Editor **DANA SIMIAN**

# MODELLING AND DEVELOPMENT OF INTELLIGENT SYSTEMS

Proceedings of the Second International Conference on
**MODELLING AND DEVELOPMENT OF INTELLIGENT SYSTEMS**

SIBIU, ROMANIA, September 29 - October 02, 2011

*Editor   DANA SIMIAN*

# MODELLING AND DEVELOPMENT OF INTELLIGENT SYSTEMS

Proceeding of  the Second International Conference
"Modelling and Development of Intelligent Systems"
September 29 - October 02, 2011, Sibiu, ROMANIA

**Lucian Blaga University Press, Sibiu**

**2011**

Lucian Blaga University Press, Sibiu, 2011

**Editor** Dana Simian

All papers in this volume were peer review by two independent reviewers

**Associate editor** Laura Florentina Stoica
**Cover design** Ralf Fabian

Proceedings of the Second International Conference
**Modelling and Development of Intelligent Systems**
September 29 - October 02, 2011, Sibiu, ROMANIA

# Preface

This volume contains refereed papers which were presented at the Second International Conference *Modelling and Development of Intelligent Systems*. The conference was held between September 29 - October 02, 2011, at Sibiu, Romania. It was organized by the Department of Informatics at the Faculty of Sciences, from " Lucian Blaga" University of Sibiu.

The aim of the conference was to bring together scientists, researchers, students, interested and working in fields which can be connected with modeling and development of intelligent systems. Specific topics of interest included but were not restricted to: evolutionary algorithms, evolutionary computing, genetic algorithms and their applications, grid computing and clustering, data mining, ontology engineering, intelligent systems for decision support, knowledge based systems, pattern recognition and model checking, motion recognition, e-learning, hybrid computation for artificial vision, knowledge reasoning for artificial vision, geometric modelling and spatial reasoning, modelling and optimization of dynamic systems, large scale optimization techniques, adaptive systems, multiagent systems, swarm intelligence, metaheuristics and applications, machine learning, self learning algorithms, mathematical models for development of intelligent systems. The talks were delivered by universities' members, researchers and students from 6 countries (Bulgaria, Ireland, Republic of Moldova, Romania, Serbia and Switzerland). During the conference a wide range of theoretical and practical problems related to the conference topics were discussed. Two plenary lectures were presented:

- Florin Boian - *Web services: models, equivalence transformations, frameworks, implementations* - "Babes-Bolyai" University of Cluj-Napoca
- Milan Tuba - *Ant colony optimization pheromone correction strategies* - Megatrend University of Belgrade

We thank all the participants for their interesting talks and discussions. We also thank the members of the scientific committee for their help in reviewing the submitted papers and for their contributions to the scientific success of the conference and to the quality of this proceedings volume.

December 2011                                                                 Dana Simian
                                                                            Conference chairman

# Scientific Committee

Prof. PhD **Octavian Agratini**, Babes-Bolyai University of Cluj Napoca, Romania

Prof. PhD **Kiril Alexiev**, Bulgarian Academy of Sciences, Bulgaria

Prof. PhD **Florian Boian**, Babes-Bolyai University of Cluj Napoca, Romania

Prof. PhD **Ioana Chiorean**, Univ. Babes Bolyai, Cluj-Napoca, Romania

Researcher **Camelia Chira**, Babes-Bolyai University of Cluj Napoca, Romania

Prof. PhD **Domenico Consoli**, Urbino University, Italy

Prof. PhD. **Oleksandr Dorokhov**, Kharkiv National University of Economics, Ukraine

Prof. PhD **Stefka Fidanova**, Institute for Parallel Processing, Bulgarian Academy of Sciences, Bulgaria

Prof. PhD **Vasile Georgescu**, University of Craiova, Romania

Prof. PhD **Dejan Gjorgjevik**, Faculty of Electrical Engineering and Information Technologies in Skopje, Macedonia

Prof. PhD **Heiner Gonska**, Duissburg – Essen University, Germany

Prof. PhD **Gheorghe Grigoras**, "Al. I. Cuza" University, Iasi, Romania

Prof. PhD **Paul Corneliu Iacob**, "Transilvania" University, Brasov, Romania

Prof. PhD **Julian Ławrynowicz**, University of Lodz, Polish Academy of Sciences, Poland

Prof. PhD **Suzana Loskovska**, Faculty of Electrical Engineering and Information Technologies in Skopje, Macedonia

Prof. PhD **Daniela Marinescu**, Transilvania University of Brasov, Romania

Prof. PhD **Nikos Mastorakis**, Hellenic Naval Academy, Greece

Prof. PhD **Ioana Moisil**, Lucian Blaga University of Sibiu, Romania

Prof. PhD ing. **Cornelia Novac**, "Dunarea de Jos" University, Galati, Romania

Prof. PhD **Ioana Cristina Plajer**, "Transilvania" University, Brasov, Romania

Prof. PhD **Dana Simian**, "Lucian Blaga" University of Sibiu, Romania

Lect. PhD **Florin Stoica**, "Lucian Blaga" University, Sibiu, Romania

Prof. PhD **Arpad Takaci**, University of Novi Sad, Serbia

Prof. PhD **Milan Tuba**, Megatrend University of Belgrade, Serbia

Prof. PhD **Cornelia Tudorie**, "Dunarea de Jos" University, Galati, Romania

Prof. PhD **Dan Eugen Ulmet**, University of Applied Sciences Esslingen, Germania

Lect. PhD **Anca Vasilescu**, Transilvania University, Braşov, Romania

Prof. PhD **Lubin Vulkov**, University "Angel Kunchev" of Rousse, Bulgaria

# Table of Contents

# A Methodology for Investigation the Change in Visual Detection of Motion in the Elderly

**Kiril Alexiev, Nadejda Bocheva, Simeon Stefanov**

**Abstract**

The current research is focused on constructing tools for modelling and analysis of changes in motion perception that occur with age. Created software tools allow conducting psychophysical experiments for estimating the sensitivity to motion direction of two age groups. The results were analysed and compared to evaluate the major changes in cognition with age. The differences in visual motion integration and in the decision-making strategies were assessed by a procedure for objective scenario estimation. In the framework of this procedure a trajectory detector estimates and classifies useful statistics for each test. A special measure is proposed for estimation of the temporal characteristics of the random scenario that determines the correctness of observer's decision. The interpretation of the results reveals new information about the age-related characteristics of visual processing of motion information. Moreover, it allows detecting subjects with significant degradation of visual processing.

## 1   Introduction

The complex process of ageing has been investigated intensively in the last years by scientists from very different fields – medical, biological, social, economic, etc. Understanding the age-related changes in the brain is important for improving the quality of life of the elderly by developing appropriate interventions and ways to slow-down and/or to postpone the deterioration in the cognitive functioning and task performance. It also provides knowledge about brain plasticity and adaptability at the neurobiological, cognitive and behavioural levels. The present study investigates the differences in visual motion perception between two age groups – younger (mean age 19.5) and older (mean age 73.9). Motion information is important for many everyday tasks like navigation, collision avoidance, figure-ground segregation, three-dimensional shape recovery, etc. These tasks are of vital significance for the survival of the individual. Therefore, the study of differences in the perception of motion of the two age groups may reveal some of the brain mechanisms that compensate the impairment of visual functions with aging.

The experiments were focused on the age-related changes in the sensitivity to motion direction of dynamic stimuli. The task of the observers was to indicate whether the mean direction of motion appeared to the left or to the right of the vertical. A fixed proportion of moving elements translated in random directions, while the direction of motion of the rest is taken from uniform distribution centred slightly to the left or to the right from the vertical. In order to perform the task

the observers have to integrate the motion direction of multiple elements. However, the task performance also depends on their ability to ignore the irrelevant information provided by the noisy elements. Age-related deficits in task performance may be expected due to the increase in the internal noise in the ageing visual system [1] or to the impaired attentional capacities of the elderly people [2] and to their inability to scale attention to the scope of relevant stimuli [3].

The experimental design used in the study resembles the classification image paradigm [4],[5] and [6] for a recent review. In a typical experiment of this type the observers classify signals presented against an additive background noise in a numerous number of trials. The noise present in the correctly classified trials and in the incorrectly classified trials is analyzed in order to reveal the perceptual "template" used for solving the task e.g. to infer the stimulus features that determine the perceptual decisions. The classification image paradigm is also useful for estimating the internal noise in the task performance [7]. Our stimuli could be described as a signal modulated by noise in the presence of background noise. This stimulation increases the perceptual load and allows not only to evaluate the perceptual abilities of the subjects, but also to reveal differences in performance due to attentional processes and different decision strategies. To achieve this goal, however, we need to create new approaches and tools of analysis. The main aim of the present paper is to present these new analytical procedures for assessing the age-related differences in discrimination of global motion direction.

Details of our experimental design and its relation to the classification image paradigm will be given in the next chapter. Perception ability investigation is a challenging problem. It requires the development of appropriate tools to obtain accurate statistical estimates of the perception of relevant information from both age groups. The specialized stochastic scenario generator was developed as a valuable tool for easy scenario generation changing noise distributions and their parameters. It is described in the third chapter. To analyze the results of the tests two samples were introduced – frame statistics and trajectory statistics. These statistics describe in some sense the objective reality i.e. what the subjects see on the monitor. The chosen statistics are explained in the fourth chapter. The processing of statistics data with joined data of subjects' answers gives us some knowledge about the human motion information processing and about the change in the cognitive processes with age. The used methods and the interpretation of results are given in the fifth chapter. The summary of the paper and data analysis are included in the conclusion of the paper.

## 2 Experiment Description

The subject sat at a distance of 114 cm from the computer screen (21" Dell Trinitron with Nvidia Quadro 900XGL graphic board). The monitor operated in a 1280 x 1024 resolution mode. A chin-rest was used to maintain a fixed distance to the screen. The refresh rate of the monitor was 85 Hz. The observation was binocular.

The number of participants in each group is 9. The age of the younger subjects is in the interval 16-24 years, 6 of them are male. The age of older participants varies between 66 to 82 years, 4 of them are male. All participants have normal or corrected to normal vision and have passed eye examination. None of them reported having any major health problems. The younger subjects gave their responses by pressing a mouse key, while the older subjects gave an oral response and their answer was recorded by the experimenter. Each observer took part in 5000 trials, the last 500 of which were the same as the first 500. So, there are 4500 unique trials. The stimuli consisted of 25 frames movie sequences showing 48 moving elements. Elements were isotropic Laplacian-of-Gaussians with radius $r = 0.5$ deg and $\sigma = 0.125$ deg. The elements moved in a circular aperture with radius of 7.0 deg, positioned in the middle of the computer screen. The stimuli were generated and presented with the help of Psychophysics Toolbox [8], which is used for user controlled scenario visualization with strict time constraints.

Due to the complexity of the problem and its dependence on numerous factors several special measures were introduced. An initial normalization of the participants in the test is carried out aiming to reach one and the same level of correct responses in participants' answers ~ 75 per cent.

# 3  Scenario Generator

Stochastic generator is designed to produce easily a video stream of random moving elements with chosen stochastic characteristics. A typical task in a classification-image experiment is to discriminate between two patterns in the presence of noise. In our experiment the patterns are defined as top-down movement at a mean angle of $-10^0$ and $+10^0$. The stochastic generator creates a number of elements in each frame of the video flow, divided in two groups – the group of elements with restricted directions of motion and the group of stimuli with randomly generated motion directions. We will label the elements of the first group "signal" and the elements of the second – "noise". The input parameters of the generator characterize the number, type of the stimuli and stochastic characteristics of the elements:

*nDots* - Specifies the number of the signal elements with generated trajectories;

*rDots* - Specifies the number of noise trajectories generated with random directions;

*DotType* - Sets the visual type of the generated stimuli. Three different visualization patterns are realized – elements with equal (flat) intensity, elements with Gaussian distributed intensity and elements with complimentary to LOG distribution of intensity. They are depicted on fig. 1:



a) element with flat intensity (equal point intensity)  b) Element with Gaussian distributed intensity  c) Element with complimentary to LOG (Laplacian Of Gaussian) distribution of intensity
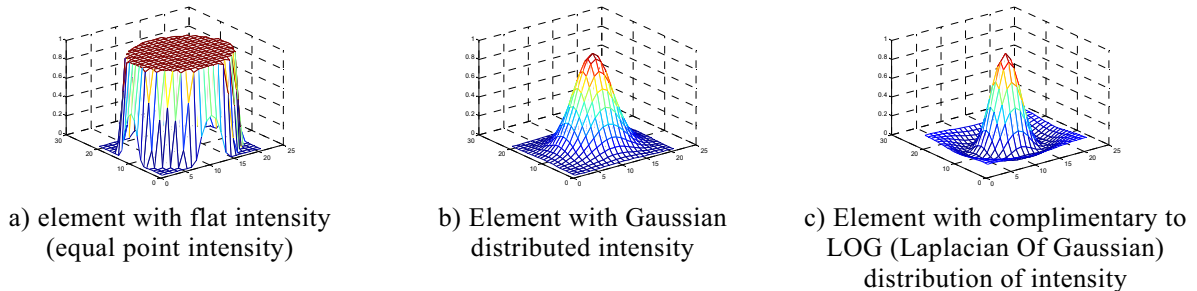
Fig. 1 Visualization patterns of the moving elements

*dotRadius* - size of the element; positive number, depending on the dot type;

*Direction* - the direction of translation in degrees;

*Speed* - the displacement of the element between two frames in pixels;

*Area* - Aperture size (in pixels);

*FrameN* - number of frames for the sequence;

*Data_flag* selects the variant of realization: 1 - Noise; 2 - Signal; 3 Signal + Noise.

The chosen parameters in the experiment are as follow: *DotType* is complimentary to LOG distribution of intensity; *dotRadius* is 16 pixels; *Area* is 550 pixels, *Speed* is equal to 10 pixels per a frame.

The total number of elements in a motion sequence was 48 and it remained constant for all frames and for all observers. The starting positions of the elements were uniformly distributed in the aperture. The stochastic characteristic of each element remained constant for a trial. If an element went outside the circular aperture the new one was generated at a new random position in the aperture, but with the same motion characteristics. The number of signal elements (respectively noisy complementing elements) is determined adaptively at the initial normalization step and it is unique for each participant.

The generated video frames were presented on fig. 2.

Several types of scenario uncertainty were introduced. First of all, every frame was randomly generated. Even in the case of constant parameters of the chosen distributions, the concrete realization (a frame) was unique. To preserve the opportunity to regenerate the same scenario, the initial values of the seeds of random generators were registered.

The group of the noisy elements was generated with random start positions (uniformly distributed) of each element and random direction (also uniformly distributed). The *Speed* is constant.

The group of the signal elements was generated in more complex manner. They had two different additional types of uncertainty – in direction and in velocity.
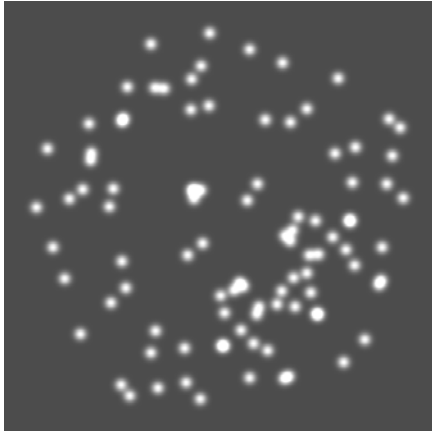


Fig. 2 A frame from generated video flow
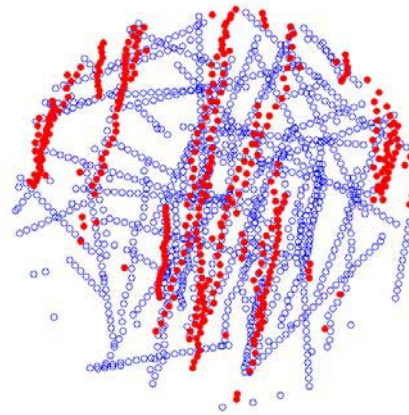


Fig. 3 The fused image of a scenario with 25 frames

The parameters of direction uncertainty were as follows:

*NoiseD* - the range of the noise distribution of the direction - in degrees;

*NoiseDtype* - noise distribution for direction with two alternatives – uniform and normal distributions and two forms of evaluation: generation for each frame or ones for the whole trial (25 frames).

The velocity uncertainty was introduced in a similar way. The mean velocity of the signal and noise elements in all trials was 6.4 ang. deg/sec.

The chosen parameters for signal stimuli in the experiment were as follow: *NoiseD* range is equal to $[-30^0, +30^0]$; *NoiseDtype* was uniformly distributed, generated for each frame; the range of velocity uncertainty was equal of [-50%, +50%] of the value of *Speed*, the value of velocity was generated uniformly in the beginning and it was constant for the test. The realized scenario consisted of signal trajectories that changed their direction from frame to frame and have unique speed.

If we have used only the noise elements, the experimental design would have been similar to the reverse-correlation methods used in physiology to describe the functional characteristics of the neurons [9]. This approach was applied also successfully by Gosselin & Schyns in psychophysical studies [10]. If we have used only the signal dots in the movie sequences for the discrimination task, our experimental design will closely resemble the study of Thomas & Knoublach [11]. They have used a temporal luminance modulation perturbed by uniform noise; the distribution of directions of the signal elements in our study was randomly perturbed on every frame due to the random generation of the directions and the relatively little number of signal elements. Our stimulation is more complex because we wanted to study not only the integration abilities of the visual system in processing motion information, but also to characterize the observer's strategies in conditions of high perceptual load.

Our experiment may be described as a discrimination task between two dynamic patterns. Under the word "dynamic" we consider stochastic realization of random process with given (constant) parameters. Every scenario includes multiple instances – 25 frames (1s video stream). On the fig. 3 the information of all 25 frames was superimposed and depicted.

Let describe the experiment mathematically. A dynamic linear process (trajectories in the experiment) can be described in the discrete Markov form:

$$s(t+1)=\Phi(t)s(t)$$

Here $s$ is n-dimensional state vector, $\Phi$ is known system behaviour model, described by transition matrix. The equation will be reshaped to correspond to noisy stimuli as follows:

$$\begin{pmatrix} x_{t+1} \\ y_{t+1} \\ v_x \\ v_y \end{pmatrix} = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_t \\ y_t \\ v_x \\ v_y \end{pmatrix},$$

where $x_{t=1}$ and $y_{t=1}$ are realizations of uniform distribution in the aperture circle; $v_{x,t=1}=Speed*cos(\Theta)$, $v_{y,t=1}=Speed*sin(\Theta)$; $\Theta$ are realizations of uniform distribution in the range $[0^0, 360^0]$ and $\Delta t=1$. If the condition $Area < \sqrt{x_{t+1}^2 + y_{t+1}^2}$ is fulfilled the trajectory restarts in a similar to the first frame way, preserving the generated angle α.

In the case of signal stimulus, the equations will be different:

$x_{t+1}=(Speed+\mu)*cos(\Theta+v_{t+1})+x_t$
$y_{t+1}=(Speed+\mu)*sin(\Theta+v_{t+1})+y_t$

Here $\mu$ is a realization of uniform distribution in the range *[0.5, 1.5]*; $\Theta$ is randomly chosen from the pair *-10^0,+10^0*; $v_{t+1}$ is realization of uniform distribution in the range *[-30^0, +30^0]* for the frame *t+1*.

# 4 Statistics

In this part we have to deal with several non-trivial problems like: 1. What kind of statistics to collect? 2. If we calculate plausible trajectories how to overcome the combinatorial problem? 3. How to analyse the collected statistics?

The perception of directed motion of elements in a stream of images is related to the discovery of close/linked elements in adjacent frames or to trajectory detection. Moreover, the observer's task is to detect multiple identical or nearly identical directed trajectories among many others chaotic ones. That is why we propose an analysis based on the trajectory detection. One estimate, which may influence observers' decisions, is the number of trajectories (or linked elements) in a scenario. We chose two types of statistics to collect. The first one consists of every pair of elements in two consecutive frames, closer than a fixed distance. Based on the physiological data about human visual system, we assume that for every couple of frames the perception is determined as an average of all directions of the paired elements. This statistic was called frame statistics, because it describes most of the elements' relations between successive frames. The boundary limits of elements' affiliation to this statistic are given on fig. 4, where α and β are corresponding minimal and maximal distance between two successive stimuli A and B.

The frame statistic is gathered in n different bins for each of $2*\pi/n$ directions. For our experiment the frame statistic is a matrix with $24*n$ size, where $n$ is equal to 32.

The main drawback of this statistic is that it is based on a very short interval of time and ignores the temporal integration of motion information by the human visual system.

The second statistic, called trajectory statistic, tries to overcome the pointed drawback. We collected all possible trajectories with different lengths, the elements of which satisfied the limitation depicted on fig. 4. This is a very loose definition of a trajectory that may include even

11

elements that do not move along a straight line, or even in a strip. The occurrence of additional trajectories is depicted on fig. 5. There are two generated trajectories, every one of which consists of four elements – A1, A2, A3, A4 and B2, B3, B4, B5 (blue trajectories). Here the subscript indicates the frame number to which the element belongs. If the distances between elements Ai and Bi+1 (i = 1,...,4) satisfy the condition shown on fig. 4, the observer may perceive the red trajectories in the picture. Moreover, the rule on fig. 4 does not reject any trajectory, realized as a combination of nearby positioned elements from both trajectories like A1, B2, B3, B4,...; A1, A2, B3, B4,...; etc.

The calculation of this statistic is not an easy task due to unlimited increase in the number of trajectories in some scenario realizations. The attempt to reduce the escalating number of trajectories using additional constraint on the position of the elements (regarding that a trajectory contains only elements, located in a strip with limited width– fig. 6) was not fruitful. The number of scenarios with enormous number of trajectories was reduced insignificantly.
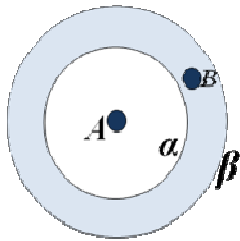


Fig. 4 Linked elements rule ($\alpha<AB<\beta$)
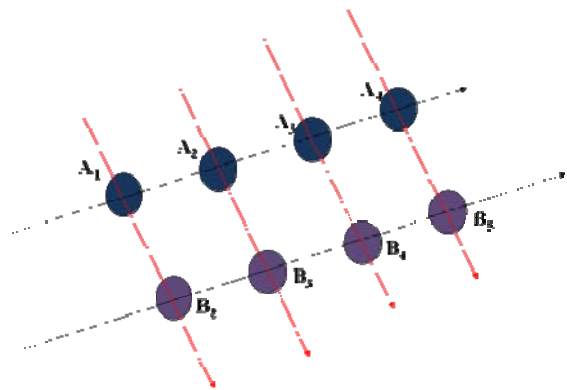


Fig. 5 Additional trajectories in the case of close trajectories



Fig. 6 Additional limitation on the stimulus
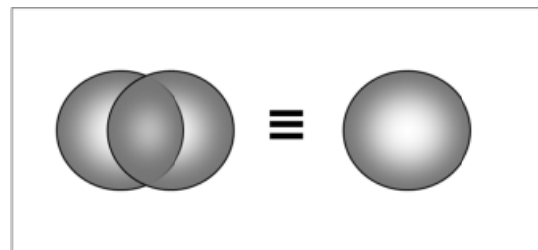


Fig. 7 The partially occluded stimulus are merged

The careful analysis of the generated trajectories shows that enormous number of trajectories occurs in the case when two randomly generated trajectories are almost collinear and very close. It was difficult to believe that the random generation of two trajectories onto one and the same place is probable, but not impossible and it happens ones or more times in the tests for almost every participant in experiments. The solution of the problem was very simple, but robust. We merged every pair/triple/... of overlapping elements in a frame and regard them as one in trajectory analysis (fig. 7). The explanation of our simplification may find additional supporting arguments due to the exposition time of a frame (40ms) and persistence of human vision. On fig. 8 the frame and trajectory statistics for one subject are shown.

To demonstrate the usefulness of the selected statistics a test was conducted. The classification image was calculated from the frame statistics, using Ahumada definitions [4]. Classification image is a linear discriminator in two hypothesis test. It is a special case of the classical

discriminant analysis [12], when stimulus pdf's are symmetric with respect to the vertical axis and the signal is corrupted by additive Gaussian noise. The classification image is calculated as linear combination of four components $\alpha_{s,r}$ for the trials segregated by the signal s (left – l or right - r) and the observer's response r (correct - c or incorrect - i). If both stimuli are equally frequent and the error rates are equal, the combination rule, originally used by Ahumada, is: CI = $\alpha_{l,i}$ - $\alpha_{l,c}$ + $\alpha_{r,c}$ – $\alpha_{r,i}$.

The most power components of the resulting image are placed in the field of the signal (fig. 9) which may be regarded as a confirmation of usefulness of the selected statistic for describing the dynamics of the perceptual template applied by the subjects.



a) frame statistic – contour image (1 – right correct detection, 2 – right incorrect detection, 3 – left correct detection, 4 – left incorrect detection)

b) trajectory statistic for all trials
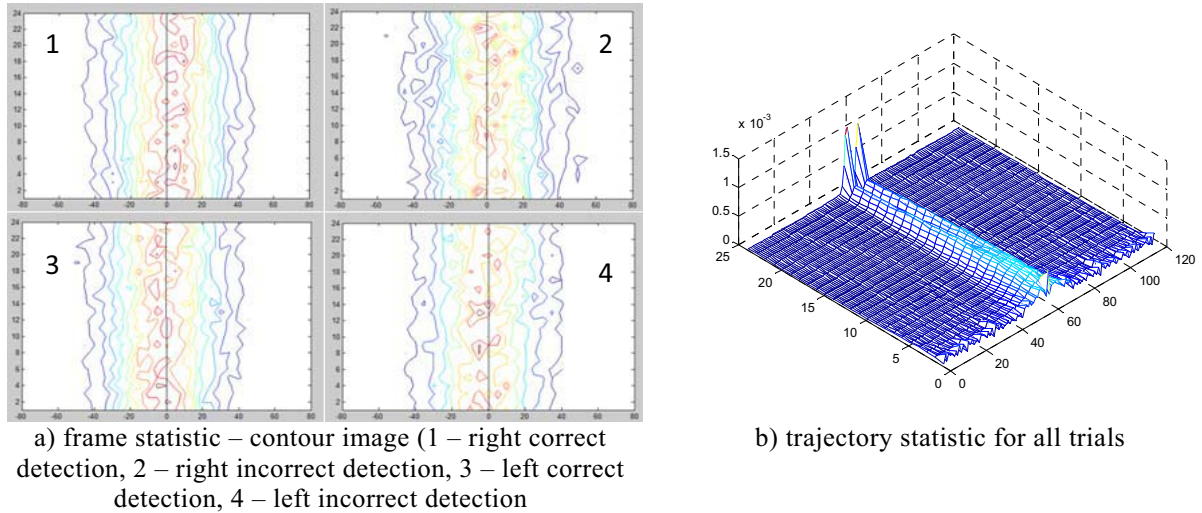
Fig. 8 The proposed statistics
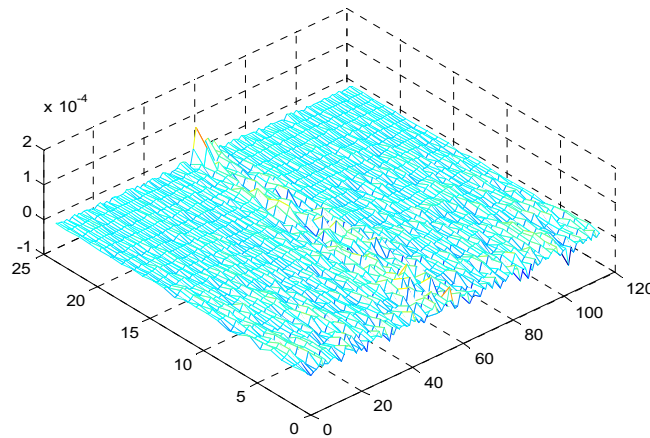


Fig. 9 Classification image (by Ahumada)

# 5  Data Analysis

Scenario analyzer detects any correlation between collected statistics and corresponding response of the observer. In this way we seek to answer the question what is the minimal duration of motion sufficient for the younger and older subjects to make the correct decision about the mean motion direction. Another important goal is to understand how a person integrates signals in the time window and detects motion in a specific direction based on the information gathered at the

actual locations of the stimuli. The most important part of the study is the detection of differences between members of "younger" and "older" groups.

We will examine the frame distributions of the mean direction (frame statistic - Fig. 8a) for different cases – correct left detection Fig. 8a-1, incorrect left detection Fig. 8a-2, correct right detection Fig. 8a-3, incorrect right detection Fig. 8a-4. The integrated results for one of participants are displayed on fig. 10. This picture reveals how the particular decision strategy of a tested person is changing in time.
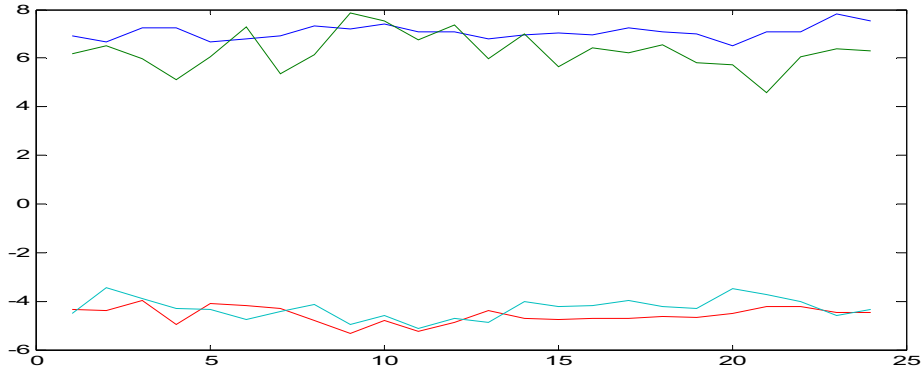


Fig. 10 Frame distributions of the mean direction (blue – correct right detection, green – incorrect right detection, red – correct left detection, cyan – incorrect left detection)

The trajectory statistic is used to assess the impact of the noise. On 3D image of this statistic a distinct area can be seen, where the distributions of the left and right trajectories are clearly distinguishable (fig. 8b). In this area, the observers have information (according trajectory statistic) to take correct decision. To determine the size of this zone it is necessary to evaluate the point (threshold) at which the unimodal distribution transforms to bimodal (fig. 11, fig. 12). For robust threshold determination a 2D Gaussian smoothing filter is applied in advance. The thresholds were evaluated for trials with participants' correct results and for trials with the incorrect ones.
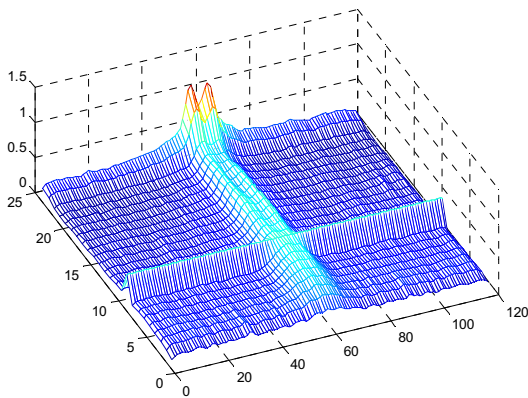


Fig. 11 Trajectory statistic – threshold detection
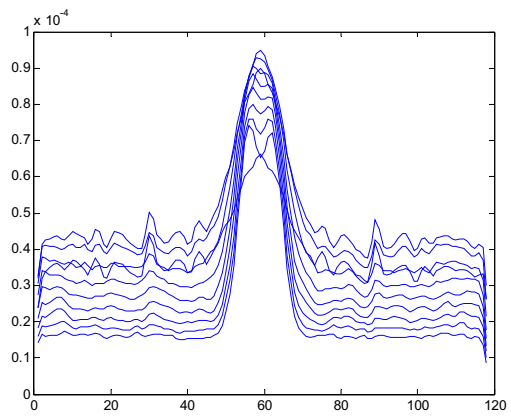


Fig. 12 Determination of the transition from unimodal distribution to bimodal

The summarized results for both groups are given in Table 1. The results show that there are objective reasons to make a wrong decision – the zone of unimodal distribution in the case of incorrect decision is always bigger than that in the case of correct decision. The two age groups have similar temporal thresholds for the correct responses; however, the older group shows greater

variability among its members. The difference in the thresholds for correct and incorrect decisions for the older group is smaller (e.g. 5.6 vs. 5.7) than that for the younger group (e.g. 5.6 vs. 6.3). This could be explained by the prior normalisation of the performance for each participant. Usually the younger participants reach the level of 75 per cent correct answers with smaller number of signal elements. The smaller samples fluctuate more around ideal distribution when the number of realizations goes to infinity.

Table 1

| YOUNG | CORRECT | INCORRECT | OLD | CORRECT | INCORRECT |
|---|---|---|---|---|---|
| b | 7 | 8 | 2 | 7 | 5 |
| c | 5 | 5 | 3 | 5 | 5 |
| d | 5 | 7 | 4 | 5 | 5 |
| emy1 | 5 | 5 | 7 | 5 | 6 |
| epy1 | 5 | 7 | 8 | 5 | 7 |
| imy1 | 6 | 7 | 11 | 8 | 7 |
| k | 5 | 6 | 12 | 4 | 4 |
| mdy1 | 6 | 5 | 13 | 6 | 7 |
| tpy1 | 6 | 7 | j | 5 | 6 |
| mean | **5.6** | **6.3** | | **5.6** | **5.7** |

We chose one representative from each group with approximately the same performance in the prior normalization test ("c" from "younger" group and "7" from "older" group) and gathered statistics on the fluctuation of their responses over time. For this purpose, the total number of experiments (4500) was divided in 9 blocks of 500 trials. The results are shown in fig. 13, 14 and 15.

The threshold fluctuation shows that in almost all cases the information about the motion direction, estimated by the temporal threshold in the trajectory statistics, was more in the case of correct answers than that in the case of wrong ones (the corresponding thresholds are lower for correct answers). The outcome of this experiment is that random generated stimuli almost always help the observer to make correct decision (7 versus 2 cases).



Fig. 13 Threshold fluctuation in time for participant "7"



Fig. 14 Variability in the number of correct and incorrect responses in time for participant "7"

15

Fig. 15 Variability in the number of correct and incorrect responses in time for participant "c"

On the next two figures the variability in the number of the correct and incorrect responses in the different blocks for the same subjects- younger – fig. 14 and older – fig. 15 are displayed. The younger person has greater variability in his performance during the course of the experiment, which can be explained by a change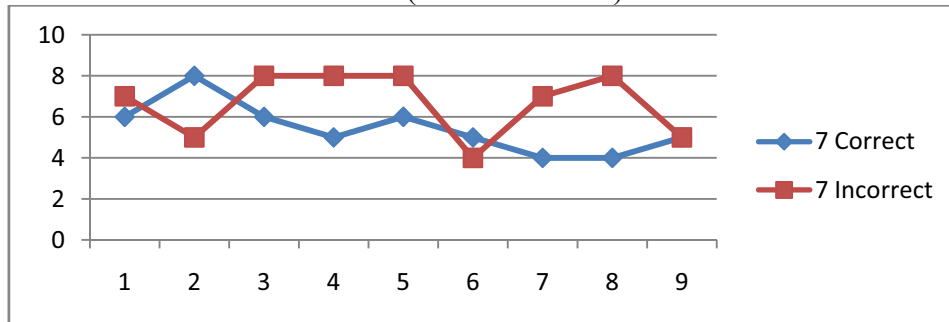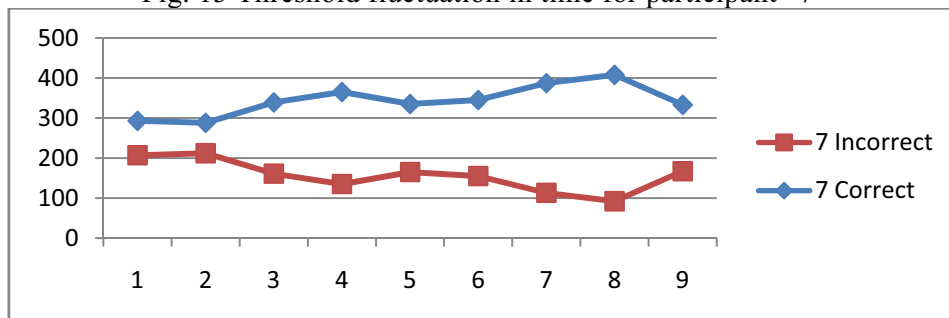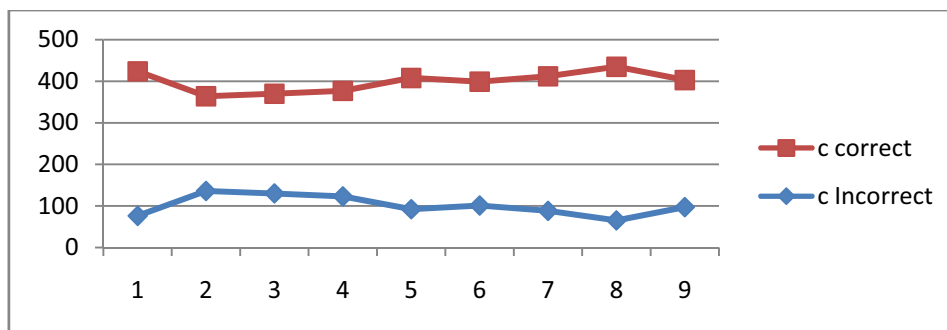 in the decision strategy of the observer in the course of the experiment. The older subject preserves almost the same level of fluctuation in task performance. It is very interesting to compare positive trends in representatives of the both groups. First of all, the positive trend in both cases means that there exists self-learning process mostly in accommodation to the task conditions and information processing (there is no feedback about correctness of answers). The positive trend is more visible for the younger subject, which indicates his greater potential.

# 6   Conclusion

This paper presents a study of the age-related changes in visual information processing. The results of the study provide a good description of the trends in the cognitive processes associated with ageing in performance of perceptual tasks. This makes them a useful tool to characterize the process of ageing and to seek association between the behavioural data and the physiological changes in the brain. The analytical procedures presented in this study and the interpretation of the results reveals new information about the age-related characteristics of visual processing of motion information. Moreover, it allows detecting subjects with significant degradation of visual processing and this may serve as a signal to initiate a search for the source of degenerative processes for that person.

Many classical statistical methods were used in this study. But it should be noted that the major contribution of the work is in the suggested procedure for objective scenario estimation. The procedure consists of three major steps – trajectory detector, collecting frame and trajectory statistics and calculation of the information threshold. Trajectory detector discovers what is displayed objectively to the observers. The suggested trajectory statistic collects trajectory records from each trial. But the most important suggestion is the proposed threshold (transition point from unimodal to bimodal distribution, or a threshold segregating the statistic in two parts – the first one where the existing evidence is not enough to make a correct decision and the second where it provides enough information. This allows us to construct a good estimator to what extent the generated noise influences over observers' decision.

The psychophysical studies have the potential not only to describe human performance but also, when combined with other data, to be used as a valuable diagnostic tool to separate normal ageing from degenerative processes. For this purpose it is important to select and to test methods that allow, based on the data from psychophysical experiments, to access the individual differences of the subjects and their deviation from the age group they belong to.

In order to be used as a diagnostic tool of degenerative processes, this approach should be tested in longitudinal studies with large set of subjects and combined with other tests on the cognitive abilities.

# References

[1] Bennett P. J., Sekuler R., Sekuler A. B., The effects of aging on motion detection and direction identification, *Vision Research,* 47*,* 799–809*,* 2007.

[2] M. Mapstone, K. Dickerson, C. J. Duffy, Distinct mechanisms of impairment in cognitive ageing and Alzheimer's disease, *Brain*, 131 (6, pp. 1618-1629), 2008.

[3] Greenwood P.M., Parasuraman R., The scaling of spatial attention in visual search and its modification in healthy aging, *Perception & Psychophysics*, 66(1), pp. 3–22, 2004.

[4] Albert J.Ahumada, Perceptual classification images from Vernier acuity masked by noise [Abstract]*, Perception*, 25, ECVP Abstract Supplement, 1996.

[5] Albert J. Ahumada, Classification image weights and internal noise level estimation, *Journal of Vision* 2, pp. 121-131, 2002.

[6] Richard F. Murray, Classification images: A review, *Journal of Vision*, 11(5):2, pp. 1-25, 2011.

[7] A. E. Burgess, B. Colborne, Visual signal detection: IV. Observer inconsistency, *Journal of Optical Society of America A*, Vol. 5, Issue 4, pp. 617-627, 1988.

[8] Brainard, D. H., The Psychophysics Toolbox, *Spatial Vision*, 10, pp. 433-436, 1997.

[9] Ringach, D., Shapley, R., Reverse correlation in neurophysiology, *Cognitive Science*, 28, pp. 147–166, 2004.

[10] Gosselin, F., Schyns, P. G., Superstitious perceptions reveal properties of internal representations, *Journal Psychological Science*, 14 (5), pp. 505–509, 2003.

[11] Thomas, J. P., Knoblauch, K., Frequency and phase contributions to the detection of temporal luminance modulation, *Journal of the Optical Society of America A*, Vol. 22, Issue 10, pp. 2257–2261, 2005.

[12] G. P. Klimov, *Applied mathematical statistic*, "Наука и изкуство", Sofia, pp. 201, 1975.(in Bulgarian)

KIRIL ALEXIEV
Institute of Information and
Communication Technologies
Mathematical Methods for Sensor
Information Processing
Sofia, 25A Acad. G. Bonchev Str.
BULGARIA
E-mail: alexiev@bas.bg

NADEJDA BOCHEVA
Institute of Neurobiology
Sensory Neurobiology
Sofia, 25 Acad. G. Bonchev Str.
BULGARIA
E-mail: nadya@percept.bas.bg

SIMEON STEFANOV
Institute of Neurobiology
Sensory Neurobiology
Sofia, 25 Acad. G. Bonchev Str.
BULGARIA
E-mail: simeon_st@percept.bas.bg

# Gallbladder description in ultrasound images ontology

**Natalie Bruc, Galina Magariu, Tatiana Verlan**

**Abstract**

The process of ontology elaboration for gallbladder ultrasound images on the base of knowledge enclosed in decision support system SonaRes is described in this article.

## 1 Introduction

In present paper we present description of the process of ontology construction for gallbladder ultrasound images. This ontology is inspired and based on the knowledge base created and being used for SonaRes – the decision support system for ultrasound diagnostics [1-5]. This system has accumulated the experience of the skillful experts-sonographists in the domain of hepato-pancreato-biliary zone examination. This experience and knowledge is well structured and formalized in this system for gallbladder and pancreas.

On the other hand, there is a powerful and attractive, from the point of view of knowledge portability, tool – ontology, which in computer science is considered as an attempt of comprehensive and detailed formalisation of some knowledge domain with the help of conceptual scheme. The ontology for gallbladder ultrasound images, presented in this paper, is an attempt to make the formalised knowledge from SonaRes system accessible in other environments and to provide interoperability.

## 2 Ontology Structure

Let's consider the structure of the proposed ontology step-by-step (see Fig. 1).

From a logical point of view there are 5 basic interrelated classes: *Organ*, *Pathology*, *Patient*, *Image* and *Complex_characteristics*. They are at the uppermost level of the ontology.

- Class *Organ* is meant for description of different organs and is related with classes *Pathology* and *Patient*:

  - with class *Pathology* – to reflect, with what pathology the certain description of organ is related;

  - with class *Patient* – to display, what patient from the given database has this pathology.

- Class *Image* is a "container" of all images of concrete patients and is related with class *Patient* to reflect the relation: which patient does the given image belong to.

- Class *Pathology* is meant for the list of pathologies structured by different areas and organs.

- Class *Patient* is a "container" of all patients of the given database. It is related with classes *Pathology* and *Image*:
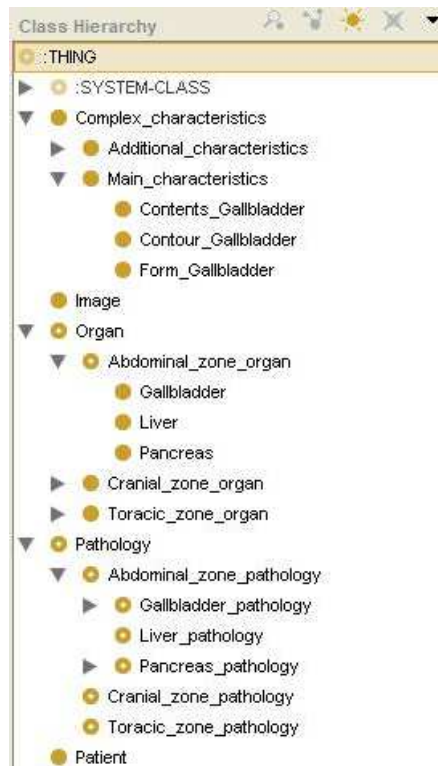
Figure 1: Class hierarchy.

– with class *Pathology* – to reflect, what pathologies the given patient has;

– with class *Image* – to reflect, what images belong to the given patient.

• Class *Complex_characteristics* is used for the description of complex organ characteristics. It is related with class *Organ*, and more precisely, with some its subclasses, depending on the organ which is described by respective characteristic. In this article we will speak about gallbladder and its characteristics. Thus, class *Complex_characteristics* contains subclasses that describe characteristics of gallbladder.

## 2.1 Basic class Organ

Class *Organ* is described by such slots as: *organ_name*, *region*, *pathology*, *image*, *patient*.

