the supervision brings a notable reduction of states and simultaneously yields the flexibility as to the escape routes. In general, the more conditions at the supervisor synthesis are imposed the less number of states of the supervised system occur. Namely, each condition eliminates some useless states of the original model.

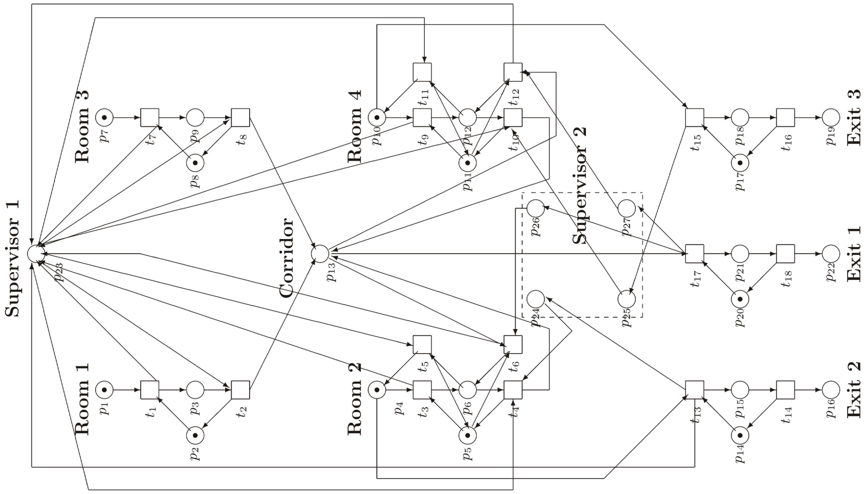


Fig. 8. The P/T PN-based model of the EA with the supervisors S1, S2

4.4 P/T PN-Based Modelling the Workflow of the Evacuation Process

Usually, it is not necessary to compute the complete state space of EA. It is useful perhaps only at the design of buildings and detailed planing the safety escape paths (e.g. before the erection of a new building). However, the P/T PN-based model of EA can also be useful at finding the evacuation paths in exististing buildings. Namely, by means of RT the model is able to yield all feasible paths from a given initial state (the building occupancy before the evacuation of EA) to the desired terminal state (the empty EA). Thus, during the evacuation process, the 'flow' of people can be directed through the most suitable escape route(s). As it was already mentioned in section 2.3, workflow is defined as a scheme of executing a more complicated activity (or process) itemized into simpler activities and their interconnections. The consecutive flow - i.e. a sequence of steps where any step immediately follows the precedent one without any gap or delay and ends just before starting the subsequent step - is emphasized by the workflow. Just P/T PN are suitable to model this. Therefore, let us create now the P/T PN-based model of the evacuation process wokflow.

In our case, the simplest form of the P/T PN-based model of the evacuation workflow is given in Fig. 9. However, here the sense of the places p_i and transitions t_j is completely different from that in the above introduced P/T PN-based model of EA, of course. Here, p_1 expresses the start of the evacuation process; p_2 - p_6 represent the rooms R1 - R5 to be evacuated; p_7, p_8, p_9 express, respectively, the exits E2, E3, E1; p_{10} represents the checking point (it permanently finds if all of the rooms were already evacuated); p_{11} expresses the end of evacuation process. The incidence matrices $\mathbf{F}$, $\mathbf{G}^T$ of the P/T PN model of the evacuation workflow are introduced in the Appendix. Just the movement of the marks in the P/T PN model of the evacuation workflow represents the 'flow' of people. As we can see, the initial state is $\mathbf{x}_0 = (1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0)^T$. After starting the evacuation (firing t_1) the rooms R1 - R4 (represented by p_2 - p_5) and the corridor R5 (represented by p_6) are evacuated. Because the doors t_2, t_4 are one-way doors, R1 (p_2) and R3 (p_4) are evacuated only through them to the corridor p_6 and then through Exit 1 (t_6) outside of EA. The room R2 (p_3) can be evacuated either through Exit 2 (t_3) directly out of EA or through t_7 to the corridor and then through Exit1 out of EA, while R4 (p_5) can be evacuated either through own Exit 3 (t_5) directly out of EA or through t_8 to the corridor and then through Exit1 (t_6) out of EA. The the state space of such a system has $N = 327$ states.

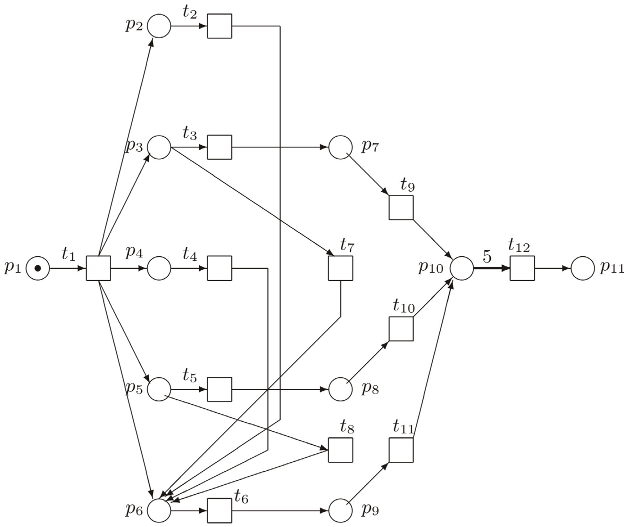


Fig. 9. The simple P/T PN-based workflow of the evacuation process from EA

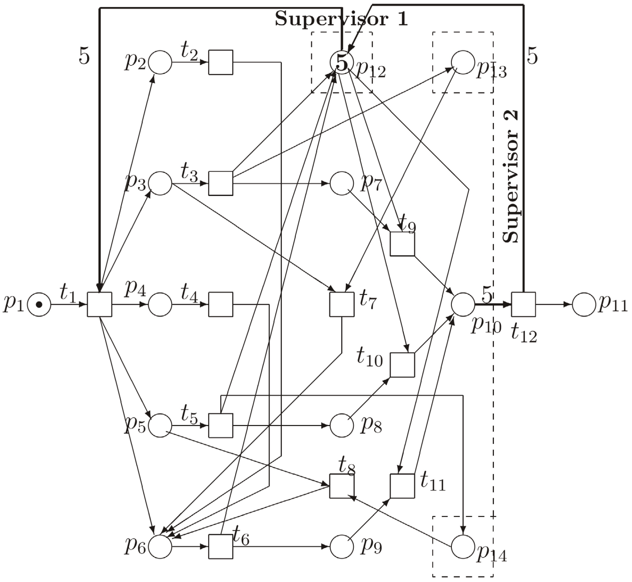


Fig. 10. The supervized P/T PN-based workflow

The model expresses all possible evacuation ways. The OR-split workflow primitives $\{p_3,\ t_3$ or $t_7\}$ and $\{p_5,\ t_5$ or $t_8\}$ complicate the situation. In general, to prescribe a particular way when there exist several options, we need a tool for this. Fortunately, also here, at the P/T PN-based model of the evacuation workflow, we can apply the supervision like in the previous case. We can synthesize the supervisor S1 by means of the condition $p_2 + p_3 + p_4 + p_5 + p_6 + p_{10} \leq 5$. It has to be satisfied, because there are 5 rooms to be evacuated and a checking point represented by p_{10}. Hence, the synthesis of the supervisor S1 is realized by the following procedure

$$
\begin{aligned}
{}^1\mathbf{x}_0 &= (1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0)^T \\
\mathbf{L}_p &= (0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0); \quad \mathbf{b} = (5) \\
{}^1\mathbf{F}_s &= (5\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0) \\
{}^1\mathbf{G}_s^T &= (0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 5); \quad \mathbf{x}_{s0} = (5)
\end{aligned}
$$

The the state space of the augmented system (i.e. the original model of the evacuation workflow together with the supervisor S1) has also $N = 327$ states but monitoring the evacuation process is performed.

However, we can also add the supervisor S2 to the S1. For synthesizing S2 the conditions $v_7 \leq v_3$; $v_8 \leq v_5$ are used. These inequalities represent the desired priorities. Hence, starting from

$$
{}^2\mathbf{x}_0 = (1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \mid 5)^T \tag{30}
$$

$$\mathbf{L}_v = \begin{pmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}; \quad {}^2\mathbf{b} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$${}^2\mathbf{F}_s = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$${}^2\mathbf{G}_s^T = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}; \quad {}^2\mathbf{x}_{s0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Hence, the model of the supervised workflow (supervised simultaneously with both supervisors S1 and S2) is obtained - see Fig. 10. The state space of the model has $N = 227$ states, i.e. less than 327 about for 1/3. This documents the fact that the supervision reduces the number of states also in case of the PN-based evacuation workflow.

In Fig. 9 the transitions $t_2 - t_5$ are understood to be fired simultaneously. However, because there exist also two OR-split primitives in Fig. 9, created by $(p_3,\ t_3,\ t_7)$ and $(p_5,\ t_5,\ t_8)$, respectively, the supervisor S2 in Fig. 10 has to ensure desired priorities between t_3 and t_7 as well as between t_5 and t_8.

However, we can set another priorities, namely $v_7 \geq v_3;\ v_8 \leq v_5$ or $v_7 \leq v_3;\ v_8 \geq v_5$ or $v_7 \geq v_3;\ v_8 \geq v_5$ and found the alternative supervisors S2 for them. Thus, in the simulation process we can analyze also the flexibility at using the escape routes in various 'dynamic' situations during the evacuation.

Wider P/T PN-Based Model of Workflow. In order to investigate the workflow in case when also the two-ways doors are used, i.e. when the rooms R2 and R3 can be accessible from the corridor R5 too, the additional transitions t_{13} and t_{14} has to be added to Fig. 9. Then, the P/T PN model of the non-supervised evacuation workflow is displayed in Fig. 11. Thus, the experiments can be extended also for usage of the transitions t_{13}, t_{14}. Such a non-supervised structure has 968 different states. Here, two particularities occur: (i) t_7 and t_{14} cannot be used simultaneously; (ii) t_8 and t_{13} cannot be used simultaneously. Namely, it is senseless to allow entering the two-ways door from both sides simultaneously. It is necessary to set a priorities. Consider e.g. the following priorities: $v_7 \leq v_3;\ v_8 \leq v_5$ like before, and $v_{14} \leq v_7;\ v_{13} \leq v_8$, and consequently $v_6 \leq v_3;\ v_6 \leq v_5$. Then, the supervisor S1 is synthesized as follows

$$\mathbf{L}_v = \begin{pmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}; \quad \mathbf{v}_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad (31)$$

$${}^1\mathbf{F}_s = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad {}^1\mathbf{G}_s^T = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

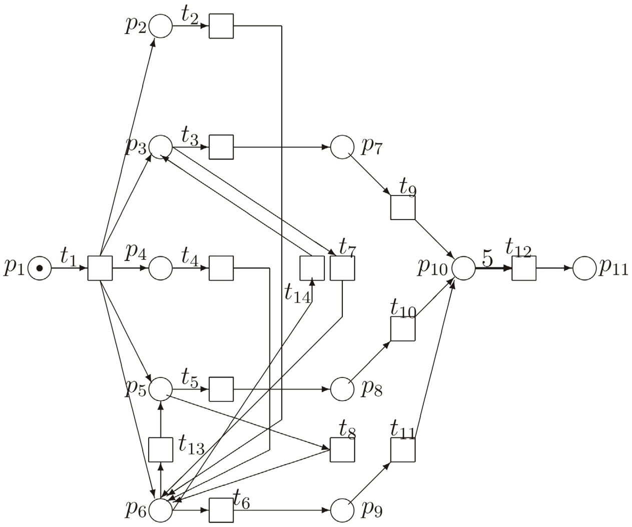


Fig. 11. The non-supervised P/T PN-based workflow in case of usage the two-ways doors

The supervisor is constituted by the 6 places, namely $p_{12} - p_{17}$, together with their interconnections with the non-supervised model expressed by ${}^1\mathbf{F}_s$, ${}^1\mathbf{G}_s^T$. In this case, when ${}^1\mathbf{x}_{s0} = (111111)^T$, the number of different states is only 69. It is about 14 times less amount than 968. The supervised evacuation workflow is given in Fig. 12. Of course, also here we can vary the priorities and investigate some (even all) of their combinations in the simulation process.

However, also in this case, another supervisor S2 (primarily the monitor) can be added. Consider the demand $p_2 + p_3 + p_4 + p_5 + p_6 \leq 5$ checking the number of evacuated rooms. The condition in the matrix form is as follows

$$\mathbf{L}_p.\mathbf{x} \leq \mathbf{b};\ \text{where}\ \mathbf{L}_p = (0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0);\ \mathbf{b} = (5) \qquad (32)$$

Hence, after synthesizing the supervisor S2 corresponding to the condition (32), the parameters and the initial state of S2 are the following

$$\begin{aligned} {}^2\mathbf{F}_s &= (0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 5\ 0\ 0)^T \\ {}^2\mathbf{G}_s^T &= (5\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0)^T;\ \mathbf{x}_s0 = (5) \end{aligned}$$

Including the supervisor (represented by the place p_{18} and all its interconnections represented by ${}^2\mathbf{F}_s$, ${}^2\mathbf{G}_s^T$) into the previous model given in Fig. 12, we obtain the model displayed in Fig. 13. Here, the number of different states is only 69 too. But there is the advantage here that the number of the evacuated rooms is checked too. It is necessary to emphasize that the supervisors can be synthesized

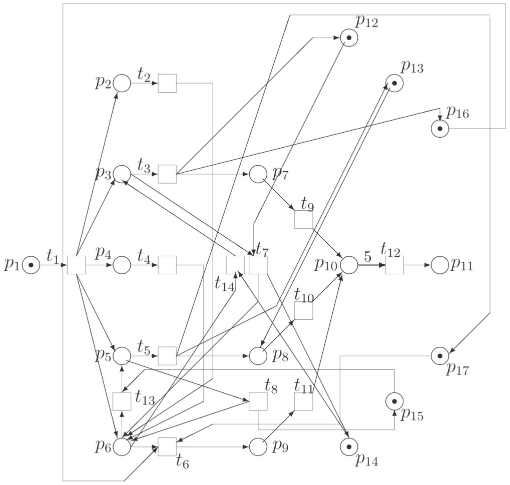


Fig. 12. The P/T PN-based workflow supervized by S1 corresponding to priorities given by the conditions (31)

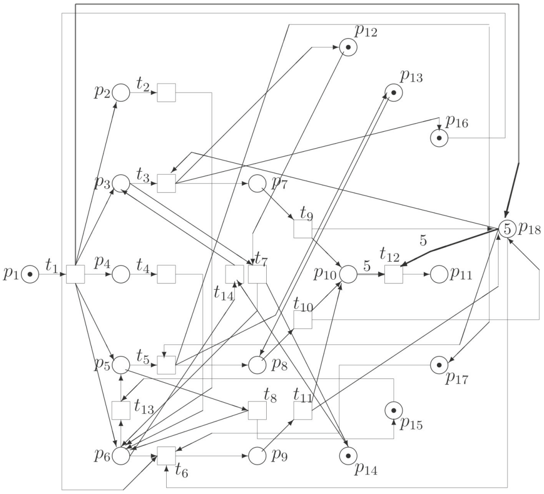


Fig. 13. The P/T PN-based workflow supervized by S1 and S2 corresponding to priorities given by the condition (32)

flexibly either jointly or separately in an arbitrary order. However, in case of the successive synthesis we always have to start from the augmented system (i.e. the system being already supervised by the supervisor synthesized before), not from the original non-supervised system.

5 Conclusion

Initially, the approach to modelling DES by means of P/T PN was introduced. The models of particular modules (agents) can be built by such an approach up. Subsequently, three possibilities how to construct bigger aggregates from the cooperating modules were offered. P/T PN-based modelling the workflow was pointed out and basic workflow primitives were introduced. Afterwards, the cooperation by means of supervisors was proposed. It was shown how the supervisors are synthesized by the help of prescribed conditions. Two approaches to the supervisor synthesis were introduced: (i) the synthesis based on P-invariants of P/T PN model expressed in analytical terms by means of the linear discrete system (3); (ii) the extended method of the synthesis, where some conditions are imposed on the state vector, control vector and Parikh's vector of the P/T PN model.

Finally, the case study was introduced. Here, the P/T PN-based model of the whole EA consisting of P/T PN-based models of the EA modules (agents) represented by two kinds of doors was created and tested. Moreover, the supervision was utilized here in order to improve the model quality and to diminish the number of states.

Then, the P/T PN-based modelling the evacuation workflow was studied. Namely, two P/T PN-based models of the evacuation workflow from EA were analyzed and corresponding effective safety escape routs were found be means of the supervision. The supervision was utilized especially in order to solve decision problems connected with OR-split primitives. The supervisors improve the quality of the evacuation process and decrease the number of states.

The P/T PN-based models can be effectively utilized in the process of simulation. The simulation can be executed either graphically (using the P/T PN graphical tool) or numerically (using MATLAB). The simulation experiments with P/T PN models shown that the advantage of the P/T PN model of the real EA is the direct analogy with the scheme of the real EA. Moreover, in the process of simulation the P/T PN model makes evolving the evacuation dynamics possible, namely, by means of the movement of the P/T PN marks. On the other hand, at this approach the number of states is usually big, of course. It is a disadvantage of such a procedure. However, the dimensionality of the model depends on the level of abstraction at the model design. Of course, the more details about the modelled object are asked the bigger dimensionality such a model acquires. The P/T PN-based model of the evacuation workflow allows to study directly the escape routes from EA. In this case the model is simpler and less in comparison with the previous one. It contains neither so much details nor so much states.

Both approaches are executed off-line. The former approach can be utilized especially at the design of new buildings (or other areas) to plan escape routs. It helps to construct them safely in order to avoid problems at solving crisis situations in future (i.e. after their erection and starting their usage). Thus, the states of the model corresponds to the states of the real building and the model allows to analyze the system in the whole. Consequently, many times new escape routes from EA can be found - especially in more complicated areas. The latter approach can be successfully utilized especially at finding the safety escape routes in buildings that already exists.

However, both approaches can be combined, of course. Thus, their advantages can complement each other and their disadvantages can be suppressed.

Acknowledgement. The work was partially supported by the Slovak Grant Agency for Science VEGA under grants # 2/0075/09 (2009-2012) and # 2/0039/13 (2013-2016) where the theoretical aspects were investigated. The author thanks VEGA for the support. Simultaneously, this publication is the result of the project implementation: Research and development of new information technologies for forecasting and solving crisis situations and for the security of populace, ITMS code: 26240220060 supported by Operational Programme Research & Development funded by the ERDF (European Regional Development Found).

References

1. Čapkovič, F.: Modelling, analysing and control of interactions among agents in MAS. Computing and Informatics 26(5), 507–541 (2007)
2. Čapkovič, F.: Automatic control synthesis for agents and their cooperation in MAS. Computing and Informatics 29(6+), 1045–1071 (2010)
3. Čapkovič, F.: Cooperation of agents in manufacturing systems. In: Jędrzejowicz, P., Nguyen, N.T., Howlet, R.J., Jain, L.C. (eds.) KES-AMSTA 2010, Part I. LNCS (LNAI), vol. 6070, pp. 193–202. Springer, Heidelberg (2010)
4. Čapkovič, F.: DES control synthesis and cooperation of agents. In: Nguyen, N.T., Kowalczyk, R., Chen, S.-M. (eds.) ICCCI 2009. LNCS, vol. 5796, pp. 596–607. Springer, Heidelberg (2009)
5. Čapkovič, F.: A modular system approach to DES synthesis and control. In: Proc. 2009 IEEE Conference on Emerging Technologies & Factory Automation, ETFA 2009, Palma de Mallorca, Spain, p. 8. CD ROM. IEEE Press, Piscataway (2009)
6. Čapkovič, F., Jotsov, V.: A system approach to agent negotiation and learning. In: Sgurev, V., Hadjiski, M., Kacprzyk, J. (eds.) Intelligent Systems: From Theory to Practice. SCI, vol. 299, pp. 133–160. Springer, Heidelberg (2010)
7. Čapkovič, F.: Supervision of agents modelling evacuation at crisis situations. In: Jezic, G., Kusek, M., Nguyen, N.-T., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2012. LNCS, vol. 7327, pp. 24–33. Springer, Heidelberg (2012)
8. Chebbi, I., Dustdar, S., Tata, S.: The view-based approach to dynamic interorganizational workflow cooperation. Data & Knowledge Engineering 56(2), 139–173 (2006)

9. Eshuis, R., Wieringa, R.: Comparing Petri net and activity diagram variants for workflow modelling - A quest for reactive Petri nets. In: Ehrig, H., Reisig, W., Rozenberg, G., Weber, H. (eds.) Petri Net Technology for Communication-Based Systems. LNCS, vol. 2472, pp. 321–351. Springer, Heidelberg (2003)
10. Hofman, U., Veichtlbauer, A., Miloucheva, I.: Dynamic evacuation architecture using context-aware policy management. International Journal of Computer Science and Applications 6(2), 38–49 (2009)
11. Iordache, M.V., Antsaklis, P.J.: Supervision based on place invariants: A survey. Discrete Event Dynamic Systems 16(4), 451–492 (2006)
12. Iordache, M.V., Antsaklis, P.J.: Supervisory Control of Concurrent Systems: A Petri Net Structural Approach. Birkhauser, Boston (2006)
13. Iordache, M.V.: Methods for the supervisory control of concurrent systems based on Petri nets abstraction. Ph.D. dissertation, University of Notre Dame, USA (2003)
14. Lino, P., Maione, G.: Applying a discrete event system approach to problems of collective motion in emergency situations. In: Klingsch, W.W.F., Rogsch, C., Schadschneider, A., Schreckenberg, M. (eds.) Pedestrian and Evacuation Dynamics 2008, pp. 465–477. Springer, Heidelberg (2010)
15. Murata, T.: Properties, analysis and applications. Proceedings of the IEEE 77(4), 541–580 (1989)
16. Sniezek, J.A., Wilkins, D.C., Wadlington, P.L., Bauma, M.R.: Training for crisis decision-making: Psychological issues and computer-based solutions. Journal of Management Information Systems 18(4), 147–168 (2002)
17. van der Aalst, W.M.P.: Three good reasons for using a Petri net-based workflow management system. In: Navathe, S., Wakayama, T. (eds.) Proc. of International Working Conference on Information and Process Integration in Enterprises, IPIC 1996, Cambridge, Massachusetts, pp. 179–201 (1996)
18. van der Aalst, W.M.P.: Three good reasons for using a Petri-net-based workflow management system. In: Wakayama, T., Kannapan, S., Khoong, C.M., Navathe, S., Yates, J. (eds.) Information and Process Integration in Enterprises: Rethinking Documents, ch.10. Kluwer International Series in Engineering and Computer Science, vol. 428, pp. 161–182. Kluwer Academic Publishers, Boston (1998)
19. van der Aalst, W.M.P.: The Application of Petri Nets to Workflow Management. Journal of Circuits, Systems and Computers 7(1), 21–66 (1998)
20. van der Aalst, W.M.P.: Workflow verification: Finding control-flow errors using Petri net-based techniques. In: van der Aalst, W.M.P., Desel, J., Oberweis, A. (eds.) Business Process Management. LNCS, vol. 1806, pp. 161–183. Springer, Heidelberg (2000)
21. van der Aalst, W.M.P., van Hee, K.M.: Workflow Management: Models, Methods, and Systems. MIT Publisher, Boston (2004)
22. van der Aalst, W.M.P., van Hee, K.M., ter Hofstede, A.H.M., Sidorova, N., Verbeek, H.M.W., Voorhoeve, M., Wynn, M.Y.: Soundness of workflow nets: Classification, decidability, and analysis. Formal Aspects of Computing 23(3), 333–363 (2011)
23. Wang, J., Rosca, D., Tepfenhart, W., Milewski, A.: Incident command system workflow modeling and analysis: A case study. In: Van de Walle, B., Turoff, M. (eds.) Proc. of 3rd International ISCRAM (Information Systems for Crisis Response and Management) Conference, Newark, NJ, USA, pp. 127–136. ISCRAM Association, Brussels (2006)
24. Kubera, Y., Mathieu, P., Picault, S.: Everything can be agent! In: Proc. of 9th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2010, Toronto, Canada, pp. 1547–1548 (2010)

25. Wagner, T., Quenum, J., Moldt, D., Reese, C.: Providing an agent flavored integration for workflow management. In: Jensen, K., Donatelli, S., Kleijn, J. (eds.) Transactions on Petri Nets and Other Models of Concurrency V. LNCS, vol. 6900, pp. 243–264. Springer, Heidelberg (2012)

Appendix

The incidence matrices of the P/T PN-based model of the endangered area are the following

$$\mathbf{F}=\begin{pmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{pmatrix}
\mathbf{G}^T=\begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&1&0&0&0&0&0&0\\
0&1&0&1&0&0&0&1&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1
\end{pmatrix}$$

The incidence matrices of the P/T PN-based model of the workflow are as follows

$$\mathbf{F} = \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&1&0&0&0&0&0\\ 0&0&0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&1&0&0&0&0\\ 0&0&0&0&0&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0&0&0&0&5\\ 0&0&0&0&0&0&0&0&0&0&0&0 \end{pmatrix} \mathbf{G}^T = \begin{pmatrix} 0&0&0&0&0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0&0&0&0&0\\ 1&1&0&1&0&0&1&1&0&0&0&0\\ 0&0&1&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0&0&0\\ 0&0&0&0&0&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&1&1&0\\ 0&0&0&0&0&0&0&0&0&0&0&1 \end{pmatrix}$$

Particle Swarm Optimization with Disagreements on Stagnation

Andrei Lihu, Ştefan Holban, and Oana-Andreea lihu

Department of Computer Science, Politehnica University of Timişoara,
Bd. Vasile Pârvan, 300223 Timişoara, Romania
andrei.lihu@gmail.com, stefan@cs.upt.ro, oana.lihu@gmail.com

Abstract. This paper[1] introduces a modified particle swarm optimization (PSO) that exhibits the so-called "extreme social disagreements" among its wandering particles in order to resolve the stagnation when it occurs during search. We provide a short theoretical introduction about particle swarm optimization, then we describe and test our modified algorithms. We conclude from tests on several optimization benchmarks that our approach may help PSO escape stagnation in most of the situations in which it was tested. This work is intended to illustrate one of the benefits of using disagreements in social algorithms like PSO.

Keywords: particle swarm optimization, disagreements, riot, stagnation, swarm intelligence.

1 Introduction

Nowadays, the *particle swarm optimization* (PSO) is a top competitor among optimization algorithms. As a population based optimization technique that implements "the social mind" metaphor, it simulates a type of social behavior found in various animal communities like birds or fish [19].

PSO belongs to the category of algorithms that has as an advantage the ability to operate without the need of gradient information. Initially described in [10], it models a swarm of particles flying iteratively through the hyperspace of solutions until a termination condition is met. Particles improve their positions during iterations based on their own personal and their group's best experience. The ability to be influenced by neighbors represents *the social component* of the algorithm and the ability to take decisions from its own experience represents *the cognitive component*. The swarm is a super-organism that acts like a "learning-machine" ([5]), mimicking the way most Earth's living beings' societies organize and operate. PSO is a *collective intelligence algorithm*. There are lots of other algorithms in this category. Canonical examples include *ant colony optimization* ([8]), *bee's algorithm* ([18]) and *stochastic diffusion search* ([3, 4]) among others.

[1] This paper is an extended and improved version of one of our previously published article at ICCCI 2011 (Gdynia, Poland 2011, — see [14]), featuring extra test scenarios and an in-depth analysis of new results.

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 171–190, 2013.

Depending on the difficulty of the problem they attempt to solve, optimizers may stagnate during the search through the hyperspace of solutions. They may not improve the current solution for an indefinite amount of time. Two of the many reasons stagnation occurs are the following:

- The algorithm is currently trapped in a local minimum and has no means to escape. Eventually, this may lead to premature convergence.
- The algorithm is either currently wandering across a large plateau, or in an equally sized dense and spiky multidimensional environment.

Stagnation is the problem that we try to solve in this paper. Because it is a common basic problem, we expect that any real-world practical application that is using our approach may benefit from it. Since the range of applications is potentially large, and also each one having its own specifics, we tested our modifications on some standardized benchmarks.

For a population-based algorithm as PSO, we think that the diversity inside the swarm should be increased in order to avoid stagnation. Simultaneously, the algorithm should still converge and it should converge in a reasonable amount of time. Following the promising results obtained with the *disagreements concept* in [13] and [12], we propose a simple mechanism to increase the diversity inside the swarm only when stagnation occurs. Since PSO is a social algorithm, we modeled and added a new social behavior to its social component: the disagreements that naturally occur in a social group. Whenever stagnation is detected, the particles from the swarm can oppose their group's way by exhibiting different opinions with a given probability. This way the particles' positions change their original trajectory and have more chances to disrupt the current stagnation period. For some generations full-blown riots may persist and change the path of the current search. In order to preserve the final ability to converge, we will apply the disagreements following a decreasing linear probability rule.

After conducting several empirical tests, we concluded that this simple enhancement might help evolutionary algorithms like PSO escape from local minima when needed, therefore making it suitable for solving multi-modal problems. If properly implemented, the extra-added computational cost would be minimal.

As an overview, in Section 2 we present general information about the standard particle swarm optimization and how to measure stagnation. In Section 3 we explain the theoretical foundation of disagreements as previously introduced in [13] and [12]. In Section 4 we introduce a practical approach to handle stagnation — *the riot-on-stagnation operator* (RS-PSOD) and in Section 5 we conduct some tests using the new operator to analyze their results. In Section 6 we conclude that the new approach can have real benefits in stagnation-prone environments.