Slot *organ_name* serves for the organ name. Slot *region* indicates the area to which the organ belongs. Slot *pathology* is related with the class *Pathology*. Every instance of the concrete organ is related with a certain pathology. Slot *image* is assigned for the image reflecting pathology, indicated in the slot *pathology*. Slot *patient* is related with the class *Patient* and it is assigned for the list of patients which have the pathology specified in the slot *pathology*.

Class *Organ* at present has three subclasses: *Abdominal_zone_organ*, *Cranial_zone_organ*, *Toracic_zone_organ* – depending on the area to which the organ can belong. All these subclasses have the same slots, as class *Organ*, differing only by the allowed values of some slots.

Class *Abdominal_zone_organ* at present has 3 subclasses: *Gallbladder*, *Pancreas*, *Liver*.

## 2.2 Description of subclass Gallbladder

In subclass *Gallbladder* the relation in slot *pathology* changes. Unlike class *Organ*, the type of slot *pathology* is declared as "Class with superclass Gallbladder_pathology".

New slots are added in subclass *Gallbladder* using the knowledge tree as basis (see Fig. 2). This knowledge tree (see Fig. 3) is the representation of knowledge about gallbladder for experts-sonographists in Expert-shell of SonaRes system [6].



Figure 2: Respective slots for gallbladder description in current state of ontology.
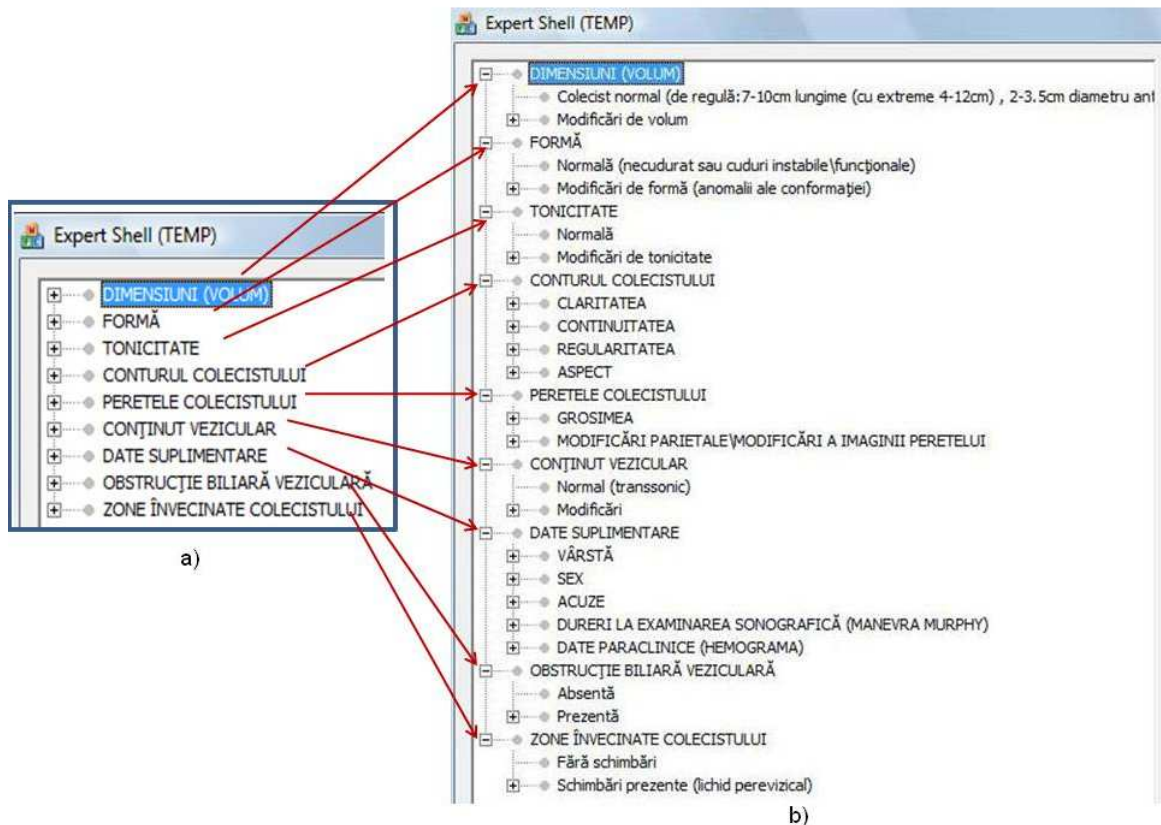


Figure 3: Knowledge tree for gallbladder in Expert-shell: a) collapsed nodes for characteristics; b) partly expanded nodes for characteristics.

At the uppermost level of the tree there are 9 basic characteristics: DIMENSIONS (VOLUME), SHAPE, TONICITY, GALLBLADDER CONTOUR, GALLBLADDER WALL, GALLBLADDER CONTENTS, SUPPLEMENTARY DATA, GALLBLADDER OBSTRUCTION, PERIVESICULAR AREA. They have different degree of complexity. Therefore the slots, corresponding to these characteristics, have different degree of complexity, too.

### 2.2.1 Slots "dimensions" and "tonicity"

For the description of such a simple characteristic as DIMENSIONS it is enough to simply introduce slot *dimensions* into subclass *Gallbladder*.

Let us give the values and argumentation for some facets for slot *dimensions* (see Fig. 4):
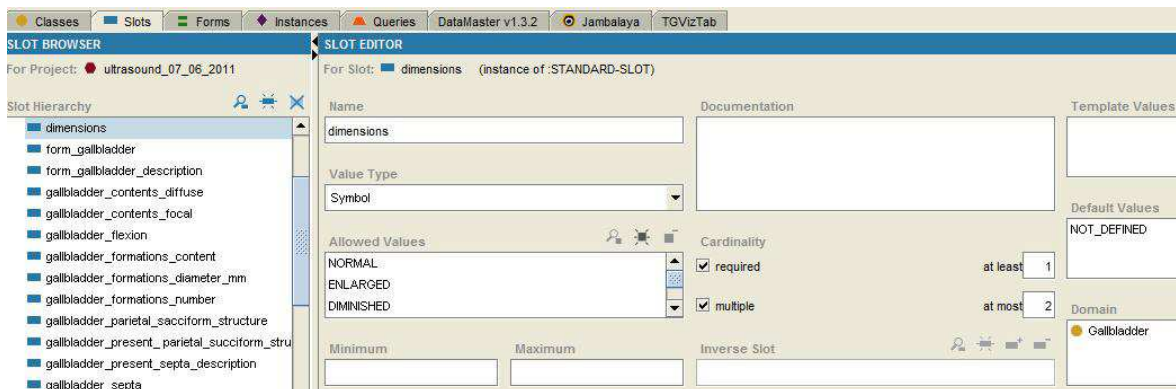


Figure 4: Facets for slot "dimensions".

- Value type = Symbol – then allowed values for this slot can be indicated. In compliance with the knowledge tree, they are the following: NORMAL, ENLARGED, DIMINISHED, NOT_DEFINED, ANY_VALUE. The value NOT_DEFINED is used in the case when the gallbladder dimension is not defined. The value ANY_VALUE is used in the case when the gallbladder dimension doesn't matter for pathology.

- Required – at least 1, since each pathology requires at least 1 value for gallbladder dimension.

- Multiple – at most 2, since, proceeding from the knowledge about gallbladder pathologies, there are pathologies for which two possible values are accepted.

Then, when representing the knowledge on certain gallbladder pathology, it is necessary to select in slot *dimensions* the respective value (one or two) in the respective instance for the class *Gallbladder* (see Fig. 5).

Characteristic of gallbladder TONICITY is as simple as DIMENSIONS. Thus, for its description it is enough to introduce the slot *tonicity*. Slots *tonicity* and *dimensions* have similar structure.

### 2.2.2 Slot "contour_gallbladder"

Characteristic GALLBLADDER CONTOUR is more complicated. Thus, for its representation it is not enough simply to introduce a slot which gets simple values. For such characteristics the ontology contains the class *Complex_characteristics*. This class contains subclasses *Main_characteristics* and *Additional_characteristics* (see Fig. 6):

- *Main_characteristics* is for characteristics from the highest level of the knowledge tree;

- *Additional_characteristics* is for characteristics which are at lower levels of the knowledge tree and are used for the description of characteristics from higher levels.

For all these characteristics, irrespective of the level, it is necessary to indicate the pathology in which they take part. Hence, for all respective classes and subclasses the slot *pathology* is needed. Therefore, such a slot is introduced in the class *Complex_characteristics* – it will inherited by all its subclasses. But it can be made more exact. Thus, for example, when describing the characteristics regarding the gallbladder, in facets of slot *pathology* it should be indicated "Allowed Superclasses = Gallbladder_pathology".
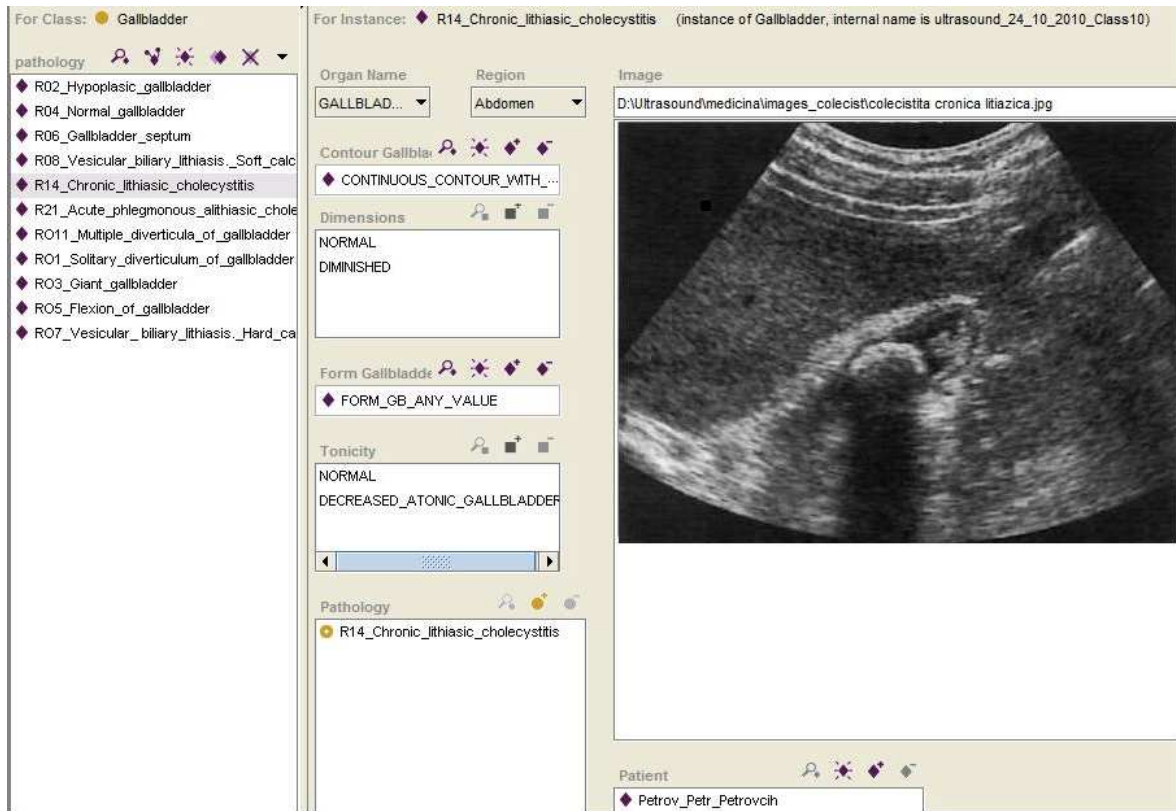
Figure 5: Representation of the knowledge on gallbladder pathology "Chronic lithiasic cholecystitis".

In the class *Main_characteristics* we create subclass *Contour_Gallbladder*, and in the class *Gallbladder* we introduce slot *contour_gallbladder*, which is related with class *Contour_Gallbladder* (the slot has the type "instance of Contour_Gallbladder"). In the class *Contour_Gallbladder* the following slots are introduced:

- slots which correspond to characteristics from knowledge tree: *aspect, clarity, continuity, regularity*. These slots have allowed values in compliance with the knowledge tree.

- For the slot *aspect*: allowed values= IMAGE_OF_HOMOGENEOUS_BAND, IMAGE_OF_DOUBLE_CONCENTRIC_CONTOUR, ANY_VALUE, NOT_DEFINED.

- For the slot *clarity*: allowed values= CLEAR, UNCLEAR, ANY_VALUE, NOT_ DEFINED.

- For the slot *continuity*: allowed values= CONTINUOUS, INTERRUPTED, ANY_ VALUE, NOT_DEFINED.

- For the slot *regularity*: allowed values= IRREGULAR, REGULAR, ANY_VALUE, NOT_DEFINED.

- slot *pathology* is related with the class *Gallbladder_pathology*. So, its type is "Class with superclass Gallbladder_pathology". Here those pathologies will be introduced, for the description of which the respective combination of slots values in *Contour_Gallbladder* takes part.

- slot *contour_description* is the string, in which per se is written somewhat like the title of the respective contour. For example, *contour_description*= CLEAR_CONTINUOUS_CONTOUR_WITH_IMAGE_OF_HOMOGENEOUS_BAND – for the case when *clarity*= CLEAR, *continuity*= CONTINUOUS, *aspect*= IMAGE_OF_ HOMOGENEOUS_BAND, *regularity*= ANY_VALUE.
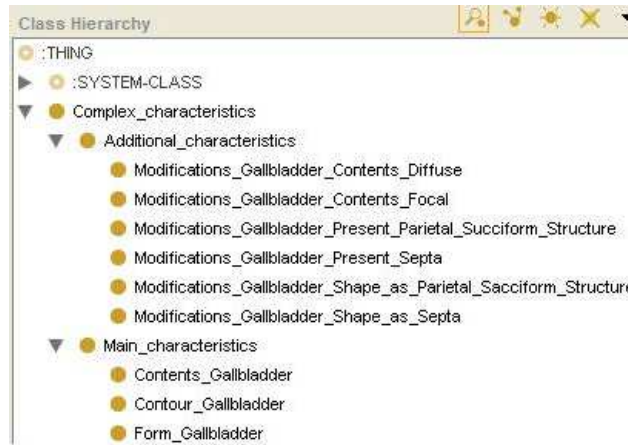
Figure 6: Classes for gallbladder description in current state of ontology.

Thereby, the instance for class *Contour_Gallbladder* is (in terms of knowledge tree) a combination of characteristics values and their relation with respective pathology (or several pathologies). For example:
- instance with the title CLEAR_CONTINUOUS_CONTOUR_WITH_IMAGE_ OF _ HO-MOGENEOUS_BAND is related with the pathology "Normal_gallbladder" since the corresponding combination of values (clarity= CLEAR, continuity= CONTINUOUS, aspect= IMAGE_OF_HOMOGENEOUS_BAND, regularity= ANY_VALUE) takes part in the description of the pathology "Normal_gallbladder".
- instance with the title CONTINUOUS_CONTOUR_WITH_IMAGE_OF_HOMOGENEOUS_BAND is related with three pathologies ("Hypoplasic_gallbladder", "Giant_gallbladder", "Acute_phlegmonous_alithiasic_cholecystitis"), since the corresponding combination of values (clarity= ANY_VALUE, continuity= CONTINUOUS, aspect= IMAGE_OF_HOMOGENEOUS_BAND, regularity= ANY_VALUE) takes part in the description of these pathologies (see Fig.7).



Figure 7: Instance for contour with title CONTINUOUS_CONTOUR_WITH_IMAGE_OF_HOMO-GENEOUS_BAND.

### 2.2.3 Slot "form_gallbladder"

Characteristic FORM is even more complicated.

23

In the class *Main_characteristics* we create subclass *Form_Gallbladder*, and in the class *Gallbladder* we introduce slot *form_gallbladder*, which is related with class *Form_Gallbladder* (the slot has the type "instance of Form_Gallbladder").

In compliance with the knowledge tree, in the class *Form_Gallbladder* the slots for the following characteristics are needed: Normal, Flexion, Septa, Parietal succiform structure. These characteristics are of different level of complexity, which is reflected in the way of their description.

**The characteristic "Normal"** gets the unique value "Normal (without flexion or instable functional flexion)". So it is represented by the slot *normal* with Value Type=Symbol, Allowed Values= NORMAL_WITHOUT_FLEXION_OR_INSTABLE_FUNCTIONAL_ FLEXION.

**The characteristic "Flexion"** can get 5 values. So it is represented by the slot *gallbladder_flexion* with Value Type=Symbol, and Allowed Values= ABSENT, MEDIO-CORPORAL.BILOBATE_GALLBLADDER, DISTAL.FUNDIC_SECTION,PHRYGIAN_CAP, FOLDED_GALLBLADDER.NECKLACE_OF_BEADS, ANNULAR.CLEPSYDRA, ANY_VALUE, NOT_DEFINED.

**The characteristic "Septa"** unlike the characteristic "Flexion" is complex. Moreover, the characteristic "Flexion" has mutually exclusive values, but values of the characteristic "Septa" are not of such type: values of leaves in subnodes can combine or be absent. Thus, we create the class *Modifications_Gallbladder_Shape_as_Septa* as the subclass of the class *Additional_characteristics*.

In the class *Modifications_Gallbladder_Shape_as_Septa* we create the following slots:
- slot *absent* with Value Type=Symbol, Allowed Values= ABSENT;
- slot *modifications_gallbladder_present_septa* related with the class *Modifications_Gallbladder_Present_Septa*, which is created also as the subclass of the class *Additional_characteristics*;
- slot *gallbladder_shape_as_septa_description* for the title of given combination of values;
- slot *pathology* in this case we make more exact: it has Allowed Superclasses = Gallbladder_pathology.

In the class *Modifications_Gallbladder_Present_Septa* we create the following slots:
- *septum_orientation* with Value Type=Symbol, Allowed Values=Longitudinal, Transversal, ANY_VALUE, NOT_DEFINED
- *number_of_septa* with Value Type=Symbol, Allowed Values=Solitary, Multiple, ANY_VALUE, NOT_DEFINED
- *gallbladder_present_septa_description* for the title of given combination of values;
- *pathology* has Allowed Superclasses = Gallbladder_pathology.

**The characteristic "Parietal sacciform structure comunicating with vesicular cavity"** is also complex and with not mutually exclusive values. Thus, we create the class *Modifications_Gallbladder_Shape_as_Parietal_Sacciform_Structure* as the subclass of the class *Additional_characteristics*.

In the class *Modifications_Gallbladder_Shape_as_Parietal_Sacciform_Structure* we create the following slots:
- slot *absent* with Value Type=Symbol, Allowed Values= ABSENT;
- slot *modifications_gallbladder_present_parietal_succiform_structure* related with the class *Modifications_Gallbladder_Present_Parietal_Succiform_Structure*, which is created also as the subclass of the class *Additional_characteristics*;
- slot *gallbladder_shape_as_formation_description* for the title of given combination of values;
- slot *pathology* has Allowed Superclasses = Gallbladder_pathology.

In the class *Modifications_Gallbladder_Present_Parietal_Succiform_Structure* we create the following slots:
- slot *gallbladder_formations_diameter_mm* with Value Type=Integer to indicate diameter (in millimetres) of formation in gallbladder;

- slot *gallbladder_formations_content* with Value Type=Symbol. Since all the subnodes for respective characteristic are mutually exclusive, and the further expanding of the knowledge tree only precise every higher level, then all the leaves can be simply enumerated (writing the names more extensively by adding the names from lower levels): LIQUID_WITH_TRANSSONIC_ECHOGENICITY, LIQUID_WITH_HOMOGENEOUS_ENHANCED_ECHOGENICITY, LIQUID_WITH_INHOMOGENE-OUS_ENHANCED_ECHOGENICITY, LIQUID_AND_SOLID_WITH_REDUCED_MOBILITY, LIQUID_AND_SOLID_IMMOBILE, LIQUID_AND_SOLID_MOBILE, ANY_VALUE, NOT_DEFINED;

- slot *gallbladder_present_parietal_succiform_structure_description* for the title of given combination of values;

- slot *pathology* has Allowed Superclasses = Gallbladder_pathology.

Thus, the class *Form_Gallbladder* for characteristic "Form" has the following slots:

- *form_gallbladder_description* for the title of given combination of values;

- *gallbladder_flexion* for characteristic "Flexion";

- *gallbladder_parietal_succiform_structure* related with class *Modifications_Gallbladder_Shape_as_Parietal_Sacciform_Structure* for characteristic "Parietal sacciform structure comunicating with vesicular cavity";

- *gallbladder_septa* related with class *Modifications_Gallbladder_Shape_as_Septa* for characteristic "Septa";

- *normal* for *characteristic* "Normal";

- *pathology* for relation with those pathologies, which have given combination of values.

### 2.2.4 Instances for classes reflecting complex characteristics

By analogy the other characteristics from the knowledge tree are to be described in the ontology – this is the subject for discussion in future. However, even if all the characteristics from the knowledge tree are described, the knowledge in the ontology will not be full. It is interesting and useful to describe gallbladder pathologies with the help of the elaborated structure.

In this purpose all the subclasses involved in the class *Complex_characteristics* have the slot *pathology*. The instances for all these classes are created. Every instance for each class, corresponding to some characteristic, is the description of some combination of values of characteristics. These combinations may be found in some pathology, and this fact is indicated by the value of the slot *pathology*, which is related to the respective subclass for pathologies.

## 2.3 What does instance for subclass Gallbladder mean?

Instance for subclass *Gallbladder*, as a matter of fact, is the pathology description.

Every instance for any subclass of class *Organ* (and particularly, instance for the subclass *Gallbladder*) contains a concrete pathology, characteristics which describe this pathology, and the corresponding image. That is the description of the specified pathology (see Fig. 8). Moreover, since class *Organ* (and all its subclasses accordingly) has the slot, related with class *Patient*, then every instance of subclass of class *Organ* specifies which patients have the given pathology.

## 2.4 Basic class Pathology

Class *Pathology* is meant for the list of all pathologies structured by organs and different areas. It is created as abstract class and divided into the subclasses corresponding to different organs. At the given stage, the subclass *Abdominal_zone_pathology* is created which contains the following 3 subclasses:

- *Gallbladder_pathology* (gallbladder pathologies);

- *Liver_pathology* (liver pathologies);

- *Pancreas_pathology* (pancreas pathologies).

The subclasses with names of corresponding organs pathologies are included into these classes (see Fig. 9). These classes are related with instances of class *Gallbladder* and of all classes which are in the class *Complex_characteristics*.

## 3 Conclusions

At the first stage of the ontology elaboration its structure differed slightly from the current one [7, 8]. At the uppermost level there were:

- - 4 interrelated classes: *Organ*, *Pathology*, *Patient* and *Image*. From a logical point of view they are the basic ones.

- - such classes as *Contents_Gallbladder*, *Contour_Gallbladder*, *Form_Gallbladder*, *Modifications_Gallbladder_Contents_Diffuse*, etc. From a logical point of view, they are not basic classes. They were used for the description of complex organ characteristics.
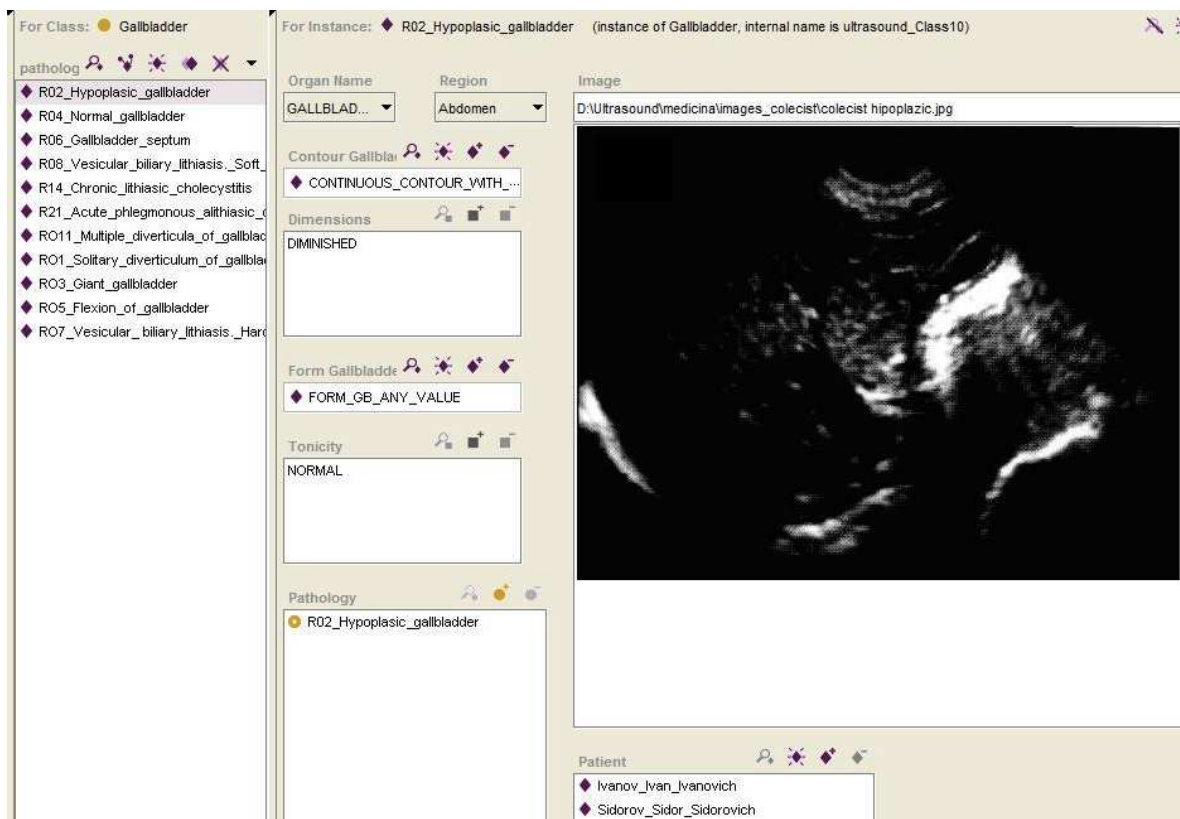


Figure 8: Instance for class *Gallbladder* as the description of the pathology "Hypoplasic_gallbladder".

This structure was ponderous enough and not sufficiently transparent and clear. Thus we decided to introduce at the uppermost level one more basic class *Complex_characteristics* with subclasses *Main_characteristics* and *Additional_characteristics*. So the classes *Contents_Gallbladder*, *Contour_Gallbladder*, *Form_Gallbladder*, *Modifications_Gallbladder_Contents_Diffuse,* etc. were introduced into this structure (see Fig. 1).

These changes in the ontology structure were made rather rapidly and smoothly due to the Protegé [9] flexibility. But the ontology became more transparent. This fact proves once more that the process of ontology elaboration is the iterative one.

Figure 9: Hierarchy of pathologies.

# References

[1] L.Burtseva, S.Cojocaru, C.Gaindric, E.Jantuan, O.Popcova, I.Secrieru, D.Sologub, SONARES – A Decision Support System in Ultrasound Investigations. *Computer Science Journal of Moldova*, nr. 2 (44), vol. 15, pp. 153–177, 2007.

[2] S. Cojocaru, C. Gaindric, Clinical Decision Support System SONARES. *International Journal Information Theories and applications*, vol.15, N 2, pp. 137–143, 2008.

[3] Iu. Secrieru, O. Popcova, S. Puiu, D. Sologub, Knowledge Structure Modeling in Ultrasound Investigation Domain. *Proceedings of the 5th European Conference on Intelligent Systems and Technologies*, Iasi, Romania, 2008, CD.

[4] Iu. Secrieru, Structured Knowledge Management Techniques for the Development of Interactive and Adaptive Decision Support System. *Computer Science Journal of Moldova*, vol.17, n.1(49), pp.58–73, 2009.

[5] O. Popcova, S. Puiu, Iu. Secrieru, Knowledge Structurization and Formalization for Gallbladder Ultrasound Investigation Domain. *Proceedings of the II International Conference on Advanced Information and Telemedicine Technologies for Health*, Minsk, Belarus, pp. 98–102, 2008.

[6] Iu. Secrieru, D. Sologub, Expert Shell Aimed at Creation of the Knowledge Base for Ultrasonic Research Intelligent System. *Revista de inventică*, N48, vol. IX AN XV-2005, pp.36–41, Iaşi, România.

[7] N. Bruc, Knowledge Representation with the Help of Ontologies in SonaRes System. *Collection of Abstracts of BIT+ "Information Technologies-2008"*, 2008, Chisinau, vol. 8, pp. 50–53, in Russian.

[8] C.V.Gaindric, S.K.Cojocaru, N.L.Brook, G.A.Magariu, T.B.Verlan, Ultrasound images ontology elaborating in ultrasound diagnostics domain. *Internet conference "Informational systems and technologies"*, Orel, 2011 (in Russian, in print).

[9] http://protege.stanford.edu/.

Bruc Natalie
Institute of Mathematics and
Computer Science of ASM
5, Academiei street, Chisinau,
Republic of Moldova, MD 2028
REPUBLIC OF MOLDOVA
E-mail: *nataliebruc@yahoo.com*

Magariu Galina
Institute of Mathematics and
Computer Science of ASM
5, Academiei street, Chisinau,
Republic of Moldova, MD 2028
REPUBLIC OF MOLDOVA
E-mail: *gmagariu@math.md*

Verlan Tatiana
Institute of Mathematics and
Computer Science of ASM
5, Academiei street, Chisinau,
Republic of Moldova, MD 2028
REPUBLIC OF MOLDOVA
E-mail: *tverlan@math.md*

# Ultrasound diagnostics system SonaRes: structure and investigation process

**Liudmila Burtseva, Svetlana Cojocaru, Constantin Gaindric, Olga Popcova, Iulian Secrieru**

### Abstract

In this paper the recent approaches to development of Diagnostic Decision Support Systems (DDSS) are examined. The solutions used in the SonaRes system, which combines knowledge-based and image-based techniques, are described.

## 1 Introduction

Diagnostics is one of the most difficult tasks clinicians may face every day. Determination of the correct diagnostics involves a multitude of factors, that characterize the patient's state of health. Omission or misinterpretation of them can lead to mistakes with unpredictable consequences. No less important is to reveal the relationships between these factors, as well as interaction between organs and its manifestation mode. In such activities it is a natural tendency to ask for a help, to make sure we proceed in the right way or to find a solution when we do not know how to proceed. A second opinion, an advice, is useful to any physician, even to an experienced one.

The problem, associated with the medical diagnostics activities, acquires a special relevance in modern circumstances. First of all, it is connected with the fact that physicians have to work with poorly structured and weakly formalized information. Besides, medical diagnosis involves interplay between a vast number of medical knowledge resources in so-called healthcare environment of important patient data, including clinical information, up-to-date status reports, medication and medical imagery.

The ultrasound investigation domain is not an exception. The appearance of new ultrasound devices or the improvement of old scanners doesn't simplify, but even complicates the physicians diagnostic thinking. So far as one has to analyze a number of diagnostic data, the time of diagnostics determination increases and the accuracy sometimes may reduce.

The subject of this study is the Decision Support Systems in ultrasound diagnostics revealing two classes: image-based and knowledge-based systems. Some interesting implementations of combination of these methods are presented. The principles and techniques of knowledge acquisition, structurization, representation and management, aiming to create an efficient logical inference and adaptive user interface, are described. Some aspects of future work are discussed.

As an illustration for the proposed approach serves the SonaRes decision support system, designed to assist the diagnostician in the ultrasound examination process of the abdominal region. It is particularly difficult because of numerous organs are located in a relatively small space, and there is a need to take into account their interaction.

## 2   Diagnostic Decision Support Systems: general characteristics

DDSS at the early stages were perceived by doctors rather simplistically as: "a machine algorithm that supports the clinician in one or more components of the diagnostic process" [1] or as "a process of definition in the process of investigation of nature and circumstances of the disease formation" [2].

In [3] there are identified three components of comprehensive information needs:

1. currently satisfied information needs (information recognized as relevant to a question and already known to clinician);

2. consciously recognized information needs (information recognized by the clinician as important to know to solve the problem, but which is not known to him); and

3. unrecognized information needs (information that is important for the clinician to know to solve a problem at hand, but not recognized as being important by the clinician).

Establishing the diagnosis is a process preceding the suggestion of therapeutic or surgical treatment. According to [4] "The process of diagnosis entails a sequence of interdependent, often highly individualized tasks: evoking the patients initial history and physical examination findings; integration of the data into plausible scenarios regarding known disease processes; evaluating and refining diagnostic hypotheses through selective elicitation of additional patient information, such as laboratory tests or serial examinations; initiating therapy at appropriate points in time (including before a diagnosis is established); and evaluating the effect of both the illness and the therapy, on the patient, over time".

Diagnosis is a process consisting of separate steps. These steps begin with establishing the certain facts in the examination process and lead to the inference that the obtained facts correspond to some conclusion or begin with some preliminary diagnosis achieving the conformity of the set of objective facts of the patient state to confirm the presumptive diagnosis or reject it, if the facts do not correspond to or contradict the assumption. Even if the start and end points of the process are identical, the steps followed by two doctors could differ very much, and at the same time, the diagnostician can take various steps in two almost identical cases. Since the investigation is a creative process based on knowledge and experience, different physicians may face the problems in valuation of the same patient. Diagnosis as an interpretation of the results of a number of observations, is potentially recursive and is essentially defined by consistently complicating diagnostic tools.

As a matter of fact, DDSS do not generate a single conclusion (diagnosis) and usually suggest several ones, based both on patient data and knowledge embodied into the system. All these conclusions correspond to the facts, observed in the process of examination. Since the general knowledge, storing in the system about the case under examination, could be supplemented with other one, specific to a given patient, the physician would be allowed to reduce, as far as possible, the number of conclusions generated by the system.

In general case DDSS are not intended to replace the physician, their role being to give clinician recommendation on his request or to draw automatically his attention to the special cases (cases of alerts), .

Obviously, DDSS are targeted on a specific area, which may be more or less broad, and the domain of their applicability is defined by a pathology (a group of pathologies) or a diagnostics method, which in its turn is oriented to certain pathologies (group of pathologies or organs).

## 3   Decision Support Systems in ultrasound diagnostics

The ultrasound examination of patients, being non-invasive and not expensive, is a basic technique of medical imaging.

Ultrasound image is the primary (input) information for every ultrasound examination. The main characteristic of these images is two-layer structure of the information contained in it. The first layer is the image itself (graphical features), and the second layer is its textual description in medical terms (medical features).

Besides the difficulties of ultrasound image processing, because of the speckle, tissue related textures and artifacts, this source of information is still significant for diagnosis decision making. However, utilization of this technique does not always come up to expectations, encountering some difficulties associated with the dependence on operator, which affects the quality of the obtained images, and the way the results are differently described and interpreted by several specialists.

It should be stressed that consecutive losses of the information accuracy are inherent to the process of ultrasound examination. An analog signal transmitted by the probe is converted into a digital one and used to construct an image that quite subjectively (depending on qualification and experience) is interpreted by the operator. To overcome such shortcomings the information systems are developed with the purpose to reduce the influence of subjective factors by assisting in the examination process [5], [6], [7]. These systems can be used to get a second opinion, helping the physician-ecographist to make better interpretations, to deduct conclusions, and to obtain higher-quality images.

Computer-aided diagnosis schemes has been subject to various research since 1980, when a number of medicine domains began to use computer assistance [8]. During these years, huge databases of medical images were created and became a powerful source for decision making. Taking into consideration these tendencies, two main approaches to create DDSS systems can be distinguished:

- Image-based systems. Such systems are aimed at making the decisions based on comparison between the newly obtained images and the images taken from a database. These systems classify new images according to existing classification methods and/or provide the possibility to determine a similarity degree with images stored in the database. In case of an acceptable resemblance with an image, this is considered as a precedent and the current decision is based on the existing decision for this particular precedent.

- Knowledge-based systems. Such systems deal with decision making based on a description given by the user, and data and rules provided by the system. Most of these systems are informational-instructive with (or without) an additional diagnosis component.

A joint realization of decision making and image retrieval in DDSS development shows itself as a fruitful research direction. Usage of the domain-specific knowledge leads to better results in many applications, in medicine as well. On the one hand, medical imaging computer systems provide the medical practitioner with imaging techniques, leading to new explicit knowledge. On the other hand, informatics research helps in standardizing the image content, enabling comparisons across populations and facilitating new ways of thinking. Moreover, research in medical image analysis tries to find links between image features and knowledge in order to fulfill quantification tasks and to answer prognostic questions [9].

## 4 Decision support based on combination of knowledge-based and image-based techniques

In this section the system architecture solutions, which use various combinations of knowledge-based and image-based techniques, represented.

The first group of solutions provides decision support based on classified images, using the technologies and methods of artificial intelligence (AI). The typical system architecture in this case consists of the following steps: preprocessing, features extraction, usage of AI methods for decision making.

Setting of regions of interest (ROIs) [10] is an aspect that is actively used and should to be noted. ROIs concept helps to facilitate the process of medical knowledge acquisition. ROIs marked by expert can be associated with some domain knowledge instances, for example, the image subjects in the ontology.

ANALYSIS [11] is a CAD system designed to assist in the interpretation of vascular ultrasound images. The role of image preprocessing in the system can be defined as a typical one for this kind of architecture. Improvement of images quality is very important for the effectiveness of subsequent tasks, such as definition of ROIs and feature extraction. According to AI technologies, ANALYSIS uses a vector of texture and motion features for fuzzy c-means classification. The selected features set is fed into a group of neural network classifiers.

A tool for finding of visual pathologies artifacts in mammographic images was proposed in [12]. It also has the typical architecture. But, the authors of this system declare a completely automated approach. The AI technologies in this case are represented by ROI characterization using textural features, computed from gray tone spatial dependence matrix, and ROI classification by means of a neural network. The system has wide possibilities for evaluation because of its large database of mammographic images.

An automated approach is also proclaimed for the system described in [13], directly aimed at retrieval of medical images to support neurology diagnosis. In this case, the differential diagnosis information related to specific image characteristics is defined by experts.

A system for classification of brain tumor in computed tomography scan brain images is proposed in [14]. The system architecture is similar to a typical one, but has several AI-related steps. The decision tree classification assists the physicians to classify unclear images. For this hybrid system (imaging and AI techniques) it is also typical, that the association rules for decision trees are constructed basing on images themselves, but not on domain knowledge.

The second group of solutions gives the decision support based on knowledge, acquired using image processing and retrieval technologies. The architecture of such systems provide a kind of parallel work of image processing/retrieval and AI-based techniques. After all required information is obtained during these parallel steps, the decision engine interprets the results and offers advice to the user.

A system for detecting a diffuse lung disease pattern in high resolution computed tomography lung images, proposed in [15], is an extended variant of the images-based systems, and uses AI technologies as a support. The architecture of this system has an additional step verification of the decision. To perform this step researchers use domain knowledge about the structure of lung, as well as expert knowledge about the appearance of diseases. A model for human lung is created basing on domain knowledge. The algorithms automatically generate visual lung regions according to the anatomical sense of notions, frequently used in disease reporting.

EchoCardio Lab [16] of the European HEARTFAID Project is an infrastructure providing integrated management of different type data of echocardiography workflows. Decision support services and image analysis facilities interaction is implemented in the frame of this infrastructure.

A research in breast cancer, grading using a knowledge-guided semantic indexing of histopathology images, is described in [17]. Its architecture is the most successful one in representation of joint usage of knowledge-based and image-based techniques. The creation of association between the meaning and features, extracted from the image, is done according to domain knowledge (the domain knowledge modeling is provided, starting with the domain ontology). The algorithm performs a segmentation of images and processes the object recognition phase (features extraction step) as an input for the semantic classification step. The conceptual annotations are rule-based defined in the grading model for every particular frame and globally transmitted for the entire case.

From the above analysis (that does not purport to be a comprehensive one) we can conclude that three points of major importance, influencing essential the success of DDSS, are the following:

- Knowledge acquisition and presentation;

- Image processing and search for similar ones (with extraction of knowledge from the images);

- Interaction with user.

The way these issues are resolved determines the functionality, user attitude and, as a result, adequacy of generated conclusions. In what follows, we will discuss these points, basing on solutions realized in the SonaRes system.

# 5   SonaRes solution

The SonaRes decision support system [7] provides a second opinion for specialists, experienced or not, with necessary explanations and images that are similar to the currently examined case.

SonaRes system combines two basic approaches of ultrasound diagnostic systems development by assisting the decision making process on the base of both rules and images.

The main components of the SonaRes system are:

- Module of knowledge acquisition and validation;

- Integrated database (of knowledge, images, annotations, examination reports templates) and tools for its management;

- Tools for the examination process support;

- Image processing module, including algorithms for fast retrieving of similar images;

- Generator of examination reports.

Here is a brief presentation of the SonaRes components.

The first module – knowledge acquisition and validation – is designed to support effective communication with experts in the development of the knowledge base. It is created as an expert shell. The main stages of the development are: problem identification, knowledge acquisition, structure definition, formalization and implementation. The experience of experts and the medical specialty literature served as the main sources of knowledge.

In order to develop the methodology and technology, which can be extended to the whole abdominal region, the gallbladder and pancreas were selected. The necessary knowledge for the examinations of these organs was obtained from physicians who have vast experience in ultrasound diagnostics, consisting of:

- Structured information about organ localization, including the method of visualization of typical areas, objective conditions for visualization, considerations about possible non-visualization, objective conditions for a difficult visualization;

- Descriptors of the main characteristics of organs (number, size or volume, shape, contour, etc.);

- Structured information about pathologies and anomalies, each of them being determined by the characteristics of organ modifications (anomalies of shape, size, quantity, etc.).

The knowledge obtained from the experts is stored in the knowledge base and presented as a tree structure.

The second component is the integrated database of images, annotations and examination reports.

The tools aimed to support the examination process, included into the third component, allow to select one of the main ways of examination, corresponding to the physician usual methods of work:

- Step by step, i.e. studying the obtained image, the physician selects the attributes from a list and sets their values. Depending on the values of selected attributes, one or more conclusions are proposed, corresponding to the rules from the knowledge base. The conclusion may be accompanied by an image, in which ROIs are highlighted, if the diagnostician thinks that this is necessary for the physician who administrates the treatment. Following a special request, the annotated images from the integrated database, similar to the one obtained in the current examination, are retrieved and presented, that allows validation of the diagnosis on the base of similar cases solved by experts. This method allows to reduce the time required to obtain a conclusion, i.e. an examination result, raises the diagnosis quality, promotes the formation of correct actions and mentality in the field of ultrasound diagnostics. This is a very important point in professional training, encouraging use of correct terminology.

- From the presumed pathology to its confirmation or refection. In this case the physician determines if the rules marked by the system as appropriate for the currently examined case correspond or not to the presumed pathology. This way can be used by more experienced physicians.

- Mixed way, which allows the clinician to alternate in the examination process both procedures.

To assist in the examination process, a thesaurus has been developed. A sufficient number of terms should be contained in this thesaurus, providing a clear picture of the full spectrum of clinical concepts. It can be used autonomously as an encyclopaedic reference book, but also for the help function that

is integrated in the examination interface for a quick access to explanations of terms that appear in the examination process. For each term its definition is given, along with synonyms and translations (for national and international users). The encyclopaedic reference book is also supplied with videos (remember that in ultrasonography the organs can be seen dynamically). Queries of the thesaurus can be made through various criteria: key words, combinations of words, search by topic.

The fourth component performs image processing (preprocessing) and search of similar images. Because of the shortcomings of ultrasound images mentioned above, for the beginner in ultrasound examination or for a physician with lack of experience it is difficult to identify the organ pathology based only on one image. Thus, it is useful to provide a quick search for images similar to one obtained in the examination process.

To provide the ROIs marking and attaching of the rules to the images, several special tools, which are part of the SonaRes toolkit for experts, were developed. These tools implement three types of visualization for the ROI-related expert actions: (i) setting the correspondence of ROI with the node of the knowledge tree for an organ; (ii) review of the all ROIs that correspond to the chosen node, on different images; (iii) review of the all ROIs (for different nodes) corresponding to the chosen image.

The SonaRes system collects a set of model images attached to the corresponding rule, with ROIs defined for a particular pathology. The role of annotated images in decision making is to provide help by illustration, in cases when the medical specialist is not sure how to interpret correctly the images.

In the case of ultrasound images, the most fruitful method for retrieving similar images seems to be the application of both medical features (obtained from a knowledge base) and visual features (obtained from images). In this case formalized knowledge is used to classify the images and reduce the retrieval. Medical similarity allows the confirmation of the diagnosis assumption by obtaining a gallery of images marked by experts as containing the supposed pathology or fact (for example: is this a stone in the kidney?). Visual similarity allows the determination of pathology or fact by presenting the list of rules in the knowledge base that correspond to graphical features found in images.

The general-purpose methods of Content-Based Image Retrieval (CBIR), used in SonaRes, are developed and tested in the frame of Image Retrieval in Medical Application (IRMA) project [18]. To provide the integration of knowledge accumulated in SonaRes (facts, rules, annotated images) and IRMA algorithms of medical images storage and processing the CASAD (Computer-Aided Sonography of Abdominal Diseases) system was developed [19]. It represents a data warehouse of standard referenced images. The scheme of interaction and data exchange between SonaRes, CASAD, IRMA systems is presented below (see Fig.1).



Fig. 1 – Scheme of three systems interactions

The CASAD as data warehouse supposes two main functions:

1. to add new image to warehouse;

2. to retrieve images similar to the pattern image uploaded by user.