2 Stagnation in Particle Swarm Optimization

2.1 Particle Swarm Optimization

As in Van den Bergh's PhD thesis, [21], we describe *the particle swarm optimization algorithm* (PSO) as follows:

Let n be the dimension of the solution hyperspace H^n, let s be the number of particles from the swarm, and let i be the index of a particle, such that $i \in \overline{1 \ldots s}$. Each particle i has the following variables: x_i – the current position, v_i – the current velocity, y_i – the current best position, $\hat{y}$ – neighborhood's best. The function f is the function to be minimized. PSO has three phases: initialization, iterations, termination.

In the initialization phase, the particles' positions and velocities are randomly spread in the search space. Best positions are initially (and afterwards) updated using (3) and (4).

In the iterations' phase, the updating principle for velocities and positions is given in (1) and (2):

$$v_{ij}(t+1) = wv_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[\hat{y}_j(t) - x_{ij}(t)] \ , \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \ , \quad (2)$$

where c_1 is the personal coefficient, c_2 is the social coefficient, $c_1, c_2 \in (0, 2]$. r_1 and r_2 are random vectors, such that: $r_1, r_2 \sim \mathcal{U}(0,1)$. The first term of (1) is the previous velocity influenced by an inertial weight w. The second term is the personal component that makes the particle move toward its best personal position found so far and the third term orients the particle toward neighborhood's best position found so far.

At each iteration, y_i and $\hat{y}$ are updated using the formulae:

$$y_i(t+1) = \begin{cases} y_i(t) & \text{if } f(x_i(t+1)) \geq f(y_i(t)) \\ x_i(t+1) & \text{if } f(x_i(t+1)) < f(y_i(t)) \end{cases} , \quad (3)$$

$$\begin{aligned} &\hat{y}(t) \in \{y_0(t), y_1(t), \ldots, y_s(t) | f(\hat{y}(t))\} = \\ &min\{f(y_0(t)), f(y_1(t)), \ldots, f(y_s(t))\} \ . \end{aligned} \quad (4)$$

When an *a priori* established criterion is met, the algorithm terminates (a number of fitness calls or generations have elapsed, stagnation, etc.).

2.2 Stagnation

Stagnation is defined as an undesirable situation in which there is no improvement of current solution for an amount of time. For PSO, it is defined in [15] as follows:

> "The particle swarm system is thought to be in stagnation, if arbitrary particle i's history best position P_i and the total swarm's history best position P_g keep constant over some time steps."

In order to measure the stagnation in PSO, [22] takes into consideration also the velocities of the particles and defines an improvement ratio as:

$$R = \left| \frac{1 - f_c/f_p}{1 - v_c/v_p} \right| \ , \quad (5)$$

where f_c is the current fitness value of the best particle, f_p is the previous value of the best particle and v_c is the current average velocity of all particles, while v_p is the previously recorded one. The system enters stagnation when R drops under a preset threshold value ϵ.

The approach from [22] assumes that when stagnation occurs the velocities tend to 0. However, this is valid in situations when particles get trapped into local minima, while for very densely spiked or plateau functions these assumptions might not be true. To decide for all cases if no improvement took place in an amount of time Δt_h between two iterations, $t_1 = t$ and $t_2 = t + \Delta t_h$, we decided to measure the Euclidean distance between the current fitness of best particle at iterations t_1 and t_2. To achieve a relative measurement, this is compared with best fitness at the initial moment multiplied with a preset threshold value, ϵ. Therefore, the following relation must hold true in order to have a state of stagnation:

$$\|\hat{y}_{t+\Delta t_h} - \hat{y}_t\| < \epsilon \cdot \|\hat{y}_t\| \quad . \tag{6}$$

3 Disagreements

Disagreements are an ubiquitous social phenomenon that leads to greater diversity and heightened awareness of current problems. Acemoglu et al. state in [2] that:

> "Disagreement among individuals in a society, even on central questions that have been debated for centuries, is the norm; agreement is the rare exception. How can disagreement of this sort persist for so long? Notably, such disagreement is not a consequence of lack of communication or some other factors leading to fixed opinions. Disagreement remains even as individuals communicate and sometimes change their opinions."

The origin of disagreements is well explained in [1]:

> "In none of these cases can the disagreements be traced to individuals having access to different histories of observations. It is rather their interpretations that differ."

We apply this concept that comes from the field of social networks to a social algorithm like PSO to increase the diversity of the swarm when stagnation is detected. For a better understanding and completeness, we will introduce the disagreements concept in PSO as we did in [13] and [12].

Although a disagreement can be defined as a function that takes values in the space of solutions $V^n \in \mathbb{R}^n$ and one can define the disagreements as the family of D functions for which the following property holds true:

$$F_{\mathrm{D}} = \{D : V^n \rightarrow V^n | \forall z \in V^n .\ D(z) \neq z\} \ , \tag{7}$$

in order to obtain a true disagreement the designer of the algorithm must take into consideration information about context.

From the particle's point of view, between two succesive iterations, there is a regular transition β_i that comes in effect from the updating principle:

Definition 1. *Given a PSO algorithm* P *with a swarm of* s *individuals whose positions are represented by* $W(t) = \{x_1(t), x_2(t), \ldots, x_s(t)\}$*, for any two successive iterations,* t *and* $t+1$*, there is a corresponding vector of update behaviors* $\mathfrak{B}(t) = \{\beta_1(x_1, t), \beta_2(x_2, t), \ \ldots, \beta_s(x_s, t)\}$ *that makes the transition from* $W(t)$ *to* $W(t+1) = \{x_1(t+1), x_2(t+1), \ldots \ x_s(t+1)\}$*.*

Definition 2. *A disagreement is defined as a function* D *that operates on an individual's* x_i *update behavior* β_i *at iteration* t*.*

The golden rule is that disagreements do not happen very frequently, therefore we define a "no-disagreement" function:

Definition 3. *The identity function (no disagreement should happen) is* $\emptyset_{\mathrm{D}}$*.*

In order to accomodate disagreements in PSO, at the iteration level the updating rule is changed by the disagreements selector which "injects" disagreements into the swarm, defined as:

Definition 4. *Let* ρ *be the disagreement selector that decides which disagreement is invoked upon an individual update behavior* β_i *at iteration* t *from a given set of disagreements* $\Delta_v \in \mathcal{P}(F_{\mathrm{D}})$*:*

$$\rho(\Delta_v, \beta_i, t) = D_j, \ \ D_j \in \Delta_v, \ \ i \in \overline{1, |\mathfrak{B}|} \ \ , j \in \overline{1, |\Delta_v|}. \tag{8}$$

The original behaviours change when disagreements are injected, therefore $\mathfrak{B}(t)$ becomes $\mathfrak{B}_{\mathrm{D}}(t) = \{\beta_{\mathrm{D}1}(x_1, t), \beta_{\mathrm{D}2}(x_2, t), \ldots, \beta_{\mathrm{D}s}(x_s, t)\}$.

Definition 5. *Let* P *be a PSO and* ρ *a disagreements apply rule. A* ***PSO with disagreements (PSOD)****,* P_{D}*, is obtained by modifying* P*'s updating principle with the rule* ρ*, as described by the disagreement injector function* Ψ*:*

$$\Psi(\mathsf{P}, \rho) = \mathsf{P}_{\mathrm{D}} \ . \tag{9}$$

In PSO a disagreement appears when a particle do not want to follow the group leader, therefore the social component from the updating principle is modified accordingly. There can be partial and extreme disagreements, but still the number of disagreements should be less than the number of "agreements". **Disagreements provide ways in which an algorithm may behave sometimes differently from normal operation, yet it is not an error, but a feature**. To illustrate the concept we replaced the first term from eq. (1) with a generic $V(t, i)$ — the velocity component, the second term with $C(t, x_i, y_i)$ — the cognitive component, the third term with $S(t, x_i, \hat{y}$ — the social component. Making the substitution in (2), where we also change the position component with a generic one, $X(t)$, we obtain the generalized updating equation, which is also the update behaviour β_i:

$$X(t+1,x_i) = X(t,x_i) + V(t,i) + C(t,x_i,y_i) + S(t,x_i,\hat{y_i}) + \zeta \ , \\ C(t,x_i,y_i) \rightarrow y_i, S(t,x_i,\hat{y_i}) \rightarrow \hat{y_i} \ , \forall i \in \overline{1,s} \ , \tag{10}$$

where $C(t,x_i,y_i) \rightarrow y_i$ should be read as "the result of $C(t,x_i,y_i)$ tends to y_i" and $S(t,x_i,\hat{y}) \rightarrow \hat{y}$ should be read as "the result of $S(t,x_i,\hat{y})$ tends to $\hat{y}$". ζ is usually 0 and can accommodate any other more elaborate variant of PSO that may consist of other components.

After we apply the injection operator Ψ_{PSO}, the updating principle of the newly obtained PSO, now called *particle swarm optimization with disagreements* (PSOD) is transformed from eq. (10) to:

$$\begin{aligned} & X(t+1,x_i) = \rho(\Delta_v,\beta_i,t) \\ = \ & \rho(\Delta_v, X(t,x_i) + V(t,i) + C(t,x_i,y_i) + S(t,x_i,\hat{y_i}) + \zeta, t) \\ = \ & X(t,x_i) + V(t,i) + C(t,x_i,y_i) + D_i(S(t,x_i,\hat{y_i})) + \zeta \ , \\ & C(t,x_i,y_i) \rightarrow y_i, S(t,x_i,\hat{y_i}) \rightarrow D_i(\hat{y_i}) \ , D_i \in \Delta_v \ , \forall i \in \overline{1,s} \ . \end{aligned} \tag{11}$$

The concept of disagreements is a special operator that can be applied to any social PSO without modifying the internals of the algorithm. The social component can vary in implementation from algorithm to algorithm, but the injection operator can be applied in any case. A disagreement operator can affect only the social component of PSO.

4 Riot-on-Stagnation Operator

Disagreements can have various implementations. Although one can imagine elaborate constructions based on machine learning techniques for example, in order to prove the practicalities and the simplicity of the concept, we have chosen to rely on a simple and computationally cheap technique: to use randomness injection, much like in the dissipative variant of PSO from [23], where the authors propose to increase the entropy of the system by adding after the updating equations in the PSO, with probabilities given by the chaotic factors c_v (for velocity) and c_l (for location) in the range $[0, 1]$, of the following equations for velocity and position update, respectively:

$$\text{IF } (rand() < c_v) \text{ THEN } v_{id} = rand() * v_{max,d} \tag{12}$$

$$\text{IF } (rand() < c_l) \text{ THEN } x_{id} = Random(l_d, u_d), \tag{13}$$

where $Random(l_d, u_d)$ is a random variable between l_d and u_d.

We instead, will divert the social component, therefore individuals can "riot" against the *status quo.*

To demonstrate how a PSOD can resolve stagnation, we defined in terms of eq. (9) the *riot-on-stagnation operator* (RS-PSOD) as follows:

Definition 6. *Let* P *be a particle swarm optimization algorithm that has a social component. The function that injects in* P *a set of disagreements —* Δ_{RS}*, with an apply rule —* ρ_{RS}*, and transforms it into a particle swarm optimization with disagreements following the RS rule, namely a RS–PSOD algorithm —* P_{RS}*, is defined as:*

$$\Psi_{\mathrm{RS-PSOD}}(\mathsf{P}) = \Psi_{\mathrm{PSO}}(\mathsf{P}, \rho_{\mathrm{RS}}) = \mathsf{P}_{\mathrm{DRS}} \ . \tag{14}$$

The subset of disagreements (the Δ_v that contains the disagreements, $D_{\mathrm{RS}i}$) is defined by:

$$\Delta_{\mathrm{RS}} = \{\emptyset_{\mathrm{D}}, D_{\mathrm{RS}}\} \ . \tag{15}$$

We modeled an "extreme disagreement", D_{RS}, which multiplies member-wise (a Hadamard product, $\otimes$) the social component S (e.g. $c_2 r_{2j}(t)[\hat{y}_j(t) - x_{ij}(t)]$) by a vector r containing random uniformly distributed values in the intervals $[-\lambda_{\mathrm{u}}, -\lambda_{\mathrm{l}}]$ and $[+\lambda_{\mathrm{l}}, +\lambda_{\mathrm{u}}]$, with $\lambda_{\mathrm{l}}, \lambda_{\mathrm{u}} \in \mathbb{R}_+^*, \lambda_{\mathrm{u}} > \lambda_{\mathrm{l}}$ and $\lambda_{\mathrm{l}} \geq 1$:

$$D_{\mathrm{RS}}(z) = r_i \otimes z, \;\; r_i = r_{i_1} + sgn(r_{i_1}) \cdot \lambda_{\mathrm{l}}, \;\; r_{i_1} \sim \mathcal{U}(-(\lambda_{\mathrm{u}} - \lambda_{\mathrm{l}}), +(\lambda_{\mathrm{u}} - \lambda_{\mathrm{l}})) \ , \tag{16}$$

where r_i is the i–th component of r and r_{i_1} is a random number for each r_i.

Fig. 1 is a very simple visualization of the concept in two dimensions.

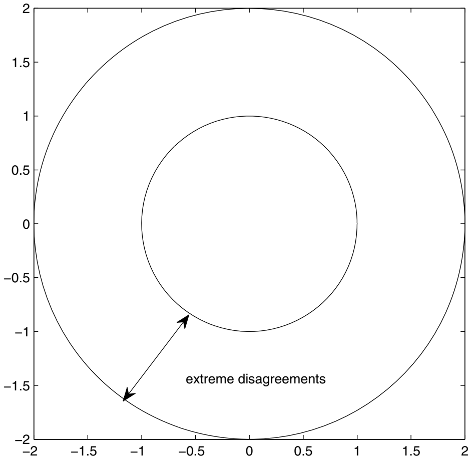


Fig. 1. Area between circles shows where r_i — that produces "extreme disagreements" — can be generated for the case when $\lambda_{\mathrm{l}} = 1$ and $\lambda_{\mathrm{u}} = 2$

Definition 7. *Let* $\theta_{\mathrm{RS}}(t,i) \sim \mathcal{U}(0,1)$ *be an uniformly distributed random variable that is generated at each iteration t for each particle i. Let* $\delta = \frac{t}{t_{max}} \in [0,1]$ *be the current execution progress indicator, where* t_{max} *is the total number of iterations. The RS–PSOD operator is defined as follows:*

$$\rho_{\mathrm{RS}}(\Delta_{\mathrm{RS}}, \beta_i, t) = \begin{cases} \emptyset_{\mathrm{D}}(S), & \textit{if}\ \theta_{\mathrm{RS}}(t,i) < \delta \text{ or } (\mathit{6}) \text{ is false} \\ D_{\mathrm{RS}}(S), & \textit{if}\ \theta_{\mathrm{RS}}(t,i) \geq \delta \text{ and } (\mathit{6}) \text{ is true} \end{cases} \ . \tag{17}$$

The updating principle from eq. (11) becomes:

$$\begin{aligned} X(t+1,x_i) &= \rho_{\mathrm{RS}}(\Delta_{\mathrm{RS}},\beta_i,t) \\ &= \rho_{\mathrm{RS}}(\Delta_{\mathrm{RS}},X(t,x_i)+V(t,i)+C(t,x_i,y_i)+S(t,x_i,\hat{y}_i)+\zeta,t) \\ &= \quad X(t,x_i)+V(t,i)+C(t,x_i,y_i)+D_{\mathrm{RS}i}(S(t,x_i,\hat{y}_i))+\zeta \ , \\ C(t,x_i,y_i) &\rightarrow y_i, S(t,x_i,\hat{y}_i) \rightarrow D_{\mathrm{RS}i}(\hat{y}_i) \ , D_{\mathrm{RS}i} \in \Delta_{\mathrm{RS}} \ , \forall i \in \overline{1,s} \ . \end{aligned} \tag{18}$$

5 Experimental Results

In the following subsections we present our two experimental sessions regarding the riot-on-stagnation operator. Session A contains 4 test functions. The random numbers that are needed at the basic operations of the algorithms were generated using the `java.lang.Random` generator that comes with from the framework Java EvA2 ([11]). We used it because in most real-world applications this is the random number generator of choice. It should be noted that using other random number generator may yield slightly different results.

Session B is based on a slightly modified configuration from session A that is run on 8 test functions based on the same random generator. We used the evolutionary framework Java EvA2 to test the algorithms and implement the above described RS-PSOD. The framework we picked to modify in order to check our assumptions is used in several real-world applications, therefore we have chosen it to offer a higher degree of realism.

The functions we used in test are standard benchmark problems used in optimization. The performance shown in these tests should approximate the performance on a wide range of real-world problems.

5.1 Session A – Setup

The experimental setup in the first session is similar to the one we used to test the 6σ-PSOD operator in [13] and [12]. In this work we have chosen to test in a single high dimension: 30. We have selected two algorithms to improve with disagreements: a standard (classic) constriction-based PSO with the configuration found by Clerc in [7], that we call here SPSO: $\chi = 0.729$ with $c_1 = c_2 = 2.05$ as in [16]; the second chosen type of algorithm is a social-only PSO, initially discovered by Kennedy in [9] and studied more by Pedersen under the name "PSO-VG" ([17]), with the following configuration: $w = 0.729$ with $c_2 = 1.494$.

As in our previous related work, [13] and [12], we transformed these two algorithms into their disagreement-enabled counterparts applying the RS-PSOD operator: $\Psi_{\mathrm{RS-PSOD}}(SPSO) = SPSOD_{\mathrm{RS}}$ and $\Psi_{\mathrm{RS-PSOD}}(PSO{-}VG) = PSO{-}VGD_{\mathrm{RS}}$, with $\lambda_{\mathrm{l}} = 1$ and $\lambda_{\mathrm{u}} = 2$. Both algorithms use the grid topology. For each algorithm, we measured the mean best fitness value with its standard deviation across 50 runs and the average number of riots (how many generations disagreed) for a swarm made of 50 particles. Algorithms terminate after 30000 fitness evaluations. Stagnation is detected for a threshold value $\epsilon = 0.005$ after $\Delta t_h = 10$ generations passed.

For the first test session, inspired by Chen and Li's work in [6], we selected the following test problems: Generalized Rosenbrock (L_1), Shifted Rastrigin (L_2), Shifted Schwefel (L_3) and Griewank (L_4). The reasons for picking each one, the same as in [6], are explained along the following enumeration:

1. We started with Generalized Rosenbrock to observe how our new PSODs behave on plateau functions:

$$L_1(X) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2),\ X \in [-5, 5]^n \ . \tag{19}$$

   Global optimum: $X_G = (1, \dots 1),\ L_1(X_G) = 0$.

2. Shifted Rastrigin was used to check for behavior of convergence:

$$L_2(X) = \sum_{i=1}^{n} (z_i^2 - 10\cos(2\pi z_i) + 10) + f_{\text{bias}},\ Z = X - O,\ X \in [-5, 5]^n \ , \tag{20}$$

   where $O = [o_1, \dots o_n]$ is the shifted global optimum.
   Global optimum: $X_G = O,\ L_2(X_G) = f_{\text{bias}} = -330$.

3. To see if the new PSODs retain the original PSO robustness, we employed Shifted Schwefel in tests:

$$L_3(X) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} z_j \right)^2 + f_{\text{bias}},\ Z = X - O,\ X \in [-100, 100]^n \ , \tag{21}$$

   where $O = [o_1, \dots o_n]$ is the shifted global optimum.
   Global optimum: $X_G = O,\ L_3(X_G) = f_{\text{bias}} = -450$.

4. Finally, with Griewank we looked if the new behavior helps escaping local minima:

$$L_4(X) = \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{s} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1,\ \ X \in [-600, 600]^n \ . \tag{22}$$

   Global optimum: $X_G = (0, \dots 0),\ L_4(X_G) = 0$.

5.2 Session A – Results

Table 1 and Table 2 contain the results of the performed tests:

As we can notice in Table 1, applying the variant of the algorithm with disagreements for the problem L_1 is yielding better results both on SPSO and PSO-VG. Fig. 2 shows the evolution of the best individual obtained when we

Table 1. Benchmark results for L_1 and L_2

	L₁			**L₂**		
Algorithm	Mean	Std. dev.	Riots	Mean	Std. dev.	Riots
SPSO	57.1260	40.4190	-	64.0869	18.2855	-
SPSOD$_{\mathrm{RS}}$	43.2560	29.5548	16.25	59.3327	15.9307	24.20
PSO-VG	67.3753	48.8016	-	112.9689	34.4212	-
PSO $-$ VGD$_{\mathrm{RS}}$	62.5470	37.6536	12.80	96.0294	31.0198	31.45

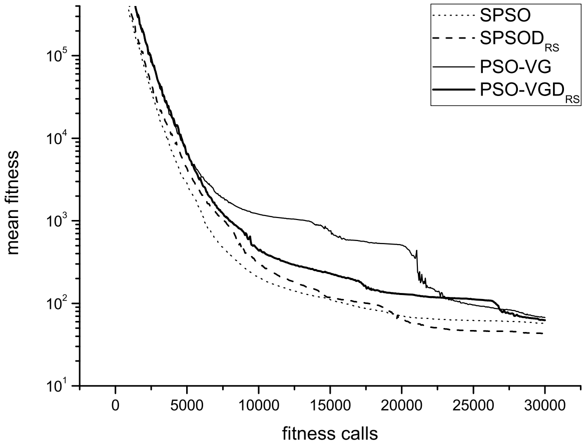


Fig. 2. Convergence graph for L_1 (30 dimensions; $s = 50$). PSODs find better solutions on plateau functions than their original PSO counterparts.

tested L_1. *The riot-on-stagnation operator is slightly beneficial on this particular problem because it can successfully mitigate stagnation on plateaus.*

In L_2's case, for the selected values of ϵ and Δt_h, the convergence of both PSOs is improved after applying the disagreements operator. Fig. 3 clearly demonstrates that PSODs converge towards a better solution in the presence of a quite high number of riots (an average of 24.20 and 31.45). PSO-VGD$_{\mathrm{RS}}$ also performs slightly better in this stagnation prone environment, but *in both cases there is a small performance improvement.*

When we tested for Shifted Schwefel (L_3), we obtained better results for our transformed PSOs than for the original variants; with the chosen parameters, *the conclusion is that the new PSODs are slightly more robust and perform better*, as can be seen in Fig. 4. Comparing the number of average riots from this test with those obtained for L_2 or L_1, we understand that a smaller but more pinpointed number of riots was generated in the latter case only when needed, therefore it did not introduce too much randomness in the swarm.

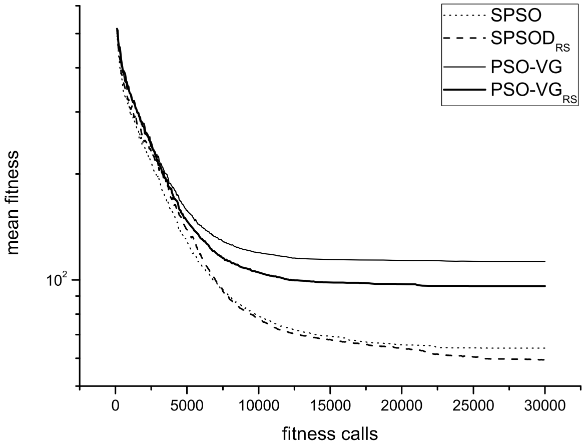


Fig. 3. Convergence graph for L_2 (30 dimensions; $s = 50$). Comparing in pairs, PSOD variants have slight better convergence than original PSOs.

Table 2. Benchmark results for L_3 and L_4

	L3			**L4**		
Algorithm	Mean	Std. dev.	Riots	Mean	Std. dev.	Riots
SPSO	1327.1377	1403.8522	-	0.0090	0.0116	-
SPSOD$_{RS}$	1122.9062	1012.4270	0.33	0.0075	0.0103	5.16
PSO-VG	5583.0198	4478.2125	-	0.0147	0.0149	-
$\mathrm{PSO-VGD_{RS}}$	4431.4144	4195.0482	7.45	0.0101	0.0078	11.38

For L_4, as it can be seen in Table 2 and from the convergence graph in Fig. 5, we got marginally improved results.

The conclusion of session A of tests is that a small amount of disagreements may give better results than no amount. We can infer that there is a range of minimum and maximum number of acceptable riots, but this needs further research to be established. ***This kind of stagnation recovery method proves itself useful in multi-modal environments and the empirical analysis confirms that a slight overall improvement is obtained for the parameters considered in the test.***

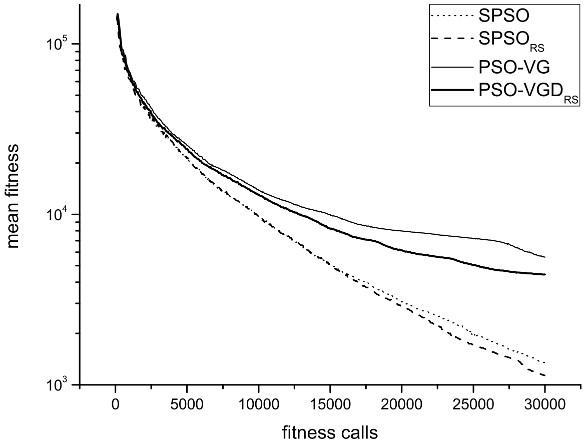


Fig. 4. Convergence graph for L_3 (30 dimensions; $s = 50$). PSODs retain the original PSO robustness.

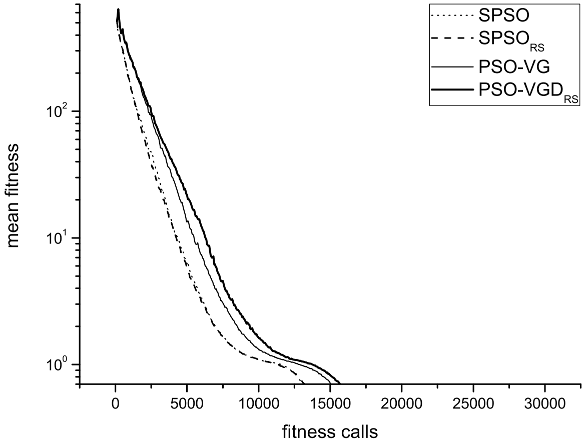


Fig. 5. Convergence graph for L_4 (30 dimensions; $s = 50$). Marginally improved convergence.

5.3 Session B – Setup

For Session B we did an extensive test on 8 benchmark functions: we added 4 more functions to those from the Session A (also used in [12]). Details about the extra functions are given below:

5. Shifted Sphere, a shifted variant of the simplest test function, the sphere (as described in [20]).

$$L_5(X) = \sum_{i=1}^{n} z_i^2 + f_{\text{bias}},\ Z = X - O,\ X \in [-100, 100]^n \ , \tag{23}$$

where $O = [o_1, \dots o_n]$ is the shifted global optimum.
Global optimum: $X_G = O,\ L_5(X_G) = f_{\text{bias}} = -450$.

6. Shifted Rosenbrock, the shifted variant of L_1, as described in [20].

$$L_6(X) = \sum_{i=1}^{n-1} (100(z_i^2 - z_{i+1})^2 + (z_i - 1)^2) + f_{\text{bias}}, \\ Z = X - O + 1,\ X \in [-100, 100]^n \ , \tag{24}$$

where $O = [o_1, \dots o_n]$ is the shifted global optimum.
Global optimum: $X_G = O,\ L_6(X_G) = f_{\text{bias}} = 390$.

7. Ackley, a multi-modal "tornado"-shaped function.

$$L_7(X) = -20 \cdot \exp\left(-0.2\sqrt{\tfrac{1}{n}\textstyle\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\tfrac{1}{n}\textstyle\sum_{i=1}^{n} \cos(2\pi \cdot x_i)\right) + \\ +\ 20 + e,\ \ X \in [-20, 20]^n \ . \tag{25}$$

Global optimum: $X_G = (0, \dots 0),\ L_7(X_G) = 0$.

8. Bohachevsky 1, an unimodal, scalable and separable function.

$$L_8(X) = \textstyle\sum_{i=1}^{n-1} (x_i^2 + x_{i+1}^2 - 0.3\cos(3\pi x_i) - 0.4\cos(4\pi x_{i+1}) + 0.7), \\ X \in [-100, 100]^n \ . \tag{26}$$

Global optimum: $X_G = (0, \dots 0),\ L_8(X_G) = 0$.

All configurations are the same as given before in Section A. The difference is that stagnation is detected quicker, after $\Delta t_h = 5$ generations passed.In this scenario, we reseeded the random number generator more often, at each iteration, therefore the results are quite different. We measured the best fitness and its standard deviation along with the median fitness; we also recorded the average number of riots. For illustration purposes, we have drawn the best fitness, worst fitness and average population distance graphs for a special case involving a plateau function.

5.4 Session B – Results

Table 3 shows the results obtained on Generalized Rosenbrock. *Here it can noticed a fair improvement in favor of PSODs.* This is of exceptional importance because we are dealing with a plateau function where stagnation occurs most of the time. *Obtaining good results on plateau functions is the main objective of a technique that aims at repairing stagnation phases.* As in Session A, both the classic PSO and the social-only one have a better performance on this function. When looking at the results we should take into consideration that the stagnation is detected earlier (in 5 generations vs. 10 in Session A).