Both functions are based on image classification criterion. In the case of new image adding classified image is added to the warehouse. In the case of image retrieval the set of images belonging to the same class that the pattern image is presented.

The last component of the system is generator of examination reports. Traditionally, the medical conclusion of the examination consists of: data about patient, image, quantitative measurements in the examination, and the physician conclusion in an arbitrary form. The examination report given out by the SonaRes contains structured data, obtained during the examination, and the conclusion consists of

the rules corresponding to the measured values. Data that can not be obtained during the investigation session, and bearing a specific nature (requiring biochemical or other analysis), are recorded by the physician in his usual free form.

The process of system development was an iterative one. As a result, 54 decision rules for the gall-bladder and 52 decision rules for the pancreas were formulated. The analysis of the obtained rules shows that, in addition to the normal state and anomalies, we also embraced all basic groups of pathologies. The specific of pancreas required additional information for 34 help files.

# 6   Conclusion

The concept of the SonaRes system was described for the first time in [7]. Further research, described in this article, has allowed to improve characteristics of all modules of the system and to create the prototype with the following peculiarities:

- guidance of the examination process, adapting it to different levels of physician's experience;

- support for reporting, assuring common standards;

- prevention of possible errors in the examination process (e.g. omitting of some important aspects, skipping some characteristics or admitting some inaccuracy in the formulation of conclusions);

- possibility to use experts' experience, collected in the SonaRes database, in the form of annotated images, similar with the one under examination;

- processing of captured images in order to increase their quality or to better distinguish some special regionas or characteristics;

- possibility to be used for professional training;

- storage of investigation records (having possibility to observe the disease dynamics, to collect statistics, etc.).

This prototype has been tested by the group of experts-physicians and can be recommended for dissemination.

The SonaRes system can be widely used by all categories of physicians as a support for the examination process. Also, the system can be used by experienced physicians when examining some difficult cases, by physicians practicing in isolated areas or those with limited access to experts, as well as by novices and those studying the ultrasound diagnosis domain.

# References

[1] Miller RA, Why the standard view is standard: people, not machines, understand patients problems. *J Med Philos*, 15: 581-591, 1990.

[2] F T de Dombal, Ethical considerations concerning computers in medicine in the 1980s. *J Med Ethics*, December 13(4): 179-184, 1990.

[3] Forsythe DE, Buchanan BG, Osheroff JA, Miller RA, Expanding the concept of medical information: an observational study of physicians information needs. *Comput Biomed Res*, 25: 181-200, 1992.

[4] Eta S. Berner, EdD, *Clinical Decision Support Systems: Theory and Practice*, Second Edition, Springer Science+Business Media, LLC, 2007.

[5] http://cabinet.fromru.com

[6] M. Huettig, G. Buscher, T. Menzel, W. Scheppach, F. Puppe, H. P. Buscher, A Diagnostic Expert System for Structured Reports, Quality Assessment, and Training of Residents in Sonography. *Medizinische Klinik*, 99, nr.3: 117-122, 2004.

[7] L. Burtseva, S. Cojocaru, C. Gaindric, E. Jantuan, O. Popcova, I. Secrieru, D. Sologub, SONARES – A decision support system in ultrasound investigations. *Computer Science Journal of Moldova*, v.15, N2(44): 153-177, 2007.

[8] Doi K., Computer-aided diagnosis in medical imaging: historical review, current status and future potential. *Comput Med Imaging Graph*, 31(4-5): 198-211, 2007.

[9] Holsapple C. W., Winston A. B., *Decision Support Systems: A Knowledge-based Approach*, West Publishing Co., Minneapolis, 1986.

[10] L. Temal, M. Dojat, G. Kassel, B. Gibaud, Towards an ontology for sharing medical images and regions of interest in neuroimaging. *Journal of Biomedical Informatics*, vol.41: 766-778, 2008.

[11] Bui A, Taira R, (eds.), *Medical Imaging Informatics*, Springer, 2010.

[12] Bellotti R, De Carlo F, Tangaro S, Gargano G, Maggipinto G, et al., A completely automated CAD system for mass detection in a large mammographic database. *Med Phys*, 33, 3066, 2006.

[13] Mei-Ju Su, Heng-Shuen Chen, Chung-Yi Yang, Sao-Jie Chen, Robert Chen, et al., Diagnostic decision support by intelligent medical image retrieval with electrical medical record enhance dementia treatment. *Medical Imaging Technology*, 25(5): 350-355, 2007.

[14] Rajendran P, Madheswaran M, Hybrid Medical Image Classification Using Association Rule Mining with Decision Tree Algorithm. *Journal of Computing*, January, Volume 2, Issue 1, 2010.

[15] Zrimec T, Wong J, Improving computer aided disease detection using knowledge of disease appearance, *Proceedings Medinfo*, 12(Pt 2): 1324-1328, 2007.

[16] M. Martinelli, D. Moroni, O. Salvetti, M. Tampucci, A Knowledge-based Infrastructure for the Management of Diagnostic Imaging. *Proc. In Heart Failure Domain Transac. on Mass-Data Analysis of Images and Signals*, vol. 2, no. 1: 3-18, 2010.

[17] Tutac AE, Racoceanu D, Leow DK, Dalle JR, Putti T, et al., Translational approach for semi-automatic breast cancer grading using a knowledge-guided semantic indexing of histopathology images. *Proceedings ot the 3rd MICCAI Workshop on Microscopic Image Analysis with Application in Biology*, New York, September 5-6, 2008.

[18] http://irma-project.org

[19] T. Deserno, L. Burtseva, Iu. Secrieru, O. Popcova, CASAD – Computer-Aided Sonography of Abdominal Diseases – the concept of joint technique impact. *Computer Science Journal of Moldova*, v.17, N3(51): 278-297, 2009.

Liudmila Burtseva
Institute of Mathematics and Computer Science of the Academy of Sciences of Moldova
Programming Systems Laboratory
5, Academiei street, Chisinau, MD 2028
Republic of Moldova
E-mail: *burtseva@math.md*

Svetlana Cojocaru
Institute of Mathematics and Computer Science of the Academy of Sciences of Moldova
Programming Systems Laboratory
5, Academiei street, Chisinau, MD 2028
Republic of Moldova
E-mail: *svetlana.cojocaru@math.md*

Constantin Gaindric
Institute of Mathematics and Computer Science of the Academy of Sciences of Moldova
Information Systems Laboratory
5, Academiei street, Chisinau, MD 2028
Republic of Moldova
E-mail: *gaindric@math.md*

Olga Popcova
Institute of Mathematics and Computer Science of the Academy of Sciences of Moldova
Information Systems Laboratory
5, Academiei street, Chisinau, MD 2028
Republic of Moldova
E-mail: *oleapopcova@yahoo.com*

Iulian Secrieru
Institute of Mathematics and Computer Science of the Academy of Sciences of Moldova
Programming Systems Laboratory
5, Academiei street, Chisinau, MD 2028
Republic of Moldova
E-mail: *secrieru@math.md*

# The Analysis of Continuous Variables in the Decision Model of Bankruptcy Risk using Bayesian Networks

**Mihaela Crăciun, Dominic Bucerzan, Crina Rațiu**

### Abstract

This paper is concerned with the modeling using bayesian network (BN) of bancruptcy prediction (BP) from the economic model proposed by Anghel [5]. Within the simulation the paper is focused on the choosing a discretizing method for the used interval, depending on the performance of the three methods chosen. Comparison is made between Bracket Medians discretizing method and Pearson-Tukey discretizing method. The BN construction process, respectively the simulation was realized using the AgenaRisk software. Simulation results were obtained from sensitivity analysis table and graphic.

**Keywords:** *Bayesian Network (BN), Normal Distribution, Discretizing, Bracket Medians Method, Pearson-Tukey Method*

## 1   Introduction

Modeling using bayesian network (BN) of bancruptcy prediction (BP) from the economic model proposed by Anghel [5], is presented in this work.
For simulation we have chosen several discretizing methods for the used interval.
Before detailing the proposed solution, we define de necessary elements for modeling using BN.

**Random variables**

Given a probability space $(\Omega, P)$, a ***random variable*** X is a function whose domain is $\Omega$. The range of X is called the space of X.
We call $P(X=x)$ the ***probability distribution*** of the random variable X. [1]

**Bayesian Network – BN**

Bayesian networks – BN consist of:
- a direct acyclic graph (DAG), whose edges represent relationships among random variables that are often (but not always) causal;
- the prior probability distribution of every variable that is a root in the DAG; and
- the conditional probability distribution of every non-root variable given each set of values of its parents. [2]

**The I. Anghel Model in Bankruptcy Risk Prediction – Anghel PM for BR**

Anghel PM for BR is based on a statistical data sample collecting during the period 1994-1998. The author used the discriminant analysis method (MDA) for selecting bankrupt enterprises. Thus is created the following function score.

$A = 5.667 + 6.3718 * X_1 + 5.3932 * X_2 – 5.1427 * X_3 – 0.0105 * X_4$, subject is founded in [5].

Where:

X1 - earning after taxes / incomes;

X2 - Cash Flow / total assets;

X3 - liability / total assets;

X4 - liability/ sales * 360

Weight coefficients were calculated according to the variation and covariation of the financial variables and according to the difference of the variables mean broken down by enterprises groups: bankrupt and good situation.

The inflection point that minimize the error rate is A=0, with an uncertainly range between 0 and 2.05.

In case of *a prior* analysis of the success rate for the score function A, it will compared the predictive classification with the known situation of the enterprises in the sample. When choosing an inflection point equal to 0 it determine a success rate of 91.2%. In case we choose two inflection points (A=minim 0 and A= maximum 2.05) and we consider a zone of uncertainty between this two points, we obtain a success rate of 96.7%.

The risk assessment in Anghel PM for BP is the following:

When A < 0, bankruptcy/failure situation;

When 0 ≤ A ≤ 2.05, uncertainty situation demanding prudence;

When A > 2.05, a good financial situation

In case of *a posterior* analysis of the success rate for the score function A, it will analyzed the degree of relevance for another sample of enterprises. It will obtain a prediction success rate without an uncertainly zone of 92.8%, respectively the success rate with an uncertainly zone of 97.8%.

The results obtained in the two analysis permit assessment that the A score is efficient and can be applied to enterprises in the Romanian economy.

**Discretizing**

Let be a BN that contains random variables that are discrete or continuous. For the continuous variable the possible values of the node are ranges and the probability of each of these ranges is specified in the network. This is called discretizing the continuous variables. [3]

*Methods for discretizing*

*A. Bracket Medians Method [4]*

In the Bracket Medians (BM) Method the mass in a continuous probability distribution function $F(x) = P(X \leq x)$ is divided into n equally spaced intervals. The method proceeds as follows. Typically we can use three, four ore five intervals. If we have more intervals, the computation is more accurate. Let be n=3 in this explanation. Next we use the BM Method to discretize the distribution into three ranges. They are four steps to follow:

Step 1: For values between 0 and 100, we consider following three intervals:

$$[0, 0.333], [0.333, 0.666] \text{ and } [0.666, 1].$$

Step2: We need to find points $x_1$, $x_2$, $x_3$ and $x_4$ such that:

$$P(X \leq x_1) = 0.0, P(X \leq x_2) = 0.333, P(X \leq x_3) = 0.666, P(X \leq x_3) = 1$$

where the values on the right in these equalities are the endpoints of the three intervals.

Using mathematics package we obtain: $x_2=43.5$ and $x_3=56.4$. Clearly we have $x_1=0$ and $x_4=100$.

Step 3: For each interval $[x_i,x_{i+1}]$ we compute the bracket median $d_i$, which is the value such that: $P(x_i \le X \le d_i)=P(d_i \le X \le x_{i+1})$.

Step 4: Define the discrete variable D with the following probabilities:
$$P (D=d_1) = 0.333, P (D=d_2) = 0.333, P (D=d_3) = 0.333$$

*B. Pearson-Tukey Method [4]*

In the Pearson-Tukey Method the mass in a continuous probability distribution function $F(x) = P(X \le x)$ is divided into three intervals. The method proceeds as follows:

Step 1: Determine points $x_1$, $x_2$ and $x_3$ such that
$$P(X \le x_1) = 0.05, P(X \le x_2) = 0.50, P(X \le x_3) = 0.95$$

Step2: Define the discrete variable D with the following probabilities:
$$P (D = x_1) = 0.185, P (D = x_2) = 0.63, P (D = x_3) = 0.185$$
Using mathematics package we obtain the cut points 36.6 respectively 63.4.

# 2   Bayesian Networks Construction

We explain the Anghel PM for BR accepting the Bayes' Theorem and the accuracy of the software AgenaRisk [6] in [7].

After the BN construction we obtain the visual model as shown in the Figure 1.
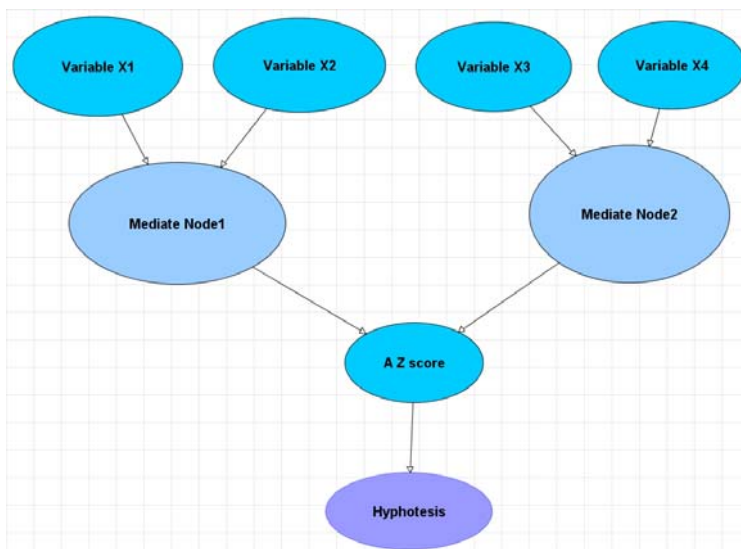


**Figure 1 – BN showing causal structure**

We use three types of nodes to model our BN: sample, result and assumption nodes.

The difference between the BN presented in [7] and the model presented in this paper refers to the lower and upper bounds we consider and the graph type we associate. Also we removed from de model the observation nodes.

The *sample nodes* are simulation nodes, with continuous interval type. The lower bound is 0 and the upper bound is 100. The NPT is a Normal Expression with mean 50 and variance 225. The graph types associated to this node are Line and represent the Probability Distribution.

The *result nodes* are simulation nodes, too. They divide in two categories. The Mediate Nodes and the A Z score node. The types of Mediate Nodes are continuous interval with values between 0 and 1.176. The NPT is an arithmetic expression $6.3718*X_1 +5.3932* X_2$, respectively $5.667-5.1427*X_3-0.0105*X_4$. The graph types associated to this node are Line and represent the Probability Distribution. The type of A Z score node is continuous interval with values between -520 and 2.700. The NPT is an arithmetical expression MN1+MN2. The graph type associated to this node is Line and represents the Probability Distribution.

The *assumption node* Hypothesis is a simulation node, with Boolean type. The state options are customised, with Positive Outcome "Good financial situation" and the Negative Outcome "Bankruptcy / failure situation". The NPT is a comparison expression: ***if(zscore<=2.05,"Bankruptcy / failure situation", "Good financial situation")***. The graph type associated to this node is Line and represents the Probability Distribution.

The statistic attached to the main risk graph is shown in Figure 2.
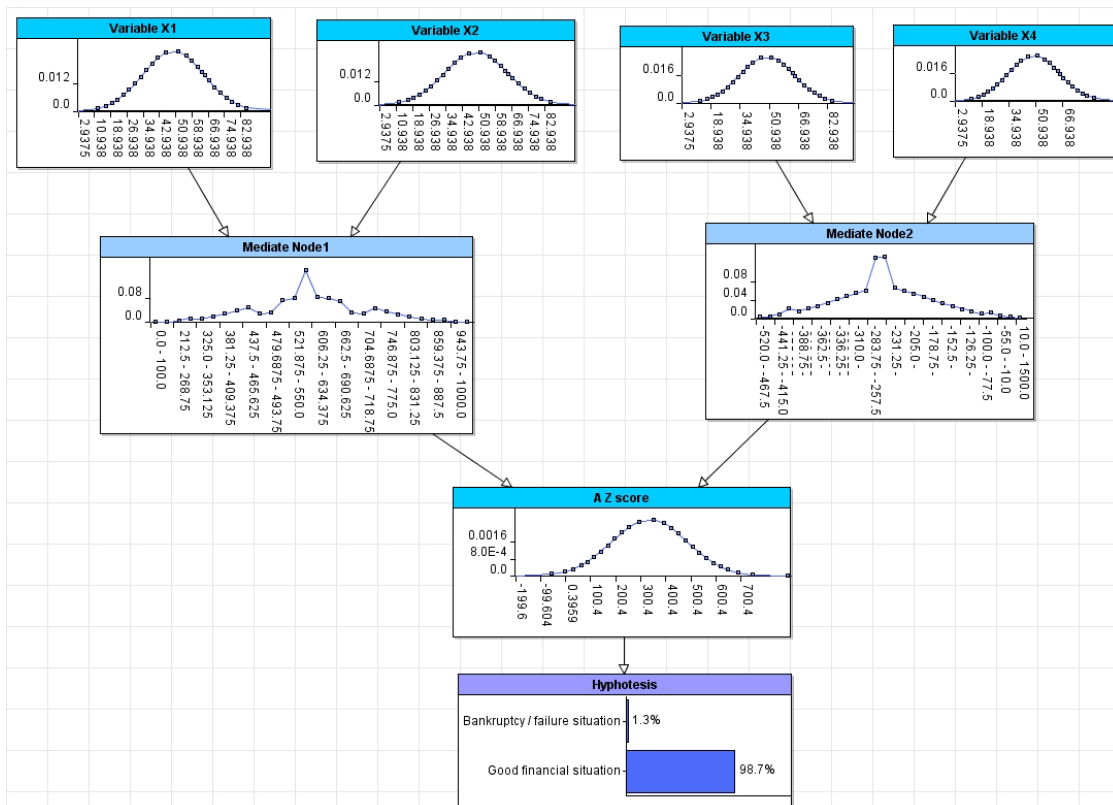


**Figure 2 - Complete Hypothesis Testing Model**

# 3   Hyphotesis Testing Model Simulation

Within the *sample nodes* we define the Node Probability Table – NPT, using the Normal Distribution expression, see the Figure 3.

**Figure 3 – The NPT defined as Normal Distribution with Mean=50 and Variance=225**

## 3.1 Simulation using one interval

This simulation case is based on the BN structure defines by section 2. It works of the interval [0, 100], defined in the *sample nodes*. In this case we obtain a percentage of 98.700 % "Good financial situation" (see Figure 2). In simulation with scenario we test as enter observation in sample node Variable X1 the value 25 (mean between 20 and 30, see values obtained in Figure 11). In this case we obtain a percentage of 93.633 % "Good financial situation".

After the Sensitivity Analysis we obtain the results shown in Figure 4.
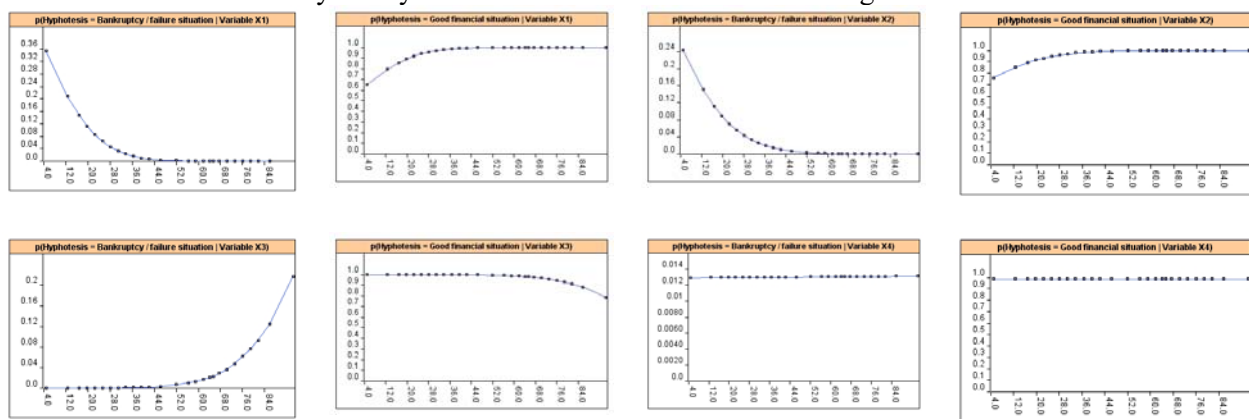


**Figure 4 – Sensitivity Analysis in the interval [0, 100]**

## 3.2 Simulation using Bracket Medians (BM) Discretizing Method

In this case, we must first build our BN. The key for this BN is defined in the *sample nodes*, more accurate in his Node States (see Figure 5) and in his Node Probability Table (see Figure 6).



**Figure 5 – Node States for the sample nodes in BM Discretizig Method**

After defining all the nodes like in the section 2, we obtain the BN shown in the Figure 7.



**Figure 6 – NPT in BM Discretizing Method**
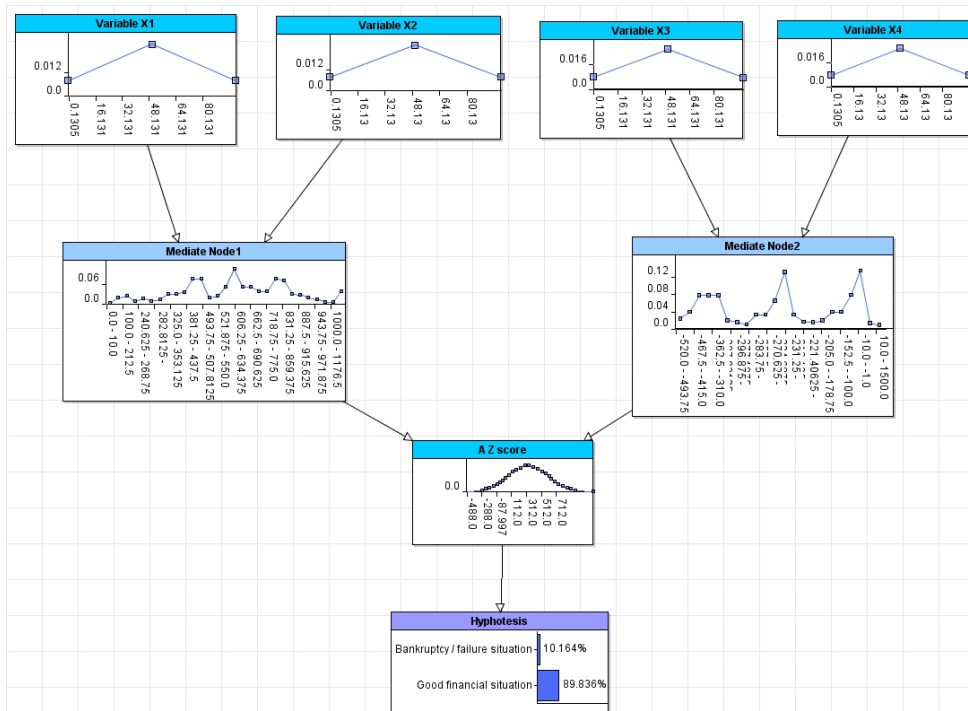


**Figure 7 – Complete Hyphotesis Testing Model using BM Discretizing Method**

In the BM Method we work within three intervals [0, 43.5], [43.5, 56.4] and [56.4, 100] defined in the *sample nodes*. In this case we obtain a percentage of 89.836 % "Good financial situation" (see Figure 7).

After the Sensitivity Analysis we obtain the results shown in Figure 8.
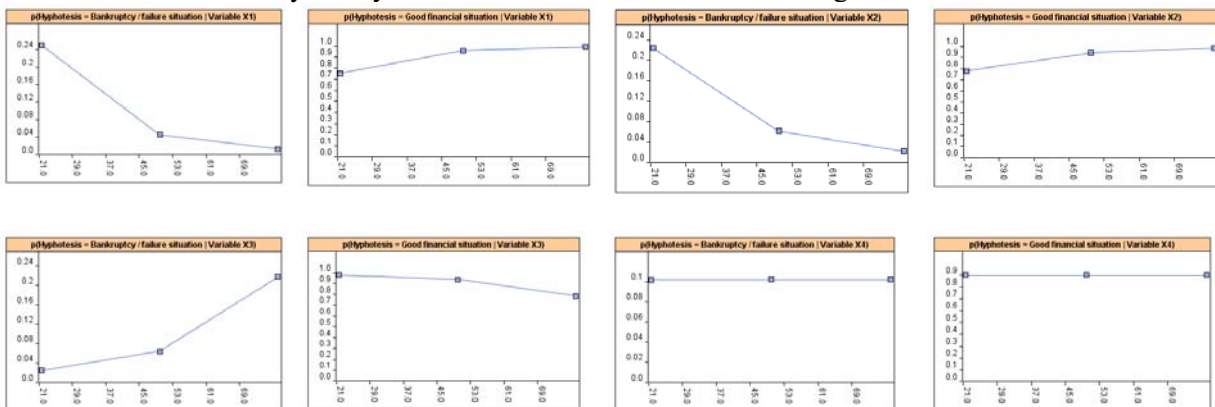


**Figure 8 - Sensitivity Analysis using Bracket Medians Discretizing Method**

## 3.3 Simulation using Pearson-Tukey (P-T) Discretizing Method

In this case, we must first build our BN, just like in the subsection 3.2. The key for this BN is defined in the *sample nodes*, more accurate in his Node States (see Figure 9) and in his Node Probability Table (see Figure 10).
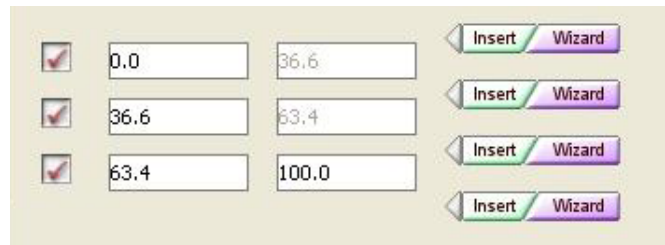


**Figure 9 - Node States for the sample nodes in P-T Discretizing Method**
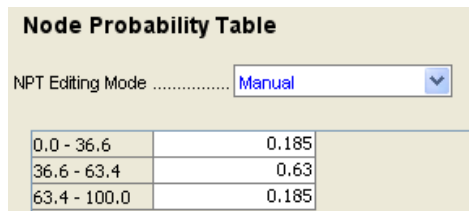


**Figure 10 – NPT in P-T Discretizing Method**

After defining all the nodes like in the section 2, we obtain the BN shown in the Figure 11.
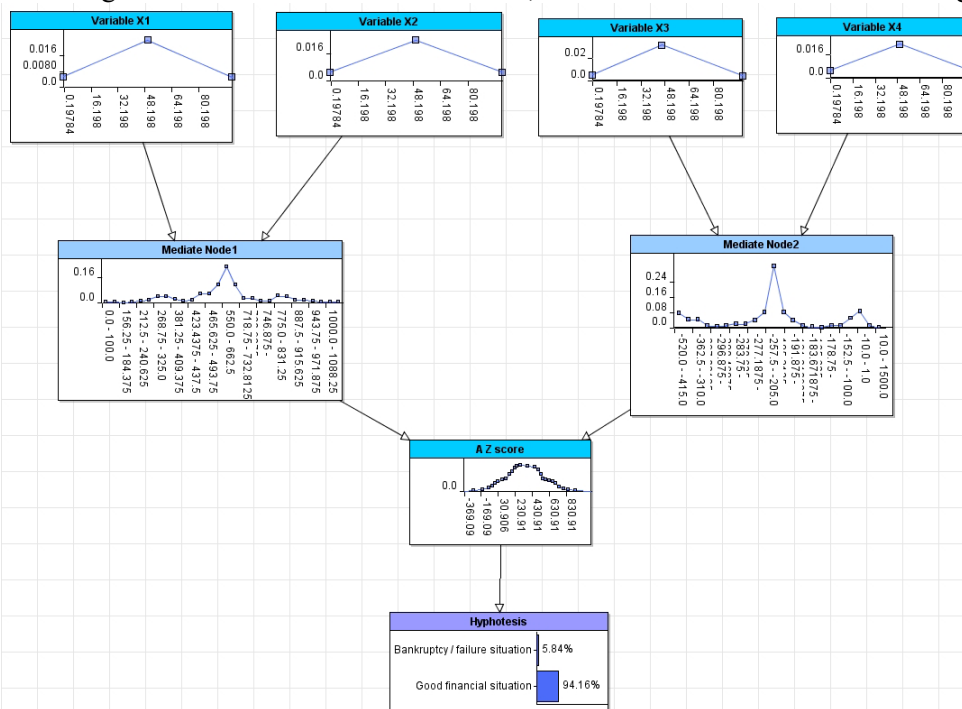


**Figure 11 – Complete Hyphotesis Testing Model using P-T Discretizing Method**

In the P-T Method we work within three intervals [0, 36.6], [36.6, 63.4] and [63.4, 100] defined in the *sample nodes*. In this case we obtain a percentage of 94.16 % "Good financial situation" (see Figure 11).

After the Sensitivity Analysis we obtain the results shown in Figure 12.
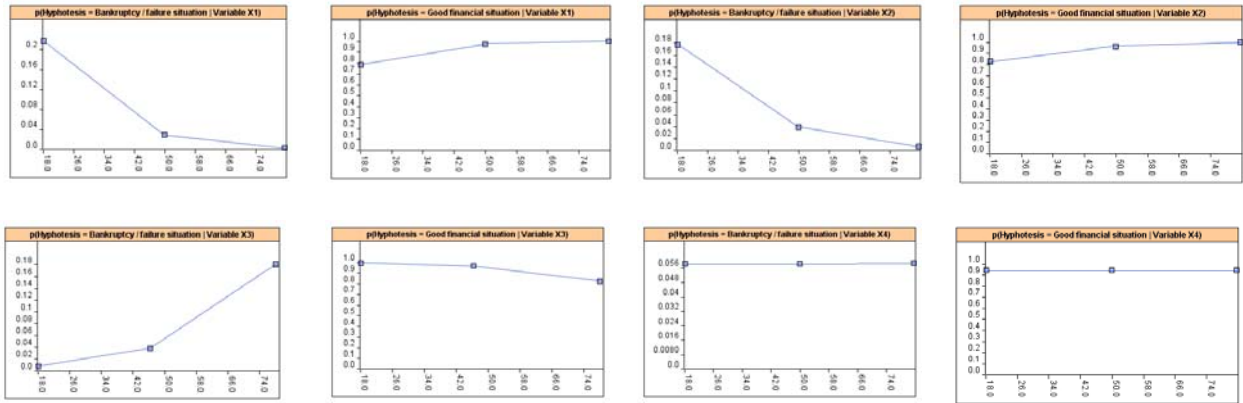


**Figure 12 - Sensitivity Analysis using Pearson-Tukey Discretizing Method**

# 4 Conclusions

The Hyphotesis Testing Model results obtained in our paper are highlighted in Table 1.

| | 1 Interval without scenario | 1 Interval with scenario | Bracket Medians | Pearson-Tukey |
|---|---|---|---|---|
| **Bankruptcy / failure situation** | *1,300%* | *6,367%* | *10,164%* | *5,84%* |
| **Good financial situation** | *98,700%* | *93,633%* | *89,836%* | *84,160%* |

**Table 1 – Comparison between Hyphotesis Testing Model results using the presented methods**
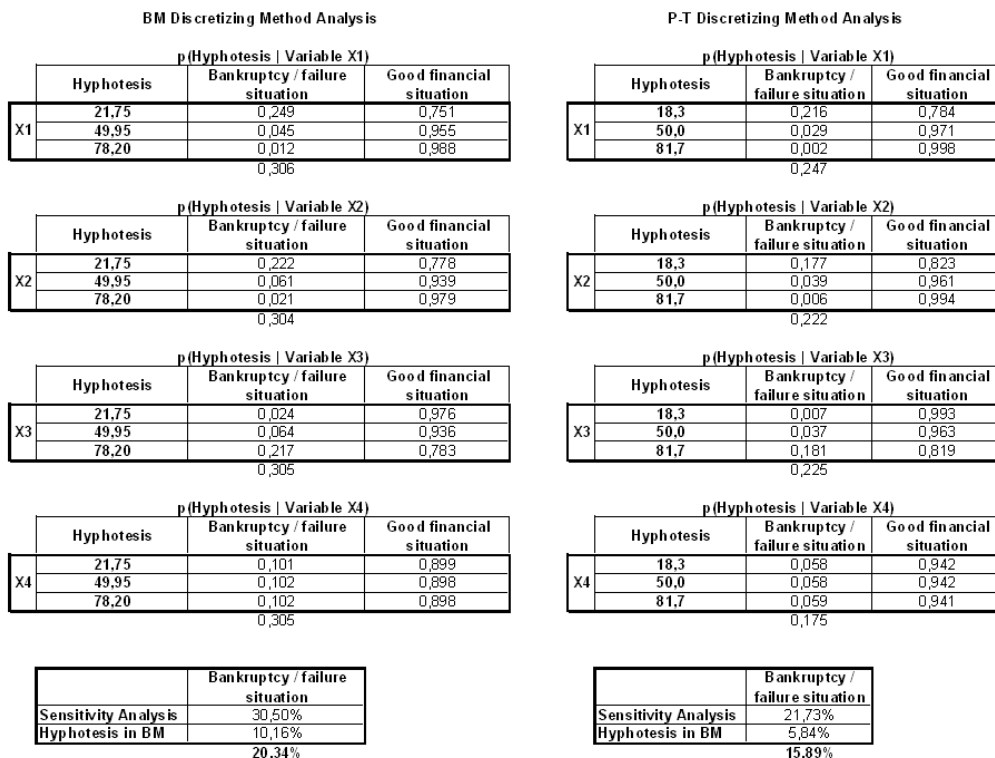


**Figure 13 – Sensitivity Analysis in BM and P-T Discretizing Methods**

The influence of sample nodes $X_i$ prediction on the assumption node Hyphotesis prediction is analyzed in Figure 13.

Mean of the bankruptcy prediction for the sample nodes $X_i$ in BM discretizing method is 30.50%. Also in this method the bankruptcy prediction for the assumption node Hyphotesis is 10.16%. Difference between the two prediction values is 20.34%.

Mean of the bankruptcy prediction for the sample nodes $X_i$ in P-T discretizing method is 21.73%. Also in this method the bankruptcy prediction for the assumption node Hyphotesis is 5.84%. Difference between the two prediction values is 15.89%.

From these results we can conclude that the bankruptcy prediction has a higher degree of accuracy using the P-T discretizing method.

# References

[1] Hogg, R.V., and A.T. Craig, *Introduction to Mathematical Statistics*, Macmillan, New York, 1972

[2] Cooper, G.F*., The Computational Complexity of Probabilistic Inference Using Bayesian Belief Networks*, Artificial Intelligence, Vol. 33, 1990

[3] Lindley, D.V., *Introduction to Probability and Statistics from a Bayesian Viewpoint*, Cambridge University Press, London, 1985

[4] Berry, D.A., *Statistics: A Bayesian Perspective*, Wadsworth, Belmont, California, 1996

[5] Anghel Ion – *Falimentul – radiografie şi predicţie*, Ed. Economică, Bucureşti, 2002

[6] Agena 2007, Press Release, *http://www.agenarisk.com/agenarisk/case_13.shtml*

[7] Crăciun Mihaela–Daciana, Bucerzan Dominic, Raţiu Crina, *Prediction Analysis of Bankrupcty using Bayesian Networks*, Analele Universităţii Maritime Constanţa, Anul XI, Volumul 14, 2010

MIHAELA CRĂCIUN
"Aurel Vlaicu University" of Arad
Dep. of Mathematics and Computer Science
Str.Elena Dragoi, No.2, 310330 - Arad,
Complex Universitar M
ROMANIA
mihaeladacianacraciun@yahoo.com

DOMINIC BUCEZAN
"Aurel Vlaicu University" of Arad
Dep. of Mathematics and Computer Science
Str.Elena Dragoi, No.2, 310330 - Arad,
Complex Universitar M
ROMANIA
dominic@bbcomputer.ro

CRINA RAŢIU
DARAMEC SRL
Şofronea, Arad
ROMANIA
ratiu_anina@yahoo.com

# On some fuzzy positive and linear operators

**Anca Farcaş**

### Abstract

The purpose of this work is to show that fuzzy Bernstein-Stancu operators introduced in [3] satisfy an A-statistical version of fuzzy Korovkin theorem. Some properties of these operators are also proved. An example of new fuzzy positive and linear operators is presented.

## 1 Introduction

Sometimes, the phenomena encountered in real life do not have a precise definition for membership criteria. For example, the class of plants obviously includes flowers, trees, grass but there are elements like amoeba or bacteria which have an ambiguous status regarding their belonging to plants class.

Similarly, we are dealing with ambiguity when we want to compare the number 30 with the class of real numbers much greater than zero, which obviously do not have a precise definition. In mathematics, this kind of uncertainty can be modeled in two ways. First way is a probabilistic one and the second way refers to fuzzy logic.

For the first time modeling uncertainty through fuzzy logic was approached by L.Zadeh [13] who introduced the fuzzy sets as basis for reasoning with multiple truth values. Later on, the concept was applied in many areas of science like finance, weather prediction, hand writing analysis, electronics, biomedicine or elevators.
Considering the wide applicability of fuzzy sets naturally arises the fuzzy functions approximation problem (which can model complicate real processes) with effectively calculable functions.

In this paper, starting from classical Bernstein-Stancu operator defined in [10],

$$(B_m^{(\alpha,\beta)} f)(x) = \sum_{k=0}^{m} \binom{m}{k} x^k (1-x)^{m-k} f\left(\frac{k+\alpha}{m+\beta}\right), m \in \mathbb{N}, x \in [0,1],$$

and considering its fuzzy variant defined in [3], we prove some properties concerning this variant and we also introduce a new class of fuzzy operators.

In Section 2, we collect some basic elements used throughout the paper. Further on, we prove that the fuzzy Bernstein-Stancu operators satisfy a fuzzy Korovkin - type theorem and this theorem holds for our new class of operators.

## 2 Preliminaries

We need the following definitions.

**Definition 1** ([12]) *Let* $\mu : \mathbb{R} \to [0,1]$ *with the following properties:*

**(i)** $\mu$ *is normal, i.e,* $\exists\ x_0 \in \mathbb{R},\ \mu(x_0) = 1$;

**(ii)** $\mu$ *is a convex fuzzy subset, i.e,* $\forall x, y \in \mathbb{R}, \forall \lambda \in [0,1],\ \mu(\lambda x + (1-\lambda)y) \geq \min\{\mu(x), \mu(x)\}$;

**(iii)** $\mu$ *is upper semicontinuous on* $\mathbb{R}$, *i.e,* $\forall x_0 \in \mathbb{R}$ *and* $\forall \epsilon > 0$, *a neighborhood* $V(x_0)$ *of* $x_0$ *exists such that* $\mu(x) \leq \mu(x_0) + \epsilon,\ \forall x \in V(x_0)$;

**(iv)** *The set* $\overline{\{x \in \mathbb{R} : \mu(x) > 0\}}$ *is compact in* $\mathbb{R}$, *where* $\bar{A}$ *denotes the closure of* $A$.

*The function* $\mu$ *is called fuzzy number.*

The set of all $\mu$ is denoted by $\mathbb{R}_{\mathcal{F}}$.
For $0 < r \leq 1$ and $\mu \in \mathbb{R}_{\mathcal{F}}$ define

$$[\mu]^r := \{x \in \mathbb{R} : \mu(x) \geq r\} \quad \text{and} \quad [\mu]^0 := \overline{\{x \in \mathbb{R} : \mu(x) > 0\}}.$$

We recall now that for each $r \in [0,1]$, $[\mu]^r$ is a closed and bounded real interval, see [8].

**Definition 2** ([12]) *Let* $u, v \in \mathbb{R}$ *and let* $\lambda \in \mathbb{R}$. *We define uniquely the sum* $u \oplus v$ *and the product* $\lambda \odot u$ *as follows*

$$[u \oplus v]^r = [u]^r + [v]^r, \quad [\lambda \odot u]^r = \lambda[u]^r, \quad \forall r \in [0,1],$$

*where* $[u]^r + [v]^r$ *means the usual addition of two intervals (as subsets of* $\mathbb{R}$) *and* $\lambda[u]^r$ *means the usual product between a scalar and a subset of* $\mathbb{R}$.

Notice that $1 \odot u = u$ and $u \oplus v = v \oplus u$, $\lambda \odot u = u \odot \lambda$ hold.

**Remark 1** *If we have* $0 \leq r_1 \leq r_2 \leq 1$, *then* $[u]^{r_1} \subseteq [u]^{r_2}$. *We denote the interval* $[u]^r = [u_-^{(r)}, u_+^{(r)}]$, *where* $u_-^{(r)} \leq u_+^{(r)}$, $u_-^{(r)}, u_+^{(r)} \in \mathbb{R}_+$, $\forall r \in [0,1]$.

In the following sections we will use the fuzzy metric given in [11],

**Definition 3** *Let* $D : \mathbb{R}_{\mathcal{F}} \times \mathbb{R}_{\mathcal{F}} \to \mathbb{R}_+$ *be defined as follows*

$$\begin{aligned} D(u, v) &:= \sup_{r \in [0,1]} \max\{|u_-^{(r)} - v_-^{(r)}|, |u_+^{(r)} - v_+^{(r)}|\} \\ &= \sup_{r \in [0,1]} \rho([u]^r, [v]^r), \end{aligned}$$

*where* $[v]^r = [v_-^{(r)}, v_+^{(r)}]$, $u, v \in \mathbb{R}_{\mathcal{F}}$ *and* $\rho$ *is the Hausdorff distance.*

Following [11], $(\mathbb{R}_{\mathcal{F}}, D)$ is a complete metric space.
Let $f, g : [a,b] \to \mathbb{R}_{\mathcal{F}}$ be fuzzy number valued functions. Then, according to [5], the distance between f and g is given by

$$D^*(f, g) := \sup_{x \in [a,b]} \sup_{r \in [0,1]} \max\{|f_-^{(r)}(x) - g_-^{(r)}(x)|, |f_+^{(r)}(x) - g_+^{(r)}(x)|\}.$$

**Definition 4** ([2], [7]) *Let* $f : [a,b] \to \mathbb{R}_{\mathcal{F}}$ *be a fuzzy real number valued function. We define the first fuzzy modulus of continuity of f by*

$$\omega_1^{(\mathcal{F})}(f, \delta) := \sup_{\substack{x, y \in [a,b] \\ |x-y| \leq \delta}} D(f(x), g(x)), \quad \delta \in (0, b-a].$$

**Definition 5** ([3]) *Let* $f \in C([0,1], \mathbb{R}_{\mathcal{F}})$. *We define the second fuzzy modulus of continuity of f by*

$$\omega_2^{(\mathcal{F})}(f, h) := \sup_{\substack{u, v \in [0,1] \\ |u-v| \leq 2h \\ h > 0}} \left\{ D\left( f(u) \oplus f(v), 2 \odot f\left(\frac{u+v}{2}\right) \right) \right\}.$$

The set of all fuzzy continuous functions on the interval $[a, b]$ is denoted by $C_{\mathcal{F}}[a, b]$.

**Remark 2** ([2]) *Let $f, g \in C_{\mathcal{F}}[a, b]$. We say that $f$ is fuzzy larger than $g$ pointwise and we denote it by $f \succeq g$ if and only if*

$$f(x) \succeq g(x) \quad iff \quad f_{-}^{(r)}(x) \geq g_{-}^{(r)}(x) \quad and \quad f_{+}^{(r)}(x) \geq g_{+}^{(r)}(x), \ \forall x \in [a, b], \ \forall r \in [0, 1].$$

**Definition 6** ([5]) *Let $L : C_{\mathcal{F}}[a, b] \to C_{\mathcal{F}}[a, b]$ be an operator. Then $L$ is said to be fuzzy linear if, for every $\lambda_1, \lambda_2 \in \mathbb{R}$, $f_1, f_2 \in C_{\mathcal{F}}[a, b]$, and $x \in [a, b]$,*

$$L(\lambda_1 \odot f_1 \oplus \lambda_2 \odot f_2; x) = \lambda_1 \odot L(f_1; x) \oplus \lambda_2 \odot L(f_2; x)$$

*holds. Also $L$ is called fuzzy positive linear operator if it is fuzzy linear and the condition $L(f; x) \preceq L(g; x)$ is satisfied for any $f, g \in C_{\mathcal{F}}[a, b]$ and all $x \in [a, b]$ with $f(x) \preceq g(x)$.*

We now recall the definition of fuzzy Bernstein-Stancu operators as it was given in [3].

**Definition 7** *Let $f \in C([0, 1], \mathbb{R}_{\mathcal{F}})$, $m \in \mathbb{N}$, $0 \leq \alpha \leq \beta$. We define*

$$({}^{\mathcal{F}}L_m^{\alpha,\beta} f)(x) = \sum_{k=0}^{m}{}^{*} p_{m,k}(x) \odot f\left(\frac{k + \alpha}{m + \beta}\right), \quad x \in [0, 1], \tag{1}$$

*where $p_{m,k}(x) = \begin{pmatrix} m \\ k \end{pmatrix} x^k (1 - x)^{m-k}$.*

Here $\sum^{*}$ stands for fuzzy summation.

For the reader's convenience, we recall the almost convergence, the statistically convergence, the A-statistically convergence of a real sequence, and the fuzzy Korovkin theorem.