Table 3. Results from Session B on L_1

Algorithm	Mean	Std. dev.	Median	Riots
SPSO	165.3517	422.1568	76.9643	0.00
SPSOD$_{\mathrm{RS}}$	109.4401	176.3451	78.1319	41.60
PSO–VG	7732.8096	30776.5721	1370.3259	0.00
$\mathrm{PSO} - \mathrm{VGD}_{\mathrm{RS}}$	5108.7225	9494.4679	987.2933	40.23

For the second test function, which is a CEC 2005 shifted Rastrigin, *the results are in the favor of the disagreements variants of the algorithms* (as shown in Table 4). This reconfirms the good results obtained in the first round. It should be noticed that the number of riots is significantly higher and keep to be higher than in Session A throughout all Session B of testing because of the smaller detection interval.

Table 4. Results from Session B on L_2

Algorithm	Mean	Std. dev.	Median	Riots
SPSO	76.2976	19.6627	75.4126	0.00
SPSOD$_{\mathrm{RS}}$	73.8956	19.3179	73.7137	64.26
PSO–VG	136.1342	33.4431	134.5831	0.00
$\mathrm{PSO} - \mathrm{VGD}_{\mathrm{RS}}$	128.4655	30.8897	125.7556	65.85

On Shifted Schwefel, whose results are found in Table 5, *the performance of the newly developed algorithms is worse than the one on the original configurations.* However the difference is not significant.

Table 5. Results from Session B on L_3

Algorithm	Mean	Std. dev.	Median	Riots
SPSO	3015.6738	3899.0972	1931.8083	0.00
SPSOD$_{\mathrm{RS}}$	3352.9564	3362.3291	2097.0997	21.08
PSO–VG	10596.2091	7690.6586	9198.2555	0.00
$\mathrm{PSO} - \mathrm{VGD}_{\mathrm{RS}}$	12087.4852	7810.2853	10893.6008	40.83

On Griewank (see Table 6), in the given conditions, we got mixed results. There is a small performance penalty when using disagreements-enabled PSOs. *On L_3 and L_4 the results do not recommend the usage of the disagreements technique, but because in the first round of testing we obtained good performance then it might be recommended that a parameter tuning should be performed.*

Table 6. Results from Session B on L_4

Algorithm	Mean	Std. dev.	Median	Riots
SPSO	0.8985	1.7813	0.3122	0.00
SPSOD$_{\mathrm{RS}}$	0.8234	1.5424	0.3228	39.00
PSO–VG	16.5073	22.7002	7.5948	0.00
$\mathrm{PSO} - \mathrm{VGD}_{\mathrm{RS}}$	17.5706	23.6572	9.2657	50.87

On Shifted Sphere also, L_5, the results in Table 7 are showing no benefit in applying the idea of disagreements. *They show that an aggressive configuration with too early stagnation detection might throw the search algorithm out of promising regions.*

Table 7. Results from Session B on L_5

Algorithm	Mean	Std. dev.	Median	Riots
SPSO	31.9002	143.8044	0.3528	0.00
SPSOD$_{\mathrm{RS}}$	36.4193	153.7945	0.2931	32.82
PSO–VG	1015.4969	1764.6044	519.7061	0.00
$\mathrm{PSO} - \mathrm{VGD}_{\mathrm{RS}}$	1084.1458	1407.9120	597.2662	45.60

For our study, the results for Shifted Rosenbrock in Table 8, the shifted variant of L_1, which is also a plateau, are of greater importance than others. *In this case the improvement brought by disagreements is significant for both types of PSO in the study.*

Table 8. Results from Session B on L_6

Algorithm	Mean	Std. dev.	Median	Riots
SPSO	3614315.8716	15605767.2823	1451.6981	0.00
SPSOD$_{\mathrm{RS}}$	269769.5854	1694392.5214	1476.0156	28.18
PSO–VG	74898901.9595	141500339.1900	36151995.7048	0.00
$\mathrm{PSO} - \mathrm{VGD}_{\mathrm{RS}}$	46759829.6188	101363339.6820	10799342.9023	41.25

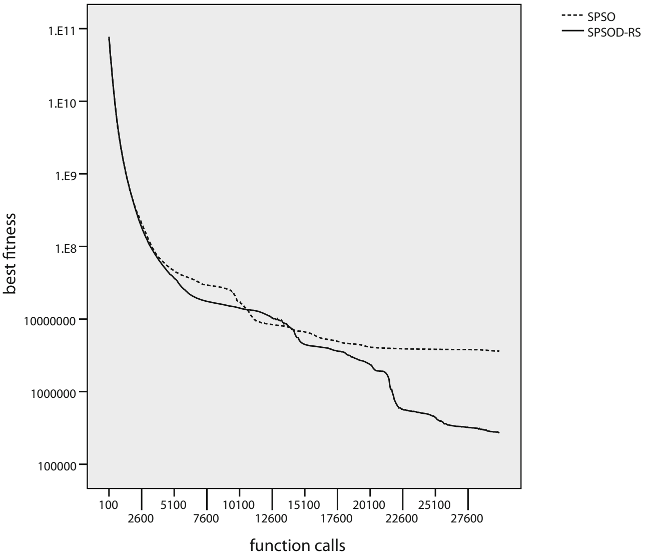


Fig. 6. Best fitness graph for SPSO on L_6

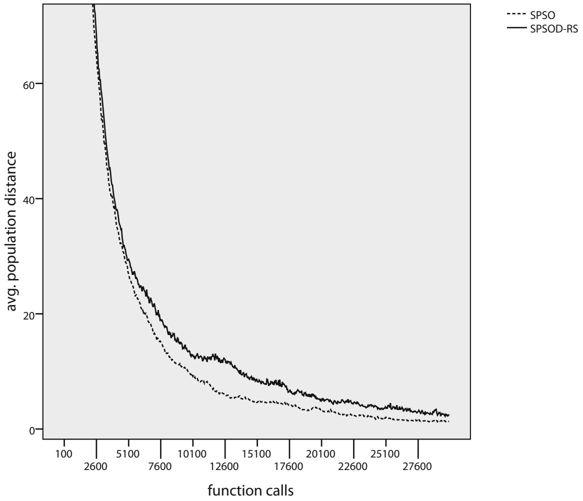


Fig. 7. Avg. pop. distance graph for SPSO on L_6

In Fig. 6 one may notice that the best individuals of SPSOD_{RS} win the competition against the standard algorithm. The diversity inside the population is higher, and the metric used in Fig. 7, the average distance between individuals, proves it.

The results for Ackley's function are given in Table 9.*There is a marginal improvement when using the disagreements.* Ackley's function is of special interest due to its funnel-like shape. It is particularly difficult because there are lots of local minima inside the funnel.

Table 10 contains the results for Bohachevsky, an unimodal function for which it seems that *the results are mixed: for the standard configuration of PSO it seems that it's better to employ disagreements, but for the social-only algorithm, it is better without them, possibly because the random changes that disagreements are bringing manage to disturb the evolution of PSO-VGs to a greater extent.*

Table 9. Results from Session B on L_7

Algorithm	Mean	Std. dev.	Median	Riots
SPSO	3.8164	2.4098	3.1609	0.00
SPSOD$_{\mathrm{RS}}$	3.3366	2.0410	2.9133	57.94
PSO–VG	11.1217	3.7621	11.0339	0.00
$\mathrm{PSO} - \mathrm{VGD}_{\mathrm{RS}}$	10.7198	3.7740	10.3852	64.78

Table 10. Results from Session B on L_8

Algorithm	Mean	Std. dev.	Median	Riots
SPSO	10.3324	12.8913	6.7159	0.00
SPSOD$_{\mathrm{RS}}$	8.1672	11.0206	5.9007	53.64
PSO–VG	60.4135	81.4418	34.5865	0.00
$\mathrm{PSO} - \mathrm{VGD}_{\mathrm{RS}}$	72.6821	75.7849	47.6797	50.55

In session B, with a special setup that makes possible an earlier detection of stagnation, the original PSOs were better in 2 out of 8 test functions: L_3 ***and*** L_5***. The disagreements-enabled PSOs outperformed in various degrees the original considered algorithms in 4 out of 8 benchmarks. For the rest of 2 test functions, we got mixed results. Analyzing the overall results, it can be noticed that PSOs with disagreements behave well in most cases and the performance penalty would not be significant even in worst scenarios. On the other hand, they clearly prove better on the most stagnation prone environments, the plateau functions.***

6 Conclusion

This paper shows that in certain conditions the method of adding the so-called "disagreements" when stagnation appears in the search process is valuable for getting improved final results.

Disagreements are a common occurrence in social groups and intuitively we applied them in a social optimization algorithm, PSO. While one can imagine many forms of disagreements, in this paper we pursued a simple approach that aims to not increase the computational cost of the algorithm: controlled randomness injection. In contrast with other similar approaches that we know about, our proposed and tested way of dealing with the problem of stagnation differs in some significant theoretical and practical realms. *The problem is solved by defining a new perspective upon modifications in PSO, the disagreements*, that in our case affect only the social component of the algorithm. They are based on the idea that some particles purposely do not want to follow the leader in the group. Although there are some similarities with mutation, the intent, mode of action and effect of disagreements are different. The mutation is mostly accidental in real-world and most evolutionary algorithms . In social groups, the disagreements appear on purpose and are the result of a particular context; the "riot-on-stagnation" operator follows this argument.

In order to find out how the derived algorithms behave on plateau test functions, if they keep their convergence rates, if they are still as robust as before and how they can help to escape local minima, we employed two different test sessions with several benchmark functions from the scientific literature to capture information about all these aspects.

It seemed that the added randomness did not impede the convergence rate and the robustness of the PSO. The extra computationally involved cost was minimal (one more use of the random generator).

Even though we obtained good results for our fixed sets of stagnation parameters, this is still a preliminary study. Future work may consist in studying the optimum amounts of disagreements (number of riots) that can be safely employed and other methods to apply them in order to improve the PSO furthermore.

Overall, we have concluded that adding disagreements when a PSO stagnates can help it tackle local minima and find a better way towards the solution in most cases for multi-modal and plateau environments. Our method yielded particularly better results for a very stagnation prone environment, the Rosenbrock's plateau. Benchmark results recommend trying this technique in various real-world applications.

We think that the results are promising enough in order to proceed doing more related research.

Acknowledgments. This work was partially supported by the strategic grant POSDRU 6/1.5/S/13, Project ID6998 (2008), co-financed by the European Social Fund Investing in People, within the Sectorial Operational Programme Human Resources Development 2007-2013.

References

[1] Acemoglu, D., Chernozhukov, V., Yildiz, M.: Learning and disagreement in an uncertain world. In: NBER Working Papers 12648, National Bureau of Economic Research Inc. (October 2006)

[2] Acemoglu, D., Como, G., Fagnani, F., Ozdaglar, A.: Opinion fluctuations and disagreement in social networks. CoRR abs/1009.2653 (2010)

[3] Bishop, J.: Stochastic searching network. In: Proceedings of the 1st IEE Conference on Artificial Neural Networks, pp. 329–331 (1989)

[4] Bishop, J., Torr, P.: The stochastic search network. In: Proceedings of the 1st IEE Conference on Artificial Neural Networks, pp. 370–387 (1992)

[5] Bloom, H.: The Lucifer Principle: A Scientific Expedition Into the Forces of History. Atlantic Monthly Press (1997)

[6] Chen, X., Li, Y.: A modified PSO structure resulting in high exploration ability with convergence guaranteed. IEEE Transactions on Systems Man and Cybernetics Part Bcybernetics 37(5), 1271–1289 (2007)

[7] Clerc, M., Kennedy, J.: The particle swarm - explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation 6(1), 58–73 (2002)

[8] Dorigo, M.: Optimization, Learning and Natural Algorithms. Ph.D. thesis, Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy (1992) (in Italian)

[9] Kennedy, J.: The particle swarm: social adaptation of knowledge. In: Proceedings of 1997 IEEE International Conference on Evolutionary Computation, ICEC 1997, pp. 303–308 (1997)

[10] Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (August 2002)

[11] Kronfeld, M., Planatscher, H., Zell, A.: The EvA2 optimization framework. In: Blum, C., Battiti, R. (eds.) LION 4. LNCS, vol. 6073, pp. 247–250. Springer, Heidelberg (2010)

[12] Lihu, A.: Disagreements – A New Social Concept in Swarm Intelligence and Evolutionary Computation. Ph.D. thesis, Politehnica University of Timişoara, Romania (2012)

[13] Lihu, A., Holban, Ş.: Particle swarm optimization with disagreements. In: Tan, Y., Shi, Y., Chai, Y., Wang, G. (eds.) ICSI 2011, Part I. LNCS, vol. 6728, pp. 46–55. Springer, Heidelberg (2011)

[14] Lihu, A., Holban, Ş.: Particle swarm optimization with disagreements on stagnation. In: Katarzyniak, R., Chiu, T.-F., Hong, C.-F., Nguyen, N.T. (eds.) Semantic Methods for Knowledge Management and Communication. SCI, vol. 381, pp. 103–113. Springer, Heidelberg (2011)

[15] Ming, J., Yupin, L., Shiyuan, Y.: Stagnation analysis in particle swarm optimization. In: Swarm Intelligence Symposium, SIS 2007, pp. 92–99. IEEE (April 2007)

[16] Parsopoulos, K., Vrahatis, M.: Particle Swarm Optimization and Intelligence: Advances and Applications. Premier Reference Source, Information Science Reference (2010)

[17] Pedersen, M.: Tuning and Simplifying Heuristical Optimization. Ph.D. thesis, University of Southampton, UK (2010)

[18] Pham, D.T., Castellani, M., Sholedolu, M., Ghanbarzadeh, A.: The bees algorithm and mechanical design optimisation. In: Filipe, J., Andrade-Cetto, J., Ferrier, J.L. (eds.) Proceedings of the Fifth International Conference on Informatics in Control, Automation and Robotics, Intelligent Control Systems and Optimization, ICINCO 2008, Funchal, Madeira, Portugal, May 11-15, pp. 250–255. INSTICC Press (2008)

[19] Reynolds, C.: Flocks, herds and schools: A distributed behavioral model. In: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1987, pp. 25–34. ACM, New York (1987)

[20] Suganthan, P., Hansen, N., Liang, J., Deb, K., Chen, Y., Auger, A., Tiwari, S.: Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. Tech. rep., Nanyang Technological University, Singapore (May 2005)

[21] Van Den Bergh, F.: An analysis of particle swarm optimizers. Ph.D. thesis, University of Pretoria, Pretoria, South Africa (2002)

[22] Worasucheep, C.: A particle swarm optimization with stagnation detection and dispersion. In: IEEE Congress on Evolutionary Computation, pp. 424–429. IEEE (2008)

[23] Xie, X., Zhang, W., Yang, Z.: Dissipative particle swarm optimization. In: Proceedings of the 2002 Congress on Evolutionary Computation, vol. 2, pp. 1456–1461. IEEE Computer Society, Washington, DC (2002)

Evolutionary Algorithm with Geographic Heuristics for Urban Public Transportation

Jolanta Koszelew and Krzysztof Ostrowski

Bialystok University of Technology, Faculty of Computer Science, Poland

Abstract. This paper presents a new evolutionary algorithm with a special geographic heuristics, which solves bi-criteria version of Routing Problem in Urban Public Transportation Networks often called Bus Routing Problem (BRP). Our solution returns a set of routes, containing at most k quasi-optimal paths with the earliest arrival in the first instance and with minimal number of transfers in the second. Effective algorithms for BRP are the heart of public transport routes planners. Proposed algorithm was compared with three another solutions for itinerary planning problem. This comparison is prepared on the base of experimental results which were performed on real-life data - Warsaw city public transport network. Conducted experiments confirm high effectiveness of the proposed method in comparison with comparable solutions for considered problem.

Keywords: multi-modal public transport networks, itinerary planning problem, time-dependent k-shortest paths problem, evolutionary algorithm, geographic heuristics.

1 Introduction

A public transport route planner provides for citizens and tourists information about available public transport journeys. The heart of such systems are effective methods for solving itinerary planning problem in a multi-modal urban public transportation networks. This paper describes a new evolutionary algorithm solving a certain version of this problem. The method returns the set of k-journeys that lexicographically optimize two criteria's: total travel time and number of transfers. A journey in a modern urban public transport network usually involves combined use of the available public transport services. Any path in such journey enhanced with a feasible schedule to traverse it is called itinerary. The itinerary planning problem in a multi-modal urban public transport network consists of finding optimal journey or set of journeys satisfying user's preferences. Generally, the itinerary planning problem constitutes a multi-criteria time-dependent routing and planning problem [12], providing a user with many alternative itineraries for a given urban journey. A public transportation journey planner is a kind of Intelligent Transportation Systems (ITS) and provides information about available public transport journeys. Users of such a system determine source and destination point of the travel, the start

N.T. Nguyen (Ed.): Transactions on CCI XII, LNCS 8240, pp. 191–207, 2013.

time, their preferences and, as a result the system returns information about optimal routes (journeys) [21]. In practice, public transport users' preferences may be various, but the most important of them are: a minimal travel time and a minimal number of changes (from one vehicle to another) [3].

The shortest path problem [2] [7] is a core model that lies at the heart of network optimization. It assumes that weight link in traditional network is static, but is not true in many fields of ITS. The optimal path problems in variable time network break through the limit of traditional shortest path problems and become foundation theory in ITS [4]. The new real problems make the optimal path computing to be more difficult than finding the shortest paths in networks with static and deterministic links, meanwhile algorithms for a scheduled transportation network are time-dependent.

Many algorithms have been developed for networks whose edge weights are not static. Most of them take into consideration a network with only one kind of link, without parallel links and returns only one route. Cooke and Halsey [7] modified Bellman's [2] "single-source with possibly negative weights" algorithm to find the shortest path between any two vertices in a time-dependent network. Dreyfus [8] made a modification to the standard Dijkstra algorithm to cope with the time-dependent shortest path problem. Orda and Rom [17] discussed how to convert the cost of discrete and continuous time networks into a simpler model and still used traditional shortest path algorithms for the time-dependent networks. Chabini [4] presented an algorithm for the problem with discrete time and edge weights are time-dependent. Ahuja [1] proved that finding the general minimum cost path in a time-dependent network is NP-hard and special approximation method must be used to solve this problem. There are some methods for BRP ehich are based on evolutionary algorithms. In 1998, Pattnaik et al. [18] formulated the BRP with fixed transit demand as an optimization problem for minimizing the overall cost, composed of the user cost plus the operator cost. In 2003, Chakroborty [5] in his paper systematically introduced the urban transit network design problem, and divided the BRP into two components: the urban transit routing problem and the urban transit scheduling problem. At the same time, the defnitions, characteristics, assessment criteria and feasible constraints of the BRP were described in detail. He also summarized the approaches for solving the BRP respectively based on the genetic algorithm in his previous publication. Finally, he published the results obtained using his methods for the Mandl's network and compared them with other researchers'.

The evolutionary algorithm with the geographic heuristics (EAG) presented in this paper generates k routes with optimal travel time and number of transfers in lexicographically order. Narrowly, resultant routes have the minimal travel time in first order and the minimal number of transfers in second order. Like the k-shortest paths algorithm Lawler, these methods generate multiple "better" paths, the user can have more choices from where he or she can select according own preferences such as total amount of fares, convenience, preferred routes and so on. Presented algorithm is an improved version of the method called simply the evolutionary algorithm (EA) described in [15]. Other EA approach to Urban Public Transportation Problem we can find in [3], [6], [22] but in this version of problem only one resultant route is returned.

The computational performance of EAG had been tested on a wide range of real-life journey planning problems defined on the urban public transport network of Warsaw, Poland. EAG was also compared with three another methods solving the itinerary planning problem. The first comparable solution realizes an cultural algorithm (CA) [20]. CAs are an evolutionary computation technique, that uses knowledge that has been generated in several times, for the same population, using a belief space. In the belief space CA stores routes with a minimal number of transfers and in each iteration of the method special operators try to improved a realization time of routes included in the population. CA is the best known evolutionary method for BRP with generation of k-routes The second method applies a local search heuristic based on a special transfer graph (TG) [9]. In this algorithm routes with smallest number of transfers are considered. Next, TG tries to add new stops to such routes, but insertion is performed only if the realization time of modified route does not increased. Computer experiments had shown that EAG generates routes as good as CA, better than TG and is significantly faster than CA. EAG is also compared with EA. The main differences between EAG and EA consist of using an geographic heuristics in the generation of the initial population step and in the mutation operator. Conducted experiments have that through the use of geographic heuristics EAG generates more efficient solutions than comparable methods. Moreover, the execution time of EAG is comparable with EA.

The remainder of this paper consists of five sections. Section 2 includes definition of itinerary planning problem and description of multi-modal urban public transport network model. In Section 3 author presents each step of EAG, and illustrates it with a simple example. Geographic heuristic which are used in EAG are described in Section 4. Section 5 is the comparison of effectiveness of EAG, EA, CA and TG methods in two aspects: computation time and quality of resultant journeys. The paper ends the section which also includes the major concluding remark.

2 Network Model and Problem Definition

A public transportation network in our model is represented as a bimodal weighted graph G $=\langle V, S, W\rangle$ [13], where V is a set of nodes, S is a set of transport links and W is a set of walk links [11]. Each node in G corresponds to a certain transport stop (bus, tram or metro stop, etc.), shortly named stop. We assume that stops are represented with numbers from 1 to n. The directed edge (i,j,l,t) is an element of the set S, if the line number l connects the stop number$i$ as a source point and the stop number j as a destination [19]. A transport link corresponds to one possibility of the connection between two stops. Each edge has a weight t which is equal to the travel time (in minutes) between nodes i and j which can be determined on the base of timetables. A set of edges is bimodal because it includes, besides directed links, undirected walk links. The undirected edge $\{i, j, t\}$ is an element of the set W, if walk time in minutes between i and j stops is not greater than $limit_w$ parameter. The value of $limit_w$ parameter has a big influence on the number of network links (density of graph). The t value for undirected edge $\{i, j, t\}$ is equal to walk time in minutes between i and j stops.

A graph representation of public transportation network is shown in Fig. 1. It is a very simple example of the network which includes only nine stops. In the real world the number of nodes is equal to 3500 for the city with about 1 million of inhabitants.

Formal definition of our problem is as follows: At the input we have:G- graph of transportation network, $timetable(l)$ - times of departures for each stops and line l, source point of the travel (o), destination point of the travel (d), starting time of the travel ($time_o$), number of the resultant paths (k) and limit for walk links ($limit_w$). At the output we want to have the set of resultant routes, containing at most k quasi-optimal paths with minimal time of realization (in minutes) in the first instance and with minimal number of transfers in the second [12].

Weight of transport link (i, j, l, t) is strongly dependent on the starting time parameter ($time_o$) and $timetable(l)$ which can be changed during the realization of the algorithm solving our problem. The t value of (i, j, l, t) link is equal to the result of subtraction: time of arrival for line l to the stop j and start time for stop i ($time_i$).

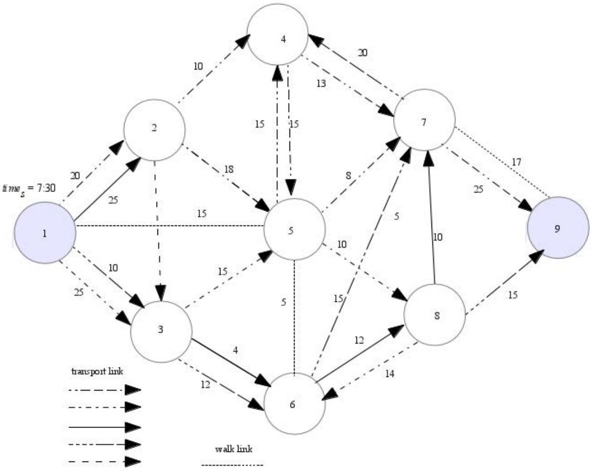


Fig. 1. Flowchart of EAG

3 Description of EAG

EAG is a hybrid algorithm which combines a standard evolutionary technique [10] with application of results of pre-computations and geographic heuristics. Such combination takes effect in achieving of a final population with a high fitness for small size of initial population and small number of generations. In Fig. 2 and Fig. 3 we see flowcharts of EAG and EA. EA [15] is a previous version of EAG. EA doesn't use a geographic heuristics. There are two main differences between these methods. The first one consists in generation of initial population step. In EAG we use a geographic heuristics to generate one of the subsets of individuals in the initial population. Second one consists in using additional kind of mutation operator called Mutation B in EAG. Mutation B also uses a geographic heuristics.

EAG routes determination is a multi-stage process involving following stepsdescribed below. This description does not contain details on the use of geographic heuristics. How to use geographic heuristics will be described in Section 4.

1. Pre-computation: Before first running of the EAG two kind of sets of routes templates are determined: $T_{o,d}$ - sets of all possible templates of routes with the minimal number of transfers for each pair of o and d stops, $R_{o,d}$ - sets of all possible templates of routes with the minimal number of stops for each pair of o and d stops. A template of a route includes only consecutive stops, without travel time value on connections. We can determine these sets on the base of transfer graph G_{tr}. G_{tr} is the time independent graph $G_{tr} =< V, E >$, where V is a set of stops and E is a set of directed transfer connections. The edge (i, j) is the transfer connection in G_{tr} , if there is at least one route without transfers between stops i and j. Methods for determining templates included in sets $T_{o,d}$ and $R_{o,d}$ are defined in [13]. We can transform very fast a given template to a route by determination travel time value for each connections of a route. An example of transformation templates to routes for the network presented in Fig. 1 is shown in Fig. 4. It's very important that time of realization of pre-computations doesn't increase the computation time of EAG, because this step is using only one time before first running of the method.
2. Initialization: EAG starts with generating an initial number of routes - P . Each route composes a chain of connections between considered consecutive stops. Two stops can be considered consecutive if they are consecutive stops for transport line or walk link. For each connection included in the route value of travel time t for a given start time $time(o)$ and timetables is determined. The initial population is generated in a special way: m_1 individuals are computed as routes with minimal number of transfers, m_2 next individuals are routes with minimal number of stops and other m_3 $(P = m_1 + m_2 + m_3)$ individuals are routes generated with application of the geographic heuristics. Individuals with minimal number of transfers are generated on the base of the set $T_{o,d}$. Chromosomes with minimal number of stops are generated on the based of the set $R_{o,d}$. The process of generation of this last part of initial routes is detail described in Section 4.

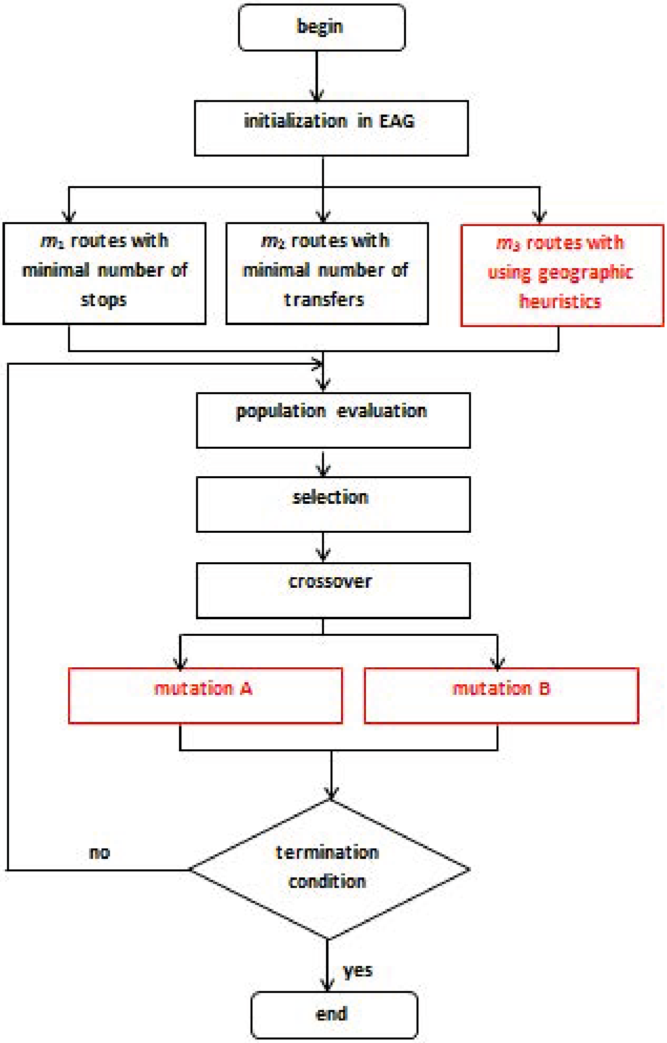

begin
initialization in EAG
m_1 routes with minimal number of stops
m_2 routes with minimal number of transfers
m_3 routes with using geographic heuristics
population evaluation
selection
crossover
mutation A
mutation B
termination condition
no
yes
end

Fig. 2. Flowchart of EAG

3. Evaluation: We calculate a value of fitness function F to evaluate the optimum nature of each route (chromosome). The fitness function should estimate the quality of individuals, according to the time of realization of the tour and number of transfers in lexicographically order.
4. Improving population: After fitness evaluation, the EAG starts to improveinitial population through ng applications of crossover and mutation. In every generation we first choose with probability pr between crossover and mutation.