Based on the result of Lorentz [9], a bounded real sequence $(x_n)_{n \in \mathbb{N}}$ is said to be almost convergent to a real number $L$ if and only if

$$\lim_{p->\infty} \frac{x_n + ... + x_{n+p-1}}{p} = L.$$

**Definition 8** ([6]) *A sequence $(x_n)_{n \in \mathbb{N}}$ of real numbers is called statistically convergent to a real number $L$, if for every $\varepsilon > 0$,*

$$\delta(\{n \in \mathbb{N} : |x_n - L| \geq \varepsilon\}) = 0,$$

*where*

$$\delta(E) := \lim_{N \to \infty} \frac{1}{N} \sum_{j=1}^{n} \chi_E(j)$$

*represents the density of the set $E \subseteq \mathbb{N}$, and $\chi_E$ is the characteristic function associated to set $E$. We denote this limit by*

$$st - \lim_n x_n = L.$$

**Definition 9** *Let $A = (a_{j,n})_{j,n \in \mathbb{N}}$ be a non-negative regular summability method. The sequence $(x_n)_{n \in \mathbb{N}}$ is said to be A-statistically convergent to a real number $L$ if for every $\varepsilon > 0$,*

$$\delta_A(\{n \in \mathbb{N} : |x_n - L| \geq \varepsilon\}) = 0,$$

*or equivalent*

$$\lim_{j \to \infty} \sum_{\{n : |x_n - L| \geq \varepsilon\}} a_{j,n} = 0.$$

*We denote this limit by*

$$st_A - \lim_n x_n = L.$$

We now give the fuzzy Korovkin theorem.

**Theorem 1** ([2]) *Let $(L_n)_{n\in\mathbb{N}}$ be a sequence of fuzzy positive linear operators from $C_{\mathcal{F}}[a,b]$ into itself. Assume that there exists a corresponding sequence $(\widetilde{L_n})_{n\in\mathbb{N}}$ of positive linear operators from $C[a,b]$ into itself with the property*

$$\{L_n(f;x)\}_{\pm}^{(r)} = \widetilde{L_n}(f_{\pm}^{(r)};x) \tag{2}$$

*for all $x \in [a,b]$, $r \in [0,1]$, $n \in \mathbb{N}$ and $f \in C_{\mathcal{F}}[a,b]$. Assume further that*

$$\lim_n \|\widetilde{L_n}(e_i) - e_i\| = 0 \ \ \text{for each} \ \ i = 0,1,2.$$

*Then, for all $f \in C_{\mathcal{F}}[a,b]$, we have*

$$\lim_n D^*(L_n(f), f) = 0.$$

# 3   Main results

In this section we prove that fuzzy Bernstein-Stancu operators satisfy the A-statistical version of fuzzy Korovkin theorem which was first obtained in [2].

First of all, we give the following lemma:

**Lemma 1** *The fuzzy Bernstein-Stancu operators defined by (1) are positive and linear operators.*

**Proof.** The linearity:

$$^{\mathcal{F}}L_m^{\alpha,\beta}(\lambda_1 \odot f_1 \oplus \lambda_2 \odot f_2; x) = \sum_{k=0}^{m}{}^{*} p_{m,k}(x) \odot \left[\lambda_1 \odot f_1\left(\frac{k+\alpha}{m+\beta}\right) \oplus \lambda_2 \odot f_2\left(\frac{k+\alpha}{m+\beta}\right)\right]$$

$$= \lambda_1 \underbrace{\sum_{k=0}^{m}{}^{*} p_{m,k}(x) \odot f_1\left(\frac{k+\alpha}{m+\beta}\right)}_{^{\mathcal{F}}L_m^{(\alpha,\beta)}(f_1;x)} \oplus \lambda_2 \underbrace{\sum_{k=0}^{m}{}^{*} p_{m,k}(x) \odot f_2\left(\frac{k+\alpha}{m+\beta}\right)}_{^{\mathcal{F}}L_m^{(\alpha,\beta)}(f_2;x)}.$$

The positivity:  Let $f \preceq g$,  where $f, g \in C_{\mathcal{F}}[0,1]$.
This implies

$$f\left(\frac{k+\alpha}{m+\beta}\right) \preceq g\left(\frac{k+\alpha}{m+\beta}\right), \ k = \overline{0,m},$$

and we deduce

$$\sum_{k=0}^{m}{}^{*} p_{m,k}(x) \odot f\left(\frac{k+\alpha}{m+\beta}\right) \preceq \sum_{k=0}^{m}{}^{*} p_{m,k}(x) \odot g\left(\frac{k+\alpha}{m+\beta}\right).$$

Consequently, $\qquad\qquad\qquad {}^{\mathcal{F}}L_m^{(\alpha,\beta)}(f;x) \preceq^{\mathcal{F}} L_m^{(\alpha,\beta)}(g;x).$ $\qquad\qquad\qquad\square$

In order to give our main result we need the following theorem.

**Theorem 2** ([5]) *Let $A = (a_{j,n})$ be a non-negative regular summability matrix and let $(L_n)_{n\in\mathbb{N}}$ be a sequence of fuzzy positive linear operators from $C_{\mathcal{F}}[a,b]$ into itself. Assume that there exists a corresponding sequence $(\widetilde{L_n})_{n\in\mathbb{N}}$ of positive linear operators from $C[a,b]$ into itself with the property (2). Assume further that*

$$st_A - \lim_n \|\widetilde{L_n}(e_i) - e_i\| = 0 \ \ \text{for each} \ \ i = 0,1,2.$$

*Then, for all $f \in C_{\mathcal{F}}[a,b]$, we have*

$$st_A - \lim_n D^*(L_n(f), f) = 0. \tag{3}$$

**Theorem 3** *If the sequence* $({}^{\mathcal{F}}L_m^{(\alpha,\beta)}f)_{m\in\mathbb{N}}$ *defined by* (1) *satisfies the conditions*

$$st_A - \lim_m \|\widetilde{L_m}^{(\alpha,\beta)}(e_i) - e_i\| = 0, \ i = 0, 1, 2, \tag{4}$$

*then*

$$st_A - \lim_m D^*({}^{\mathcal{F}}L_m^{(\alpha,\beta)}, f) = 0. \tag{5}$$

**Proof.** Since

$$(\widetilde{L_m}e_0)(x) = \sum_{k=0}^m \binom{m}{k} x^k(1-x)^{m-k} = 1, \tag{6}$$

clearly

$$st_A - \lim_m \|\widetilde{L_m}e_0 - e_0\| = 0. \tag{7}$$

We can also write

$$\begin{aligned}
(\widetilde{L_m}e_1)(x) &= \sum_{k=0}^m \binom{m}{k} x^k(1-x)^{m-k}\frac{k}{m+\beta} + \sum_{k=0}^m \binom{m}{k} x^k(1-x)^{m-k}\frac{\alpha}{m+\beta} \\
&= x + \frac{\alpha - \beta x}{m+\beta}.
\end{aligned} \tag{8}$$

Consequently, we get

$$\|\widetilde{L_m}e_1 - e_1\| \le \frac{\alpha}{m+\beta} + \frac{\beta}{m+\beta}.$$

For a given $\varepsilon > 0$, we consider the sets

$$D := \{m : \|\widetilde{L_m}e_1 - e_1\| \ge \varepsilon\},$$

$$D_1 := \{m : \frac{\alpha}{m+\beta} \ge \frac{\varepsilon}{2}\}, \quad D_2 := \{m : \frac{\beta}{m+\beta} \ge \frac{\varepsilon}{2}\}.$$

It is obvious that $D \subset D_1 \cup D_2$. Consequently, for each $j \in \mathbb{N}$, we get

$$\sum_{m\in D} a_{j,m} \le \sum_{m\in D_1} a_{j,m} \le \sum_{m\in D_2} a_{j,m}. \tag{9}$$

Since $st_A - \lim_m \dfrac{\alpha}{m+\beta} = st_A - \lim_m \dfrac{\beta}{m+\beta} = 0$, taking in (9) the limit as $j$ tends to infinity, we conclude

$$\lim_{j\to\infty} \sum_{m\in D} a_{j,m} = 0.$$

This identity implies

$$st_A - \lim_m \|\widetilde{L_m}e_1 - e_1\| = 0. \tag{10}$$

Further on, we obtain

$$\begin{aligned}
\widetilde{L_m}(e_2, x) &= \sum_{k=0}^m \binom{m}{k} x^k(1-x)^{m-k}\frac{k^2}{(m+\beta)^2} + \sum_{k=0}^m \binom{m}{k} x^k(1-x)^{m-k}\frac{2k\alpha}{(m+\beta)^2} + \\
&\quad + \sum_{k=0}^m \binom{m}{k} x^k(1-x)^{m-k}\frac{\alpha^2}{(m+\beta)^2} \\
&= \frac{m^2}{(m+\beta)^2}\left(x^2 + \frac{x(1-x)}{m}\right) + \frac{2\alpha m}{(m+\beta)^2}x + \frac{\alpha^2}{(m+\beta)^2} \\
&= x^2 + \frac{xm - x^2m + 2\alpha xm + \alpha^2 - \beta^2 x^2 - 2\beta x^2 m}{(m+\beta)^2} \\
&= x^2 + \frac{mx(1-x) + (\alpha - \beta x)(2mx + \alpha + \beta x)}{(m+\beta)^2},
\end{aligned} \tag{11}$$

which implies

$$\|\widetilde{L_m}e_2 - e_2\| \le \frac{m}{4(m+\beta)^2} + \frac{(\alpha+\beta)(2m+\alpha+\beta)}{(m+\beta)^2}.$$

Reasoning in a similar manner as above, one obtains

$$st_A - \lim_m \|\widetilde{L_m}e_2 - e_2\| = 0. \tag{12}$$

On the basis of $(7), (10), (12)$, taking into account Theorem 2, we arrive at $(5)$. The proof is complete. $\square$

# 4    Examples

In the following section we give an example of fuzzy positive and linear operators starting from fuzzy Bernstein-Stancu operators. These new operators satisfy a generalized fuzzy Korovkin theorem.

First of all, we recall a generalization of Theorem 1 using matrix summability method.

**Theorem 4** ([4]) *Let* $\mathcal{A} = (A^n)_{n \ge 1}$ *be a sequence of infinite non-negative real matrices such that*

$$\sup_{n,k} \sum_{j=1}^{\infty} a_{kj}^n < \infty \tag{13}$$

*and let* $(L_j)_{j \in \mathbb{N}}$ *be a sequence of fuzzy positive linear operators from* $C_{\mathcal{F}}[a,b]$ *into itself. Assume that there exists a corresponding sequence* $(\widetilde{L_j})_{j \in \mathbb{N}}$ *of positive linear operators from* $C[a,b]$ *into itself satisfying* $(2)$. *Assume further that*

$$\lim_{k \to \infty} \left\| \sum_{j=1}^{\infty} a_{kj}^n \widetilde{L_j}(e_i) - e_i \right\| = 0, \quad for\ each\ \ i = 0,1,2, \tag{14}$$

*uniformly in n. Then, for all* $f \in C_{\mathcal{F}}[a,b]$, *we have*

$$\lim_{k \to \infty} D^* \left( \sum_{j=1}^{\infty} a_{kj}^n L_j(f), f \right) = 0,$$

*uniformly in n.*

**Example 1** *Let* $(u_j)_{j \in \mathbb{N}}$ *a sequence almost convergent to zero such that* $u_j \ge 0,\ \forall\ j \in \mathbb{N}$.
*For example,* $u_j = 1 + (-1)^j$ *is a special sequence which is almost convergent but is not statistically convergent.*
*By using our general sequence* $(u_j)_{j \in \mathbb{N}}$ *and the fuzzy Bernstein-Stancu operators we give an example of fuzzy positive and linear operators defined on* $C_{\mathcal{F}}[0,1]$ *which satisfy Theorem 4.*
*So, let*

$$\mathcal{N}_j(f;x) = u_j \odot^{\mathcal{F}} L_m^{(\alpha,\beta)}(f;x),\ j \in \mathbb{N},\ x \in \mathbb{R},\ f \in C_{\mathcal{F}}[0,1]. \tag{15}$$

*The corresponding real positive linear operators are given by*

$$\widetilde{N_j}(f_{\pm}^{(r)};x) = u_j \sum_{k=0}^{j} \binom{j}{k} x^k (1-x)^{j-k} \cdot f_{\pm} \left( \frac{k+\alpha}{j+\beta} \right),\quad j \in \mathbb{N},\ x \in \mathbb{R},\ f_{\pm}^{(r)} \in C[0,1]. \tag{16}$$

*On the basis of* $(6), (8), (11)$, *we observe that*

$$\widetilde{N_j}(e_0, x) = u_j,$$

$$\widetilde{N_j}(e_1, x) = u_j \left[ x + \frac{\alpha - \beta x}{m + \beta} \right],$$

$$\widetilde{N_j}(e_2, x) = u_j \left[ x^2 + \frac{mx(1-x) + (\alpha - \beta x)(2mx + \alpha + \beta x)}{(m + \beta)^2} \right].$$

*Since $(u_j)_{j \in \mathbb{N}}$ is almost convergent to zero, we get*

$$\lim_{k \to \infty} \| \sum_{j=1}^{\infty} a_{kj}^n \widetilde{N_j}(e_i) - e_i \| = 0, \quad i = 0, 1, 2, \quad uniformly \ \ in \ n.$$

*Consequently, from Theorem 4 we have*

$$\forall f \in C_{\mathcal{F}}[0, 1], \quad \lim_{k \to \infty} D^* (\sum_{j=1}^{\infty} a_{kj}^n N_j(f), f) = 0, \quad uniformly \ \ in \ n.$$

**Remark 3** *According to ([3; Theorem 29]), for our operators defined by (15) we have*

$$\sup_{x \in [0,1]} D\left( (\mathcal{N}_j f)(x), f(x) \right) \le \left[ 3 + \frac{m^3 + 4m\alpha^2(m - \beta^2)}{4(m - \beta^2)(m + \beta)^2} \right] \omega_2^{(\mathcal{F})} \left( f, \frac{1}{\sqrt{m}} \right) + \frac{2(\alpha + \beta)\sqrt{m}}{m + \beta} \omega_1^{(\mathcal{F})} \left( f, \frac{1}{\sqrt{m}} \right).$$

# 5 Conclusions and Future Work

In this paper we have proved that the fuzzy Bernstein -Stancu operators satisfy a fuzzy Korovkin theorem and we have presented an example of fuzzy positive and linear operators defined on $C_{\mathcal{F}}[0, 1]$ which satisfy the fuzzy Korovkin theorem.

In the future we will explore the area of fuzzy positive and linear operators and we will introduce new classes of fuzzy positive and linear operators. Moreover, we will study the statistical convergence of sequences of these type of operators and we will extend these properties to multidimensional case.

# References

[1] O. Agratini, *Approximation by linear operators*, Presa Universitară Clujeană, 2000 (in Romanian).

[2] G. A. Anastassiou, On basic fuzzy Korovkin theory, *Studia Univ. "Babeş - Bolyai" Math.*, 4, 3-10, 2005.

[3] G. A. Anastassiou, *Transfers of real approximations to vectorial and fuzzy setting*, in Intelligent Mathematics: Computational Analysis, Springer, 2011.

[4] G. A. Anastassiou, $\mathcal{A}$ - *summability and fuzzy Korovkin approximation*, in Fuzzy Mathematics:Approximation Theory, Springer, 2010.

[5] G. A. Anastassiou, O. Duman, Statistical fuzzy approximation by fuzzy positive linear operators, *Computers and Mathematics with Applications*, 55, 573-580, 2008.

[6] J. A. Fridy, On statistical convergence, *Analysis* 5, 301-313, 1985.

[7] S. G. Gal, *Approximation theory in fuzzy setting, in: Handbook of Analytic-Computational Methods in Applied Mathematics*, Chapman and Hall/CRC, 2000.

[8] R. Goetschel Jr., W. Voxman, Elementary fuzzy calculus, *Fuzzy Sets and Systems*, 18, 31-43, 1986.

[9] G. G. Lorentz, A contribution to the theory of divergent sequences, *Acta Math.*, 80, 167-190, 1948.

[10] D. D. Stancu, On a generalization of Bernstein polynomials, *Studia Univ. "Babeş-Bolyai", Math.*, vol. 14, fasc.2, 31-45, 1969 (in Romanian).

[11] Congxin Wu, Ming Ma, On embedding problem of fuzzy number space: Part 1, *Fuzzy Sets and Systems*, 44, 33-38, 1991.

[12] Congxin Wu, Zengtai Gong, On Henstock integral of fuzzy number valued functions, *Fuzzy Sets and Systems*, 3, 523-532 2001.

[13] L. A. Zadeh, Fuzzy Sets, *Inform. and Control*, 8, 338-353, 1965.

Anca Farcaş
Babeş Bolyai University
Faculty of Mathematics and Computer Science
Kogălniceanu Street, No.1, 400084 Cluj-Napoca
ROMANIA
E-mail: *anca.farcas@ubbcluj.ro*

# A Parameter Adjustment Tool for CompuCell3D

**Xuefeng Gao, Sabin Tabirca**

### Abstract

The aim of this paper is to present a graphic user interface (GUI) tool for parameter adjustment in CompuCell3D, an open source multi-cell modeling framework and software packages. We provide an overview of this software and its underlying *Glazier-Graner-Hogeweg* (GGH) algorithm. To enhance the parameter adjustment approach in CompuCell3D simulations, we develop an easy-to-use tool - ParaAdjuster - using Python and PyQt. We briefly present a case study of using ParaAdjuster combined with CompuCell3D to simulate a solid tumor development.

## 1   Introduction

Mathematical and computational models have become more accepted by the biological community both as a means to motivate experimentation but also as a route to integrate multiple experimental measurements to generate testable predictions. CompuCell3D is able to facilitate biological simulations by defining spatially-extended *generalized cells*, which can represent clusters of cells, single cells, sub-compartments of single cells or small subdomains of noncellular materials [1]. A significant advantage of this flexible definition is to allow tuning of the level of detail in a simulation from intracellular to tissue without switching framework to examine the effect of changing the level of detail on a macroscopic outcome.

In this paper, we present an overview of the CompuCell3D modeling framework and explore the use of a parameter adjustment tool, ParaAdjuster, for the purpose of modifying parameters of CompuCell3D models. Both CompuCell3D and ParaAdjuster use Python (www.python.org), which is a popular scripting language for developing interactive scientific applications [3], and PyQt toolkit (wiki.python.org/moin/PyQt) in the user interface design.

## 2   CompuCell3D

CompuCell3D (www.compucell3d.org) is an open-source, multi-cell, multi-scale modeling framework for developing biomedical simulations [1]. CompuCell3D is widely used in modeling a broad range of phenomena in biology and physics, e.g. biological cells, foams and ferromagnetism. It implements *Glazier-Graner-Hogeweg* (GGH) model (a.k.a. Cellular Potts Model), and operates on a regular lattice in two-dimensional or three-dimensional. GGH follows an *energy-minimization* philosophy and has *generalized cells* as its most primitive objects. Users can define the properties and behaviors of a *generalized cell* type to mimic a real cell or cell-like object in the real world, e.g. a cancer cell. A typical CompuCell3D simulation includes a list of objects, a description of their interactions and dynamics, and appropriate initial conditions. Cells evolving in time by interacting with each other as well as the surrounding environment, which tends to minimize the total *effective energy*. Additional fields can also be defined that may affect the dynamics of the *generalized cells*. These fields can represent chemical diffusants and
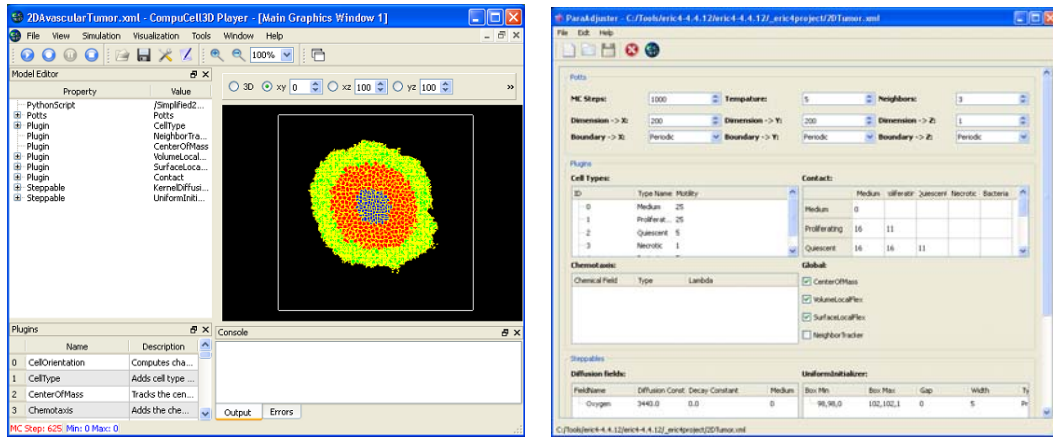
Figure 1: (a)An interactive CompuCell3D application shows a solid tumor growth model. (b)The ParaAdjuster GUI displaying the tumor model parameters using different widgets according to specific data type.

non-diffusing extracellular matrix.

CompuCell3D can be invoked either as an interactive application displaying the evolution of cells and the underlying fields in a GUI (see Figure 1(a)), or running a batch program that outputs raw data that can be post-processed. A CompuCell3D model is created textually using either an eXtensible Markup Language (XML) or Python script, in which a user can define the cell types, contact energy, fields and the global environmental parameters. The system dynamics can only be defined in a Python file, wherein a user can define the mechanisms controlling the cell behaviors, cell-cell and cell-environment interactions. To explore the effects of changing parameters in a model a user must modify the XML or Python files textually, which is inefficient and lack of error control. One may expect a tool to manage the parameters in a compact view with only the parameters themselves but all the tags in an XML file. It is this functionality that we wish to address in ParaAdjuster.

# 3 ParaAdjuster

ParaAdjuster is a light tool for CompuCell3D users to manage parameters in a graphical way (see Figure 1(b)). Similar tools like VisTrails, known as a scientific workflow and provenance management system [7], offers parameter exploration and data management for CompuCell3D. VisTrails sweeps parameters in a CompuCell3D model through the input and output ports of a module that executes the related CompuCell3D simulation. Compared to VisTrails, ParaAdjuster offers a direct modification feature with less additional effort. The workflow of using ParaAdjuster for an existed ComCell3D project is as follows:

1. Load a XML file that contains CompuCell3D definitions.

2. ParaAdjuster parses the loaded file and displays the data on a GUI using different widgets (e.g. spinner, text box, tree, etc.) according to the data types (e.g. integer, string, double, etc.).

3. Modify parameters in the widget and lunch a CompuCell3D simulation.

ParaAdjuster is designed to read and parse the CompuCell3D XML configuration file, and allow users to modify the basic simulation structure in a GUI. In an CompuCell3D XML file, each tag contains some descriptions of the model objects (e.g. CellType plugin uses the parameter syntax <CellType TypeName="Medium" TypeId="0" / > map verbose *generalized-cell*-type names to numeric cell TypeIds for all *generalized cell* types) or environment configurations (e.g <Boundary_x>, <Boundary_y> and <Boundary_z> tag pairs define the boundary conditions in the system). Whenever users need to modify some parameters in their simulations, they have to locate the target parameters in the XML and then
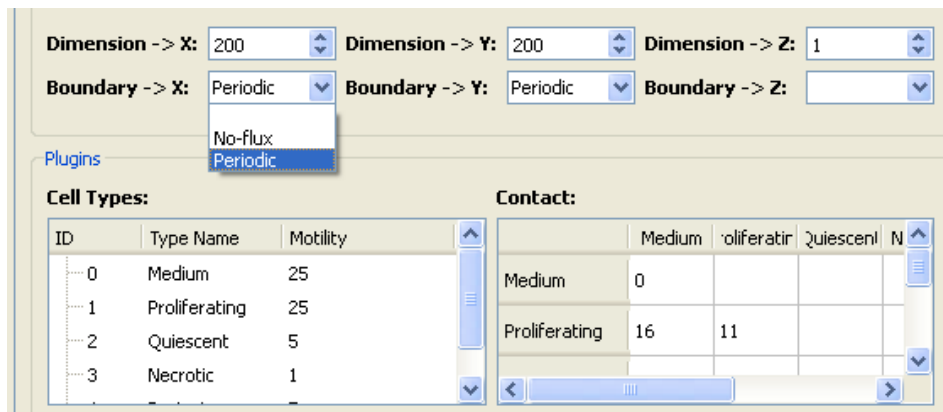
Figure 2: Widgets show CellType and Boundary Condition plugins in ParaAdjuster.

change it textually. This can bring some potential issues into the simulation including misspelling, variable type and value range errors etc.. In ParaAdjuster, all the parameters are presented using GUI widgets (see Figure 2), which are bound with a set of constraints to protect against some potential input errors. Take the TypeId in CellType plugin for instance, all CellType data are contained in a tree widget, in which each node element stands for a cell type showing a cell type ID, Name and Motility. A user can assign a unique TypeId for each cell type in the positive integer format that ranges from 0 - 1000. Therefore a repeated number, negative number, or non-integer data type is not allowed to input in the TypeId widget. The boundary condition in each direction is contained in a drop-down list widget, where users can choose either *no-flux* or *periodic*.

Users can also create a new CompuCell3D definition file based on a default template offered by ParaAdjuster, and further modify the objects, fields and related parameters to make a specific model. Starting a CompuCell3D simulation of the current loaded model in ParaAdjuster is as simple as clicking on the 'CC3D' button. We present a case study in next section, in which we use ParaAdjuster combined with CompuCell3D to simulate nutrient limited avascular tumor growth.

# 4 Case study: Avascular Tumor Growth

Our primary goal in using ParaAdjuster here is to perform parameter control for CompuCell3D models. We present a simplified version of a tumor growth model based on our previous work [5], [4]. This model contains three cancer cell types: proliferating, quiescent, and necrotic. Each cell on the grid behaves according to its genotype and the microenvironmental factors, which is oxygen in this case. Proliferating cells consume oxygen to support their activities, e.g. migration and mitosis, and become quiescent when experiencing hypoxia (oxygen deficiency region, wherein the oxygen concentration is below some threshold). A quiescent cell consumes less oxygen (1/5 of the base consumption rate), has low motility, and does not perform mitosis due to the cell-cycle arrest in $G_0$ phase [2]. Both proliferative and quiescent cells will turn into necrosis when the oxygen concentration further drops below another threshold value.

Each simulation is started with a single proliferating cell at the center of the lattice, and the initial oxygen concentration is homogeneous across the tissue. The formation of a layered-structure tumor can be observed with a central necrotic region that is surrounded by a layer of quiescent cells and an outer rim of proliferating cells. The oxygen field is described as a partial differential equation (PDE) in CompuCell3D XML file, and the related parameters including diffusion constant, consumption rate and decay rate per cell type, are parsed in ParaAdjuster and displayed in a tree widget, in which each tree element has a constraint relating to its data contained. Take oxygen constant for instance, its value is confined between 0 - 10000 in double data type; integer input will be converted into the double type; and character input is not allowed.

(a) $O_2$ consumption rate $= r$



(b) $O_2$ consumption rate $= 2r$



(c) $O_2$ consumption rate $= 3r$

| T = 200 | T = 400 | T = 600 | T = 800 |
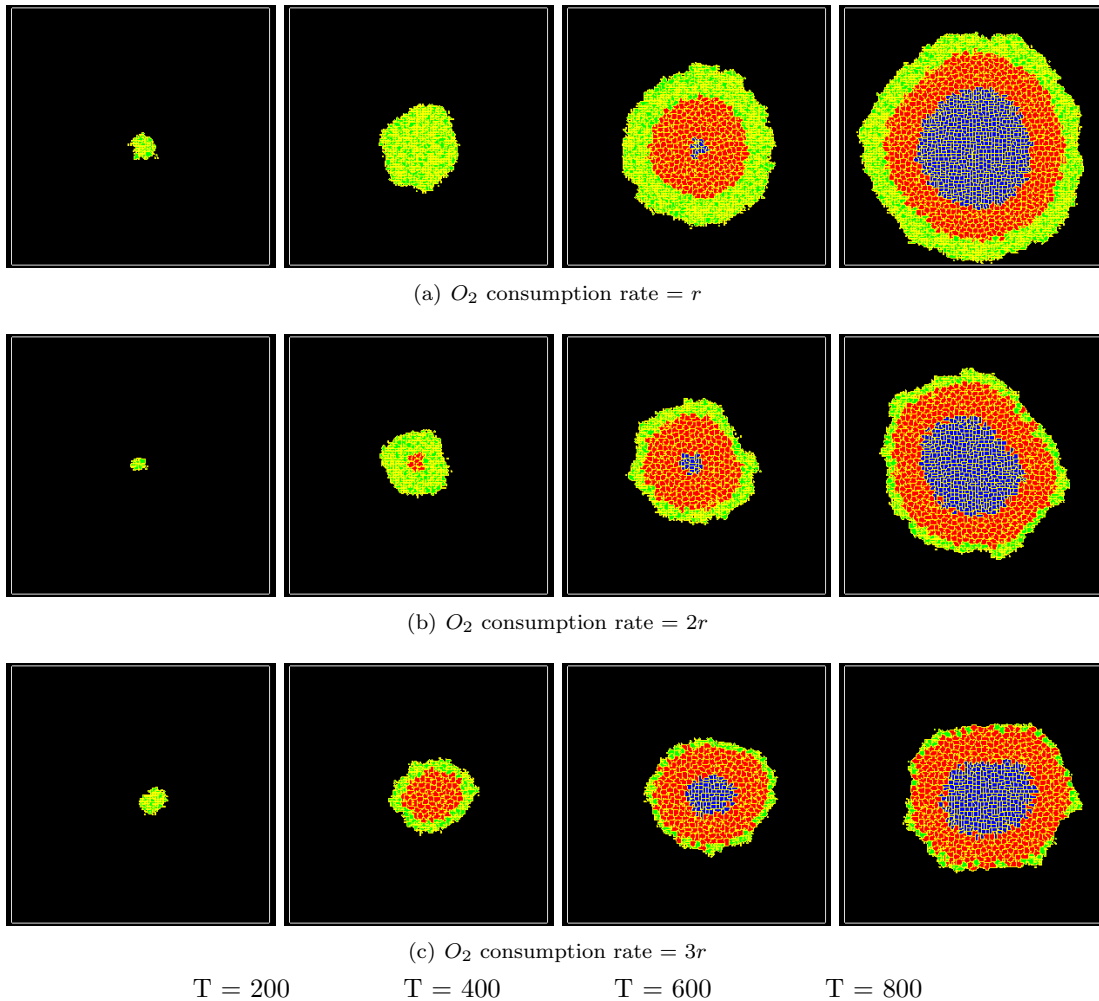
Figure 3: Exported images at regular intervals for development of a solid tumor. Proliferating cells are colored green, quiescent red, and necrotic blue. With a normal oxygen consumption rate, the tumor grows rapidly with larger final size, while with higher oxygen consumption rate the tumor forms a quiescent layer and a necrotic core earlier and appears a decrease in the final size.

Simulation results are captured at regular time intervals, 200 *Monte Carlo Steps*. Figure 3(a) shows an avascular tumor growing under a normal oxygen consumption rate. We then modified the oxygen consumption rate using ParaAdjuster, and the consequences of simulations are shown in Figure 3(b), and (c). Tumor growth is regulated through modulation of the consumption of oxygen, which appears to be a rate limiting substrate for tumor proliferation. From Figure 3 we can see that the increased oxygen consumption rate slows down tumor growth. This outcome is in agreement with the *in vivo* observations reported by Chen *et al.* [6]. Another important observation is that higher oxygen consumption rate can induce an earlier emergence of the quiescent and necrotic region. This decides the cell type distribution in a tumor, which will affect the tumor morphology in avascular phase and the following angiogenesis stage (data not shown).

It is important to note that the efficiency of metabolism has not been considered in this model, because there is no oxygen utilization contributing to individual cell's growth. Hence the future improvement of this model is to consider single cell growth and division speed by utilizing the constantly changing oxygen.

# 5 Conclusions and Future Work

ParaAdjuster is an easy-to-use tool, which offers graphically control parameters for CompuCell3D models with a compact view. Users can modify model parameters by using GUI widgets rather than textually edit the XML file. This approach can help users avoid some potential mistakes, for example typing mistakes and data type errors. Clearly, it is more friendly to users who are non-professional programmers and/or not familiar with the markup language like XML. In addition, users can launch a CompuCell3D simulation from ParaAdjuster, which can avoid some operation steps and directly see the consequent results after the parameter modification.

ParaAdjuster has not yet supported Python file, which is an important format for system dynamics definition in a CompuCell3D model. The first step in our future work is to integrate Python format as another type of entity in ParaAdjuster. Furthermore, we would like to enhance ParaAdjuster by displaying the relationships of cell-cell and cell-microenvironment by using widgets, such as maps.

# References

[1] M. H. Swat, S. D. Hester, R. W. Heiland, B. L. Zaitlen and J. A. Glazier, Multi-Cell Simulations of Development and Disease Using the CompuCell3D Simulation Environment. *Methods Mol Biol.*, 500, 361-428, 2009.

[2] C. J. Sherr, Cancer Cell Cycles. *Science*, 274, 1672-1677, 1996.

[3] T. E. Oliphant, Python for Scientific Computing. *Computing in Science and Engineering*, 9, 10-20, 2007.

[4] X. Gao, C. K. Baban, M. Tangney and S. Tabirca, Computer simulation of hypoxia regulates avascular tumor growth through p27 expression. *Complex Medical Engineering (CME), 2011 IEEE/ICME International Conference*, 15-22, 2011.

[5] X. Gao, S. Tabirca and M. Tangney, A Multiscale Model for Hypoxia-induced Avascular Tumor Growth. *2011 International Conference on Bioscience, Biochemistry and Bioinformatics*, 53-58, 2011.

[6] Y. Chen, R. Cairns, I. Papandreou, A. Koong, N. Denko and C. Nicholas, Oxygen Consumption Can Regulate the Growth of Tumors, a New Perspective on the Warburg Effect. *PLoS ONE*, e7033, 2009.

[7] S. P. Callahan, J. Freire, E. Santos, C. E. Scheidegger, C. T. Silva and H. T. Vo, VisTrails: visualization meets data management. *Proceedings of the 2006 ACM SIGMOD*, 745-747, 2006.

Xuefeng Gao
University College Cork
Dept. of Computer Science
WGB, UCC, Western Rd., Cork
IRELAND
E-mail: *xfg1@cs.ucc.ie*

Sabin Tabirca
University College Cork
Dept. of Computer Science
WGB, UCC, Western Rd., Cork
IRELAND
E-mail: *s.tabirca@cs.ucc.ie*

# Argumentation-Based Ontology Maintanance

### Adrian Groza, Raluca Mechno

#### Abstract

The study proposes a method for ontology update based on the large amount of semi-structured data available from the semantic wikis. An argumentation reasoning process evaluates the contradictory pieces of knowledge posted by different users in order to adjust the axioms of an ontology or to decide each individual to what concept it belongs.

## 1  Introduction

The potential of combining Web 2.0 with Web 3.0 is advocated in literature [1]. At the moment, we are at the beginning of developing the social computing science [7]. In this line, the current study applies the active social machine behind semantic wikis to the hard task of ontology maintenance.

Ontologies are continuously confronted to the evolution problem. Our goal here is to develop a method for ontology maintenance by exploiting the work done in argumentation theory. An argumentation debate aims at increasing or decreasing the acceptability of a controversial standpoint for the listener or reader, by conveying a set of arguments which support or attack the issue in hand.

The paper presents an argumentation framework, which provides straightforward tool to reason about arguments posted by usual users. The proposal of this framework keeps the abstraction from the logic used to represent knowledge inside arguments while specifying a logic scheme to give some structure to arguments [12].

## 2  Technical Instrumentation

### 2.1  Description Logic

Description logics[1] (DL) are a family of formal knowledge representation languages. They are more expressive than propositional logic but have more efficient decision problems than first-order predicate logic. DLs are used in Artificial Intelligence for formal reasoning on the concepts of an application domain, having an important role for defining, integrating, and maintaining ontologies.

In the description logic $\mathcal{ALC}$, concepts are built using the set of constructors formed by negation, conjunction, disjunction, value restriction, and existential restriction, as table 1 bears out. Here, $C$ and $D$ represent concept descriptions, whilst $r$ a role name. The semantics is defined based on an interpretation $I = (\Delta^I, \cdot^I)$, where the domain $\Delta^I$ of $I$ contains a non-empty set of individuals, whilst the interpretation function $\cdot^I$ maps each concept name $C$ to a set of individuals $C^I \in \Delta^I$ and each role $r$ to a binary relation $r^I \in \Delta^I \times \Delta^I$. The last column of table 1 illustrates the extension of $\cdot^I$ to arbitrary concepts. The syntax of a member of the description logic family is characterized by its recursive definition, in

---

[1]http://en.wikipedia.org/wiki/Description_logic

| Constructor | Syntax | Semantics |
|---|---|---|
| negation | $\neg C$ | $\Delta^I \setminus C^I$ |
| conjunction | $C \sqcap D$ | $C^I \cap D^I$ |
| disjunction | $C \sqcup D$ | $C^I \cup D^I$ |
| existential restriction | $\exists\, r.C$ | $\{x \in \Delta^I \mid \exists\, y : (x,y) \in r^I \wedge y \in C^I\}$ |
| value restriction | $\forall\, r.C$ | $\{x \in \Delta^I \mid \forall\, y : (x,y) \in r^I \rightarrow y \in C^I\}$ |

Table 1: Syntax and Semantics of $ALC$ concepts.

which the constructors that can be used to form concept terms are stated. Some constructors are related to logical constructors in first-order logic (FOL) such as intersection or conjunction of concepts, union or disjunction of concepts, negation or complement of concepts, universal restriction and existential restriction. Other constructors have no corresponding construction in FOL including restrictions on roles for example, inverse, transitivity and functionality.

**Definition 1** *A concept $C$ is satisfiable if there exists an interpretation $i$ such that $C^I \neq \varnothing$. The concept $D$ subsumes the concept $C$ ($C \sqsubseteq D$) if $C^I \subseteq D^I$ for all interpretations $I$.*

**Definition 2** *An ABox is a finite set of concept assertions $C(a)$ or role assertions $r(a,b)$, where $C$ represents a concept, $r$ a role, and $a$ and $b$ are two instances. Usually, the unique names assumption holds within the same ABox. A TBox is a finite set of terminological axioms of the form $C \equiv D$ or $C \sqsubseteq D$.*

## 2.2   Semantic Wikis

*Semantic wikis* provide users the capability to annotate their text with specific concepts and roles from a set of imported ontologies, in order to be processed against semantic queries. Among the available semantic wikis, such as DBpedia [4], ACEWiki [10], or OntoWiki [2], we drive our attention towards Semantic Media Wiki (SMW), due to its success in terms of number of users.

Semantic MediaWiki[2] (SMW) is an extension of MediaWiki, the wiki application best known for powering Wikipedia, that helps to search, organise, tag, browse, evaluate, and share the wiki's content. While traditional wikis contain only text which computers can neither understand nor evaluate, SMW adds semantic annotations that allow a wiki to function as a collaborative database.

To create the template, used for the argumentation scheme, it was used the "special" page, *Special : CreateTemplate*, defined by *SpecialForms*[3], an extension of SMW. With this extension it was created also the properties and categories . Another extensions: *Halo* could be used to import the ontology, *ExportRDF* to export the data in RDF format.

The information collected from template, based on Semantic MediaWiki, relates to the ontology language [17]. In MediaWiki the method used for entering information into a wiki is *wikitext*, which is transformed into HTML pages. For the interrelation between pages, hyperlinks are used. All the defined pages are classified in *namespaces*, which cannot be defined by wiki users. Every page can have one or more categories.

Like in MediaWiki, semantic data in Semantic MediaWiki is structured by pages and every page corresponds to an ontology entity. Different ontology entities can be represented by *namespaces*: *individuals* are represented by the majority of the pages, *classes* by *categories* in MediaWiki, which classify individuals an also create subcategories, *properties* are relationships between two individuals or an individual and a data value and *types* for distinguish different kinds of properties. Categories are available in MediaWiki, but properties and types are introduces by SemanticMediaWiki. The elements of wikitext are easy to interpret: triple quotes "'...'" are used for bold text, the text within brackets square [[...]] is transformed into link. For example [[England]] is just a link which do not carry any machine-understandable semantics yet. For instance, asserting a property called *capital of* to London can be done by writing [[*capital of* :: *England*]], where property *capital of* has value *England*.

---

[2] http://semantic-mediawiki.org
[3] http://www.mediawiki.org/wiki/Extension:Semantic_Forms

| Owl | Semantic MediaWiki |
|---|---|
| Owl individual | normal article page |
| owl:Class | article in namespace Category |
| owl:ObjectProperty | article in namespace Relation |
| owl:DatatypeProperty | article in namespace Attribute |
| **Statement about element page** | **Syntax in wiki-source of page** |
| objectProperty | [[propertyName::objectArticle]] |
| attributeProperty | [[propertyName:=valueString]] |
| rdf:type className | [[Category:className]] (on article pages) |
| rdfs:subClassOf className | [[Category:className]] (on category pages) |

Figure 1: Description and relationships between individuals
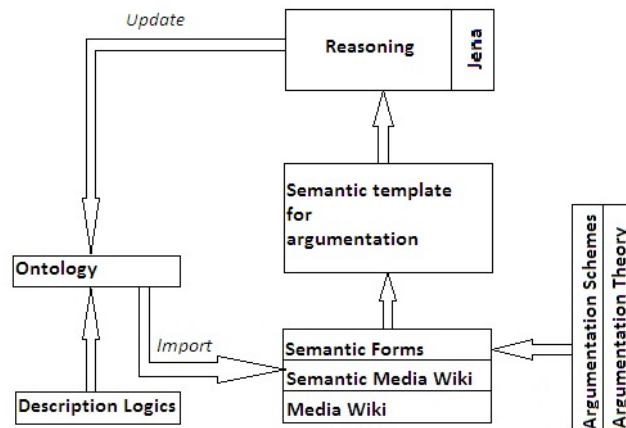


Figure 2: System Architecture.

# 3  System Architecture

## 3.1  Engineering the *Restaurant* ontology

To model the interaction between arguments, an ontology made in Protege[4] is used. In the development process of "Restaurant" ontology several engineering steps were followed. The first step is to determine the scope, by creating some competency questions like: *Which is the best restaurant from Cluj? What features should be taken into account, when you choose an restaurant? What kind of kitchen does one prefer? Is it close to work? Exists room for smoking? How long does it take to be served?.*

In the second step several knowledge repositories have been considered for re-use: the *restaurant.owl*[5], *time.owl*[6] and *food.owl*[7] ontologies. In the third step relevant terms are identified: *Cuisine, Asian Cuisine, African Cuisine, location, kitchen, Tusa, Shangai, has time of service, has location.* The nouns form are the basis for the class names and the verb phrases form the basis for property names. The classes are created in the fourth step: *SmokingRoom, Restaurant, Location, Cuisine, African.* In the fifth five, a property named *hasServiceTime* was created and some object properties: *hasCuisine, hasAddress, isCuisine, hasSmokingRoom.* Finally, the instances for the classes were added and the ontology was checked for anomalies.

---

[4]http://protege.stanford.edu/
[5]http://gaia.fdi.ucm.es/ontologies/restaurant.owl
[6]http://www.w3.org/TR/owl-time/
[7]http://krono.act.uji.es/Links/ontologies/food.owl/view

1    $tusa : Restaurant$

2    $hasName(tusa,"TusaSRL")$

3    $Restaurant \sqsubseteq SmokingRoom \sqcup Location \sqcup Cuisine$

4    $Restaurant \sqsubseteq \exists\, hasName.Name \sqcap \exists\, hasAddress.Location$
        $\sqcap \forall\, hasCuisine.Cuisine \sqcap \forall\, hasServiceTime.ServiceTime$

5    $Name \sqsubseteq Restaurant$

6    $Cuisine \sqsubseteq Asian \sqcup African \sqcup European \sqcup CuisineStyle$

7    $hasKitchen^{-} \equiv (\neg isKitchen)$

Figure 3: Part of the restaurant ontology.

According to axiom 1, in figure 3, *tusa* is an instance of class *Restaurant*. The instance *tusa* has the name "*TusaSRL*" based on axiom 2. In axiom 3 the class *Restaurant* is included in the result of intersection of classes *SmokingRoom*, *Location* and *Cuisine*. Axiom 4 describes that the class *Restaurant* must have instances of classes *Name*, *Address*, *Cuisine* and *ServiceTime*. Class *RestaurantName* is a subclass of *Restaurant*(axiom 5). Axiom 6 define class *Cuisine*, which is included in the result of reunion of classes *Asian, African, European, Oceania, American, EthnicAndReligiousCuisine* and *CuisineStyle*. The inverse role of *isKitchen* shown by axiom 7 is *hasKitchen*.

## 3.2   Argument Representation

For the abstraction of the debate, we use the theoretical model of Walton based on argumentation schemes [18]. Argument schemes encapsulate common patterns of human reasoning such as: *argument from position to know, argument from evidence, argument from sign, etc.* Argumentation schemes are defined by the following items: a name, a set of premises ($A_i$), a conclusion ($C$). Figure 5 details these attributes of *Argument from position to know* scheme.