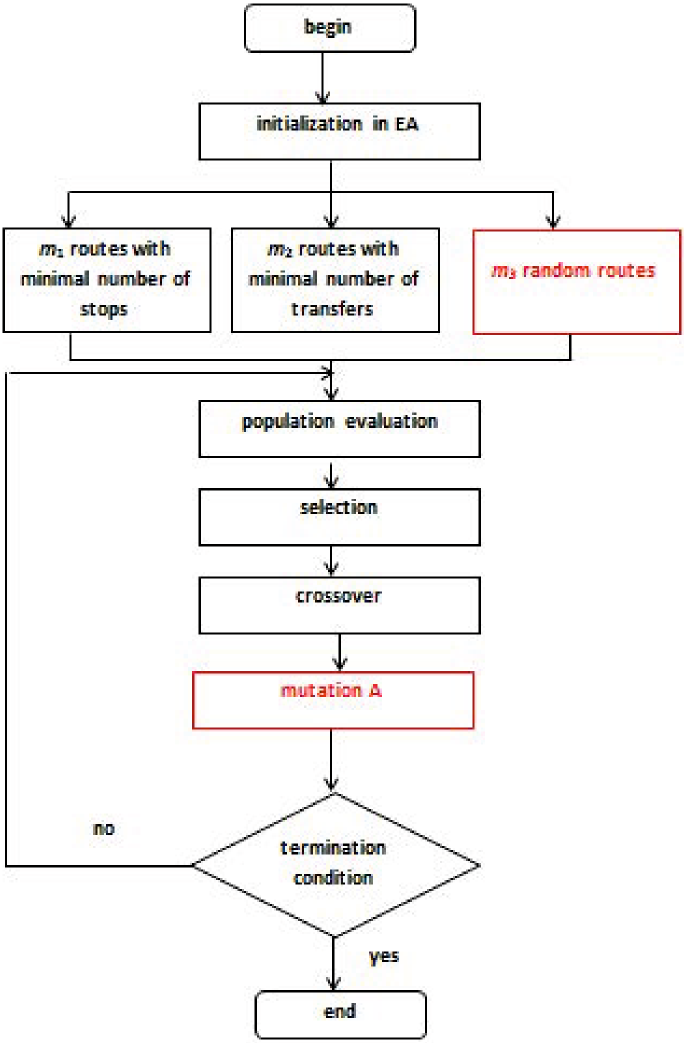


Fig. 3. Flowchart of EA

4.1 Crossover: We first select two parent individuals, according to the fitness value: the better an individual is, the bigger chance it has to be chosen. Since chromosomes lengths are different, we presented a new heuristic crossover operator, adjusted to our problem. In the first step we test if crossover can take place. If two parents do not have at least one common bus stop, crossover can not be done and parents remain unchanged. Crossover is implemented in the following way. First we choose one common bus stop, it will be the crossing point. If there are more than one

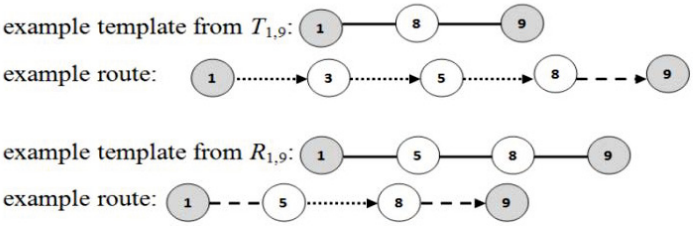

example template from $T_{1,9}$:
1
8
9
example route:
1
3
5
8
9
example template from $R_{1,9}$:
1
5
8
9
example route:
1
5
8
9

Fig. 4. Example templates included in sets $T_{1,9}$ and $R_{1,9}$ and routes obtained from these templates for network shown in Fig. 1

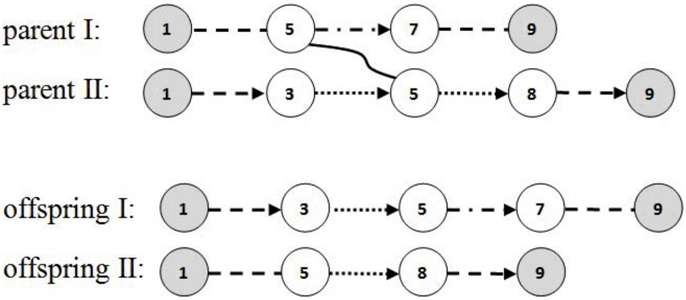


Fig. 5. Example parents and offspring individuals after crossover for network shown in Fig. 1

crossing points, we randomly choose one of them. Then we exchange fragments of tours from the crossing point to the end bus stop in two parent individuals. After crossover, we must correct offspring individuals in two ways. First we eliminate bus stop loops, then we eliminate line loops. The next step is to compute fitness function for these new individuals. Finally, we choose two best individuals from mutated chromosomes and offspring and add them to the population. Best individuals are individuals with the smallest travel time in first order and with minimal number of transfers in second order. The example of parents and offspring individuals after crossover operator is shown in Fig. 5.

4.2 With probability equals to 0.5 Mutation A operator is realized: We first choose randomly one chromosome. The next step is to randomly select two bus stops, denoted as o_1 and d_1 from the route $(o_1, d_1 \neq o, d)$. Then we randomly choose k templates of routes from o_1 to d_1 with minimal number of transfers. From these templates we select a route with minimal time of realization. If there are more than one route with minimal travel time we select one with the smallest number of transfers. This best route exchanges the fragment of a route from o_1 to d_1 in a chromosome being mutated. Then we compute fitness function for this individual and add it to the population. The example of parent individuals and offsprings after mutation operator is shown in Fig. 6.

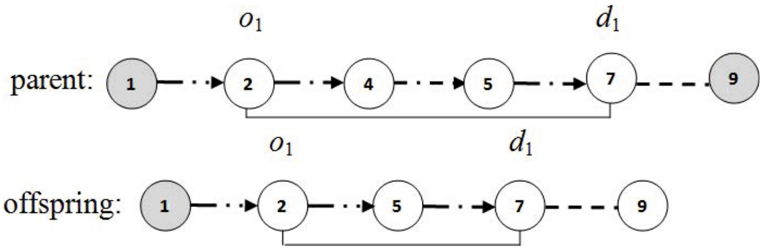


Fig. 6. Example parent and offspring individual after Mutation A for network shown in Fig. 1

4.3 With probability equals to 0.5 Mutation B operator is realized: We first choose randomly one chromosome. The next step is to randomly select two bus stops, denoted as o_1 and d_1 from the route ($o_1, d_1 \neq o, d$). Then we generate new route from o_1 and d_1 using the geographic heuristic. This generated route exchanges the fragment of a route from o_1 to d_1 in a chromosome being mutated. Then we compute fitness function for this individual and add it to the population. The process of using the geographic heuristics in Mutation B operator is described in Section 4.

5. Determining results: k resultant routes are selected from the final population by choosing k routes with the best fitness.

EAG applies results of pre-computations and the geographic heuristics during the initialization step and the second kind of mutation operator called Mutation B. Therefore, initial routes and offsprings individuals after genetic operators have much better fitness than randomly generated chromosomes. Therefore, even for small size of population and number of generations its possible to determinegood quality of resultant routes. Small values of P and ng parameters have a big influence on the reduction of a computation time of the method.

4 Geographic Heuristics

EAG in an improved version of EA presented in [15]. Differences between these methods consist on using a geographic heuristics in EAG in an initial step and during a mutation operator (Mutation B). One of the part of an initial population (m_3 routes) is generated not in the whole area of the network (like in EA) but in a geographic neighborhood of the $o - d$ specification. A geographic neighborhood of the $o - d$ specification is an geographic area designated as a collection of geographic adjacent sectors along the path between the o and d stops. More specifically, geographic neighborhood of the $o - d$ specification consists of all bus stops included in the sectors that have non-empty intersection with the section of the map ends in o and d or are neighbors of sectors (they have a common side or vertex). It is very important that we can determine a neighborhoods for each

$o - d$ specification during the pre-computation phase of the EAG. The size of sector is a parameter of the algorithm and depends on the size of the geographic area in which the public transport network is included. The example of a geographic neighborhood of the given $o - d$ specification is presented in Fig. 7.

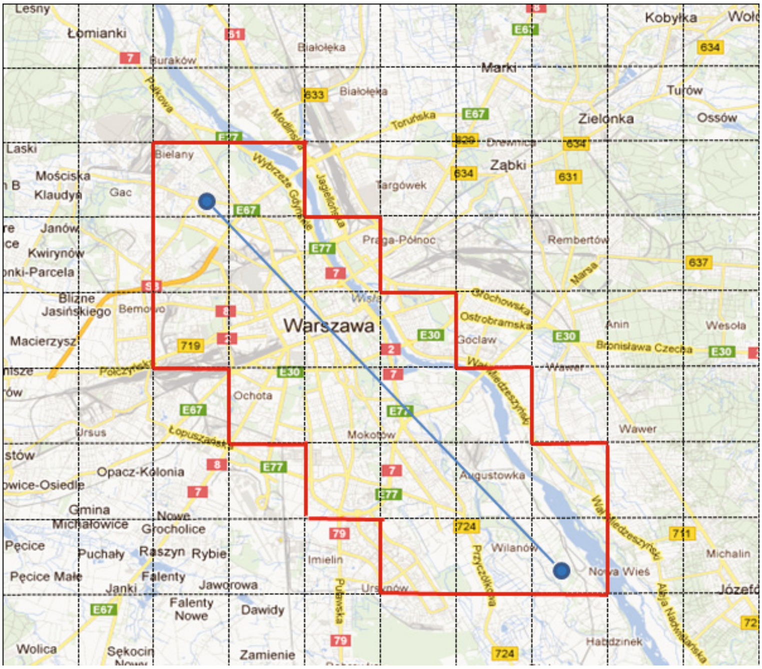


Fig. 7. Example of a geographic neighborhood (red frame) of the $o - d$ specification (blue dots)

In EA the third part of routes (m_3) included in the initial population is randomly generated in the whole area of the public transportation network. In EAG we also use random generated routes in this part of routes but we add stops which are included only in the geographic neighborhood of the $o-d$ specification. By the fact that the EAG is taken into account only the area neighborhood routes, the average fitness of the initial population is significantly higher than for EA. We see that the average travel time of routes included in the initial population is about 25% better for EAG than for EA.

The second difference between EAG and EA consists on inserting to EAG an additional type of mutation called Mutation B. This new mutation operator randomly selects two bus stops, denoted as o_1 and d_1 from the route ($o_1, d_1 \neq o, d$). Then we generate new route from o_1 and d_1 using the geographic heuristic. This generated route exchanges the fragment of a route from o_1 to d_1 in a chromosome being mutated. Then we compute fitness function for this individual and add it to the population. Geographic heuristic using for generation of a new route from o_1 and d_1 works as follows:

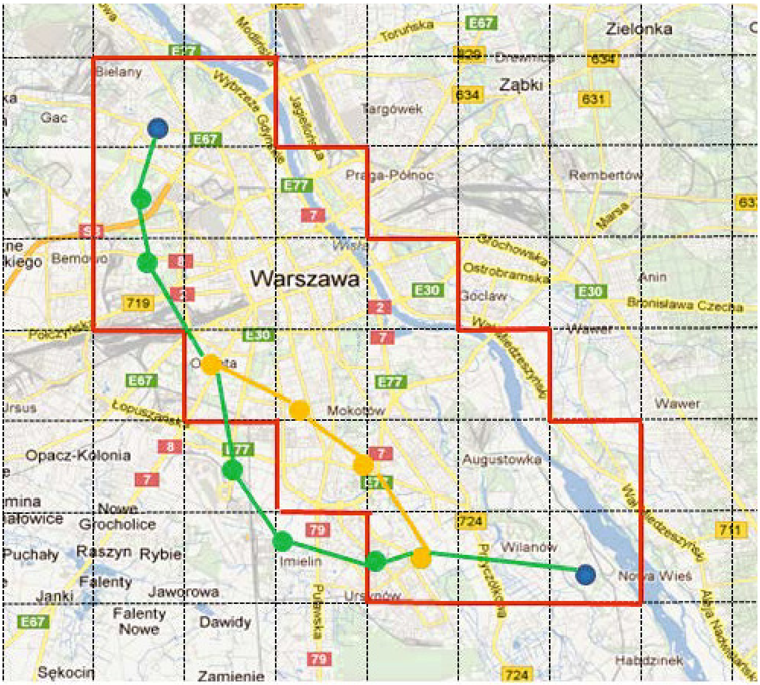


Fig. 8. Example of Mutation B realization for the given $o - d$ (blue dots) and $o_1 - d_1$ (yellow dots) specifications. Green connections - route before Mutation B, yellow connections - connections between stops o_1 and d_1) in the route after Mutation B.

1. We determine the geographic neighborhood of the $o - d$ specification.
2. For each pair (s_i, s_{i+1}) consecutive stops on the route between o_1 and d_1 we check if these stops belong to the geographic neighborhood determined in the previous step.
3. If at least one pair of adjacent vertices of the above condition is not satisfied (s_i or s_{i+1} does not belong to the geographic neighborhood for some stop s_i), then we replace the fragment from stop o_1 to stop d_1 by the shortest path (with the minimal transfers) between these stops in the transfer graph G_{tr}.

The example of realization of Mutation B operator for the given $o_1 - d_1$ specification is presented in Fig. 8. We use $T_{o,d}$ - sets of all possible templates of routes with the minimal number of transfers for pair (o_1, d_1). It's very important that all templates are determined during pre-computation step. This allows the execution time of Mutation B is comparable to the implementation of the Mutation A.

With the additional Mutation B operator significantly increased the probability of improving the quality of individuals in the population. The use ofMutation B routes in population are effectively shortened in those fragments that do not belong to the neighborhood of $o_1 - d_1$ specification.

5 Experimental Results

There were a number of computer tests conducted on real data of ransportation network in Warsaw city. This network consists of about 4200 stops, connected by about 240 bus, tram and metro lines. Values of common parameters for EGA, EA, TG and CA algorithms were following: $k = 3$, $limit_w = 15$, $P \in \{10, 20, 30\}$, $ng \in \{10, 20, 30, 40, 50\}$. The value of $limit_w$ is very important because it influences the density of network. The bigger value of $limit_w$, the more possibilities of walk links in a network. Density of network is of a key importance for time-complexity of algorithms. Additional parameters only for EAG and EA were following: $pr = 0.5$, $m_1 = 30\%P$, $m_2 = 30\%P$, $m_2 = 40\%P$. We examined routes from the center of the city to the periphery of the city (set CP), routes from the periphery of the city to the center of the city (set PC) and routes from the periphery of the city to the periphery of the city (set PP). Each of these sets includes 30 specifcation of first (o) and last (d) stops in the route which are difficult cases for each algorithm. First matter is a long distance from o and d (PP set), the second is a high density of the network in o or d localization (CP and PC sets). Algorithms was tested in a computer equipped with an Intel core2 Duo T7300, 2GHz and 2 GB RAM.

In Tab. 1 and Tab. 2 are presented the average fitness values (average travel time of routes included in the initial population) generated by EAG and EA, for $P = 30$, $limit_w = 15$, $m_1 = 30\%P$, $m_2 = 30\%P$, $m_3 = 40\%P$, $ng = 40$ (Tab. 1) and $ng = 50$ (Tab. 2). The size of sector in the geographic neighborhood was equal to 10% of the area which includes tested network. In Tab. 3 and Tab. 4 are presented arithmetic averages of travel time of the best resultant route generated by each comparable method, for considered sizes of population, numbers of generations and kinds of routes. We see that average value of the travel time of the best resultant routes for EAG is less about 10% than the corresponding values of the EA. Moreover, the fitness value of the best individual for the EA is significantly better (averageabout 19%) than the corresponding values for TG and comparable with CA(average about 1,4%). In Tab. 5 and Tab. 6 are presented

Table 1. Comparison between an average fitness value of the initial population for EAG and EA, $ng = 40$

Routes-P	EA	EAG
PC-10	99,98	66,34
CP-10	98,66	71,28
PP-10	123,14	93,33
PC-20	69,23	49,59
CP-20	99,56	82,33
PP-20	127,66	98,90
PC-30	64,67	46,55
CP-30	89,44	64,18
PP-30	116,15	97,14

Table 2. Comparison between an average fitness value of the initial population for EAG and EA, $ng = 50$

Routes-P	EA	EAG
PC-10	95,18	33,34
CP-10	126,28	110,18
PP-10	135,55	101,23
PC-20	139,30	99,99
CP-20	134,15	99,53
PP-20	154,81	129,34
PC-30	96,12	64,45
CP-30	86,35	75,28
PP-30	145,23	104,36

Table 3. Travel time of the best resultant routes generated by EAG (for each value of ng), TG and CA

Routes-P	$ng = 10$	$ng = 20$	$ng = 30$	$ng = 40$	$ng = 50$	TG	CA
PC-10	51,26	49,78	49,56	50,18	47,66	65,98	53,84
CP-10	75,12	71,14	65,34	62,43	61,23	96,28	71,28
PP-10	89,88	85,13	82,12	82,24	81,44	106,55	93,33
PC-20	45,19	36,24	35,13	33,16	34,43	59,30	39,59
CP-20	67,28	58,43	50,04	52,76	52,23	74,15	62,33
PP-20	87,15	82,18	82,24	81,65	80,98	88,90	88,90
PC-30	41,32	35,54	34,56	32,12	34,21	44,32	36,55
CP-30	57,23	55,54	54.34	55,43	53,45	76,35	55,18
PP-30	85,34	81,23	79,21	79,21	82,12	99,11	89,14

arithmetic averages of number of transfers of the best resultant route generated by EAG and EA, for considered sizes of population, numbers of generations and kinds of routes.

We see that average number of transfers of the best resultant routes for EAG is less about 8% than the corresponding values of the EA. Moreover, the average number of transfers of the best individual for the EA is significantly better (average about 38%) than the corresponding values for TG and comparable with CA (average about 11%). We compared results obtained for EAG with analogous results for EA, CA and TG and it turn out that even than ng was equal 20 and P was equal 20 EAG returns the best routes with a shorter travel time and smaller number of transfers than each comparable method. Moreover, for these values of parameters the computation time of EAG was significantly shorter than for CA and TG and comparable with execution time of EA. In Tab. 7 are shown average values of travel time (Avg-tt) and minimal number of transfers (Avg-tr) of the best route generated by EAG, EA, TG and CA for considered kind of routes. Results of EAG and EA are determined for $ng = 20$ and $P = 20$.

Table 4. Travel time of the best resultant routes generated by EA (for each value of *ng*), TG and CA

Routes-P	$ng = 10$	$ng = 20$	$ng = 30$	$ng = 40$	$ng = 50$	TG	CA
PC-10	55,98	53,98	50,67	50,67	49,98	65,98	53,84
CP-10	80,02	76,78	70,87	69,24	68,56	96,28	71,28
PP-10	96,78	94,23	90,24	88,14	88,14	106,55	93,33
PC-20	46,23	39,59	39,55	39,34	39,23	59,30	39,59
CP-20	72,76	61,00	60,05	59,67	59,56	74,15	62,33
PP-20	92,24	88,90	88,05	87,56	87,56	88,90	88,90
PC-30	45,89	39,45	39,35	38,67	38,67	44,32	36,55
CP-30	66,67	60,08	60,2	60,2	59,34	76,35	55,18
PP-30	91,89	87,24	87,12	86,25	86,25	99,11	89,14

Table 5. Number of transfers of the best resultant routes generated by EAG (for each value of *ng*), TG and CA

Routes-P	$ng = 10$	$ng = 20$	$ng = 30$	$ng = 40$	$ng = 50$	TG	CA
PC-10	3,01	3,43	3,25	3,10	3,31	4,77	3,64
CP-10	4,59	3,98	3,80	3,25	3,56	5,25	4,15
PP-10	5,38	5,20	5,10	4,92	5,27	6,53	5,45
PC-20	3,15	1,93	2,11	2,04	2,02	3,13	2,33
CP-20	3,56	2,33	2,12	2,16	2,12	2,63	2,63
PP-20	4,82	3,50	3,21	3,02	2,96	5,80	3,64
PC-30	2,15	2,01	2,15	2,15	2,15	4,15	2,01
CP-30	2,88	2,13	2,02	2,12	2,10	6,75	2,55
PP-30	4,37	3,51	3,37	3,24	3,28	5,86	3,60

Table 6. Number of transfers of the best resultant routes generated by EA (for each value of *ng*), TG and CA

Routes-P	$ng = 10$	$ng = 20$	$ng = 30$	$ng = 40$	$ng = 50$	TG	CA
PC-10	4,08	3,67	3,55	3,55	3,46	4,77	3,64
CP-10	4,89	4,15	4,10	3,94	3,87	5,25	4,15
PP-10	5,78	5,50	5,30	5,27	5,27	6,53	5,45
PC-20	3,45	2,03	2,11	2,04	2,02	3,13	2,33
CP-20	3,56	2,63	2,63	2,66	2,66	2,63	2,63
PP-20	5,02	3,70	3,63	3,24	3,24	5,80	3,64
PC-30	2,45	2,01	2,15	2,15	2,15	4,15	2,01
CP-30	3,08	2,65	2,45	2,45	2,23	6,75	2,55
PP-30	4,67	3,71	3,67	3,56	3,56	5,86	3,60

The last experiment was focused on comparison of computation time of algorithms. The results of this comparison are presented in Fig.9 and in Fig. 10.

In this experiment we tested examples of routes with a minimal number of stops, between 5 and 61. On the horizontal axis there are points representing the minimal

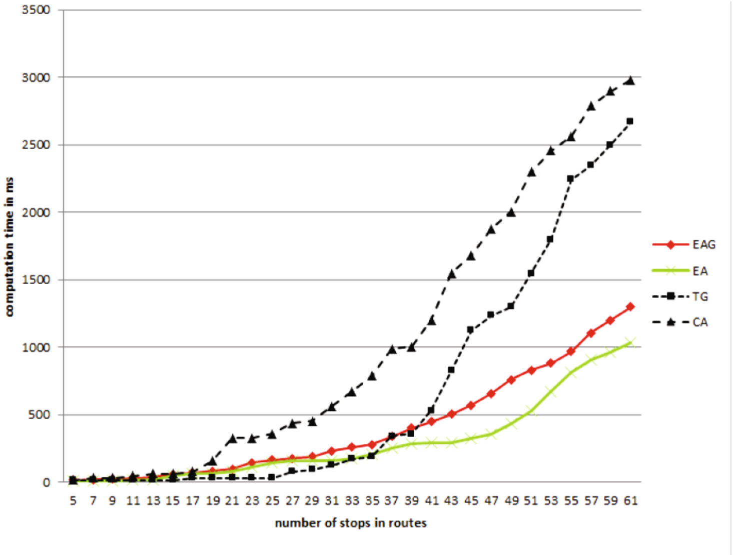


Fig. 9. The comparison of the execution time of EAG, EA, TG and CA

Table 7. The comparison of EAG, EA, TG and CA

Routes-method	Avg-tt	Avg-tr
PC-EA	39,59	2,03
CP-EA	61,00	2,63
PP-EA	88,90	3,70
PC-EAG	31,18	1,78
CP-EAG	54,05	2,33
PP-EAG	76,32	3,25
PC-TG	57,11	2,63
CP-TG	75,60	6.70
PP-TG	115,40	6,81
PC-CA	38,58	1,48
CP-CA	61,45	2,2
PP-CA	90,80	2,33

number of stops on a route. These values were computed as a result of standard *BFS* graph search method and they are correlated with difficulty of the route.

On the vertical axis there is marked time of execution. Each possible route with a given number of the minimal number of stops was tested by three algorithms at starting time at 7:30 a.m., weekday. The computation time of algorithms was averaged over every tested routes. One can see in Fig. 9 that EAG performs in significantly shorter time than CA and TG, especially for routes with minimal number of stops greater than 43. In Fig. 10 the comparison of the execution time between EAG and EA is presented. We see that using of geographic heuristic in EAG increased the execution time of this method but the differences is not significant.

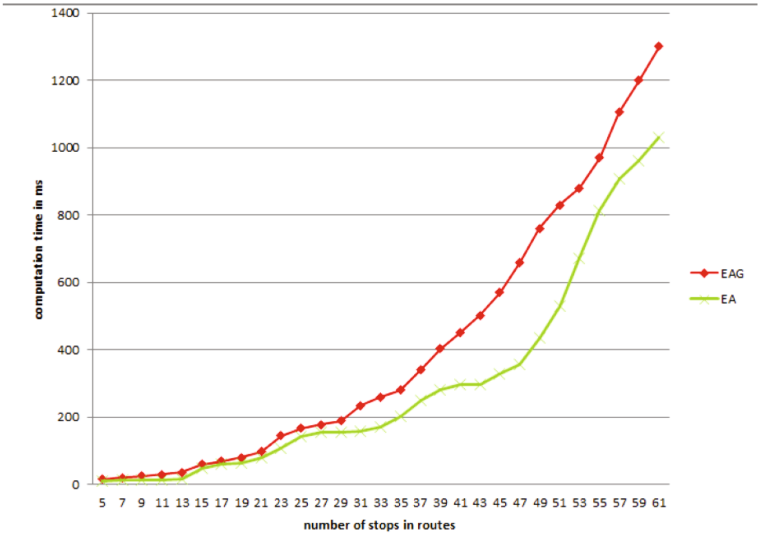


Fig. 10. The comparison of the execution time of EAG and EA

6 Conclusions

Computer experiments have shown that EAG performs much better than comparable methods. The use of geographic heuristics to generation of the initial population and the additional mutation operator resulted in considerable improvement in the fitness function values of the final population. As future work, it is intended to expand experimentation with other instances: big metropolises or regions with number of stops exceeding 5000 and small and/or rare networks. There are many centers of transfers, so called hubs, in public transportation network. The direction of future work on improvement of EAG is to develop method for detecting of hubs during the pre-computation step and to apply modified geographic heuristics which takes into account the existence of hubs.

Acknowledgments. This research was supported by S/WI/1/11.

References

[1] Ahuja, R.K., Orlin, J.B., Pallotino, S., Scutella, M.G.: Dynamic shortest path minimizing travel times and costs. Networks 41(4), 197–205 (2003)

[2] Bellman, R.E.: On a Routing Problem. Journal Quarterly of Applied Mathematics 16, 87–90 (1958)

[3] Boryczka, U., Boryczka, M.: Multi-cast ant colony system for the bus routing problem. Metaheuristics, Applied Optimization 86, 91–125 (2004)

[4] Chabini, I.: Discrete dynamic shortest path problems in transportation applications. Complexity and Algorithms with Optimal Run Time, Journal Transportation Research Records, 170–175 (1998)

[5] Chakroborty, P.: Genetic algorithms for optimal urban transit network design. In: Computer-Aided Civil and Infastructure Engineering, vol. 18, pp. 184–200 (2003)

[6] Chen, H.K., Feng, G.: Heuristics for the dynamic user-optimal route choice problem. European Journal of Operational Research 126, 13–30 (2000)

[7] Cooke, K.L., Halsey, E.: The shortest route through a network with time-dependent intermodal transit times. Journal Math. Anal. Appl. 14, 493–498 (1998)

[8] Dreyfus, S.E.: An Appraisal of Some Shortest-path Algorithms. Journal Operations Research 17, 395–412 (1969)

[9] Galves-Fernandez, C., Khadraoui, D.: remainder: Distribited Aproach for Solving Time-Dependent Problems in Multimodal Transport Networks. Advanced in Operation Research, Article ID 512613, 15 pages (2009), doi:10.1155/2009/512613

[10] Goldberg, D.E.: Genetic algorithms and their applications. WNT, Warsaw (1995)

[11] Hartley, J.K., Wu, Q.: Accommodating User Preferences in the Optimization of Public Transport Travel. International Journal of Simulation Systems, Science and Technology: Applied Modeling and Simulation, 12–25 (2004)

[12] Hansen, P.: Bicriterion path problems. In: Multicriteria Decision Making: Theory and Applications. Lecture Notes in Economics and Mathematical Systems, vol. 177, pp. 236–245 (1980)

[13] Koszelew, J.: Approximation method to route generation in public transportation network. Polish Journal of Enviromental Studies 17, 418–422 (2008)

[14] Piwonska, A., Koszelew, J.: Evolutionary algorithms find routes in public transport network with optimal time of realization. In: Mikulski, J. (ed.) TST 2010. CCIS, vol. 104, pp. 194–201. Springer, Heidelberg (2010)

[15] Koszelew, J.: An Evolutionary Algorithm for the Urban Public Transportation. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 234–243. Springer, Heidelberg (2011)

[16] Lawler, E.L.: A procedure for computing the K best solutions to discrete optimization problems and its application to the shortest path problem. Management Science 18, 401–405 (1972)

[17] Orda, A., Rom, R.: Shortest path and minimum - delay algorithms in networks with time-dependent edge-length. Journal Assoc. Computer Mach. 37(3), 607–625 (1990)

[18] Pattnaik, S.B., Mohan, S., Tom, V.M.: Urban bus transit route network design using genetic algorithm. Journal of Transportation Engineering 124, 124–368 (1998)

[19] Pyrga, E., Schultz, F., Wagner, D., Zaroliagis, C.D.: Efficient models for timetable inforamtion in public transformation systems. Journal of Experimental Algorithms 12, Article No. 2.4 (2008)

[20] Reyes, L.C., Zezzatti, C.A.O.O., Santillán, C.G., Hernández, P.H., Fuerte, M.V.: A Cultural Algorithm for the Urban Public Transportation. In: Corchado, E., Graña Romay, M., Manhaes Savio, A. (eds.) HAIS 2010, Part II. LNCS, vol. 6077, pp. 135–142. Springer, Heidelberg (2010)

[21] Wellman, M.P., Ford, M., Larson, K.: Path planning under time-dependent uncertainty. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, pp. 532–539 (1995)

[22] Zhang, Y.: Shiying Ch., Jinfeng L., Fu D.: The Application of Genetic Algorithm in Vehicle Routing Problem. In: Electronic Commerce and Security, International Symposium, International Symposium on Electronic Commerce and Security, pp. 3–6 (2008)

Author Index