The template that was created for the argumentation scheme is described in figure 6. The user completes the following fields: *Premise* represents the premise of the argument, *ValueOfCredibilityOfPremise* represents the certainty of the premise, *Conclusion* is the text that represents the conclusion, made based on premise, *ValueOfCredibilityOfConclusion* represents the value that will be computed with equation 1, *ValueOfCredibilityOfUser* represents value of credibility of user, calculated with equation 3, *User* is the name of the user that complete the fields of the template.

$$credibility(c) = min(v_i) * \alpha_f * \mu_u \tag{1}$$

where $v_i$ represents the values of the premises, $i$ could have values between *1 and 5* , user can choose how many premises will use, the template illustrated in figure 7 have just one premise, if the premise is an existing argument, the value taken will be the value of credibility of that argument, otherwise, the value taken will be 1, $\alpha_f$ represent the value of the strength with which it is sustained the premise, this value is computed with equation 2 and $\mu_u$ is the value of credibility of the user, calculated based on 3 .

$$\alpha_f = \sigma_v * (\rho_t - \rho_f)/\rho_t \tag{2}$$

To compute the value of the force with which the premise is sustained it was created an tree, illustrated in figure 7 , which contains in nodes the word used in premise. Each node is initiated with a value, $\sigma_v$ represents this value, $\rho_t$ represents all the premises in which the word was used, $\rho_f$ represents the premises in which the word was used, but the argument was not valid.

$$\mu_u = (\mu_t - \mu_f)/\mu_t \tag{3}$$

where $\mu_t$ represents all the arguments posted by the user $u$ and $\mu_f$ all the arguments posted by the user $u$, but was not valid.
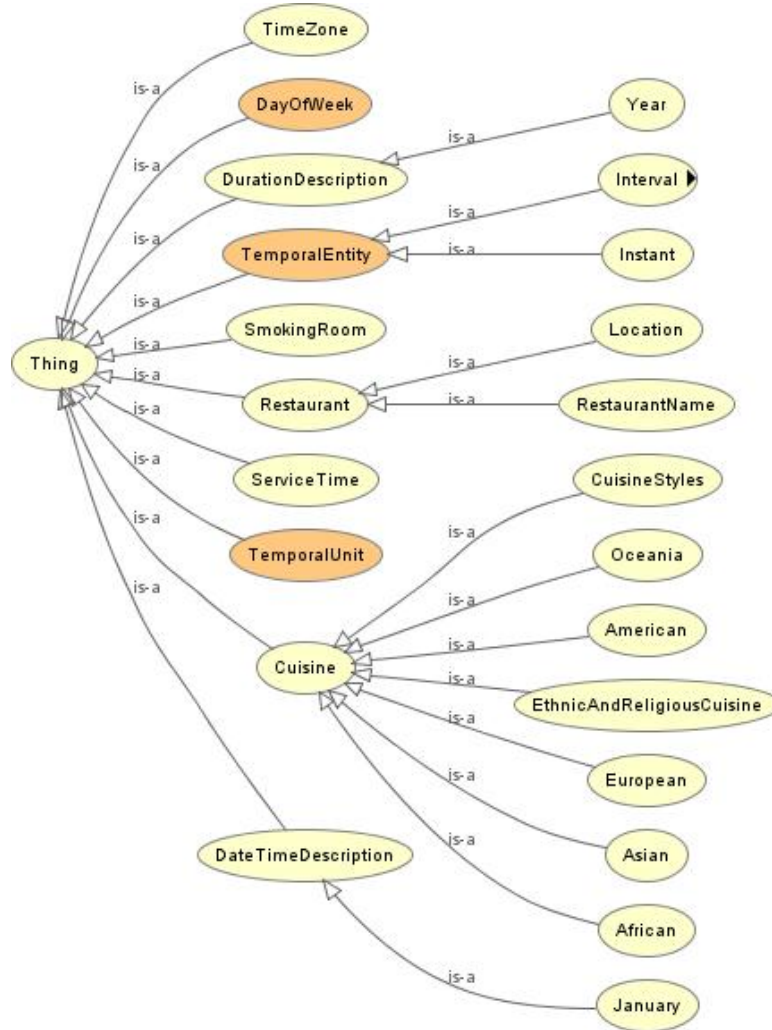
Figure 4: Graphical representation of the restaurant ontology.



Figure 5: Using argumentation scheme for structured arguments.

```
{{Argumentation
|Premise=
|ValueOfCredibilityOfPremise=
|Conclusion=
|ValueOfCredibilityOfConclusion=
|User=
|ValueOfCredibilityOfUser=
}}
```

Figure 6: Semantic template for the argumentation scheme.
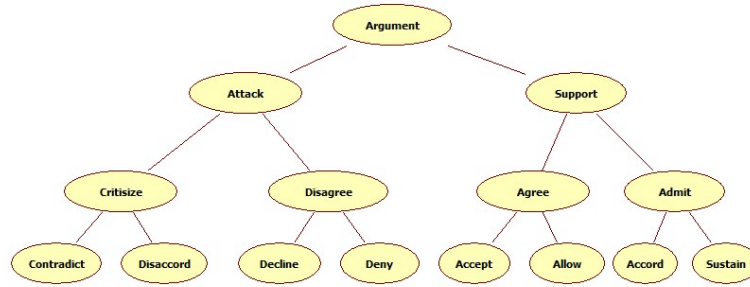
Figure 7: Word tree.

## 4 Running Scenario

This section presents a usage scenario and the application results.

### 4.1 Knowledge elicitation

The user can choose the number of premises, for the argumentation scheme that wants to sustain. We suppose that some users posted the following arguments, about the ontology *restaurant.owl*:

```
{{Argumentation
|Premise= Monday, 4PM, sustain quick service at tusa
|ValueOfCredibilityOfPremise= 0,28
|Conclusion=quick service at tusa
|ValueOfCredibilityOfConclusion= 0,084
|User= u1
|ValueOfCredibilityOfUser= 0,3
}}
```

The user $u_1$ with value of credibility $0,3$, obtained based on equation 3, where $\mu_t$=10 and $\mu_f$=6, uses the word *sustain*, which has the value $0,7$, that *Monday* at restaurant *Tusa*, on $4PM$ it is always *quickservice*. The force with which the premise is sustained is computed with equation 2 where, value of the word from tree is $0,7$, $\rho_t = 10$ and $\rho_f = 6$, then the final value is $0,28$, so the value of credibility of conclusion computed with equation 1 will be $0,084$.

```
{{Argumentation
|Premise= Monday, 4PM, attack quick service
|ValueOfCredibilityOfPremise= 1
|Conclusion=not quick service
|ValueOfCredibilityOfConclusion= 0,020
|User= u2
|ValueOfCredibilityOfUser= 0,17 }}
```

The user $u_2$ with value of credibility $0,17$, obtained based on equation 3, where $\mu_t$=17 and $\mu_f$=7, uses the word *attack*, which has the value $0,15$, that *Monday* at restaurant *tusa*, on $4PM$ it is always *quickservice*. The force with which the premise is sustained is computed with equation 2 where, value of the word from tree is $0,15$, $\rho_t = 6$ and $\rho_f = 1$, then the final value is $0,83$, so the value of credibility of conclusion computed with equation 1 will be $0,020$.

```
{{Argumentation
|Premise= quick service at tusa , agree, high quality
|ValueOfCredibilityOfPremise= 0,084
|Conclusion=high quality at tusa
|ValueOfCredibilityOfConclusion= 0,102
|User= u3
|ValueOfCredibilityOfUser= 0,42 }}
```

1.  $hasServiceTime(tusa,\ Medium)$
2.  $hasServiceTime(tusa,\ Medium) \sqsubseteq \forall\, hasServiceTime(Monday \sqcup Tuesday$
    $\sqcup\, Wednesday \sqcup Thursday \sqcup Friday \sqcup Saturday \sqcup Sunday)$
3.  $hasServiceTime.Monday \sqsubseteq (\forall\, hasServiceTime(MI \sqcap MII \sqcap MIII))$
4.  $Monday \sqsubseteq (\exists\, hasServiceTime.Quick)$
5.  $MI \sqsubseteq (\exists\, hasServiceTime.Quick)$
6.  $MII \sqsubseteq (\exists\, hasServiceTime.Medium)$
7.  $MIII \sqsubseteq (\exists\, hasServiceTime.Quick)$

Figure 8: Axioms for the initial ontology

The user $u_3$ with value of credibility $0,42$, obtained based on equation 3, where $\mu_t$=7 and $\mu_f$=4, uses the word *agree*, which has the value $0,5$, that at restaurant *tusa*, it is *highquality* based on the valid argument, *quickservice* at *tusa*. The force with which the premise is sustained is computed with equation 2 where, value of the word from tree is $0,5$, $\rho_t = 15$ and $\rho_f = 6$, then the final value is $0,30$, so the value of credibility of conclusion computed with equation 1 will be $0,102$.

```
{{Argumentation
|Premise= Sunday, 6PM, decline quick service at tusa
|ValueOfCredibilityOfPremise= 0,57
|Conclusion=not quick service at tusa
|ValueOfCredibilityOfConclusion= 0,15
|User= u4
|ValueOfCredibilityOfUser= 0,41 }}
```

The user $u_4$ with value of credibility $0,41$, obtained based on equation 3, where $\mu_t$=17 and $\mu_f$=10, uses the word *decline*, which has the value $0,65$, that *Sunday* at restaurant *tusa*, on $6PM$ it is *quickservice*. The force with which the premise is sustained is computed with equation 2 where, value of the word from tree is $0,65$, $\rho_t = 7$ and $\rho_f = 3$, then the final value is $0,57$, so the value of credibility of conclusion computed with equation 1 will be $0,15$.

## 4.2   Ontology Enrichment

This subsection illustrates how the arguments posted affects the ontology or not. Consider part of the initial ontology illustrated in figure 8.

By the axiom 1, figure 8, the service time for restaurant *tusa* is *Medium*. The role *hasServiceTime* of restaurant *Tusa* is composed of union of values of *hasServiceTime* for all days from the week. Based on axiom 3, the value of *hasServiceTime* for day *Monday* results from the intersection of values of *hasServiceTime* for all 3 parts of the day, where *MI* is between $8AM$ and $12AM$, *MII* between $12AM$ and $3PM$ and *MIII* between $3PM$ and $9PM$.

According to axiom 4, the value of *hasServiceTime* for *Monday* is *Quick*. This value results from the comparison of the values obtained in those 3 parts of the day. The value of the role *hasServiceTime* for part *MI* is initial *Quick* (axiom 5).The value of the role *hasServiceTime* for part *MII* is initial *Medium* (axiom 6). The value of the role *hasServiceTime* for part *MIII* is initial *Medium* (axiom 7).

The arguments are taken into account if the value of conclusion of the argument is bigger then the actual value of the proper part. The arguments which are not taken into account will be saved and will be proposed for improvement. If other users will sustain the arguments proposed for improvements, the value of the sustained argument will be updated, by adding the current value. To support the decision of change, ontology engineers will use metrics for ontology evaluation and validation.

The value of credibility of conclusion for the argument proposed by the user $u1$ is taken into account by the ontology engineers. This argument will change the value of the *hasServiceTime* for *MII*, because the initial value for *MII* is *Medium* and it is different from the conclusion of the argument. The axiom from the new ontology become:

$$MII \sqsubseteq (\exists\, hasServiceTime.Quick)$$

The value of the role *hasServiceTime* for part *MII* will be *Quick*. In this case, this result will not affect the value of *hasServiceTime* of day *Monday*, for restaurant *Tusa* because it is already *Quick*.

# 5   Discussions and Related Work

Argumentation is a vital aspect of intelligent behaviour by humans. Consider diverse professionals such as politicians, journalists, clinicians, scientists, and administrators, who all need to collate and analyse information looking for pros and cons for consequences of importance when attempting to understand problems and make decisions [8].

There are a lot of approaches for the argumentation like: real arguments [16], fuzzy argumentation [9], arguments in OWL [14]. In case of real arguments, they do not have enough explicitly presented premises for the entailment of the claim and the proponent of an argument encode an argument into a real argument by ignoring the common knowledge, and it allows a recipient of a real argument to decode it, also by ignoring the common knowledge. In our view the user is forced to complete all the fields in the created template, so there will not be empty fields about the common knowledge before he create arguments.

The fuzzy approach enriches the expressive power of the classical argumentation model by allowing to represent the relative strength of the attack relationships between arguments, as well as the degree to which arguments are accepted [9]. In our case, a method for computing the relative strength of the attack relationships between arguments and also the support relationships between them is proposed. The computation method is based on a word tree, in which each node has a degree of support ($degre\epsilon[0, 1]$) attribute, assessed based on the degree of support of the sons and a *historic* attribute. The *historic* attribute represents the normalization of the arguments in which the current node was used.

The unambiguous and effective delivery of data and knowledge on the Web relies heavily on the correct representation and understanding of the associated contexts. However, the current way of encoding contexts of data and knowledge on the Web is largely ad hoc. Contexts are often embedded in the application programs or are implied by the application or community-specific agreements. This makes the linking and reusing of data and knowledge, and thus the integration of Web applications, a difficult problem. Therefore, building the architectural support for contexts is one of the major challenges for the Web, and in particular, for the Semantic Web [3].

Similar to our paper, ArgDF [11] is a framework which addresses the issue of argument representation in Semantic Web, aiming at developing the Argumentative Web, a large scale network of interconnected arguments created by human agents in a structured manner [15], based on the argument interchange format (AIF) ontology. The main aims of the AIF ontology are: i) to facilitate the development of (closed or open) multi-agent systems capable of argumentation based reasoning and interaction using a shared formalism; and ii) to facilitate data interchange among tools for argument manipulation and argument.

Design patterns are used in [13] to manage ontologies in an easier manner. ODPs are building blocks for ontology management representing small ontologies that can be extended and adapted to a specific application. An initial ontology is enriched based on ODPs in [5, 6]. The process is semi-automatic and has been implemented in two phases: i) element extraction - uses an initial ontology in order to extract elements together with a confidence ii) patterns matching and ranking - evaluates against ODPs the ontology elements previously extracted based on words metrics or using WordNet. The ontology is evaluated and enriched with the best new elements. In our case, the up to date knowledge is extracted from semantic wikis, where the decision to add an element or the change a axiom is taken after an argumentation process

Several approaches for extracting concepts, instances and relationships exploit separately or integrate statistical methods, semantic repositories such as WordNet, natural language processing libraries such as OpenNLP, or lexicon-syntactic patterns in form of regular expressions [19]. Instead of NLP techniques we exploit here argumentation schemes for extracting structured information. When creating an application based on semantic knowledge it is necessary to guarantee that the considered ontology meets the application requirements. In this line, ontology evaluation is important in cases where the ontology is automatically populated from different resources that might not be homogeneous, leading to duplicate instances, or instances that are clustered according to their sources in the same ontology [20].

# 6 Conclusion

This research addresses the task of ontology maintenance by exploiting the large amount of structured information available in semantic wikis. The proposed solution makes use of argumentation theory, using as an argumentation platform Semantic MediaWiki, aiming at enacting the argumentative web as envisaged in [14].

Ongoing work regards empirical evaluation of the method by using several evaluation metrics. Word-Net ontology can be exploited even more, and with the help of OpenNLP, relationships between concepts from the ontology or new domain concepts could be discovered even when the context of use causes word ambiguity.

# References

[1] Anupriya Ankolekar, Markus Krtzsch, Thanh Tran, and Denny Vrandecic. *The two cultures: Mashing up web 2.0 and the semantic web. Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1):70 – 75, 2008. Semantic Web and Web 2.0.

[2] Sören Auer, Sebastian Dietzold, and Thomas Riechert. *Ontowiki - a tool for social, semantic collaboration.* In *International Semantic Web Conference*, pages 736–749, 2006.

[3] Jie Bao, Jiao Tao, and Deborah L. McGuinness. *Context representation for the semantic web.* In *Proceedings of the WebSci10: Extending the Frontiers of Society On-Line, April 26-27th, 2010, Raleigh, NC: US.*, 2010.

[4] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. *Dbpedia - a crystallization point for the web of data. J. Web Sem.*, 7(3):154–165, 2009.

[5] Eva Blomqvist. *Ontocase-automatic ontology enrichment based on ontology design patterns.* In *Proceedings of the 8th International Semantic Web Conference*, ISWC '09, pages 65–80, Berlin, Heidelberg, 2009. Springer-Verlag.

[6] Marius Georgiu and Adrian Groza. *Ontology enrichment using semantic wikis and design patterns. Studia Universitatis Babes-Bolyai Informatica*, LVI(2):31–36, 2011.

[7] Jim Hendler and Tim Berners-Lee. *From the semantic web to social machines: A research challenge for ai on the world wide web. Artif. Intel.*, 174(2):156 – 161, 2010.

[8] Anthony Hunter. *Real arguments are approximate arguments.* In *AAAI*, pages 66–71, 2007.

[9] Jeroen Janssen, Martine De Cock, and Dirk Vermier. *Fuzzy argumentation frameworks.* In *IPMU*, pages 513–520, 2008.

[10] Tobias Kuhn. *Acewiki: Collaborative ontology management in controlled natural language.* In *SemWiki*, 2008.

[11] Sanjay Modgil and Jarred McGinnis. *Towards characterising argumentation based dialogue in the argument interchange format.* In *ArgMAS*, pages 80–93, 2007.

[12] Martín O. Moguillansky, Nicolás D. Rotstein, and Marcelo A. Falappa. *Generalized abstract argumentation: A first-order machinery towards ontology debugging. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, 14(46):17–33, 2010.

[13] Valentina Presutti and Aldo Gangemi. *Content ontology design patterns as practical building blocks for web ontologies.* In *ER*, pages 128–141, 2008.

[14] Iyad Rahwan and Bita Banihashemi. *Arguments in OWL: A progress report.* In *COMMA*, pages 297–310, 2008.

[15] Iyad Rahwan, Fouad Zablith, and Chris Reed. *Laying the foundations for a world wide argument web. Artificial Intelligence*, 171(10-15):897–921, 2007.

[16] Christoph Tempich, Elena Simperl, Markus Luczak, Rudi Studer, and H. Sofia Pinto. *Argumentation-based ontology engineering. IEEE Intelligent Systems*, 22:52–59, 2007.

[17] Denny Vrandecic and Markus Krötzsch. *Reusing ontological background knowledge in semantic wikis.* In Max Völkel and Sebastian Schaffert, editors, *SemWiki*, volume 206 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2006.

[18] Douglas Walton. *Argumentation schemes for presumptive reasoning. Argumentation*, 13(3):338–343, 1999.

[19] Shu Wang and Enhong Chen. *An instance learning approach for automatic semantic annotation.* In *CIS*, pages 962–968, 2004.

[20] Jonathan Yu, James A. Thom, and Audrey M. Tam. *Requirements-oriented methodology for evaluating ontologies. Inf. Syst.*, 34(8):766–791, 2009.

Groza Adrian
Technical University of Cluj-Napoca
Computer Science Department
Memorandumului 28, Cluj-Napoca
ROMANIA
E-mail: *adrian.groza@cs.utcluj.ro*

Raluca Mechno
Technical University of Cluj-Napoca
Computer Science Department
Memorandumului 28, Cluj-Napoca
ROMANIA
E-mail: *raluca.mechno@student.utcluj.ro*

# Decision-making system in E-marketing strategies

### Valentina Lazăr, Cristina Răulea

**Abstract**

The project aims the marketing process from the moment when the customer interact with marketing environment until the construction of new marketing strategies to enable sales and gain the customer loyalty, qualities absolutely necessary for the existence and profitability of business. That for, in developing process it was concluded that the use of applications based on threads is more expensive in terms of query time, effort and updating user data, than if it would use a system based on intelligent agents.

All this led to the development of an applications based on intelligent agents that is easy to use and exclude the detailed reports and cumbersome.

## 1    Introduction

Given that we live in a world in constant motion, a world where time is the main problem, the present application is built precisely as an alternative to this problem in the banking field. It is known that in banks, as large firms, the size of the database is a reasonable size, therefore, browsing and extracting data representing their interests is a problem. Thus, the present application comes with the following improvements: not only that the data are strictly filtered to obtain what it seeks, in this case that speaks about marketing on the Internet (E-Marketing), data are taken very on the market where the company operates, generating a double advantage: market knowledge and improvement of the marketing plan. The first approach to bank marketing evolution is performed by Philip Kotler. It involves five steps, which were identified according to the diversity and effectiveness of marketing techniques implemented.

- The first stage is designed as a bank marketing technique of advertising and sales promotion, image banks (providing small gifts, but useful: umbrellas, flashlights, pens.). The emphasis is on friendly atmosphere within the bank.

- In the second phase of banking marketing is defined as relaxed smile - to stop officers are trained how to have a more open attitude towards the customers. Is rearranged so that the

interior banks to be more pleasant atmosphere and the outside is "humanized"
Benevolence has ceased to be the deciding factor in choosing a bank.

- In the third stage of development, marketing segmentation and upgrading banking means. Banks have discovered a new competitive tool when they began to segment their markets and develop new products for each target market segment.

So what keeps the bank's ability to develop new products continuously determines the position of market leader. Marketing has entered the insurance and investment services. Innovations have resulted in widening globalization of financial markets and banking business through the provision of less traditional services and the use of information technologies and computer networks.
At the fourth stage, marketing means bank positioning. Banks realize that no commercial bank is unable to provide all products and be the best for all customers. It concludes that a bank must "occupy a specific position, taking account of its possibilities.

**Client user interface design**

It begins with the realization of user interface, and more specifically the Web Site.The first point is to be sought for interface design. To motivate customers to view the site presented is necessary, from choosing colors for presentation to the selection information to be presented.

Such colors were chosen that did not look tired (shades of blue, gray and white) and the information was compressed to avoid losing the essence when making customer interest to the attention of those present.

Another way to attract customers is interactivity and data are presented that not much effort on the part of customers to reach the desired information.

One can easily see how the environment and efficient way scroll to the desired benchmark. This presentation is especially necessary if speaking of a world in constant motion in time is most precious.

## 2      User Interface as marketing component

In terms of marketing, creating and developing a web site but should have to support the existence of specific needs, specific, related in most of the way marketing communication takes place to the bank

It is important to note that in general, it does not create products, does not set prices, not ships products to the recipient, not make the bank's reputation doubles and triples sales.

The website can contribute significantly to the manner in which the organization makes the exchange of information or marketing environment, mainly with customers (and potential) and its competitors.

In the first case is about providing a volume of information attractive enough to cause a favorable reaction to clients: marketing buying or ordering products (loans) promoted or at least request additional information about them.

In the second case is about differentiating the bank in relation to its competitors in general and especially in terms of marketing communications.

Interface provides access from the first page of this information by following the user to view what you are looking for.

It is easy to understand why they are made for, from the first page of this information, namely that, especially for banking institutions, the legislation is very thorough and contains clauses which are omitted may result in the existence of disputes between customer and bank, even the loss credibitatii to customers.

# 3    User interface design - marketing specialist

### 3.1.Marketing strategies that results from database

The application itself is built to develop marketing strategies to differentiate between firms in the market, ie between banks.

This starts from the idea of knowledge by understanding the market segment of its characteristics: the size of this segment, placing the physical market, etc..

To know this requires a market study and that study is better than their own resources.

Therefore intended to provide some queries in the database are recorded data from customers. This process is the result of a program in Java programming language that allows data to obtain a result of queries.

Thus was born the MDC, that means Developer Marketing Strategies application.

**If you wish to obtain a marketing report, you must follow several steps:**

**Step 1**: After the opening application type is determined by the person whose address query.

If it is an individual process is as follows:

**a)** click on the individual in open dialogue with fields that query is:

**b)** Once open dialogue, make default in the analysis fields as required.

As it can be seen, the dialog box fields are of two types: text fields and fields of type list.

In text fields, the user must fill in the analysis is done by allowing it to leave the fields blank.

If field-list to choose according to existing data in database does.

**c)** After completing field Query button is pressed the query but does not send data to open a box that allows a return to the current box by pressing NO, or the query itself, by pressing YES.
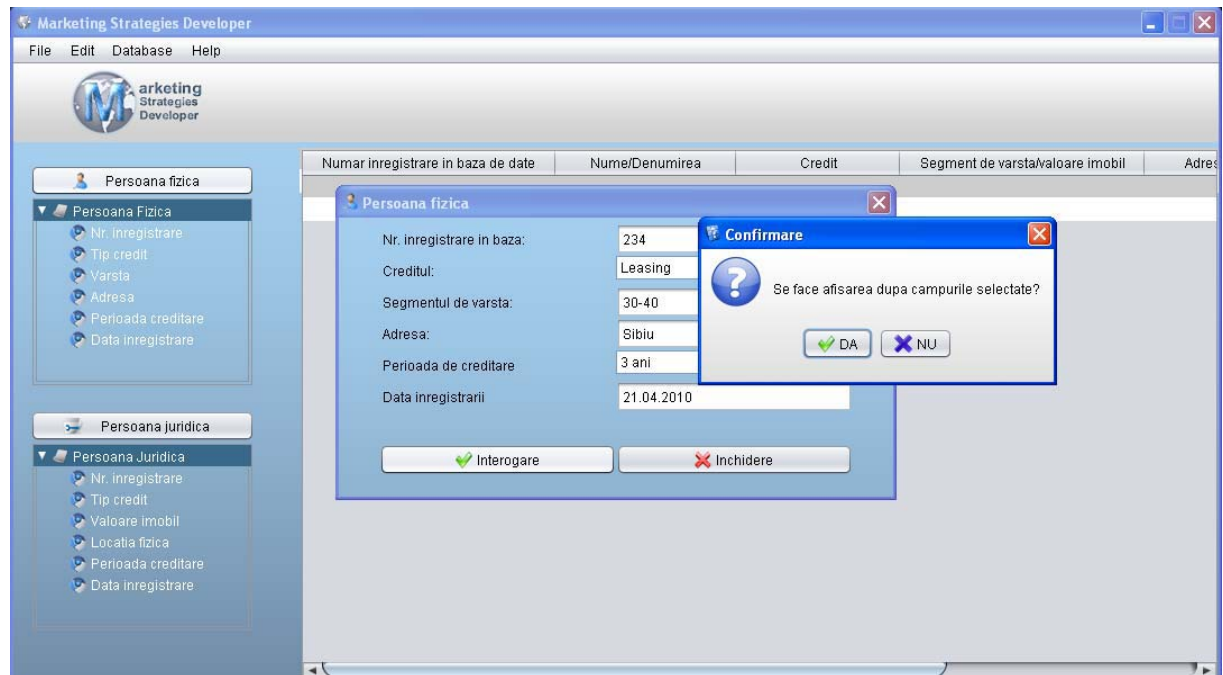


*Figure 3.1. Confirmation of the current query..*

**d)** After the interrogation, is displayed in the table in the main window, the result according to records in the data.

**Step 2:** When the fields were generated in the table, you can choose Save As from the File menu option to save the results into a text document.

These steps can be schematically represented as follows:
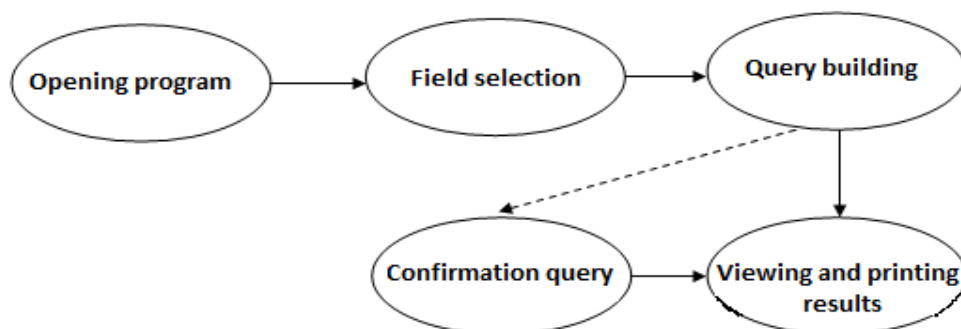
It may be noted that ease of use application and the user is assisted at every step to avoid mistakes from queries.

**3.2.Marketing reports**

The core of any marketing information system is the repository for the data center. They contain solid information on all customers and potential customers. For the sake of brevity, in this paper uses the generic term for a specific site database where information is stored. In addition, to gain maximum advantage by exploiting this information is necessary for each system to provide reliable and documented data on targeted markets and customers.

Applications for marketing uses information from the database of clients, but there are some features of this system functions:

- Making simultaneous queries in multiple tables;

- Queries made by a serial number placed in the database;

- Displaying tabular results of queries;

- Enabling the rescue of the results;

The main advantage of the customers is that provides a comprehensive picture of business relations with a client or group of customers. This overview allows a deeper understanding of consumer behavior and market in general, a point of departure, among others, marketing research, especially in studying the market.

The main characteristic of the system lies in the possibility of handling large volumes of information from various sources, internal and external, in order to develop models of market segments or commercial behavior. Analysis reports can be quickly generated, modeled and saved for future use, and business plans and scenarios improved and refined.

# 4     A new approach - intelligent agents

When architecture development concluded that the use of applications based on threads is more expensive in terms of query time, effort and updating user data, unless they would use a system based intelligent agents.

So, was concluted that is better to develop five intelligent agents:
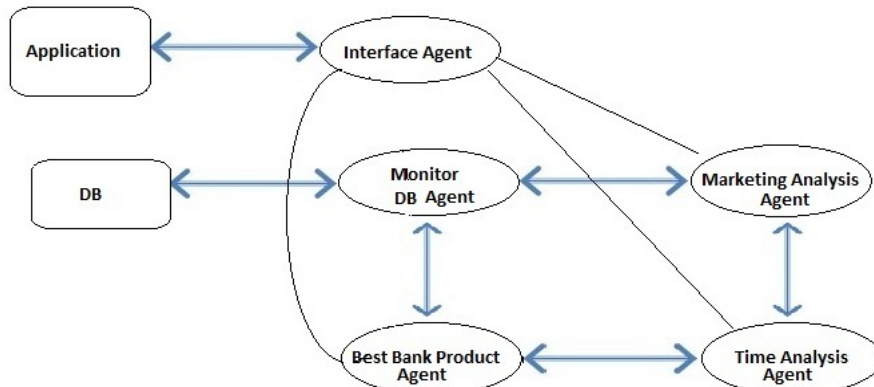
*Figure 3.2. Agencies working with database*

As it can be seen from the figure above, agents are interconditioned and connect database agents and agent application is done using the interface.

- ➢ It collects information from other agents and bring them into the application date:
- ➢ DB Monitor Agent monitors inputs, outputs and changes in the database;
- ➢ Product Best Agent Bank determines which product is most used;
- ➢ Analysis Marketing Agent determines increased or decreased interest as recorded in DB for a period of time;
- ➢ Time Analysis Agent monitors travel time from the last inspection;

The four agents mentioned above are active throughout the existence of the database. They continuously monitors the database and interface with the agent in topical bring results.

It can be seen that the need to achieve ease of use and exclusion of detailed reports and cumbersome.

Regarding the Marketing Agent Analysis should be noted that it establishes increased or decreased interest in receiving the DB Agent Monitor the number of credits in a period.

For exemple, let's asume the following situaton:

| Credit for personal needs | Mortgage | Real estate loan |
|---|---|---|
| **Prediction:** -for 300 demands in 30 days: interest is 15% | **Prediction:** - for 50 demands in 30 days : interest is 18% | **Prediction:** - for 60 demands in 30 days : interest is 15% |
| **Possible situations:** - for less than 300 demands in 30 days : interest is also 15% - for less than 200 demands interest is 14% - for less than 100 demands interest is 13% | **Possible situations:** - for less than 50 demands : interest is also 18% - for less than 35 demands : interest is 16% - for less than 15 demands : interest is 15% | **Possible situations:** - for less than 60 demands : interest is 16% - for less than 40 demands : interest is 15% - for less than 20 demands : interest is 14% |

The agent will known witch is the best decision that the bank have profit and to hierest the number of clients.

Those fluctuations can be realised without an human agent intervention.

## 4.1    Application design

The developement of the application is based on the ideea of frendly using of an interface like in the next image:
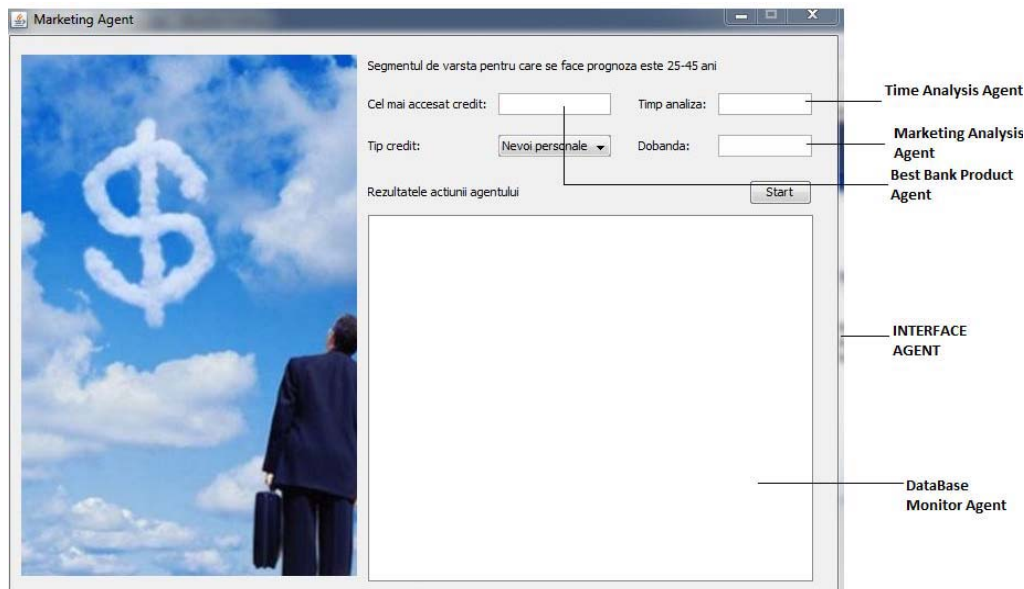


Interface agent-oriented will display the results of all the others agents and they, also, can be activated sequential or simultanuous.

A fiew **advantages** that an aplication based on intelligent agents have , comparring with the one based on threads:

1. Cooperation between agents: marketing analysis is made (*information is gathered from DB*) by an assistant agent which interacts (through ACL messages) with specialized agents: an agent who establishes, for example, increasing or decreasing rates, an agent which settles the most attractive bank product (loan), an JDBC-based *transducer agent* which serves as an interface between the database server and the other agents in the system, etc.
2. Another advantage is the continuous monitoring of the DB, which makes possible recording fluctuations and storing them for later reporting to the assistant agent. This is not possible using an application based on threads.

   Specialized agents are autonomous, and their life cycle is not related to the running time of the marketing analysis application. It may be recalled that these agents are independent of the application, so they can monitor the DB after stopping the application and even before starting the application.

3. Agents are reactive, providing information by request, and proactive, signaling when some conditions are accomplished.

## 4.2 Futures developing

As future developing, it can be said that boath aplications will be part of one to optmize the results, to add more functionalities and assistance on the process duration.

# References

[1] Michael E. Porter, *Competitive Strategy: A manual for survival and growth of firms in market economy conditions* , Teora, Bucharest ,2007
[2] Michael E. Porter, *Competitive advantage: A manual for survival and growth of firms in market economy conditions*, Teora, Bucharest ,2007
[3] Cetină I., *Banking and Financial Marketing*, Economic Publishing House, Bucharesti, 2005
[4] Dumitru, L., *Banking and Financial Marketing*, Dimitrie Cantemir University Press, Târgu-Mureş, 2001
[5] Gheorghiu, L. , *Ways of promoting banking services in Romania, marketing view*, ASE Bucharesti, 2003
[6] Odobescu E., *Marketing of modern banking*, Sigma House, Bucharest, 2007
[7] Kotler Philip, Armstrong Gary. *Principles of Marketing.* Ed. a 3-a. Bucharesti: Teora, 2004.
[8] Cetina Iuliana, Odobescu Emanuel, *Bank marketing strategy*, Economic Publishing House, Bucharest, 2007.
[9] Matt Haig, *E-marketing guide*, "Rentrop & Straton" Publishing House, 2005
[10] Gay R, Charlesworth A., Esen R., *Online marketing, customer-oriented approach,* "All" Publishing House, 2009
[11] Strauss J., El-Ansary A., Frost R., *E-Marketing*, Pearson Education, Inc Upper Saddle River, 2003
[12] Ioan-Gheorghe Ratiu, Nicoleta David, *Working with databases in Java*, „George Bariţiu" University, Braşov
[13] David Powers, *PHP Solutions.Dynamic Web Design Made Easy*.pdf
[14] Brad Bulger, Jay Greenspan, David Wall, *Mysql/Php Database Applications*, 2Nd Ed - 2004 - Wiley.pdf
[15] David M. Geary, *Graphics Java 2 - Mastering the JFC,* 3RD Edition, vol. II, Swing
[16] Jason Hunter, (E-Book ) Java Servlet Programming
[17] Andy Harrys, *PHP 6-MySQL Programming for the Absolute Beginner*.pdf
[18] Tricia Ballad, William Ballad, *Securing PHP Web Applications*.pdf
[19] Mircea Neamtu, *Distributed Programming Course*, "Lucian Blaga",University of Sibiu, Science Faculty 2010

[20] Ralf Fabian, *Course of Graphics and Image Precessing*, ”Lucian Blaga”,University of Sibiu, Science Faculty 2010
[21] E-marketing http://en.wikipedia.org/wiki/Internet_marketing
[22] Marketing Environment http://www.studentie.ro/Cursuri/Marketing/Curs-Marketing-MEDIUL-DE-MARKETING_i47_c1038_55121.html
[23] E-Marketing Strategies http://cerex.unitbv.ro/teze/rezumate/2010/rom/DajAlexisIon.pdf
[24] E-Marketing Plan , http://www.dotcom.ro/imarket.php
[25] Competition Analysis, http://www.scribd.com/doc/23835544/analiza-concuren%C5%A3ei
[26] Competitive Strategies http://www.preferatele.com/docs/marketing/4/analiza-strategica-a1.php
[27] Sales optimization techniques, http://www.newschannel.ro/categorii/vanzari-marketing/4/
[28] About safe online transactions http://www.talcioc.net/secure.php
[29] Advantages and disadvantages of PHP , http://kitsoftware.ro/Item/Framework-ul-PHP-care-sunt-avantajele-si-care-sunt-dezavantajele-lui/Section/Dezvoltare
[30] Working with the SQL database, http://www.ub-filosofie.ro/~solcan/minf/d06_07 /minf.pdf#page=135
[31] About Web Technology, http://tehnologii-web.ro/
[32] About JavaScript, http://en.wikipedia.org/wiki/JavaScript
[33] About CSS, http://en.wikipedia.org/wiki/Css
[34] About HTML, http://www.w3schools.com/html/default.asp
[35] Working with databases in Java http://software.ucv.ro/~aion/nad/jdbc.pdf
[36] PHP Web Security, http://www.slideshare.net/calin.iepure/securitatea-web-in-php-presentation
[37] Agent-oriented systems. Agent oriented analysis and design applications. http://profs.info.uaic.ro/~busaco/publications/articles/agenti2.pdf
[38] Florin Stoica, *Generating JADE agents from SDL specifications*, Int. Journal of Computers, Communications & Control, Supplementary Issue, Volume I, ISSN 1841-9836, pp. 429-438, 2006

VALENTINA LAZAR
”Lucian Blaga” University of Sibiu
Faculty of Sciences
Informatics Departament
Dr. Ion Ratiu st., No. 5-7,  550012 Sibiu, Romania
E-mail: lyah_valy@yahoo.com

CRISTINA RĂULEA
”Lucian Blaga” University of Sibiu
Faculty of Sciences
Informatics Departament
Dr. Ion Ratiu st., No. 5-7,  550012 Sibiu, Romania
E-mail:cristina.raulea@yahoo.com

# Center of a Set of Points in Three-dimensional Space Using Triangular Metric

**Petko Lalov, Stefan Dimitrov**

**Abstract**

The triangular distance between two points A(xa, ya, za) and B(xb, yb, zb) in three-dimensional space is defined as follows:

$$R_T = |x_a - x_b| + |y_a - y_b| + |z_a - z_b|$$

In the present paper is discussed and solved the following problem: given a set of points   to be found such a point  , denoted center, that minimizes:

$$\max R_T(M, A_i) \tag{1}$$

Based on the results in [1], where the two-dimensional case is analyzed, the authors prove that (1) has not a single solution and in the general case it is a two-dimensional convex simplex which is an intersection of n octahedra. The graphical solution is obtained by a developed computer program.

## 1   Introduction

The problem of finding the center of a set of points is an important problem in a number of mathematical models in transport, computer networks and communications and other engineering networks. The location of the points, on the vertices of a given graph, on the nodes of rectangular two-dimensional or three-dimensional network, is important [2]. For this reason a given metric is suitable in one case and different metric is suitable in other case. In the present paper triangular (Manhattan) metric is used that is suitable for finding the center of points in rectangular networks. The application of the latter metric helps to avoid the complex mathematical methods for nonlinear optimisation in solving the problem just stated which are not able to give all the possible geometrical solutions.

## 2   Formulation of the Problem

Triangular distance between two points A(xa, ya, za) and B(xb, yb, zb) in three-dimensional space is defined as follows:

$$R_T(A, B) = |x_a - x_b| + |y_a - y_b| + |z_a - z_b|$$

Given a set P of points  $A_i(x_i, y_i, z_i)$   $i = 1, 2, ......, n$ , the center of P will be such a point $M(x, y, z)$ , that minimizes

$$\max R_T(M, A_i) \quad i = 1, 2, \ldots, n \tag{1}$$

In [1] this problem is solved for the two-dimensional case. There it is defined as follows:

$$\min \max \rho_i \left( |x - x_i| + |y - y_i| \right) \tag{2}$$

where $\rho_i \geq 0$ are weights.

The following problem proves the equivalence of (2):

$$\min r \tag{3}$$

for:

$$|x - x_i| + |y - y_i| \leq \frac{r}{\rho_i}, \quad i = 1, 2, \ldots, n \tag{4}$$

## 3  The Two-dimensional Case

The algorithm for the implementation of (3) and (4), described in [1], is applied in the three-dimensional case.

The limitation (4) defines a set of points (x, y) whose triangular distance to $(x_i, y_i)$ is not greater than $\frac{r}{\rho_i}$. It is not difficult to estimate that this set is the interior of a square whose diagonals intersect at point $(x_i, y_i)$ and is rotated about the coordinate system by $45°$. And the square is a set of geometric points at an equal triangular distance from the point $(x_i, y_i)$ ( Figure 1).
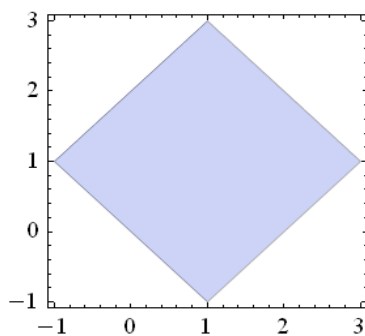


Figure 1

Considering the above, the problem is reduced to finding the smallest value of r, for which the intersection of the squares $|x - x_i| + |y - y_i| \leq \frac{r}{\rho_i}, \quad i = 1, 2, \ldots, n$ is not empty (Figure 2).
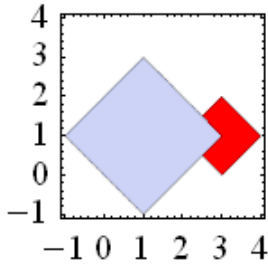
Figure 2

The basic steps of the algorithm for finding $\min r$ are:

- Coordinate system Oxy is rotated by $45^{\circ}$. In the new coordinate system Ox'y' the areas (4) are defined by the inequalities:

$$x_i^{'} - \frac{\sqrt{2}r}{2\rho_i} \le x' \le x_i^{'} - \frac{\sqrt{2}r}{2\rho_i}; \qquad y_i^{'} - \frac{\sqrt{2}r}{2\rho_i} \le y' \le y_i^{'} - \frac{\sqrt{2}r}{2\rho_i} \quad i = 1, 2, \ldots, n \qquad (5)$$

- What is this min r, for which the intervals (5) have non-empty intersection. Following [1] min r = max($r_1$, $r_2$), where

$$r_1 = \max \eta_{ij}; \qquad r_2 = \max \delta_{ij}, i = 1, 2, .., n-1, \quad j = i+1, \ldots, n$$

$$\eta_{ij} = \frac{\rho_i \rho_j}{\rho_i + \rho_j} \sqrt{2} |x_i^{'} - x_j^{'}|; \quad \delta_{ij} = \frac{\rho_i \rho_j}{\rho_i + \rho_j} \sqrt{2} |y_i^{'} - y_j^{'}|$$

- When $r_1 = r_2 = r$ the solution is one and only one and this is the point M with coordinates

$$x_M^{'} = x_{i_1}{'} + \frac{\sqrt{2}r}{2\rho_{i_1}}; \quad y_M^{'} = y_{i_2}{'} + \frac{\sqrt{2}r}{2\rho_{i_2}}$$

- When $r_1 > r_2$ $(r_2 > r_1)$ the solution is a line segment bounded by the two end points

$$(x_M^{'} = x_{i_1}{'} + \frac{\sqrt{2}r_1}{2\rho_{i_1}}, y_{M_1}^{'} = \max(y_i{'} - \frac{\sqrt{2}r_1}{2\rho_i})), \quad (x_M^{'} = x_{i_1}{'} + \frac{\sqrt{2}r_1}{2\rho_{i_1}}, y_{M_1}^{'} = \min(y_i{'} - \frac{\sqrt{2}r_1}{2\rho_i}))$$

then the transformation $Ox'y' \rightarrow Oxy$ is repeated

## 4   The Three-dimensional Case

In the three-dimensional case problem (1) can be formulated as follows:

$$min\ R \qquad\qquad (6)$$

for:

$$|x - x_i| + |y - y_i| + |z - z_i| \le R; i = 1, 2, ....., n \qquad (7)$$

Each of the limitations (7) is a regular octaedron, which image is shown in Figure 3.
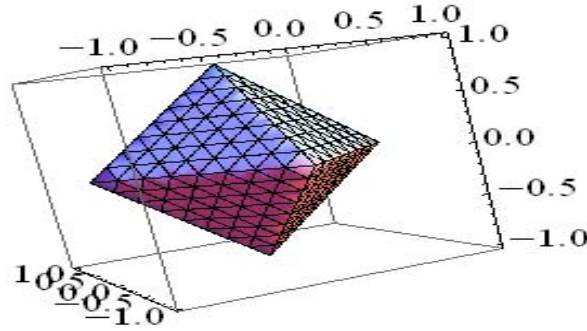


Figure 3

(6)-(7) are equivalent to

$$min \ R \qquad (8)$$

$$|x - x_i| + |y - y_i| \le R - d_i(z); \quad d_i(z) = |z - z_i|; \qquad i = 1, 2, ....., n \qquad (9)$$

If **z** is fixed, the above problem is two-dimensional.
The coordinate system Oxy is rotated by $45^{\circ}$ and the problem is reduced to finding the minR(z), where the intersection of the squares

$$x_i^{'} - k(R(z) - d_i(z)) \le x^{'} \le x_i^{'} + k(R(z) - d_i(z))$$
$$y_i^{'} - k(R(z) - d_i(z)) \le y^{'} \le y_i^{'} + k(R(z) - d_i(z)) \qquad (10)$$

is not empty.( $k = \dfrac{\sqrt{2}}{2}$ ).

## 4.1 Three-dimensional algorithm

1. For each pair of points $x_i^{'}$ $u$ $x_j^{'}$ is seeked $\alpha_{ij}(z) = \min(R(z))$, for which the intervals with centres $x_i^{'}$ $u$ $x_j^{'}$ have a common point. Considering (10), it is easy to prove that:

$$Rx_{ij}(z) = k|x_i^{'} - x_j^{'}| + 0.5|d_i(z) + d_j(z) \mid \qquad (11)$$

The same result is obtained on the axis Oy:

$$Ry_{ij}(z) = k|y_i^{'} - y_j^{'}| + 0.5|d_i(z) + d_j(z) \mid \qquad (12)$$

These functions look like as shown in Figure 4.
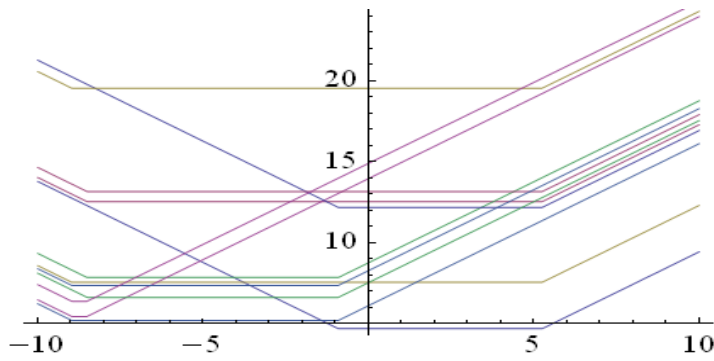
Figure 4

2. Considering the two sets of functions (11), (12), is seeked
$$R(z) = \max\{\max_{i,j} Rx_{ij}(z), \max_{i,j} Ry_{ij}(z)\}$$

(13)

It is obvious that $R(z)$ is a line segment, parallel to the axis Oz, and $z_{\min} \leq z \leq z_{\max}$

3. Defining $z_{\min}$ and $z_{\max}$

The two end points of the interval $z_{\min} \leq z \leq z_{\max}$ are defined as the intersection of $R(z)$ with the functions (11), (12).

4. From (10) are found the intervals (as function of z) within which x' and y' vary.
The boundaries of the area of variation of x' and y' for a particular instance are shown in Figure 5.
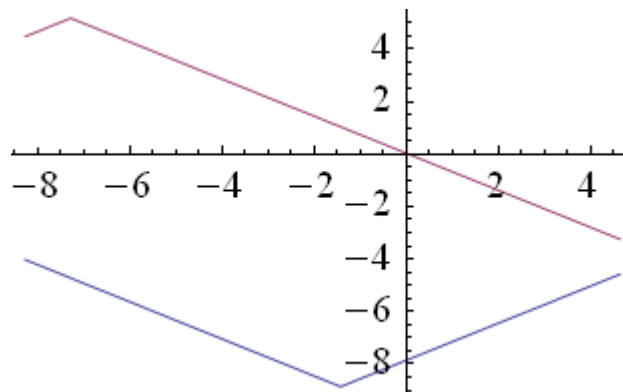


Figure 5

5. The center of the set of given points is drawn, which in the general case is a two-dimensional simplex in the three-dimensional space.
In Figure 6 is given an instance of computed center of 10 points, generated at random.

81
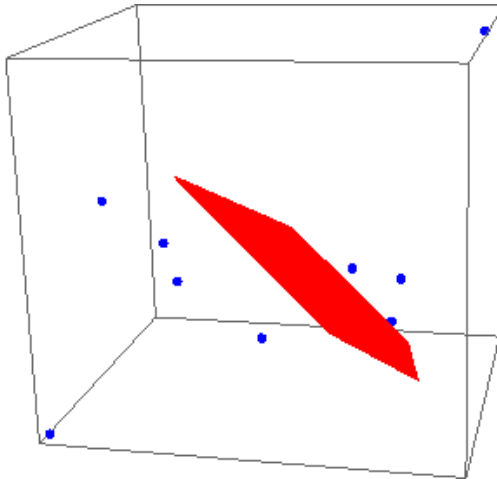
Figure 6

## 5  Conclusion

A computer program is developed that implements the proposed algorithm. The instances presented are generated by this program. For this purpose the program system Mathematica 7.0 with its rich graphical functionality is used.

## References

[1]  P. Lalov, T. Vasileva, *Extremal Problems of Triangular Metric, University Annual Applied Mathematics*, Volume 21, Sofia, 1985.

[2]  Thomas L. Saaty, *Optimization in Integer and Related Extremal Problems*, McGraw Hill, New York, 1970.

Petko Lalov
University of Mining and Geology "St.
    Ivan Rilski"
Head of Dept. of Informatics
Students' Town, Sofia
BULGARIA
E-mail: petko@mgu.bg

Stefan Dimitrov
University of Sofia "St. Kliment
    Ohridski"
FMI, Computing Systems Dept.
5, J. Bourchier, Str., Sofiq
BULGARIA
E-mail: stefan@ucc.uni-sofia.bg

# Fuzzy Expert System Design for Medical Diagnosis

## Man Diana Ofelia

### Abstract

In recent years, the methods of artificial intelligence have largely been used in the different areas including the medical applications. In the medicine area, many fuzzy expert systems (FES) were designed.

We study the possibilities of using fuzzy logic in building agent software assuming the role of an experienced medical person, which benefits of a vast medical knowledge regarding symptoms and diseases and has the role to orientate the young resident doctors in the process of diagnosis establishment.

## 1  Introduction

Fuzzy expert systems have been successfully applied in fields such as automatic control, data classification, decision analysis, expert systems, and computer vision.

Fuzzy expert systems can be used to aid in diagnosing medical cases. Symptoms and test results can be given to the expert system, which then searches its knowledge base in an attempt to match these input conditions with a particular malady or disease. This results in a conclusion about the illness and some possible suggestions on how to treat it. Such an expert system can greatly aid a doctor in diagnosing an illness and prescribing treatment. It does not replace doctors, but helps them confirm their own decisions and may provide alternative conclusions.

The Medical Diagnostics System is intended to be a software application mainly destined to orientate the resident doctors in the diagnostic process for patients" examinations. The system will be implemented for psychiatry diseases, but the architecture will try to make it flexible for further modules covering other medical areas.

The knowledge base of the system contains a few main entities: symptom with associated symptom values, disease and disease symptom value.

A *symptom* in our model is not equivalent with a medical symptom; it is an independent characteristic, like temperature, pulse, etc. The symptom has associated a name, a description - attribute that will be used to formulate the questions, and a list of two or more symptom values.

*The symptom value* in conjunction with a symptom on the other hand is similar with a medical symptom e.g. temperature of 38 degrees, pulse of 80, etc. For symptom "temperature" the symptom values would be a list of values: "36 degrees", "37 degrees", etc. Some of the symptom may have only the yes/no values.

A disease symptom value represents a symptom value associated to a disease. It contains a symptom and a symptom value associated. The diseases are stored under a hierarchical form following the model already defined in medicine. Important attributes are a list of disease symptom values for that disease. A patient should have most if not all the disease symptom values associated with a disease (at least all the mandatory ones) to be diagnosed with that disease [1].

# 2 Theoretical backgrounds

## 2.1 Fuzzy logic

Fuzzy logic is a form of multi-valued logic derived from fuzzy set theory to deal with reasoning that is approximate rather than precise. In contrast with binary sets having binary logic, also known as crisp logic, the fuzzy logic variables may have a membership value of not only 0 or 1.

Just as in fuzzy set theory with fuzzy logic the set membership values can range (inclusively) between 0 and 1, in fuzzy logic the degree of truth of a statement can range between 0 and 1 and is not constrained to the two truth values {true (1), false (0)} as in classic propositional logic.

Paradoxically, one of the principal contributions of fuzzy logic is its high power of precisiation of what is imprecise. This capability of fuzzy logic suggests, as was noted earlier, that it may find important applications in the realms of economics, linguistics, law and other human-centric fields [2].

## 2.2 Fuzzy sets

In fuzzy sets an object can belong to a set partially. The degree of membership is defined through a membership function: $\mu_A : U \rightarrow [0,1]$

where U is called the universe, and A is a fuzzy subset of U. Each value of the function is called a membership degree [3].

### 2.2.1 The basics notions of fuzzy sets

*A support of a fuzzy set A* is the subset of the universe U, each element of which has a membership degree to A different from zero

$$\text{supp}(A) = \{u|u \ U, \mu A(u) > 0\} \quad (1)$$

*Cardinality* of a fuzzy set M(A) is defined as follows:

$$M(A) = \sum \mu_A(u), u \in U \quad (2)$$

*Power set of A* is called the set of all fuzzy subsets of A.

*Normal fuzzy set* if the membership function of the fuzzy set has a grade of 1 at least for one value from the universe U.

*x-cut* of a fuzzy set A is a subset Aa of the universe U which consists of values that belong to the fuzzy set A with a membership degree greater (weak cut), or greater or equal (strong cut) than a given value x from [0,1]

*Subsethood* (introduced by Kosko in 1992). It measures the degree to which the whole universe U belongs to any of its fuzzy subsets [3].

### 2.2.2 Operations with fuzzy sets

*Union*, A∪B:

$$\mu_{A \cup B}(u) = \mu_A(u) \vee \mu_B(u) \text{ for all u from U, where } \vee \text{ means MAX} \quad (3)$$

*Intersection*, A ∩B:

$$\mu_{A \cap B}(u) = \mu_A(u) \wedge \mu_B(u) \text{ for all u from U, where } \wedge \text{ means MIN;} \quad (4)$$

The De Morgan's laws are valid for intersection and union.

*Equality*, A = B:

$$\mu_A(u) = \mu_B(u) \quad \text{for all u from U} \quad (5)$$

*Set complement*, notA, $\neg$ A:

$$\mu_{notA}(u) = 1 - \mu_A(u) \text{ for all u from U} \quad (6)$$

*Concentration*, CON(A):

$$\mu_{CON(A)}(u) = (\mu_A(u))^2 \text{ for all u from U;} \quad (7)$$

This operation is used as a linguistic modifier "very"

*Dilatation*, DIL(A):

$$\mu_{DIL(A)}(u) = (\mu_A(u))^{0.5} \text{ for all u from U;} \quad (8)$$

This operation is used as a linguistic modifier "more or less" [3].

### 2.2.3 Fuzzy terms

A linguistic variable denote a variable which takes fuzzy values and has a linguistic meaning.

Linguistic variables can be:

- *Quantitative*, for example, "temperature" (low, high); time (early, late); spatial location (around the corner);

- *Qualitative*, for example, "truth," "certainty," "belief."

The process of representing a linguistic variable into a set of linguistic values is called fuzzy quantization.
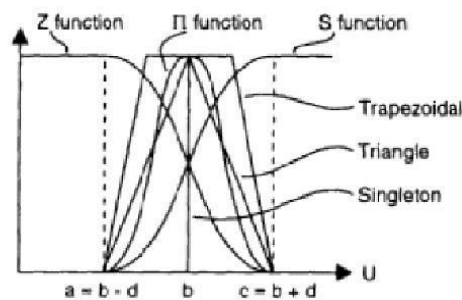


**Figure 1- The types of membership function**

## 2.3 Fuzzy propositions

*Fuzzy propositions* are propositions which contain fuzzy variables with their fuzzy values. The truth value of a fuzzy proposition "X is A" is given by the membership function .

Examples A person is a "heavy smoker." The temperature is "high." The speed is "moderate."

The fuzzy connectives are the same as in propositional logic, but here applied differently

$$\text{AND:} \quad \mu_{AandB} = \mu_A \wedge \mu_B \quad (9)$$

$$\text{OR:} \quad \mu_{AorB} = \mu_A \vee \mu_B \quad (10)$$

$$\text{NOT:} \quad \mu_{\neg A} = 1 - \mu_A \quad (11)$$

## 2.4 Fuzzy rules

The concept of "computing with words"(CW) is rooted in several papers, starting with Zadeh paper in 1973 – "Outline of a New Approach to the Analysis of complex Systems and Decision Processes", where the concepts of linguistic variable and granulation were introduced.
Computing with words evolved in a distinct methodology during time and it reflects many advantages of fuzzy logic and soft computing, advantages that took place within the past few years. A key aspect is that it involves a fusion of natural languages and computation with fuzzy variables [4].

The machinery of linguistic variables and fuzzy if–then rules is unique to fuzzy logic. This machinery has played and is continuing to play a pivotal role in the conception and design of control systems and consumer products.

- *Zadeh-Mamdani's fuzzy rules*:

IF x is A, THEN y is B,          (12)

A generalized form of the fuzzy rule is the following:

IF x1 is A1 AND x2 is A2 AND . . . AND xk is Ak, THEN y is B,    (13)

A set of fuzzy rules has a general form:

Rule 1: IF x1 is A1,1 AND x2 is A2,1 AND . . . AND xk is Ak,1, THEN y is B1, ELSE

Rule 2: IF x1 is A1,2 AND x2 is A2,2 AND . . . AND xk is Ak,2, THEN y is B2, ELSE

. . .

Rule n: IF x1 is A1,n AND x2 is A2,n AND . . . AND xk is Ak,n, THEN y is Bn,   (14)

In general, every rule that has two condition elements in its antecedent part connected by an OR connective can be represented by two rules, for example, the rule

IF x1 is A1 or x2 is A2, THEN y is B        (15)

is logically equivalent to the following two rules:

IF x1 is A1, THEN y is B   and   IF x2 is A2, THEN y is B.    (16)

- *Fuzzy rules with confidence degrees*:

If x is A, then y is B (with a CF).       (17)

Example

IF (current economic situation is good) and (current political situation is good) and (the predicted value for tomorrow is up), THEN (action—buy) (CF = 0.9)

- *Takagi-Sugeno's fuzzy rules* (1985*)*:

Rule i: IF x is Ai and y is Bi, THEN z is fi(x, y)      (18)

*If the function is linear, the rule takes the following form:*

Rule i: IF x1 is A1,i and x2 is A2,i and…xm is Am,i, THEN z = C0,i + C1,i· x1 + ⋯ + Cm,i · xm (19)

Example   *IF x is A and y is B, THEN z = 5x -2y + 3.*

- *Gradual fuzzy rules*: These are rules of the Zadeh-Mamdani type, but instead of using fuzzy values for the fuzzy variables in the rule, they use fuzzy representation of gradual properties, for example, "the more a tomato is red, the more it is ripe"[5].

## 2.5 Fuzzy inference methods

Different reasoning strategies over fuzzy rules are possible. Most of them use either the generalized modus ponens rule or the generalized modus tolens inference rule.

*(IF x is A, THEN y is B) and (x is A'), then (y is B')* should be inferred.

Some of the main else-links between fuzzy rules are:

*OR-link*: Max operator is used

*AND-link*: Min operator is used

*Truth qualification-link*: A coefficient Ti' is calculated for the inferred fuzzy set Bi' by every rule Ri. The result obtained by a rule Rj with the maximum coefficient is taken as a final result:

$$T_j = MAX\{T_i\}, i = 1,2,...n$$

$$T_i = \sum \mu_{B_i'}(v) / \sum \mu_B(v), \forall v \in V \qquad (20)$$

*Additive link*: The fuzzy results Bi' inferred by the rules Ri are added after being multiplied to weighting coefficients:

$$\mu_{B'} = \sum \mu_{B_i'} \cdot w_i, i = 1, 2, \ldots n \qquad (21)$$
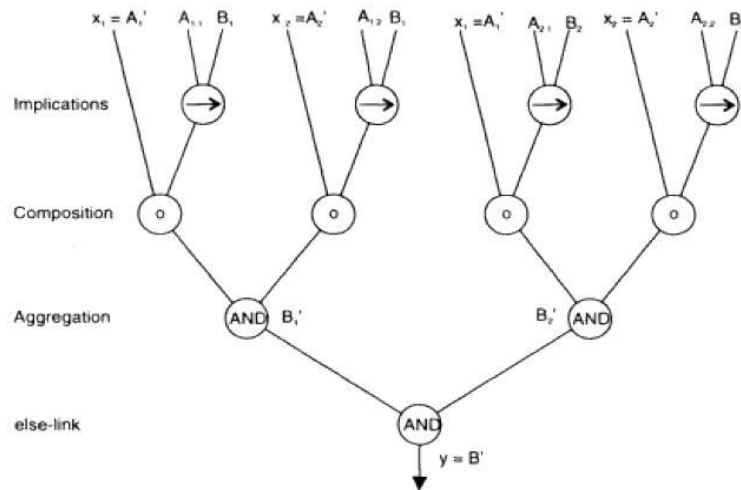


**Figure 2- Fuzzy Inference Methods**

## 2.6 Fuzzification, rule evaluation, defuzzification

*Fuzzification* is the process of finding the membership degrees and to which input data x'1 and x'2 belong to the fuzzy sets A1 and A2 in the antecedent part of a fuzzy rule.

*Rule evaluation* takes place after the fuzzification procedure. It deals with single values of membership degrees and produces output membership function B'. There are two major methods which can be applied to the rule above:

*Min inference*: $\quad B' = B \cdot \min\{\mu_{A1}(x_1'), \mu_{A2}(x_2')\}$ $\qquad (22)$

*Product inference*: $\quad B' = B \cdot \mu_{A1}(x_1') \cdot \mu_{A2}(x_2')$ $\qquad (23)$

where · denotes algebraic multiplication.

A *T-norm* is a binary mapping T: [0, 1] x [0, 1] → [0, 1], which has the following properties: commutativity, associativity, monotonicity. A boundary condition is held: T(a,1) = a. The following T-norm operators have been used in practice:

$$T(a,b) = \min\{a, b\}; \qquad (24)$$

$$T(a, b) = a * b \text{ (product)}; \qquad (25)$$

$$T(a, b) = \max\{0, a + b - 1\}. \qquad (26)$$

A T-conorm S(a, b) differs from a T-norm in that it has the property of S(a, 0) = 0 instead of the boundary property of the T-norms. Widely used T-conorms are:

$$S(a, b) = \max\{a, b\}; \qquad (27)$$

$$S(a, b) = a + b - ab; \qquad (28)$$

$$S(a, b) = min\{1, a + b\} \qquad (29)$$

*Defuzzication* is the process of calculating a single-output numerical value for a fuzzy output variable on the basis of the inferred resulting membership function for this variable.

Two methods for defuzzication are widely used:

- *The center-of-gravity method(COG).* This method finds the geometrical centre y' in the universe V of an output variable y, which center "balances" the inferred membership function B' as a fuzzy value for y.

  The crisp value is calculated by the method center of gravity defuzzifier by the formula:

$$D^* = \frac{\int D_{\mu \ middle(D)} dD}{\int \mu_{middle(D)} dD}$$

- *The mean-of-maxima method(MOM).* This method finds the value y' for the output variable y which has maximum membership degree according to the fuzzy membership function B' [3].

# 3  The structure of the Fuzzy Expert System

We have developed a rule-based Fuzzy Expert System for determination of the possibility of the diagnosis of schizophrenia, that uses symptoms data and simulates an expert-doctor's behavior.

As symptoms, Anxiety and Terror (AT), Age (A) and Isolation (I) are used. Using this data the fuzzy rules to determine the risk factor was developed. The developed system gives to the user the patient possibility ratio of the schizophrenia.

For the design process AT, A and I are used as input parameters and schizophrenia risk (SR) is used as output.
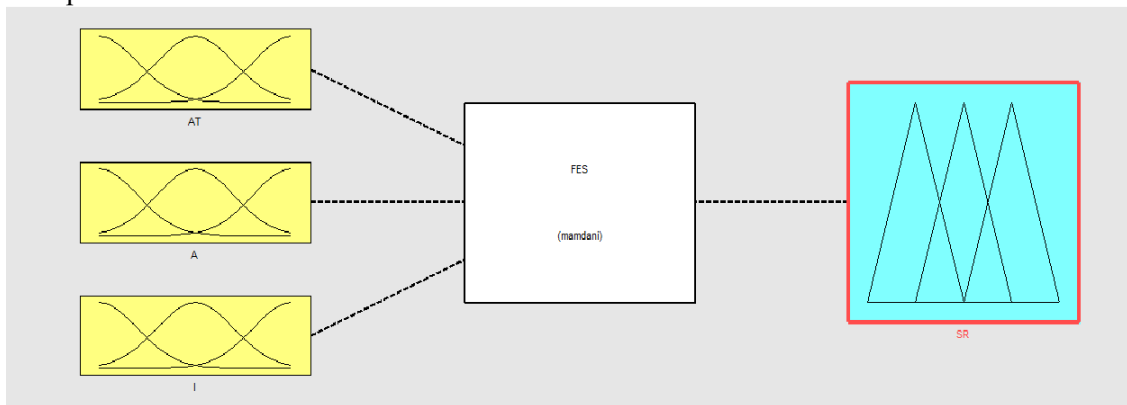


**Figure 3- The structure of Fuzzy Expert System**

Fuzzy inference process comprises of five parts: fuzzification of the input variables, application of the fuzzy operator (AND or OR) in the antecedent, implication from the antecedent to the consequent, aggregation of the consequents across the rules, and defuzzification.

The first step is to take the inputs and determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions. The input is a crisp numerical value limited to the universe of discourse of the input variable (in this case the interval between 0 and 10 for AT and I, respectively the interval between 0 and 100 for A) and the output is a fuzzy degree of membership in the qualifying linguistic set (always the interval between 0 and 1).

For fuzzification of AT and I factors the linguistic variables Low, Medium and High are used and for fuzzification of A factor, the linguistic variables Very Young, Young, Middle Age and Old are used.
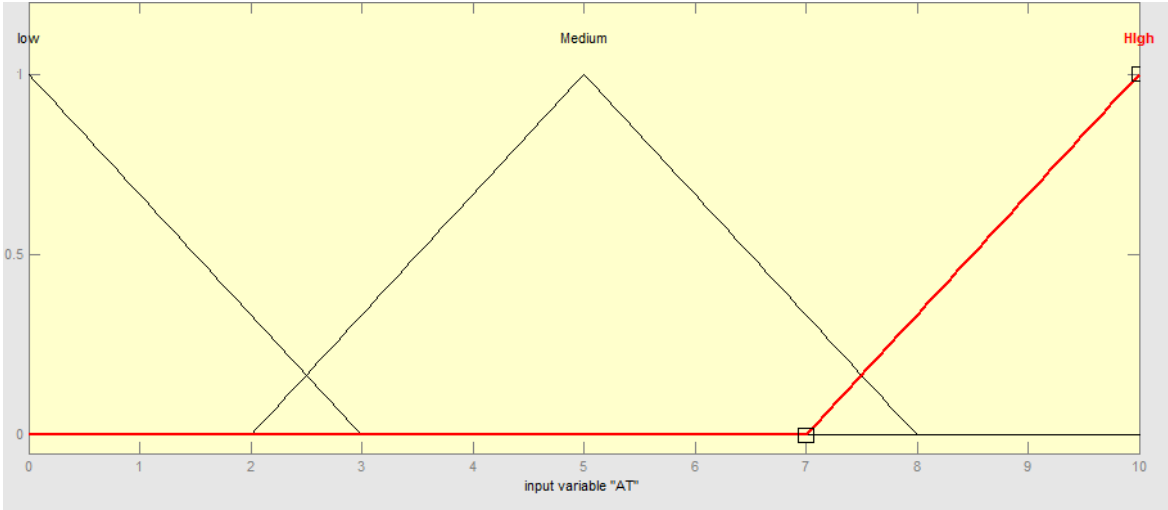


**Figure 4- Membership function of the AT**
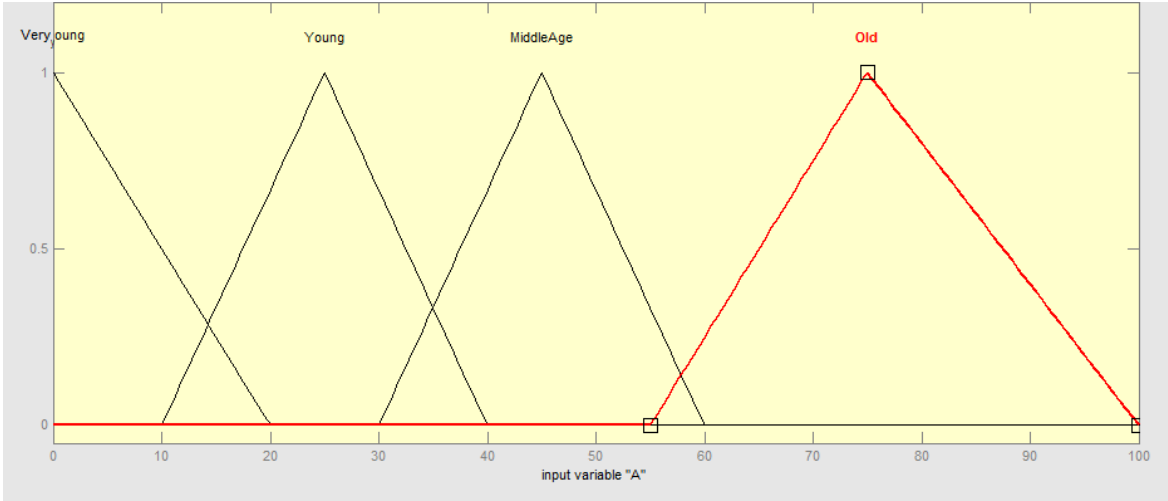


**Figure 5- Membership function of the Age**
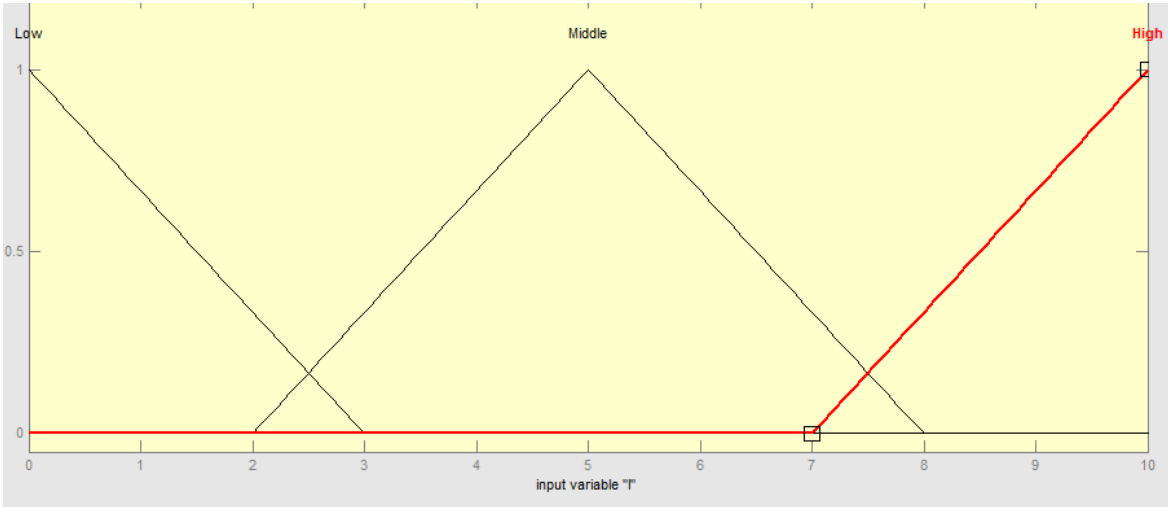


**Figure 6- Membership function of the I**
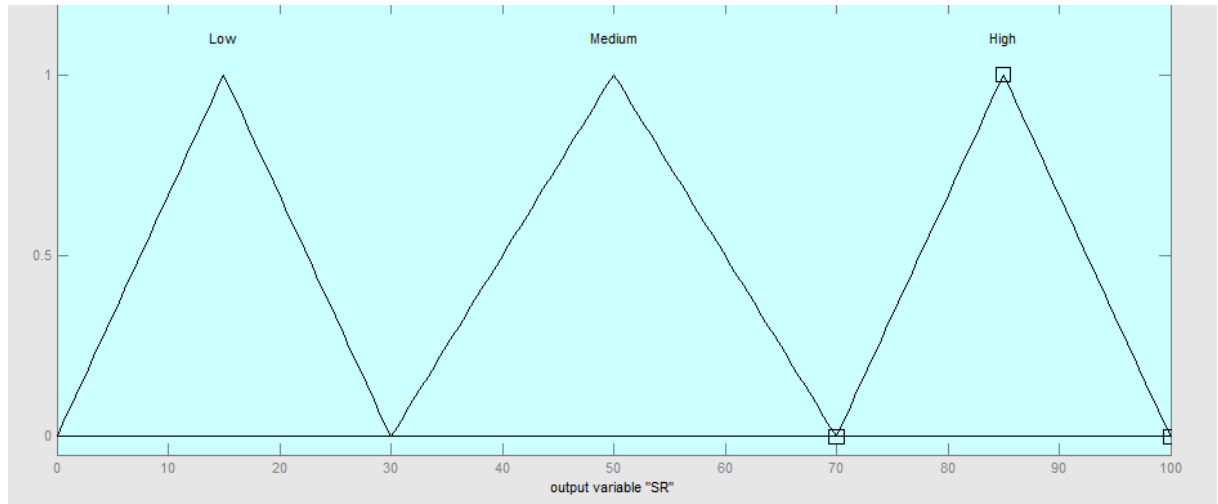
**Figure 7- Membership function of the SR**

After the inputs are fuzzified, you know the degree to which each part of the antecedent is satisfied for each rule. If the antecedent of a given rule has more than one part, the fuzzy operator is applied to obtain one number that represents the result of the antecedent for that rule. This number is then applied to the output function. The input to the fuzzy operator is two or more membership values from fuzzified input variables. The output is a single truth value [6].

Total of 36 rules are formed. For example, Rule 1, Rule 15 and Rule 18 can be interpreted as follows:

*Rule 1*: If AT is low and A is Very Young and I is Low then SR is low, i.e., if the patient's AT is low and patient is very young and patient's I is Low, then schizophrenia risc is low.

*Rule 15*: If AT is Medium and A is Very Young and I is High then SR is Medium, i.e., if the patient's AT is Medium and patient is very young and patient's I is High, then schizophrenia risc is medium.

*Rule 18*: If AT is Medium and A is Young and I is High, then SR is High, i.e., if the patient's AT is medium and patient is young and patient's I is high, then patient's SR is high.
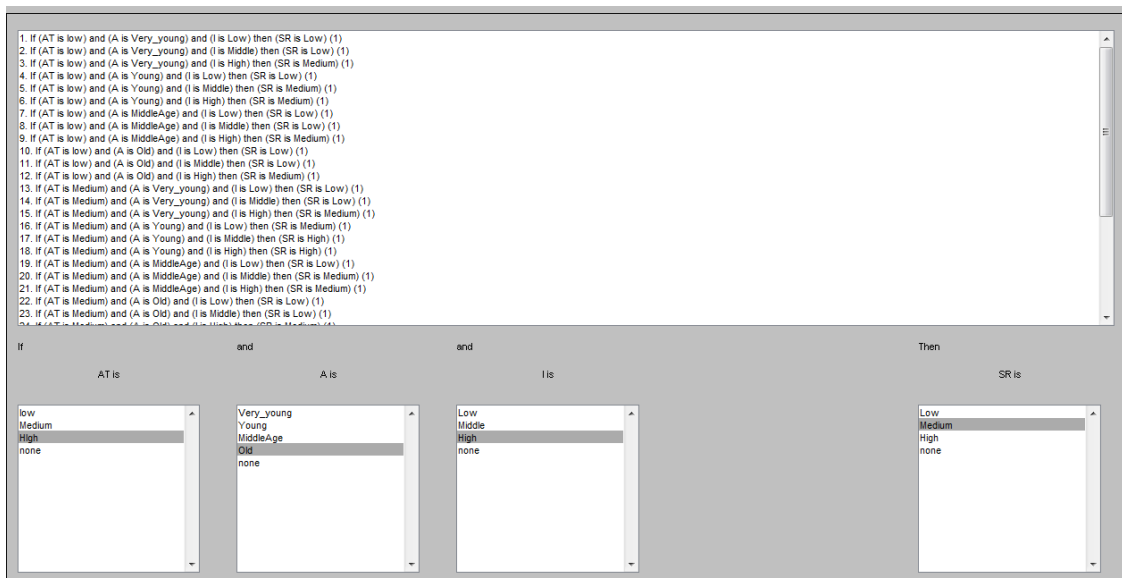


**Figure 8- Fuzzy rules**

For the inference mechanism the Mamdani max-min inference was used.

Because decisions are based on the testing of all of the rules in a FIS, the rules must be combined in some manner in order to make a decision. Aggregation is the process by which the fuzzy sets that represent the outputs of each rule are combined into a single fuzzy set. Aggregation only occurs once for each output variable, just prior to the fifth and final step, defuzzification.

The input for the defuzzification process is a fuzzy set (the aggregate output fuzzy set) and the output is a single number. As much as fuzziness helps the rule evaluation during the intermediate steps, the final desired output for each variable is generally a single number. However, the aggregate of a fuzzy set encompasses a range of output values, and so must be defuzzified in order to resolve a single output value from the set. Perhaps the most popular defuzzification method is the centroid calculation, which returns the center of area under the curve.

For the inputs AT=8, A=25, I=9, the output SR=85, i.e for the patient with AT equals 8, 25 years old and I equals 9, the SR is 85%.
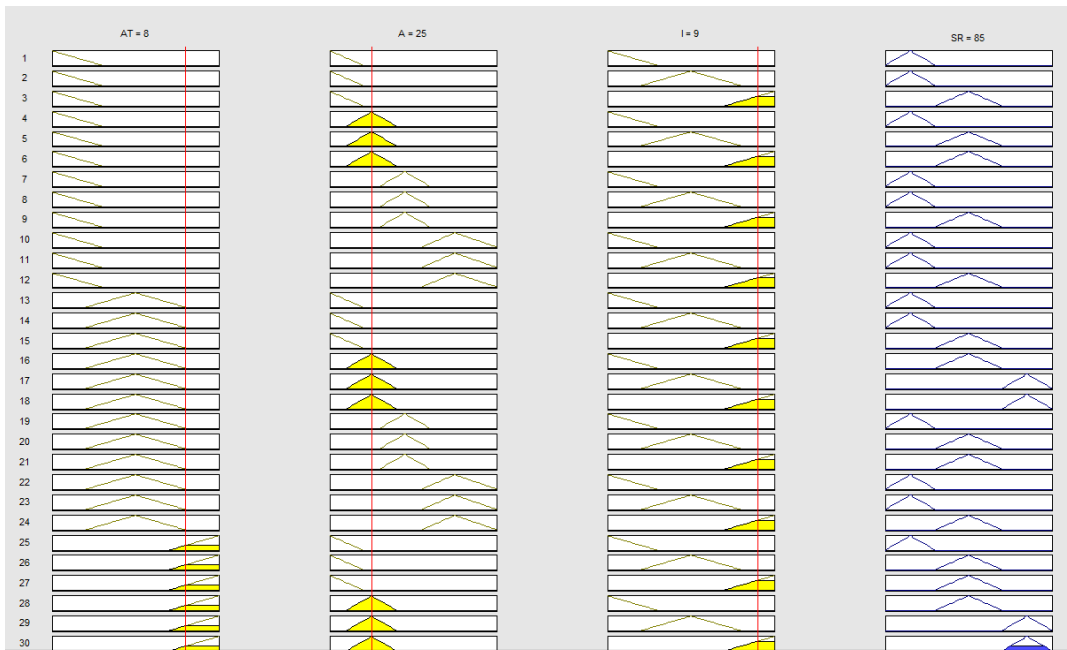


**Figure 9- Calculation of the value SR for the values AT=8, A=25 and I=9**
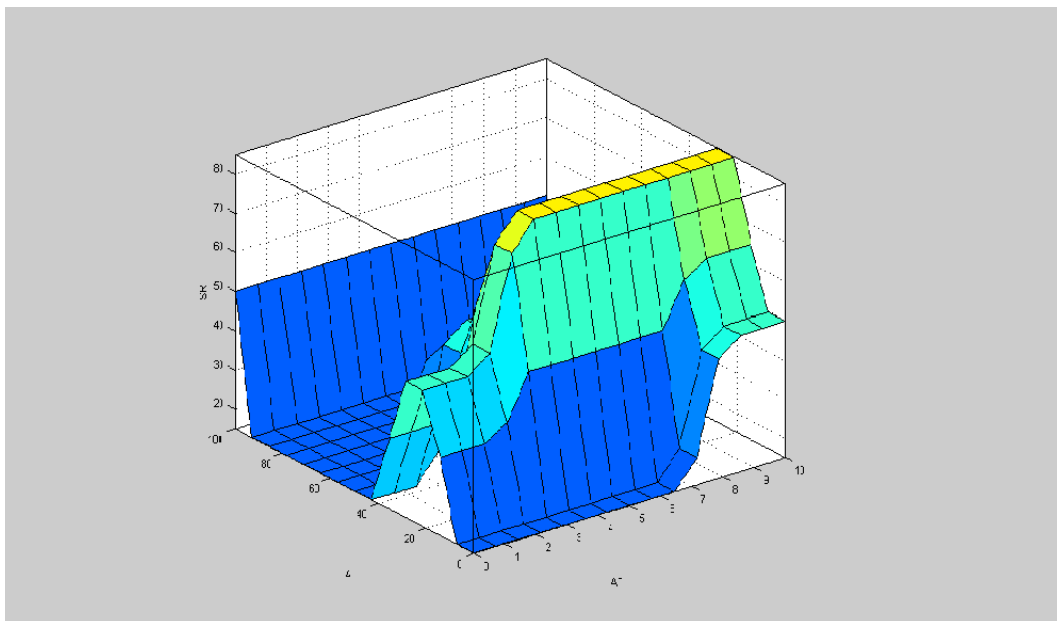


**Figure 10- Surface Viewer**

# 4   Conclusions

This paper presents a Fuzzy Expert System for Medical Diagnosys. We study the possibilities of using fuzzy logic in building agent software assuming the role of an experienced medical person, which benefits of a vast medical knowledge regarding symptoms and diseases and has the role to orientate the young resident doctors in the process of diagnosis establishment.

The last section describes a design of a Fuzzy Expert System for determination of the possibility of the diagnosis of schizophrenia, which can be used by the specialist doctors for treatment and by the resident doctors for learning the scope.

# References

[1] M. Tosa-Abrudan, D. Man, V. Zdrenghea, *Fuzzy Agent in a Medical Diagnosis Support System*, in Proceedings of the First International Conference on Modelling and Development of Intelligent Systems, Sibiu, Romania, October 22-25, 2009

[2] V. Novák,, I. Perfilieva and J. Mockor  *Mathematical principles of fuzzy logic* Dodrecht: Kluwer Academic. ISBN 0-7923-8595-0, 1999

[3] N.Kasabov, *Fuzzy Systems*,   Foundations of Neural Networks and Knowledge Engineering 1996 Massachusetts Institute of Technology, ISBN 0-262-11212-4

[4] L.A Zadeh , *From computing with numbers to computing with words – from manipulation of measurements to manipulation of perceptions*, Int. J. Appl. Math. Comput. Sci., 2002, Vol 12, No 3, 307-324.

[5] K. M. Passino and S. Yurkovich, *Fuzzy control*, 1998 Addison Wesley Longman, Inc., ISBN 020118074X

[6] S. N. Sivanandam, S. Sumathi and S. N. Deepa, *Introduction to Fuzzy Logic using MATLAB,* Springer-Verlag Berlin Heidelberg 2007,  ISBN-10 3-540-35780-7

Man Diana-Ofelia
Babes-Bolyai University
Department of Computer Science
400084 Cluj-Napoca
ROMANIA
E-mail: mandiana77@yahoo.com

# A cluster analysis for recommender systems evaluation metrics

**Ionela Maniu, George Maniu**

**Abstract**

Evaluation of recommender systems is a challenging task due to the many possible scenarios in which such systems may be deployed. Comparison between recommender systems it becomes difficult to achieve due to the large diversity of published metrics that have been used to quantitatively evaluate the accuracy of recommender systems. In this paper, we present a cluster analysis whose goal is to offer a classification of binary evaluation function in order to establish standardization within this field.
Keywords: cluster analysis, recommender systems, evaluation, binary metrics

## 1. Introduction

Recommender systems are define as ones in which "people provide recommendations as inputs", and the system then "aggregates and directs" on appropriate items (Resnick & Varian, 1997). Comparison between recommender systems it becomes difficult to achieve due to the large diversity of published metrics. To illustrate this, we quote some appropriate paragraphs from Herlocker, Konstan, Terveen, and Riedl (2004): "The challenge of selecting an appropriate metric is compounded by the large diversity of published metrics that have been used to quantitatively evaluate the accuracy of recommender systems. This lack of standardization is damaging to the progress of knowledge related to collaborative filtering recommender systems. With no standardized metrics within the field, researchers have continued to introduce new metrics when they evaluate their systems. With a large diversity of evaluation metrics in use, it becomes difficult to compare results from one publication to the results in another publication. As a result, it becomes hard to integrate these diverse publications into a coherent body of knowledge regarding the quality of recommender system algorithms."

In this paper, we present a cluster analysis whose goal is to offer a classification of binary evaluation function in order to establish standardization within this field.

This paper is organized as follows. Section 2 goal consists of achieving a logical framework with a common terminology in recommender systems domain. Section 3 describes the definitions of 27 binary similarity (dissimilarity) measures. Section 4 discusses the grouping of those measures using hierarchical clustering. Section 5 concludes this work.

## 2. A general framework for recommender systems

Recommender systems main objective is to guide the user to useful/interesting objects. For this, a number I of items are available to be recommended. In order to start the

recommendation process some of those items must be rated, these ratings are obtained explicit or implicit (inferred from other users interactions). Once the recommender system has enough ratings, it can start the process. For each recommendation, a number N < I of objects are chosen by the recommender, and show to the target user. Some recommender systems also rank the marked-out objects in order to show them as an ordered list, and in this case, the user will investigate these items starting at the top of this list.

In order to evaluate the performance of the recommender system for each object shown to a particular user, we must measure how close the utility of the shown object is with respect to the preferences of the user. In the case of an ordered list, additionally, we should take into account, the place that each recommended object has in this list.

In order to measure the closeness of predictions to users' real preferences, a numerical representation is normally used. This representation uses the predictions of a recommender system for every particular user u and item i, and the real preferences of user u for item i.

# 3. Binary similarity evaluation metrics

Once we have obtained the formal structures, in this section, we will study several metrics that can be applied into this framework. To compute these metrics, a confusion matrix is expected such as the one in table below. This table reflects the four possibilities of any recommendation decisions.

| | 1 successful recommendation | 0 Non - successful recommendation | sum |
|---|---|---|---|
| 1 recommended | a | b | a + b |
| 0 not recommended | c | d | c + d |
| sum | a + c | b + d | n=a +b +c +d |

Table 1: Confusion matrix for recommender systems

A binary vector X with N dimensions is defined as $x = ( x_1, x_2, \ldots, x_N)$ where $x_i$ has the value 0 or 1 and N represents the number of futures or dimension of the feature vector.

In confusion matrix, a is the number of occurrences of matches 1 in the first pattern and 1 in the second pattern at the corresponding positions (positive matches), b is the number of occurrences of matches 1 in the first pattern and 0 in the second pattern at the corresponding positions, c is the number of occurrences of matches 0 in the first pattern and 1 in the second pattern at the corresponding positions and d is the number of occurrences of matches 0 in the first pattern and 0 in the second pattern at the corresponding positions (negative matches). The diagonal sum a+d represents the total number of matches between patterns and the other diagonal sum b+c represent the total number of mismatches between patterns. The total sum of the table, a+b+c+d is always equal to n.

Numerous binary similarity evaluation metrics have been described in the literature [3]. Many papers discuss their properties and features. Table above lists formulas of 27 binary evaluation metrics used in cluster analysis, in this paper, in the case of recommender systems algorithms.

| Metrics | Formula | Metrics | Formula |
|---|---|---|---|
| Precision | $d(x,y) = \dfrac{d}{b+d}$ | Faith | $d(x,y) = \dfrac{a+0.5d}{n}$ |
| Recall | $d(x,y) = \dfrac{d}{c+d}$ | Gower&Legendre | $d(x,y) = \dfrac{a+d}{a+0.5(b+c)+d}$ |

| Vari | $d(x,y) = \dfrac{b+c}{4n}$ | Sokal&Michener | $d(x,y) = \dfrac{a+d}{n}$ |
|---|---|---|---|
| Shapedifference | $d(x,y) = \dfrac{n(b+c)-(b-c)^2}{n^2}$ | Sorgenfrei | $d(x,y) = \dfrac{a^2}{(a+b)(a+c)}$ |
| Patterndifference | $d(x,y) = \dfrac{4bc}{n^2}$ | Otsuka | $d(x,y) = \dfrac{a}{((a+b)(a+c))^{0.5}}$ |
| Lance&Williams | $d(x,y) = \dfrac{b+c}{2a+b+c}$ | Mountford | $d(x,y) = \dfrac{a}{0.5(ab+ac)+bc}$ |
| Hellinger | $d(x,y) = \sqrt{(1 - \dfrac{a}{\sqrt{(a+b)(a+c)}})}$ | Mcconnaughey | $d(x,y) = \dfrac{a^2 - bc}{(a+b)(a+c)}$ |
| Cosinus | $d(x,y) = \dfrac{a}{(a+b)(a+c)}$ | Tarwid | $d(x,y) = \dfrac{na-(a+b)(a+c)}{na+(a+b)(a+c)}$ |
| Ochiai-I | $d(x,y) = \dfrac{a}{\sqrt{(a+b)(a+c)}}$ | Driver&Kroeber | $d(x,y) = \dfrac{a}{2}(\dfrac{1}{a+b} + \dfrac{1}{a+c})$ |
| Jaccard | $d(x,y) = \dfrac{a}{a+b+c}$ | Jhonson | $d(x,y) = \dfrac{a}{a+b} + \dfrac{a}{a+c}$ |
| Dice | $d(x,y) = \dfrac{2a}{2a+b+c}$ | Simpson | $d(x,y) = \dfrac{a}{\min(a+b,a+c)}$ |
| Sokal&Sneath-I | $d(x,y) = \dfrac{a}{a+2b+2c}$ | Braun&Banquet | $d(x,y) = \dfrac{a}{\max(a+b,a+c)}$ |
| Sokal&Sneath-II | $d(x,y) = \dfrac{2(a+d)}{2a+b+c+2d}$ | Sokal&Sneath IV | $d(x,y) = \dfrac{\frac{a}{a+b}+\frac{a}{a+c}+\frac{d}{b+d}+\frac{d}{b+d}}{4}$ |
| MAE | $d(x,y) = \dfrac{b+c}{a+b+c+d}$ | | |

Table 2: Binary evaluation metrics

# 4. Hierarchical clustering

Hierarchical clustering is conducted to estimate the similarity among 27 measures collected. The correlation coefficient values between two measures are used to build a *dendrogram*. The average linkage between groups with squared Euclidian distance clustering method is used.

We used data from the well-known MovieLens project(http://movielens.umn.edu). Data sets consists of 100,000 ratings (1-5) from 943 users on 1682 movies. The dataset was divided into training set (90%,80% of the data) and test set (10%, 20% of the data) five times.

The dendrogram in figure 1 provides intuitive semantic grouping of 27 binary evaluation metrics[4], used to measure the prediction results of our algorithms and the real rating.
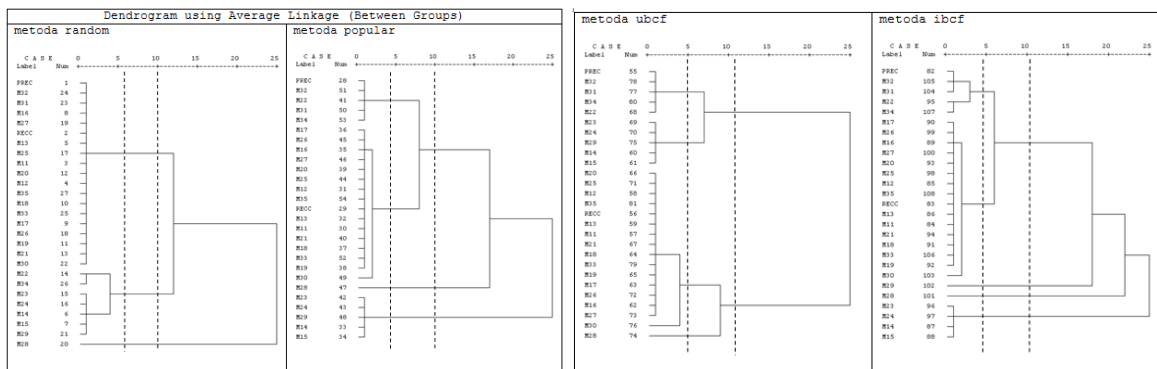


Fig. 1: Classification of 27 binary evaluation metrics, using cluster analysis, the case of random, popular, ubcf, ibcf recommendation algorithm

Binary measures with high correlation: m14-Lance&Williams, m15-Hellinger, m23-Gower&Legendre, m24-Sokal&Michener are categorized in first group, while m11-Vari, m12-Shapedifference, m13-Patterndifference, m16-cosinus, m17-Ochiai I, m18-Jaccard, m19-Dice, m20-Sokal&Sneath I, m21-Sokal&Sneath II, m25-Sorgenfrei, m26-Otsuka, m27-Mountford, m30-Driver&Kroeber, m31-Jhonson, m32-Simpson, m33-Braun, m35-MAE, recc – recall binary measures are categorized in second group, in all 4 different recommender systems algorithms. Apart from this group is m28-Wcconnanghey measure. It is interesting that additive form of negative match measures such as Jaccard, Dice have high correlation with the cosine based measures such as Ochiai I or Sorgenfrei.

# 5. Conclusions

In this survey, we used 27 binary similarity and distance metrics for evaluating 4 different recommender systems algorithms and classified them (metrics) through hierarchical clustering. In all cases, there are 3 main classes of metrics, at aggregation level smaller than 10, and there are 2 main classes of metrics, at aggregation level smaller than 15. There are differences between clusters elements (metrics) in the case of ubcf algorithm versus case of ibcf, random, popular algorithm.

# References

[1] Herlocker, J. L., Konstan, J. A., Terveen, L. G., Riedl, J. T. *Evaluating collaborative filtering recommender systems*. ACM Transactions on Information Systems, 22(1), 5–53,2004
[2] Resnick, P., Varian, H. R. , *Recommender systems*. Communications of the ACM, 1997
[3] Cha, S.H, Tappert, C.C., *Enhancing Binary Feature Vector Similarity Measures,* Journal of Pattern Recognition research I, 2006
[4] Maniu, I., *Mechanisms of economic language modeling with applications in recommender systems*, doctorate dissertation, ASE Bucharest, 2011

Ionela Maniu
Faculty of Science
Department of Informatics
Str. Ion Ratiu, nr. 5-7
Romania
E-mail: mocanionela@yahoo.com

George Maniu
Faculty of Management
Department of Management
Str. Turnului, nr. 7
Romania
E-mail: georgemaniu@yahoo.com

# Versatile integration of data mining techniques of description and prediction in Web informatics systems of Business Intelligence

**Mircea – Adrian MUSAN**

### Abstract

Using data mining techniques in computer applications for the digital economy, but not only, opened new possibilities for handling information in real time and their movement between all the factors involved. Computer applications based on these techniques assist successfully entrepreneurs in making decisions to achieve a higher degree of economic efficiency at company / organization level. Through this paper Web framework proposed, presented and developed is a versatile application, designed in a flexible way to integrate data mining techniques that are represented by RapidMiner processes applied in e-business. My application proposed here is in fact a management system of users and its processes assigned, build by data mining techniques, preprogrammed in RapidMiner.

## 1 Introduction

Application proposed here through this work is a departure point, one component of the subsequent developments, which can be used in the construction of Web informatics systems with application in economic areas, but not only.

### 1.1 Technology for extraction and knowledge management: data mining

Data mining techniques provides a useful and broad perspective in developing and using information systems in the field of e-business. Diverse areas and purposes of use, and the need to remote access, are elements of departure in this work. Thus, by subject matter, it wants to join the existing international concerns in the field of information technologies and their integration into economic applications.

Data mining is the process of extracting the knowledge on the databases / data warehouses, knowledge previously unknown, valid and operational at the same time [1].

Data mining, the analysis step of the Knowledge Discovery in Databases process, or KDD [5] is the process of extracting patterns from large data sets, by combining methods from statistics and artificial intelligence with database management. [6] With recent tremendous technical advances in processing power, storage capacity, and inter-connectivity of computer technology, data mining is seen as an increasingly important tool by modern business, to transform unprecedented quantities of digital data into business intelligence giving an informational advantage. It is currently used in a wide range of profiling practices, such as marketing, surveillance, fraud

detection, and scientific discovery. The growing consensus that data mining can bring real value has led to an explosion in demand for novel data mining technologies. [7]

Data mining deals with large complex data processing. Robust tools are needed to recover weak signals. These instruments require extremely efficient algorithms to achieve the desired processing. Data mining software combines artificial intelligence, statistical analysis and systems management databases to try to extract knowledge from data stored.

## 1.2 RapidMiner, "engine" in development processes through data mining techniques

Rapid Miner is an environment for machine learning, data mining, text mining, predictive analytics, and business analytics. It is used for research, education, training, rapid prototyping, application development and industrial applications. [8]

It allows experiments to be made up of a large number of arbitrarily nestable operators, described in XML files which are created with RapidMiner's graphical user interface. It provides a GUI to design an analytical pipeline (the "operator tree" in RapidMiner parlance). The GUI generates an XML (eXtensible Markup Language) file that defines the analytical processes the user wishes to apply to the data. This file is then read by RapidMiner to run the analyses automatically. [2]

For my application I realized the construction of a series of processes RapidMiner (an open source environment that provides complex processing to achieve modular operators) by using data mining techniques specific to e-business, using techniques of description and prediction. A data mining process is viewed as a complex process which consists in carrying out a sequence of steps: data cleaning, integration of different sources, selecting relevant data, the summary data transformation and aggregation, knowledge extraction and presentation templates.

## 2 Building data mining processes as tools for the investigation of specific e-business applications

### 2.1 Data Mining Technologies

The tasks of data mining process can be classified by types of knowledge sought by the user. The most common types of data mining tasks are [2][3]:

- *Summarizing*: A set of relevant data is summarized and abstracted resulting in a smaller set of data providing an overview of the aggregated information. A summary table can be generalized to different levels of abstraction and seen from different angles. For example, a company can be summarized sales by product, region or years and seen various levels of abstraction in any combination thereof.

- *Classification*: In the process of classification is given to analyzing a training data set or a set of objects whose class label is unknown. The model can be used to classify future data and develop a better understanding of each class in the database. For example, a model can be built based on the classification of disease symptoms and characteristics that can be used to diagnose new patients.

- *Clustering*: Clustering is the process of site identification for a variety of classes of objects based on attributes unclassified them. Objects that are classified as inter-class similarities are minimized by relying on certain criteria. Once you have decided clusters with common features of objects in a cluster, they are summarized to form a description of the class. For example, a company can classify its customers into several categories based on similarities of their age, income or address, and the common characteristics of customers in a category can be used to describe that group of customers.

- *Analysis of trends*: Templates and regularities in the behavior of data changes are detected during data binding (which they attribute data time). The data are analyzed over time and trace data are compared and appropriate modification. Trends, such as periods of growth or decay times, which happens frequently reported. For example, sales of companies can be analyzed each year, quarterly or monthly sales and to discover models to analyze the reasons behind them.

- *Mining association rules based*: An association rule reveals the associative relationships between objects, especially in a transactional database. For example, an association rule, "expected message, a message appears", says if a customer subscribes to the service "call waiting" is very likely that he or she also have the service "call display". Databases are searched to identify associations between objects and data. Another example, a retail store may discover that a set of goods is often bought with a different set of goods. This finding can then be used to design the sales strategy.

## 2.2 Development of RapidMiner processes through data mining techniques

Trough my work I realized the construction of a series of processes data mining through RapidMiner by using data mining techniques of description and prediction. A data mining process is viewed as a complex process which consists in carrying out a sequence of steps: data cleaning, integration of different sources, selecting relevant data, the summary data transformation and aggregation, knowledge extraction and presentation templates.

Following the versatility – flexibility axis, the processes are developed from a wide area, using such classification and regression techniques, clustering and association analysis techniques. The techniques used in the construction processes of classification and regression methods are: k-NN, DecisionTree, SuportVectorMachine, clustering methods: K-Means and BDScan, by association analysis methods: W-Apriori and FPGrowth – Association Rules.

Before developing the application was needed to build a series of processes with data mining techniques, by programming in RapidMiner, using such predictive techniques but also description techniques.
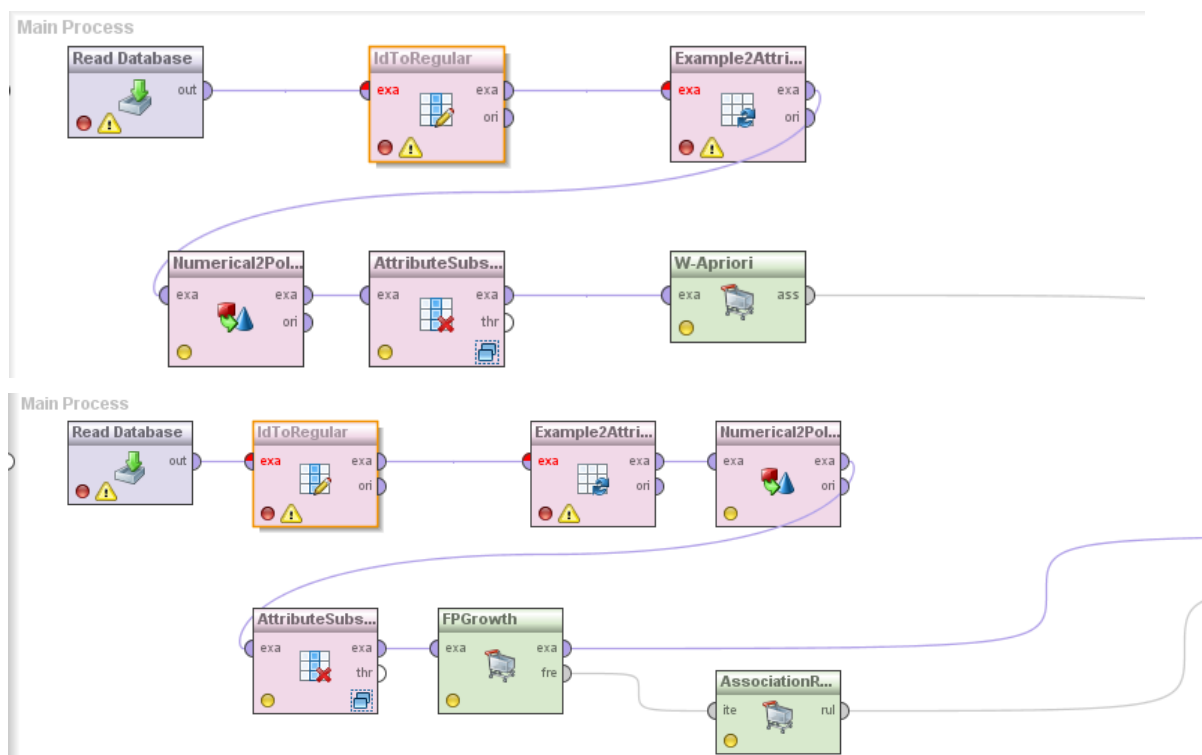


**Figure 1** – Such examples built on description techniques
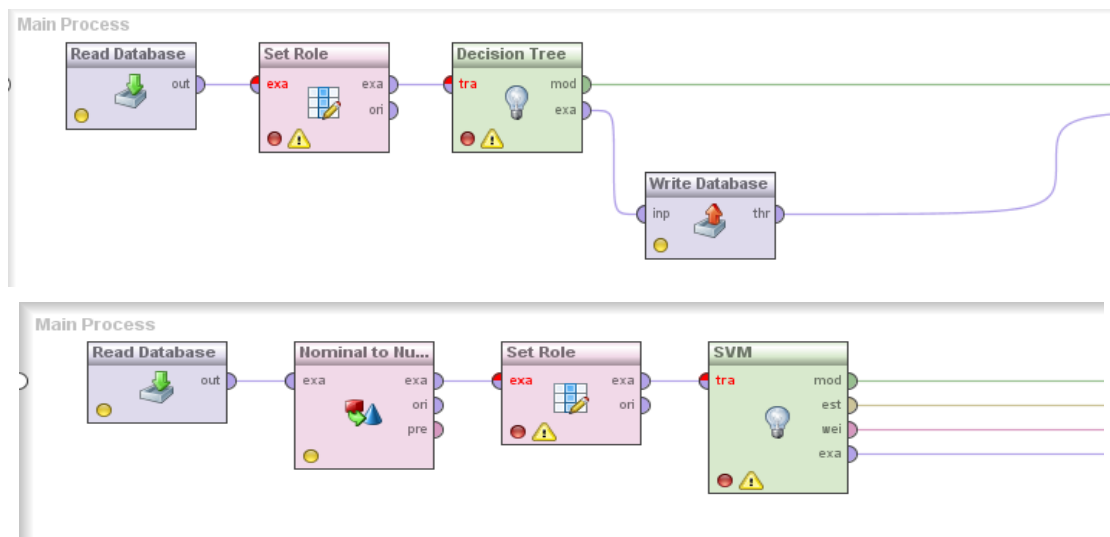
**Figure 2** – Such examples built on prediction techniques

# 3 The Web application developed for integration of data mining generic techniques

## 3.1 The application structure

The application submitted through this work is created by two client – side components:

- application instantiator enables the instantiation of new data mining web applications and the registration of users for which the access to application resources is granted;

- application manager allows registered users to manage all the actions for which they granted permission, such as connecting to the data mining engine hosted by a server, connecting to remote databases, editing, running and modifying complex data mining experiments, visualizing the results, and so on. [4]
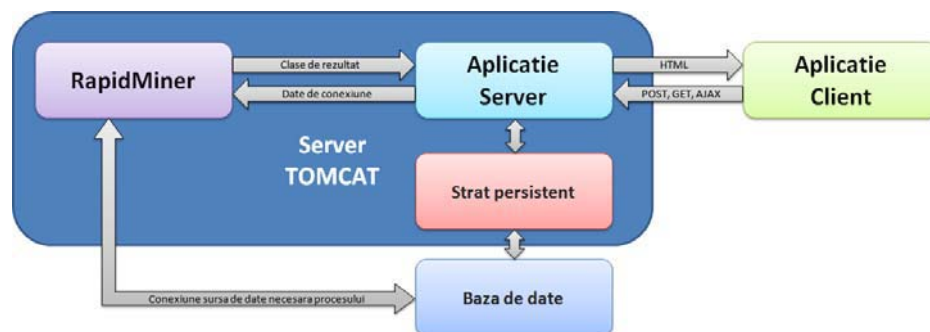


**Figure 3** – The application structure

In Figure 3 we can see structural form of the application:

- *Persistent leyer* (*Hibernate*): intermediate trading of commands SQL and assist in handling data

- *RapidMiner*: compose from RapidMiner library it running and execute processes

- *Server application*: the answers transpose into HTML and sends it to client application.

## 3.2 Versatility – flexibility – generality axis conferred

The following features bring the application in versatility – flexibility – generality axis:

- Any user, according to his membership in a group, can load and run a desired process from the categories of processes integrated in the application.

- There was processed a number of data mining processes developed in RapidMiner and rendering of the results is specific for each particular type of process.

- The user can add new processes, which, if built on some classes of operators, after scanning, application recognizes respective process, can fit into a particular class and can send that result.

- There is the possibility of introducing of other types of processes, but, given the characteristic way of execution and the results of its own way of rendering, this can only be programmed into the application environment.

- Any existing process on the application server can be accessed through its XML code.

- There can be used different datasets as storage diversified through files in various formats (xls,. arff, .aml, .csv, .sql, c4.5, .dat, .xrff, etc..). Any set of data may be retrieved from a host computer, from a URL, from server application's database or from another database server, and then it is converted to an SQL database and added to current server database.

- It can use the data sets diversified in content, provided by a structure that would be acceptable to the process.

## 3.3 Presentation of the application

The developed framework is an easy management users and their associated processes system, for analysing data and making decision, designed to run any data mining process, by using the algorithms from the specific e-business application classes. Through its architecture, the attached application provides a highly flexibility, can easily be modified in content, distributed and improved, as a new element in advanced use of intelligent systems in economic activities. Such an application can be regarded as a flexible informatics system, due to its techniques and due to its heterogeneous data sets used for work, as well as to its accessing mode, for anybody and from anywhere, through the "browser" tool used for integrating, adding, accessing and view of the results rendered particularly for each implemented technique, and the possibility of modifying, and deleting the realised processes, respectively.

It is presented in detail how to use the application, entry window, with options for user's access, management options for user groups, data mining process management option, the option of running processes and logout option. For each of these options, the content panel displays the controls for execution and views the results.



**Figure 4** – The user's list

**Figure 5** – Management of processes



**Figure 6** – Possibility of modifying the existing processes wrote in XML code



**Figure 7** – Running a RapidMiner process

Update dataset can be done by providing a link where the *.csv* file containing the new set of data or providing the address and authentication data on a server database and an order of selection set. Running process is done by clicking on the link associated *Run process*. This will generate a form dynamically depending on the current process.

The results of a process are stored in the database as HTML text, which is created dynamically according to each process. Processes can have one or more results, but they all share a schematic representation of the data upon which it was run.

I will present here viewing mode of the results for two any processes loaded on applications: one of prediction and one of description.
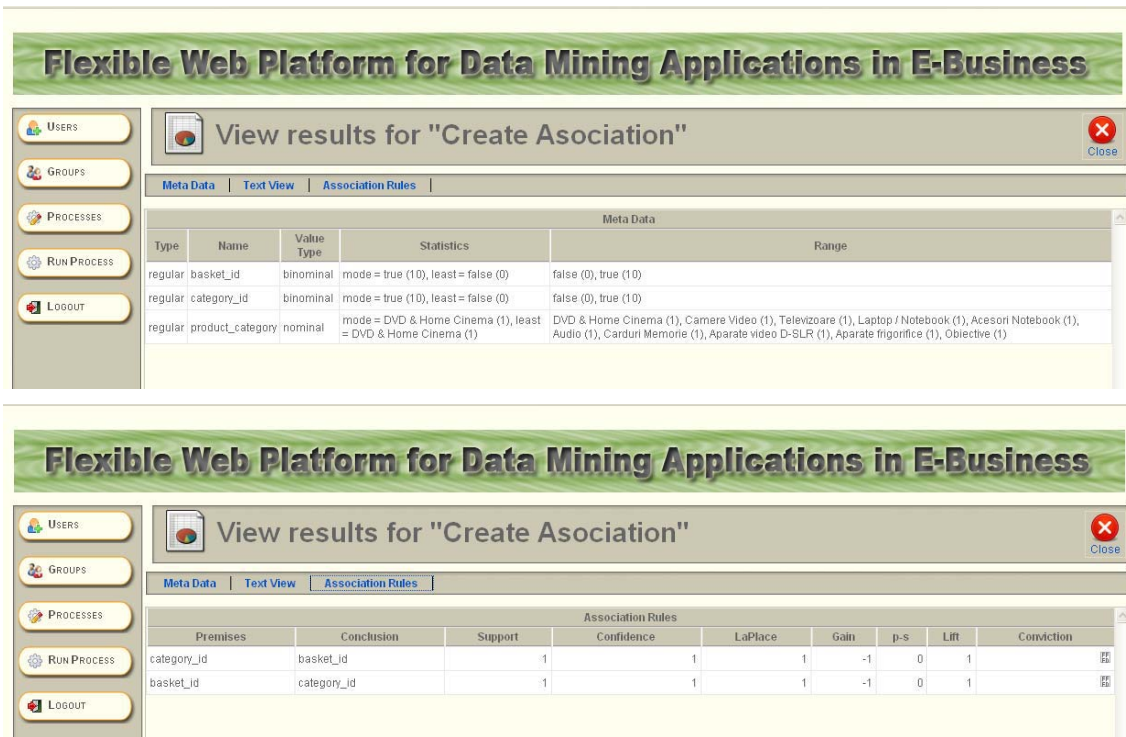


**Figure 8** – Ways to view results for a particular process built by description techniques
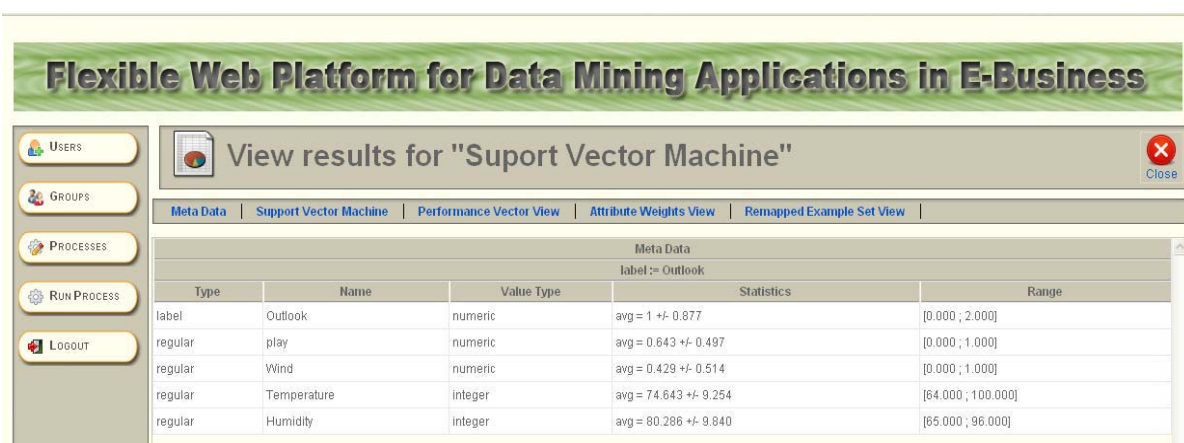
**Figure 9** – Ways to view results for a particular process built by prediction techniques

As it can see each mode of rendering the results is particular for each technique, regardless of the class to which it belongs. All these modes of rendering the results are those proposed by RapidMiner at viewing its results and integrated by my application.

# 4 Conclusions

In terms of practical usefulness, such an application developed can be regarded as a system usable in e-business for a company with several departments (development, innovation, marketing, promotion, etc.), for each of them being possible combinations of processes built into RapidMiner, or it can be regarded as a Web platform that can be used by anyone, anywhere, for the integrating, adding and accessing data mining techniques build processes.
The originality of this application lies in the idea of building it, and in the implemented methodology, by developing of an application of process integration, built by using data mining techniques in a flexible and general mode, but with the possibility of assimilation to e-business systems.

# References

[1] Giudici P., *Applied Data Mining: Statistical Methods for Business and Industry*, J. Wiley & Sons, 2005
[2] Mușan M., Hunyadi D., *Opportunities in development of business systems by building a web flexible framework for integrating generic data mining techniques in economic activities*, Analele Universității "Eftimie Murgu" Reșița, Fascicolul II Stiințe Economice, CNCSIS tip B+, pp. 380-389, 2010, ISSN 1584-0972 (CNCSIS B+)
[3] Hunyadi D., Mușan M., *Integration of data mining techniques in e-commerce applications*, Analele Universității "Eftimie Murgu" Reșița, Fascicolul II Stiințe Economice, CNCSIS tip B+, pp. 568-576, 2010, ISSN 1584-0972 (CNCSIS B+)
[4] Georgescu V., *WSRP-Enabled Distributed Data Mining Services Deliverable over a Knowledge-Driven Portal*, Proceedings of the 7th WSEAS International Conference on Applied Computer & Applied Computational Science (ACACOS'08), April 6-8, 2008, Hangzhou, China, 1790-5117, 150-156
[5] Christopher Clifton, *Encyclopedia Britannica: Definition of Data Mining*, http://www.britannica.com/EBchecked/topic/1056150/data-mining  Retrieved 2010-12-09, 2010
[6] http://www.anderson.ucla.edu/faculty/jason.frand/teacher/technologies/palace/datamining.htm
[7] http://www.sigkdd.org/curriculum.php
[8] http://en.wikipedia.org/wiki/RapidMiner

Mircea – Adrian MUSAN
"Lucian Blaga" University from Sibiu
Department of Mathematics and Informatics
Sibiu, Ion Ratiu Street, No. 5 – 7
Romania
E-mail: musanmircea@yahoo.com

# A prolongation technique for solving partial differential equations with a multigrid method

**Gabriela Nuţ**

### Abstract

The purpose of this paper is to introduce a new prolongation method for solving partial differential equations by a numerical method of multilevel type. This new technique is compared with others already existing in the literature, by means of some numerical results.

## 1  Introduction

The partial differential equations (PDE's) are used for modeling various real processes and phenomena in many different fields (fluid mechanics, thermodynamics, economics,...). Solving a PDE is thus very important, practically and theoretically. Because the analytical solution is not always available, it is important to consider numerical methods for approximating the solution of such an equation.

It is well known from the literature that the most used discretization methods are the finite difference and the finite element methods (see [1], [3]). In this paper, we use them both. In order to do this, as in [2], [4] and [5], the domain is divided in rectangular subdomains, having the same step on both $Ox$ and $Oy$ directions. The solution of the systems generated through discretization is obtained by Gauss full elimination method. The first level on wich the solution is computed is denoted by $l_0$, then this particular solution is used for generating the solutions on higher order levels, according with [4] and [5]. The grid on the $l$ level $l$ is divided by the one from the $l_0$ level in subdomains. On each of these, the system of linear equations obtained through the discretization method will be solved.

## 2  The problems

**1. Convection-diffusion equations**

It is known that the general expression of a convection-diffusion problem in two dimensions is given by:

$$\begin{cases} m\Delta u + \mathbf{n} \bigtriangledown u = f, & (x,y) \in \Omega, \\ u = g, & (x,y) \in \partial\Omega, \end{cases}$$

where $\mathbf{n} = (n_1, n_2)$ a flow velocity field and $m$ is the coefficient of diffusion or viscosity.

As an example, let's consider the following problem:

$$\begin{cases} -e\Delta u + au_x = f, & (x,y) \in (0,1) \times (0,1) = \Omega, \\ u = 0, & (x,y) \in \partial\Omega, \end{cases} \tag{1}$$

where $f = x(1-x)\sin qy$.

The exact solution in this case can be determined analytically and has the expression:

$u_{ex} = \left(Ax^2 + Bx + C\right)\sin qy$, with $A = \frac{1}{t}, B = \frac{1-2aA}{t}, C = \frac{2eA-aB}{t}, t = eq^2$.

**2. Poisson's equation**

The second problem we consider is given by a Poisson equation of the form:

$$\begin{cases} -\Delta u = f, & (x,y) \in \Omega, \\ u = g, & (x,y) \in \partial\Omega. \end{cases} \tag{2}$$

**Remark.** Even if the exact solutions of these two problems are relatively easy to be determined, in the following paragraphs we shall compute also their numerical approximations, in order to introduce our prolongation method and establish its efficiency.

# 3 Finite difference and finite element discretizations

The partial differential equations will be replaced by a liniar system of equations through the discretization methods such as finite difference and finite element discretization.

In order to achieve this, and keeping the notations used in [4], we choose a grid step $h_l = \frac{1}{2^{l+1}}$, $l$ being the number of the level. The corresponding number of grid points is $n_l = 2^{l+1}-1$ on each direction. The grid that has a step $h_l = \frac{1}{n_l+1} = \frac{1}{2^{l+1}}$ will contain the points $(x_i, y_j), i, j = 1, 2, \ldots, n_l$, and will be denoted by $G_l$. The value of the exact solution in the point $(x_i, y_j)$ is denoted by $u_{i,j}$.

**Remark.** The numerical solution together with all discretizations involved are made for the convection-diffusion equation (1), because the Poisson's equation (2) is obtained by replacing parameter $a$ by zero.

## 3.1 Second order finite difference discretization

Expanding in Taylor series the values of the function in the grid points, as in [4], one can compute approximations of the derivatives from the differential equation:

$$u_{i+1,j} = u_{i,j} + \frac{h}{1!}\frac{\partial u}{\partial x}(x_i, y_j) + \frac{h^2}{2!}\frac{\partial^2 u}{\partial x^2}(x_i, y_j) + \frac{h^3}{3!}\frac{\partial^3 u}{\partial x^3}(x_i, y_j) + ...; \tag{3}$$

$$u_{i-1,j} = u_{i,j} - \frac{h}{1!}\frac{\partial u}{\partial x}(x_i, y_j) + \frac{h^2}{2!}\frac{\partial^2 u}{\partial x^2}(x_i, y_j) - \frac{h^3}{3!}\frac{\partial^3 u}{\partial x^3}(x_i, y_j) + .... \tag{4}$$

The approximation for the second order partial derivative is then:

$$\frac{\partial^2 u}{\partial x^2}(x_i, y_j) = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + O(h^2). \tag{5}$$

An analogous result holds for the $y$-direction derivative:

$$\frac{\partial^2 u}{\partial y^2}(x_i, y_j) = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2} + O(h^2). \tag{6}$$

The approximation for the first order partial derivative in the $x$-direction is :

$$\frac{\partial u}{\partial x}(x_i, y_j) = \frac{u_{i+1,j} - u_{i-1,j}}{2h} + O(h^2). \tag{7}$$

Equation (1) will have the following discrete formulation:

$$\begin{cases} -e\left(\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2}\right) + a\frac{u_{i+1,j} - u_{i-1,j}}{2h} = f_{i,j}, \\ \qquad\qquad\qquad\qquad\qquad\qquad i, j = 1, 2, ..., n_l + 1, \\ \\ \qquad u_{i,j} = 0, \qquad i \in \{0, n_l + 2\} \text{ or } j \in \{0, n_l + 2\}. \end{cases} \tag{8}$$

The difference equations above are abbreviated by the stencil notation:

$$\frac{1}{h^2} \left[\!\!\left[ \begin{array}{ccc} & -e & \\ -e - \frac{ah}{2} & 4e & -e + \frac{ah}{2} \\ & -e & \end{array} \right]\!\!\right] u = f, \tag{9}$$

where :

$$\left[\!\!\left[ \begin{array}{ccc} a & b & c \\ d & e & f \\ g & h & k \end{array} \right]\!\!\right] u(i,j) = \begin{aligned} & au(i-1, j+1) + bu(i, j+1) + cu(i+1, j+1) + \\ & + du(i-1, j) + eu(i, j) + fu(i+1, j) + gu(i-1, j-1) \\ & + hu(i, j-1) + ku(i+1, j-1). \end{aligned} \tag{10}$$

## 3.2 Finite element discretization

According to [1], in order to apply the finite element discretization, some transformations of the given equation have to be made. So, the equation to be discretized is multiplied by a test function $v$, then is integrated on the domain $\Omega$:

$$-e \iint_\Omega v\Delta u\, dxdy + a \iint_\Omega v\frac{\partial u}{\partial x} dxdy = \iint_\Omega fv\, dxdy.$$

Using Green's formula, the equation above becomes :

$$e \iint_\Omega \bigtriangledown u \bigtriangledown v\, dxdy - e \int_{\partial\Omega} v\frac{\partial u}{\partial n} ds + a \iint_\Omega v\frac{\partial u}{\partial x} dxdy = \iint_\Omega fv\, dxdy,$$

$$u, v \in H^1(\Omega), \iint_\Omega \left[\left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 + v^2\right] dxdy < \infty. \tag{11}$$

The functions $u$ and $v$ are approximated using some continuous functions, $\Phi_i$ ($\Phi_i(x_j, y_j) = \delta_{ij}, i, j = 1, ..., N$, $N = n_l^2$ being the number of interior points of the grid on level $l$), through the relations: $u \approx \sum_{i=1}^{N} u_i\Phi_i, v \approx \sum_{j=1}^{N} v_j\Phi_j$, where $u_i = u(x_i, y_i), i = 1, ..., N$. Replacing these approximations in equation (11), the system obtained is:

$$\sum_{j=1}^{N} K_{ij}u_j = F_i, i = 1, ..., N, \tag{12}$$

where:

$$K_{ij} = \iint_\Omega \left[e\left(\frac{\partial\Phi_i}{\partial x}\frac{\partial\Phi_j}{\partial x} + \frac{\partial\Phi_i}{\partial y}\frac{\partial\Phi_j}{\partial y}\right) + a\Phi_i\frac{\partial\Phi_j}{\partial x}\right] dxdy, \tag{13}$$

$$F_i = \iint_\Omega f\Phi_i\, dxdy. \tag{14}$$

As $\Omega = \bigcup_{i=1}^{(n_l+1)^2}\Omega_i$, the above integrals can be rewritten as sums of integrals on the interior domains. Because the functions $\Phi_i$ are different from zero only on the subdomains immediately next to the point $(x_i, y_i)$, the sums corresponding to the node $i$ in the system (23) will only contain the integrals on the $\Omega_A$, $\Omega_B$, $\Omega_C$ and $\Omega_D$ domains:
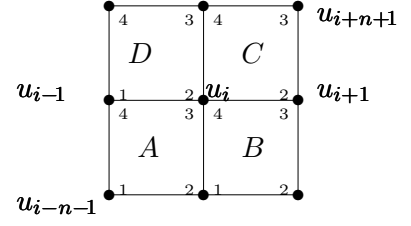


Fig. 1.

$$K_{i,i-n-1}^{(A)}u_{i-n-1} + K_{i,i-n}^{(A+B)}u_{i-n} + K_{i,i-n+1}^{(B)}u_{i-n+1} + K_{i,i-1}^{(A+D)}u_{i-1} + K_{i,i}^{(A+B+C+D)}u_i +$$
$$+K_{i,i+1}^{(B+C)}u_{i+1} + K_{i,i+n-1}^{(D)}u_{i+n-1} + K_{i,i+n}^{(C+D)}u_{i+n} + K_{i,i+n+1}^{(C)}u_{i+n+1} = F_i^{(A+B+C+D)},$$
$$i = 1, ..., N. \quad (15)$$

In the equation (15) the superscript (A) or (A+B) means that the corresponding integrals in (13) and (14) are computed on $\Omega_A$ or $\Omega_A \cup \Omega_B$.

Further we denote the restrictions on $\Omega_A$ with $\Psi_3^A$ for $\Phi_i$, $\Psi_4^A$ for $\Phi_{i-1}$, $\Psi_1^A$ for $\Phi_{i-n-1}$ and $\Psi_2^A$ for $\Phi_{i-n}$. If the domain $\Omega_A$ is $[a, b] \times [c, d]$, then the following expresions can be obtained:

$$\Psi_1^A = \frac{(x-b)(y-d)}{(b-a)(d-c)}, \Psi_2^A = -\frac{(x-a)(y-d)}{(b-a)(d-c)}, \Psi_3^A = \frac{(x-a)(y-c)}{(b-a)(d-c)}, \Psi_4^A = -\frac{(x-b)(y-c)}{(b-a)(d-c)}$$
$$\left(\Psi_k^A = \epsilon\frac{(x-\alpha)(y-\beta)}{(b-a)(d-c)}, k = 1, ..., 4\right). \quad (16)$$

The restrictions of $K$ and $F$ on a domain $\Omega_A$ are:

$$k_{ij}^A = \iint_{\Omega_A}\left[e\left(\frac{\partial\Psi_i^A}{\partial x}\frac{\partial\Psi_j^A}{\partial x} + \frac{\partial\Psi_i^A}{\partial y}\frac{\partial\Psi_j^A}{\partial y}\right) + a\Psi_i^A\frac{\partial\Psi_j^A}{\partial x}\right]dxdy, \quad i, j = 1, ..., 4. \quad (17)$$

$$f_k^A = \iint_{\Omega_A} f(x, y)\Psi_k^A(x, y)dxdy, \quad k = 1, ..., 4, \quad (18)$$

With these notations, in equation (15) all the coefficients can be determined like in the following model:

$$K_{i,i-n}^{(A+B)} = \iint_{\Omega_A\cup\Omega_B}\left[e\left(\frac{\partial\Phi_i}{\partial x}\frac{\partial\Phi_{i-n}}{\partial x} + \frac{\partial\Phi_i}{\partial y}\frac{\partial\Phi_{i-n}}{\partial y}\right) + a\Phi_i\frac{\partial\Phi_{i-n}}{\partial x}\right]dxdy =$$
$$= \iint_{\Omega_A}\left[e\left(\frac{\partial\Phi_i}{\partial x}\frac{\partial\Phi_{i-n}}{\partial x} + \frac{\partial\Phi_i}{\partial y}\frac{\partial\Phi_{i-n}}{\partial y}\right) + a\Phi_i\frac{\partial\Phi_{i-n}}{\partial x}\right]dxdy +$$
$$+ \iint_{\Omega_B}\left[e\left(\frac{\partial\Phi_i}{\partial x}\frac{\partial\Phi_{i-n}}{\partial x} + \frac{\partial\Phi_i}{\partial y}\frac{\partial\Phi_{i-n}}{\partial y}\right) + a\Phi_i\frac{\partial\Phi_{i-n}}{\partial x}\right]dxdy =$$
$$= \iint_{\Omega_A}\left[e\left(\frac{\partial\Psi_3^A}{\partial x}\frac{\partial\Psi_2^A}{\partial x} + \frac{\partial\Psi_3^A}{\partial y}\frac{\partial\Psi_2^A}{\partial y}\right) + a\Psi_3^A\frac{\partial\Psi_2^A}{\partial x}\right]dxdy +$$
$$+ \iint_{\Omega_B}\left[e\left(\frac{\partial\Psi_4^B}{\partial x}\frac{\partial\Psi_1^B}{\partial x} + \frac{\partial\Psi_4^B}{\partial y}\frac{\partial\Psi_1^B}{\partial y}\right) + a\Psi_4^B\frac{\partial\Psi_1^B}{\partial x}\right]dxdy = k_{32}^A + k_{41}^B. \quad (19)$$

Thus the equation (15) can be rewritten as:

$$\left[\left[\begin{array}{ccc} k_{24}^D & k_{23}^D + k_{14}^C & k_{13}^C \\ k_{34}^A + k_{21}^D & k_{33}^A + k_{44}^B + k_{11}^C + k_{22}^D & k_{43}^B + k_{12}^C \\ k_{31}^A & k_{32}^A + k_{41}^B & k_{42}^B \end{array}\right]\right] u_i = f_3^A + f_4^B + f_1^C + f_2^D, i=1,...,N.(20)$$

If $x = \frac{d-c}{b-a}$, $y = \frac{d-c}{h_0}$, and replacing (16) in (17),(18), the values on $\Omega_A$ for the model problem (1) are as follows:

$$\left(k_{ij}^A\right)_{i,j=\overline{1:4}} = \frac{e}{6}\left[\left[\begin{array}{cccc} 2x+\frac{2}{x} & -2x+\frac{1}{x} & -x-\frac{1}{x} & x-\frac{2}{x} \\ -2x+\frac{1}{x} & 2x+\frac{2}{x} & x-\frac{2}{x} & -x-\frac{1}{x} \\ -x-\frac{1}{x} & x-\frac{2}{x} & 2x+\frac{2}{x} & -2x+\frac{1}{x} \\ x-\frac{2}{x} & -x-\frac{1}{x} & -2x+\frac{1}{x} & 2x+\frac{2}{x} \end{array}\right]\right] + y\frac{ah_0}{12}\left[\left[\begin{array}{cccc} -2 & 2 & 1 & -1 \\ -2 & 2 & 1 & -1 \\ -1 & 1 & 2 & -2 \\ -1 & 1 & 2 & -2 \end{array}\right]\right] (21)$$

$$f_k^A = \frac{\epsilon}{(b-a)(d-c)}\left(\frac{x^4}{4} - (1+\alpha)\frac{x^3}{3} + \alpha\frac{x^2}{2}\right)\Big|_a^b \left(\frac{(y-\beta)\cos qy}{q} - \frac{\sin qy}{q^2}\right)\Big|_c^d. \qquad (22)$$

# 4 Prolongation methods

The systems of equations (9) or (20) generated in the previous section can be written on any level $l$. Each system contains $n_l^2$ unknowns. The solution is exactly computed on a level $l_0$, for example on $l_0=2$ or $l_0=3$ using Gauss elimination method with partial pivoting. Thus the exact solution on the level $l_0$, for the problem is approximated by $u_i, i \in \{1, 2, ..., n_{l_0}^2\}$ (Fig. 2.), wich only contains an error due to the discretization.

In order to solve problem on the level $l$, the grid already obtained has to be further divided. Thus, each domain from the grid, $\Omega_k, k = 1, ..., (n_0 + 1)^2$, will be splitted into $(n_i + 1)^2$ subdomains, where $n_i = 2^{l_i+1}-1$, and $l_i = l-l_0-1$. On each subdomain



Fig. 2.

$\Omega_k$, the discretization of the differential equation leads to a system whose matrix has the same form as the one on $l_0$ level. But on the level $l_0$ the boundary values were given in the hypothesis. For the systems on the level $l$ to be precisely solved on $\Omega_k$, one has to determine as accurate as possible the $n_i$ interior values on each of the sides of the domain $\Omega_k$. Two possible ways to accomplish this are given in the following subsections.
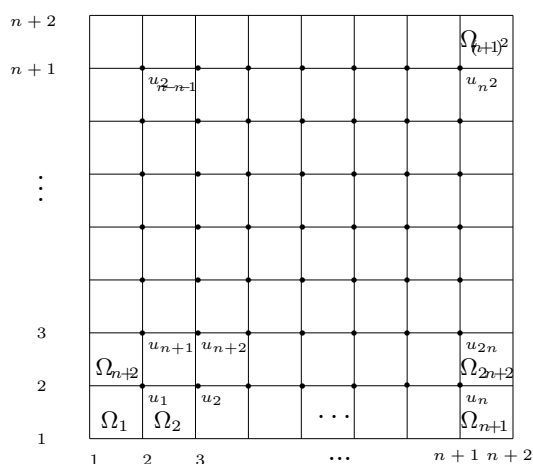
## 4.1 Pondered arithmetic mean prolongation

As in [2], the value of the approximation on level $l$ is denoted by $u^{(l)}$. On the borders of $\Omega_k$, they are defined through the following relations ($n = n_{l_0}$, $n_i = n_{l_i}$, $l_i = l-l_0-1$, $N=n_i+1$):

$u_{jN+1,iN+1}^{(l)} = u_{(i-1)n+j}^{(l_0)}, i = 0,..,n, j = 1,..,n$ for the common points of the grids $G_{l_0}$ and $G_l$.
For the grid points of $G_l$ that do not belong to $G_{l_0}$:

$u_{jN+1,iN+1+k}^{(l)} = \frac{1}{N}\left(ku_{in+j}^{(l_0)} + (N-k)u_{(i-1)n+j}^{(l_0)}\right), i = 0,...,n, j = 1,...,n;$

$u_{jN+1+k,iN+1}^{(l)} = \frac{1}{N}\left(ku_{(i-1)n+j+1}^{(l_0)} + (N-k)u_{(i-1)n+j}^{(l_0)}\right), i = 1,...,n, j = 0,...,n, k = 1,...,n_i.$

## 4.2   Stellar prolongation

In what follows, we introduce a new type of prolongation which we call "stellar prolongation" because the nodes involved in computation are in the shape of a star.

In order to determine more accurately the values of the solution on the borders of $\Omega_k$, instead of pondered arithmetic mean prolongation one can use the solutions of the systems obtained discretizing the initial equation in the grid points corresponding to the values $a_i$ and $b_i$, $i = 1, 2, ..., n^2 + n, n = n_{l_0}$ from Fig. 4 and Fig. 5.
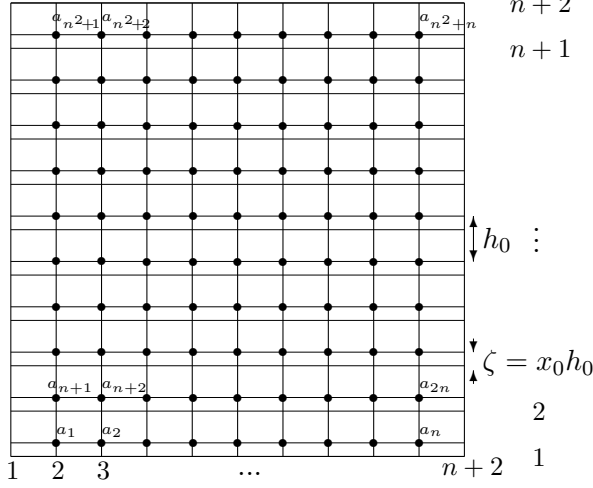


Fig. 4.
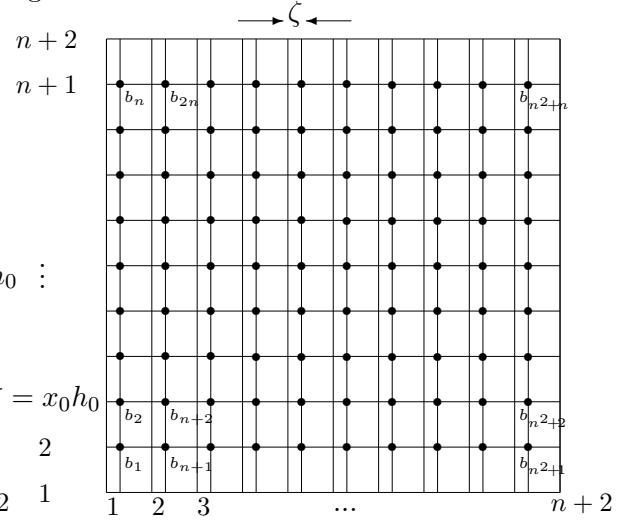


Fig. 5.

### 4.2.1   Finite difference discretization

The values $a_k, k = 1, 2, ..., n_0(n_0 + 1)$ depend on their vertical distance, $\zeta$, from the old grid $G_0$ (marked with the thin lines in Fig.4) and will be further denoted by $a_k(\zeta)$, $\zeta = jh, j = 1, ..., n_i$. They are the solutions of the following system:

$$Aa = T \tag{23}$$

where the matrix $A$ is:

$$A = \begin{bmatrix} C & D & \Theta & ... & \Theta & \Theta \\ S & C & D & ... & \Theta & \Theta \\ \Theta & S & C & ... & \Theta & \Theta \\ \vdots & & & \ddots & & \\ \vdots & & & & \ddots & \\ \Theta & \Theta & \Theta & ... & S & C \end{bmatrix}, \tag{24}$$

$$C = \begin{bmatrix} q_c & q_r & 0 & ... & 0 \\ q_l & q_c & q_r & ... & 0 \\ 0 & q_l & q_c & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_c \end{bmatrix}, D = \begin{bmatrix} q_u & 0 & 0 & ... & 0 \\ 0 & q_u & 0 & ... & 0 \\ 0 & 0 & q_u & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_u \end{bmatrix}, S = \begin{bmatrix} q_d & 0 & 0 & ... & 0 \\ 0 & q_d & 0 & ... & 0 \\ 0 & 0 & q_d & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_d \end{bmatrix},$$
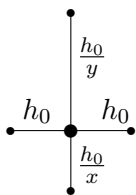
For the first and the last line of blocks in $A$, the neighbours of the node where the discretization is made are illustrated in Fig. 6. For the first line: $x = \frac{h_0}{\zeta}$, $y = 1$, for the last one: $x = 1$, $y = \frac{h_0}{h_0 - \zeta}$, whereas for the remaining lines: $x = 1$ and $y = 1$.

Fig. 6.

$$\left[\!\left[ \begin{array}{ccc} & q_u & \\ q_l & q_c & q_r \\ & q_d & \end{array} \right]\!\right] = \left[\!\left[ \begin{array}{ccc} & -e\alpha y & \\ -e - \frac{ah_0}{2} & e[2 + \alpha(x+y)] & -e + \frac{ah_0}{2} \\ & -e\alpha x & \end{array} \right]\!\right].$$

The vector of constant terms, $T$, has the components:

$$t_{in+j} = h_0^2 f\left(jh_0, (ih_0 + \zeta)\right) - \left[\!\left[ \begin{array}{ccc} & q_u & \\ q_l & q_c & q_r \\ & q_d & \end{array} \right]\!\right] u_{fr}\left(jh_0, (ih_0 + \zeta)\right),$$

$$i = 0, ..., n_0, j = 1, ..., n_0,$$

$u_{fr}$ being a function which is zero inside the domain $\Omega$ on wich the system is solved and equal to the border values on $\partial\Omega$ and $h_0$ is the grid step on $l_0$ level.

According to the kind of discretization that is used, the values of the parameter $\alpha$ are: $\alpha = \frac{xy}{x+y}$ (symmetric finite differences), $\alpha = x$ (backward finite differences), $\alpha = y$ (forward finite differences).
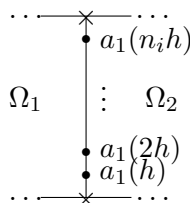
Fig. 7.

As $\zeta$ takes the values $h, 2h, ..., n_i h$, the values $a_k(\zeta)$ obtained from the system (23) will be used as border data on the vertical sides of $\Omega_k$ (for example, on the right vertical side of $\Omega_1$ and the left side for $\Omega_2$, they are corresponding to the points marked with a dot in Fig.7).

The values $b_k(\zeta), k = 1, 2, ..., n_0(n_0 + 1)$ depend on their horizontal pozition, $\zeta$ (see Fig.5) and are computed by solving a system whose matrix is also of the form (24), but in which:

$$C = \begin{bmatrix} q_c & q_u & 0 & ... & 0 \\ q_d & q_c & q_u & ... & 0 \\ 0 & q_d & q_c & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_c \end{bmatrix}, D = \begin{bmatrix} q_r & 0 & 0 & ... & 0 \\ 0 & q_r & 0 & ... & 0 \\ 0 & 0 & q_r & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_r \end{bmatrix}, S = \begin{bmatrix} q_l & 0 & 0 & ... & 0 \\ 0 & q_l & 0 & ... & 0 \\ 0 & 0 & q_l & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & q_l \end{bmatrix},$$
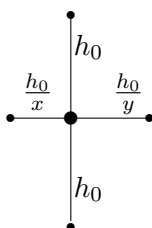
For the first and the last line of blocks in $B$, the nodes involved in the discretization are illustrated in Fig. 8. For the first line $x = \frac{h_0}{\zeta}$, $y = 1$, for the last one $x = 1$, $y = \frac{h_0}{h_0 - \zeta}$, and for the remaining lines: $x = 1$ and $y = 1$.

Fig. 8.

$$\left[\!\left[ \begin{array}{ccc} & q_u & \\ q_l & q_c & q_r \\ & q_d & \end{array} \right]\!\right] = \left[\!\left[ \begin{array}{ccc} & -e & \\ -e\alpha x + ah_0\rho\delta & e[2 + \alpha(x+y)] + a\gamma\rho & e\alpha y + ah_0\rho\beta \\ & -e & \end{array} \right]\!\right].$$

The constant terms vector now has the components:

$$t_{in+j} = h_0^2 f\left((ih_0 + \zeta), jh_0\right) - \left[\!\!\left[\begin{array}{ccc} & q_u & \\ q_l & q_c & q_r \\ & q_d & \end{array}\right]\!\!\right] u_{fr}\left((ih_0 + \zeta), jh_0\right),$$

$$i = 0, ..., n_0, j = 1, ..., n_0.$$
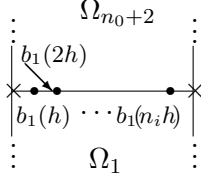


Fig. 9.

The values $b_k(\zeta), \zeta = 1, ..., n_i$ obtained from the system (23), with the matrix components described above, will be used as border data on the horizontal sides of $\Omega_k$ (for example, on the lower horizontal side of $\Omega_{n_0+2}$ and the upper side for $\Omega_1$, they are corresponding to the points marked with a dot in Fig.9).

### 4.2.2   Finite element discretization

If the discretization is made by the finite element method, the same computing method is used, the only changes are in the matrices components. If we denote:

$$\left[\!\!\left[\begin{array}{ccc} l_1 & l_2 & l_3 \\ l_4 & l_5 & l_6 \\ l_7 & l_8 & l_9 \end{array}\right]\!\!\right] = \left[\!\!\left[\begin{array}{ccc} k_{24}^D & k_{23}^D + k_{14}^C & k_{13}^C \\ k_{34}^A + k_{21}^D & k_{33}^A + k_{44}^B + k_{11}^C + k_{22}^D & k_{43}^B + k_{12}^C \\ k_{31}^A & k_{32}^A + k_{41}^B & k_{42}^B \end{array}\right]\!\!\right],$$

where $k_{ij}$ is given by (21), than the matrix $A$ has:

$$C = \begin{bmatrix} l_5 & l_6 & 0 & ... & 0 \\ l_4 & l_5 & l_6 & ... & 0 \\ 0 & l_4 & l_5 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_5 \end{bmatrix}, D = \begin{bmatrix} l_2 & l_3 & 0 & ... & 0 \\ l_1 & l_2 & l_3 & ... & 0 \\ 0 & l_1 & l_2 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_2 \end{bmatrix}, S = \begin{bmatrix} l_8 & l_9 & 0 & ... & 0 \\ l_7 & l_8 & l_9 & ... & 0 \\ 0 & l_7 & l_8 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_8 \end{bmatrix},$$

and the components of the constant terms vector:

$$t_{in+j} = f_3^A + f_4^B + f_1^C + f^D - \left[\!\!\left[\begin{array}{ccc} l_1 & l_2 & l_3 \\ l_4 & l_5 & l_6 \\ l_7 & l_8 & l_9 \end{array}\right]\!\!\right] u_{fr}\left(jh, (i + x_0)h\right),$$

$$i = 0, ..., n, j = 1, ..., n.$$

For the $B$ matrix:

$$C = \begin{bmatrix} l_5 & l_2 & 0 & ... & 0 \\ l_8 & l_5 & l_2 & ... & 0 \\ 0 & l_8 & l_5 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_5 \end{bmatrix}, D = \begin{bmatrix} l_6 & l_3 & 0 & ... & 0 \\ l_9 & l_6 & l_3 & ... & 0 \\ 0 & l_9 & l_6 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_6 \end{bmatrix}, S = \begin{bmatrix} l_4 & l_1 & 0 & ... & 0 \\ l_7 & l_4 & l_1 & ... & 0 \\ 0 & l_7 & l_4 & ... & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & ... & l_4 \end{bmatrix}.$$

The components of the constant terms vector:

$$t_{in+j} = f_3^A + f_4^B + f_1^C + f_2^D - \left[\!\!\left[\begin{array}{ccc} l_1 & l_2 & l_3 \\ l_4 & l_5 & l_6 \\ l_7 & l_8 & l_9 \end{array}\right]\!\!\right] u_{fr}\left((i+x_0)\,h, jh\right),$$

$$i = 0, ..., n, j = 1, ..., n,$$

In the matrix $A$: $x = x_0$ and $y = x_0$ for the first line of blocks in (21), on $\Omega_A$ and $\Omega_B$, $x = 1$ and $y = 1$ on $\Omega_C$ and $\Omega_D$, while $x = 1$ and $y = 1$ for the last line of blocks on $\Omega_A$ and $\Omega_B$, and $x = 1 - x_0$, $y = 1 - x_0$ on $\Omega_C$ and $\Omega_D$. For the remainder of the lines: $x = 1$ and $y = 1$.

For the matrix $B$: $x = \frac{1}{x_0}$ and $y = 1$ for the first line of blocks on $\Omega_A$ and $\Omega_D$, $x = 1$ and $y = 1$ on $\Omega_B$ and $\Omega_C$. The last line has: $x = 1$ and $y = 1$ on $\Omega_A$ and $\Omega_D$, and on $\Omega_B$ and $\Omega_C$ $x = \frac{1}{1-x_0}$ and $y = 1$. For the other lines: $x = 1$ and $y = 1$.

## 5 Solving method

The differential equation is first discretized on a grid $G_0$ and the solutions obtained solving the system resulted are the values $u_i, i = 1, ..., n_0^2$ situated in the corners of the subdomains $\Omega_k$, $k = 1, ..., n_0(n_0 + 1)$ (Fig. 2 ).

These values are then used to compute $a_k$ and $b_k, k = 1, ..., n_0(n_0 + 1)$ for each $\zeta = jh$, $j = 1, ..., n_i$ (as in Section 4.1 or 4.2).

Thus, on every subdomain $\Omega_{iN_0+j}$, $i=0, ..., n_0, j=1, ..., n_0, N_0=n_0+1$ the values on the frontiers are now known:



Fig. 10.

$$u^{(l)}_{(j-1)N+1,iN+1+k} = a_{in_0+j-1}(kh), u^{(l)}_{(j-1)N+1+k,iN+1} = b_{(j-1)n_0+i}(kh),$$

$k = 1, ..., n_0(n_0 + 1), j = 1, ..., n_i, h = \frac{1}{n_l+1}$ (see Fig. 10).

The problem is now discretized the on each subdomain $\Omega_{iN_0+j}, i = 0, ..., n_0, j = 1, ..., n_0$ and the solution obtained will represent the components of the final solution on the grid $G_l$.

Reuniting the solutions computed on the grid corresponding to the level $l_0$ and the ones from every subdomain, the final solution on the work level $l$ is obtained.

## 6 Numerical results

In order to give some numerical results, we denote by:
FD-PAM: the finite difference discretization with pondered arithmetic mean prolongation,
FD-SP: the finite difference discretization with stellar prolongation,
FEM-SP: the finite element discretization with stellar prolongation.

We have computed the infinity norm of the difference between the computed solution and the exact solution. If the grid on the level $l$ is $G_l = \{(x_i, y_j), i, j = 1, 2, ..., n_l\}$, then the error is:

$$\xi = ||u_{ex} - u_l||_\infty = max\{|u_{ex}(x,y) - u_l(x,y)|, (x,y) \in G_l\}.$$
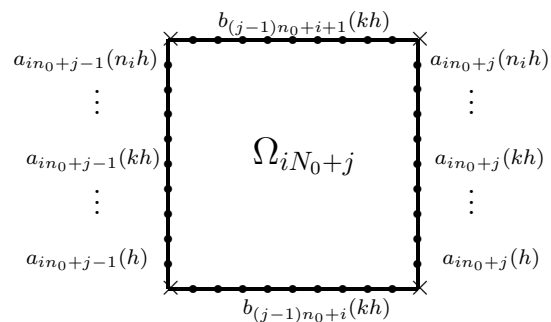
In the following table there are the errors for the convection-diffusion problem (1) ($u_{ex} \leq 1.7839 \cdot 10^4$).

| Level | $\xi_{FD-PAM}$ | $\xi_{FD-SP}$ | $\xi_{FEM-SP}$ |
|---|---|---|---|
| $l=3$ | 32.8788 | 25.3345 | 0.7798 |
| $l=4$ | 32.8788 | 25.3345 | 0.7877 |
| $l=5$ | 32.8788 | 25.3345 | 0.7877 |

In order to compare the previous methods applied on a Poisson problem, we consider the following problems of this type, and their exact solutions.

**Pr.1**
$$\begin{cases} -\Delta u = -4, & (x,y) \in (0,1) \times (0,1), \\ u(x,y) = x^2 + y^2 = u_{ex}(x,y), & (x,y) \in \partial\Omega. \end{cases}$$

**Pr.2**
$$\begin{cases} -\Delta u = -\frac{3y}{x+1} - \frac{y^3}{(x+1)^3}, & (x,y) \in (0,1) \times (0,1), \\ u(x,y) = 0.5\frac{y^3}{x+1} = u_{ex}(x,y), & (x,y) \in \partial\Omega. \end{cases}$$

**Pr.3**
$$\begin{cases} -\Delta u = -\frac{3y}{x+0.1} - \frac{y^3}{(x+0.1)^3}, & (x,y) \in (0,1) \times (0,1), \\ u(x,y) = 0.5\frac{y^3}{x+0.1} = u_{ex}(x,y), & (x,y) \in \partial\Omega. \end{cases}$$

**Pr.4**
$$\begin{cases} -\Delta u = 2\pi^2 \sin \pi x \cos \pi y, & (x,y) \in (0,4) \times (0,1), \\ u(x,y) = 0, & (x,y) \in \partial\Omega; \end{cases}$$
$$u_{ex}(x,y) = \sin \pi x \cos \pi y.$$

**Pr.5**
$$\begin{cases} -\Delta u = \alpha \sin \frac{\pi y}{b}, & (x,y) \in [0,\lambda] \times [0,b], \\ u(x,y) = 0, & (x,y) \in \partial\Omega. \end{cases}$$
$$u_{ex}(x,y) = -\alpha \left(\frac{b}{\pi}\right)^2 \sin \frac{\pi y}{b} \left(e^{\frac{\pi x}{b}} - 1\right),$$
$$\alpha = \frac{F\pi}{Rb}, \lambda = 10^7, b = 2\pi \cdot 10^6, F = 0.3 \cdot 10^{-7}, R = 0.6 \cdot 10^{-3}.$$

| **Pr.1** | $\xi_{FD-PAM}$ | $\xi_{FD-SP}$ | $\xi_{FEM-SP}$ | **Pr.2** | $\xi_{FD-PAM}$ | $\xi_{FD-SP}$ | $\xi_{FEM-SP}$ |
|---|---|---|---|---|---|---|---|
| $l=3$ | $3.9 \cdot 10^{-03}$ | $3.3 \cdot 10^{-03}$ | 0.5968 | $l=3$ | $4.9075 \cdot 10^{-3}$ | $3.4025 \cdot 10^{-3}$ | $2.2406 \cdot 10^{-4}$ |
| $l=4$ | $3.4 \cdot 10^{-03}$ | $3.3 \cdot 10^{-03}$ | 0.5968 | $l=4$ | $4.9075 \cdot 10^{-3}$ | $3.4025 \cdot 10^{-3}$ | $2.2406 \cdot 10^{-4}$ |
| $l=5$ | $3.5 \cdot 10^{-03}$ | $3.3 \cdot 10^{-03}$ | 0.5967 | $l=5$ | $4.9075 \cdot 10^{-3}$ | $3.4428 \cdot 10^{-3}$ | $2.2406 \cdot 10^{-4}$ |
| **Pr.3** | $\xi_{FD-PAM}$ | $\xi_{FD-SP}$ | $\xi_{FEM-SP}$ | **Pr.4** | $\xi_{FD-PAM}$ | $\xi_{FD-SP}$ | $\xi_{FEM-SP}$ |
| $l=3$ | 0.4407 | 0.1930 | 0.0195 | $l=3$ | 0.0170 | 0.0185 | 0.0042 |
| $l=4$ | 0.4407 | 0.4493 | 0.0195 | $l=4$ | 0.0170 | 0.0185 | 0.0042 |
| $l=5$ | 0.4407 | 0.4600 | 0.0195 | $l=5$ | 0.0170 | 0.0185 | 0.0042 |

| **Pr.5** | $\xi_{FD-PAM}$ | $\xi_{FD-SP}$ | $\xi_{FEM-SP}$ |
|---|---|---|---|
| $l=3$ | $2.4991 \cdot 10^{-20}$ | $1.7752 \cdot 10^{-20}$ | $1.1379 \cdot 10^{-20}$ |
| $l=4$ | $2.4991 \cdot 10^{-20}$ | $1.7752 \cdot 10^{-20}$ | $1.3814 \cdot 10^{-20}$ |
| $l=5$ | $2.5688 \cdot 10^{-20}$ | $1.7752 \cdot 10^{-20}$ | $1.5718 \cdot 10^{-20}$ |

# 7 Conclusions

The numerical results indicate that our stellar prolongation method is more efficient than others used in the literature. Even if we applied it on some particular cases, we expect this behavior

to be the same on other more general problems, too.

## References

[1] E.B. Becker, G.F. Carey, J.T. Oden, *Finite Elements. An Introduction*, Texas Institute for Computational Mecanics, UT Austin, 1981.

[2] A. Brandt, Multilevel Adaptive Solutions to Boundary Value Problems. *Mathematics of Computation*, 31, 333-390, 1977.

[3] W. Hackbush, *Elliptic Differential Equations*, Springer-Verlag, New York, 1992.

[4] W. Hackbush, *Multigrid Method and Applications*, Springer, Berlin, Heidelberg, 1985.

[5] U. Trottenberg, C. Oosterlee, A. Schuller, *Multigrid*, Elsevier Academic Press, London, 2001.

Gabriela Nuț
Babeș-Bolyai University, Cluj-Napoca
Faculty of Mathematics and Computer Science
1, Mihail Kogălniceanu str., Cluj Napoca, 400084
ROMANIA
E-mail: *djantai@yahoo.com*

# Methodological aspects concerning digital libraries for children

**Alina Elena Pitic**

**Abstract**

Digital libraries are nowadays offering a huge quantity of resources in different formats, for a wide spectrum of users. Making some of these resources available to children poses a number of specific challenges, especially when it comes to designing the user interface and to modelling the user interaction. This paper is part of my Ph.D. thesis and describes research aspects concerning digital library interfaces for children between 7 and 12 years old. The adopted methodology involves using the creative input from a number of children, in order to improve the specificity of the application. Children were considered design partners. The results were analysed in terms of user experience and accessibility.

# 1 Digital libraries for children

## 1.1 What is a digital library?

According to [1], a digital library can be defined as: "A potentially virtual organisation, that comprehensively collects, manages and preserves for the long depth of time rich digital content, and offers to its targeted user communities specialised functionality on that content, of defined quality and according to comprehensive codified policies."

In the Romanian scientific literature, the digital library is furthermore defined as: "a library with a similar content to one of written publications, but in which the documents have been digitally stored and can be accessed remotely" [8].

The virtual library is defined as:

- a library whose resources reside in electronic documents
- a medium that is structured by multimedia materials in such a way, as to enable rapid and informed access to the resources.
- a generic term designating all libraries on electronic media [8]

## 1.2 Actual studies on Digital libraries for children

The majority of today's digital libraries were never aimed towards young children. These children would commonly need to build complex text keyword searches were they to access these libraries. Few libraries offer visual tools for browsing the information in addition to querying, so as for young children to be able to explore them intuitively. Children also organize and present the information that they have found differently from adults. [2].

Recent research in technology for children has mostly focused on the resultant technology instead of the children involved in the design process. Within HCI there is some research exploring design processes. However, these works mostly avoid focusing on the impact that the process has on its participants, and rather dwell on the process itself. A few of the papers that report on the process used to design technology informally report benefits to children involved as design partners to technology design processes, such as [3],[4],[5],[6]. The discussion upon the impact that children have if involved in the design process tends to be secondary in these papers to the discussion of the design methods or process itself.

We find an interesting study in [7]. This study describes the children's interest in books, libraries, technology and the world around them by conducting interviews, alongside with collecting drawings and book reviews. The findings from this study includes the increased amount of books that these young people read online; the value they still place in the physical libraries as spaces for social interaction and reading; the increased reading motivation, as well as an increased interest in exploring different cultures.

## 2   Initial case study

In 2009 we commenced our work on a case study on children aged 7 to 12, by utilising a number of questionnaires and software products to test the preferences of students towards user interfaces. The study was performed on a population of 395 children from the county of Sibiu. The age, gender and the living environment breakdown is shown below:
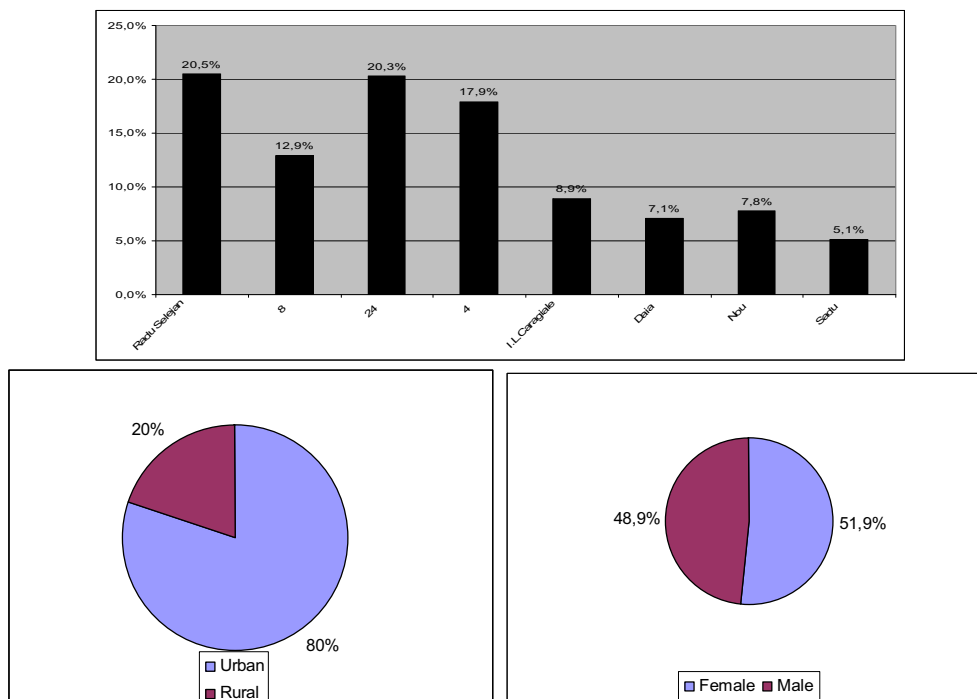


Fig. 1:  Age, gender and living environment breakdown

Furthermore, we found that there is a high level of interest in using computers and working with them.
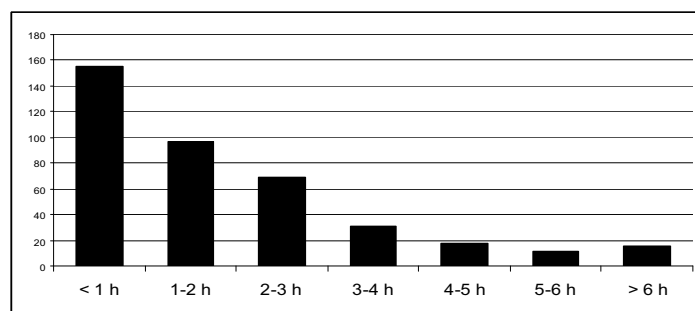


Fig. 2: Time spent working with computers (weekdays)

# 3 A digital library as a 3D interactive world

One of the used applications underwent a number of transformations, as the children provided their input to it. The application seeks to determine whether the use of a digital library leads to an increased reading appetite (both physical and digital media). We will further describe the phases we went through.

## 3.1 Application architecture
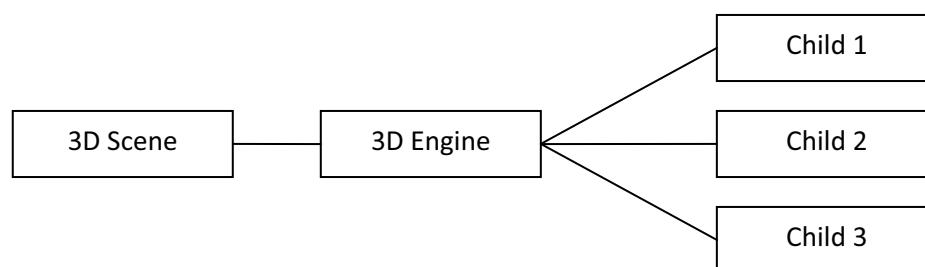


Fig. 3: Initial application architecture

The 3D scene can be built in any 3D editing software and must be saved as an .x file. Our study started with a 3D model of the library of the "Lucian Blaga" University of Sibiu.
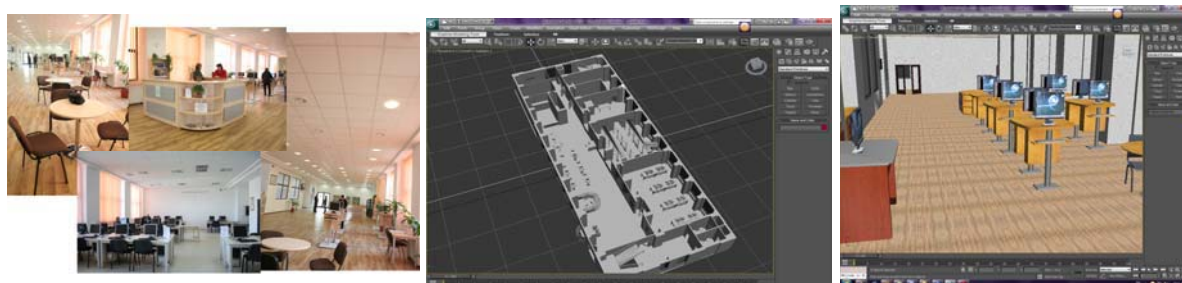


Fig. 4: Building the 3D scene

Our initial data was the 2D plan of the building, as well as pictures of the library. The first step was the modelling of the rooms on all the floors of the building, based on the 2D plans,. The next step was to insert a number of different objects in the scene, like bookshelves, tables and

computers. The final touch was setting the right textures and lightings. In this initial phase the children were not involved in any way. The result was an .x file with a simplified model of the library.

The 3D engine was written in C# and uses the Microsoft XNA library. The initial version of the 3D engine allowed us to:

- load and display a scene saved as a .x file
- rotate the viewpoint
- move through the scene
- define points of interest

## 3.2 Methodology

We decided to involve the children on different stages of the design and implementation.

The first stage involved 12 children of aged 8 to 9. We let the children to explore the 3D Library, using the same test computer and without any supplementary information. From the very beginning, all of them figured out by themselves how to navigate through the  virtual world. There were some interesting results after a mere 10 minutes of testing the virtual world:

- All the children who tested the application were delighted by it and requested a copy of it for home.
- 7 children did not open the books integrated in the environment and preferred only to explore the 3D space.
- All of them entered the virtual toilet.

After several testing sessions, the application was improved step by step:

- it now offers a greater interactivity
- in the 3D environment there is currently a single book, which is split into individual pages. The children receive information as to where to find the next page.

The architecture of the application has become:
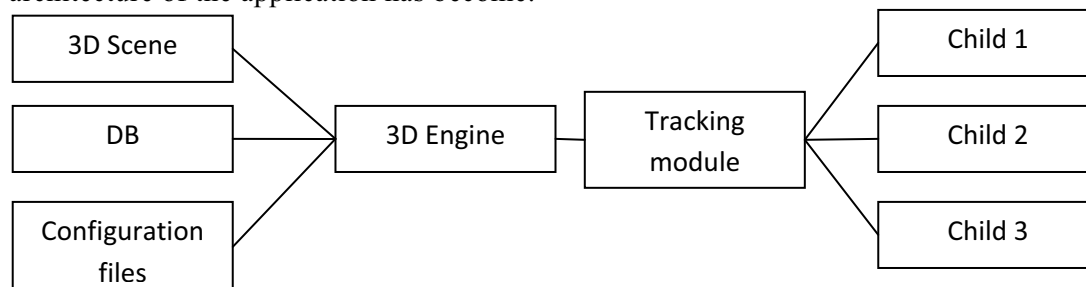


Fig. 5: The final application architecture

A tracking module was introduced in order to retain extra user data, as for example the keyboard use and the time required to complete a certain imposed task.

## 3.3 Results

The evaluation was performed using

a) cognitive evaluation: the children receive the task of going through a document and answering questions regarding what they read
b) pluralist evaluation (directed evaluation)

While working with the children, the application underwent a number of changes. We will mention a few of these:

- The interface was modified 4 times between evaluation rounds
- The interactivity had to be improved
- We came up with the idea of a scenario - a number of steps that have to be performed in order to discover all the pages of a book
- The idea of a reward at the end appeared

There are a few requests from the children that still have t be implemented:

- audio support for books
- information on how many of their colleagues have read the book (a social aspect)

## Conclusions

The designing of an interface for a digital library for children must be done in accordance with the children's needs and with the influence of the educational environment in mind.

The children can be easily involved in designing and using novel applications, as responsibility creates motivation. The next logical step is to develop specialised collaborative tools to help children use and develop interfaces for digital libraries.

Further studies regarding the decisional behaviour of children aged 7 to 12 must be performed in order to establish an optimal pattern of working with and for them.

## References

[1] L. Candela, G. Athanasopoulos, D. Castelli, K. El Raheb, P. Innocenti, Y. Ioannidis, A. Katifori, A. Nika, G. Vullo, S. Ross: *The Digital Library Reference Model*. April 2011, 17

[2] *http://www.cs.umd.edu/hcil/kiddesign/searchkids.shtml*

[3] Druin, A. What *children can teach us: Developing digital libraries for children with children*. Library Quarterly, 75 (1), 2005, 20-41.

[4] 4. Farber, A., Druin, A., Chipman, G., Julian, D. and Somashekhar, S. *How young can our design partners be?* Participatory Design Conference, CSPR, Malmo, Sweden, 2002.

[5] Hourcade, J.P., Beitler, D., Cormenzana, F. and Flores, P. *Early OLPC Experiences in a Rural Uruguayan School*. Proceedings of Computer Human Interaction 2008: Art, Science, Balance. 2503-2511

[6] McElligott, J. and van Leeuwen, L. *Designing sound toys for blind and visually impaired children*. Proceedings of Interaction Design and Children 2004: Building a Community. 65 – 72

[7] Allison Druin, Ann Weeks, Sheri Massey, Benjamin B. Bederson , *Children's Interests and Concerns When Using the International Children's Digital Library: A Four-Country Case Study*, JCDL '07 Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries

[8] Mircea Regneala, *Dictionar explicativ de biblioteconomie si de stiinta informarii*, vol. 1, Ed. a 2-a rev. București : FABR, 2001

PITIC ELENA ALINA
" Lucian Blaga" University of Sibiu
Faculty Of Science, Department of Mathematics and Informatics
Dr. Ion Ratiu street, no 5-7
E-mail: alinap29@yahoo.com; alina.pitic@ulbsibiu.ro

# Several methods of approximation for second order nonlinear boundary value problem with boundary conditions at infinity

**Daniel N. Pop, Radu T. Trîmbiţaş**

**Abstract**

Consider the problem:

$$\begin{cases} y''(x) + f(x,y) = 0, & 0 < x < \infty, \\ \quad y(0) = \infty, & y(\infty) = 0 \end{cases}$$

where $f(x,y) \in C([0,\infty] \times \mathbb{R})$, $y(x) \in C^1(0,\infty)$. This is not a classical two-points boundary value problem since $y(0) = \infty$, $y(\infty) = 0$. To solve this kind of problems we need to know the values in two inner points $a, b \in (0,\infty)$, $a \neq b$. The aim of this work is to present three approximation procedures:

1. A combined method using collocation method on B-splines of order $(k+2)$ with a $(k+1)$ order *Runge-Kutta* method.

2. A pseudospectral collocation method with *Chebychev* extreme points combined with a *Runge-Kutta* method.

3. MATLAB function `bvp4c` combined with a *Runge-Kutta* method.

Then we give a numerical examples and compare the costs (time U.C) using `MATLAB` functions `tic-toc`.

## 1 Introduction

Consider the problem (PVP):

$$y''(x) + f(x,y) = 0, \qquad x \in (0,\infty) \tag{1}$$
$$y(a) = \alpha \tag{2}$$
$$y(b) = \beta, \qquad a, b \in (0,\infty), \ a < b. \tag{3}$$

where $f(x,y) \in C((0,\infty) \times \mathbb{R})$, $a, b, \alpha, \beta \in \mathbb{R}$.

We try to solve this problem using three approximation methods:

1. A combined method based on collocation with B-splines of order $(k+2)$ and a *Runge-Kutta* method order $(k+1)$.

2. A pseudospectral collocation method with *Tchebychev* extreme points combined with a *Runge-Kutta* method.

3. MATLAB function `bvp4c` combined with a *Runge-Kutta* method.

*The methods are new in this context since the conditions are stated at the interior points of the interval*
$(0, 1)$. Problems of type $(1)+(2)+ (3)$ occurs in practice. Examples are in semiclassical description of
the charge density in atoms of high atomic number (Thomas-Fermi equation) [19, pp.155-156], reaction-
diffusion equation [9], frequency domain equation for the vibrating string (Greengard-Rokhlin problem)
[11], electromagnetic self interaction theory [4, pp.336-337], the model of the steady concentration of a
substrate in an enzyme-catalyzed reaction (Michaelis-Menten kinetics)[19, page 145].

We also consider the problem (BVP):

$$y''(x) + f(x, y) = 0, \qquad x \in [a, b] \tag{4}$$
$$y(a) = \alpha \tag{5}$$
$$y(b) = \beta, \tag{6}$$

Also it is shown that the *Runge-Kutta* method does not degrade the accuracy provided by the col-
location method for the (BVP) problem [4, Theorem 5.73 pp219, Theorem 5.140 pp253]. To apply the
collocation theory we need to have an isolated solution of (BVP) problem and this occurs if the above
linearized problem for $y(x)$ is uniquely solvable.

*R.D. Russel* and *L.F. Shampine [18]* study the existence and the uniqueness of the isolated solution.

Our methods consists into decomposition of the problem $(1)+(2)+(3)$ into three problems:

1. A (BVP) problem on $[a, b]$ (problem $(4)+(5)+(6)$).

2. Two (IVPs) on $(0, a]$ and $[b, +\infty)$.

For the existence and uniqueness of an (IVP), see [14, pp: 112-113].

If the problem (BVP) has the unique solution, the requirement $y(x) \in C^1(0, +\infty)$ ensure the existence
and the uniqueness of the solution of the problem (PVP). Our choice to use these methods is based on
the following reasons:

1. We write the code using the function `spcol` in `Matlab-Spline Toolbox` [15] and the functions
   `cebdif, cebint, cebdifft` contained in `dmsuite` [21].

2. Theoretical results on the convergence of collocation method are given in ( [12], [13]).

3. The accuracy of spectral method is superior to finite elements method (FEM) and finite difference
   methods (FDM) (the rate of convergence associated with this problems with smooth conditions are
   $\mathbb{O}(\exp(-CN))$ or $\mathbb{O}(\exp(C \sqrt[2]{N})$ where $N$ is the number of degrees of freedom in the expansions).

4. For each *Newton* iteration, the resulting linear algebraic system of equations (after using *Newton*
   method with quasilinearization) is solved using method given in [8].

## 2 A combined method using B-splines and Runge-Kutta methods

First we solve the (BVP) problem using the collocation method with B-splines of order $(k + 2)$
presented in [17, Section 2].

Consider the mesh of $[a, b]$:
$$\overline{\Delta} : a = x_0 < x_1 < \cdots < x_N = b, \tag{7}$$

where the multiplicity of $a$ and $b$ is $(k + 2)$ and the multiplicity of inner points is $k$. So the dimension
of spline space is $n = Nk + 2$. Also we construct the collocation points $\xi_j$, $j = 1, 2, ..., n - 2$ like in [17,
Section 1] and [2].

We wish to find an approximate solution of the (BVP) problem, having the following form:

$$u_{\overline{\Delta}}(x) = \sum_{i=0}^{n-1} c_i B_{i,k+1}(x), \tag{8}$$

where $B_{i,k+1}(x)$ is the B-spline of order $(k+2)$.

Our approximation method is inspired from [7, Chapter 2,5]. We impose the conditions:

**c1** The approximate solution (8) satisfies the differential equation (4) at collocation points:

$$\xi_j, \ j = 1, 2, \ldots n-2 \ .$$

**c2** The solution satisfies $u_{\overline{\Delta}}(a) = \alpha, \ u_{\overline{\Delta}}(b) = \beta$.

The above conditions yield a nonlinear system with $n$ equations:

$$\begin{cases} \sum_{i=0}^{n-1} c_i B_{i,k+1}(a) = \alpha, \\ \sum_{i=0}^{n-1} c_i B_i^{''},_{k+1}(\xi_j) + f(\xi_j, \sum_{i=0}^{n-1} c_i B_{i,k+1}(\xi_j)) = 0, \ j = 1, 2, \ldots, n-2, \\ \sum_{i=0}^{n-1} c_i B_{i,k+1}(b) = \beta, \end{cases}$$

with unknowns $c_i$, $i = 0, \ldots, n-1$. If $F = [F_0, F_1, \ldots, F_{N-1}]$ are the functions defined by the equations of the nonlinear system, using the quasilinearization of *Newton* method [4, pp: 52-55], we find the next approximation by means of

$$c^{(k+1)} = c^{(k)} - w^{(k)},$$

where $c^{(k)}$ is the vector of unknowns obtained at the $k-$th step and $w^{(k)}$ is the solution of the linear system

$$F'(c^{(k)})w = F(c^{(k)}).$$

To solve the (BVP) problem we use the method presented in [20] and the initial approximation $u^{(0)} \in C^1[0,1]$ is required. The successful stopping criterion [3] is:

$$\left\| u^{(k+1)} - u^{(k)} \right\| \le abstol + \left\| u^{(k+1)} \right\| reltol,$$

where *abstol* and *reltol* is the absolute and the relative error tolerance, respectively and the norm is the usual uniform convergence norm. The reliability of the error-estimation procedure being used for stopping criterion was verified in [8]. For the solution of two IVPs on $(0,a]$ and $[b,+\infty)$ we use a *Runge-Kutta* method of appropriate order, this need good approximation of $y'(a)$ and $y'(b)$, which could be obtained with noadditional effort during the collocation method.

The stability and convergence of *Runge-Kutta* method are guaranteed in [10, Theorem 5.3.1 page 285, Theorem 5.3.2 page 288]. A $(k+1)$ order explicit *Runge-Kutta* method is consistent and stable, so is convergent. The convergence and accuracy of our combined method to whole interval $(0, +\infty)$ was proved in [17, Section 3, Theorem 3.1] and the total costs of this method was studied in [17, Section 4].

## 3 A combined method using a pseudospectral collocation with Tchebychev extreme points and Runge-Kutta methods

Consider the grid:

$$\Delta : 0 = x_{-q} < \ldots < x_{-1} < a = x_0 < x_1 < \ldots < x_N = b < x_{N+1} < \ldots < x_{N+p}. \tag{9}$$

Our second method is a combined a pseudospectral method for the (BVP) problem and a *Runge-Kutta* method for the two IVPs on $(0,a]$ and $[b,+\infty)$. The approximate solution of (BVP) problem follow the ideas presented in [5]. Let $y(x)$ of this problem and considering the *Lagrange* basis $(l_k)$ we have:

$$y(x) = \sum_{k=0}^{N} l_k(x)y(x_k) + (R_N y)(x), \ x \in [a,b]$$

where :

$$(R_N y)(x) = \frac{y^{(N+1)}(\xi)}{(N+1)!}(x - x_0)...(x - x_N)$$

is the remainder of *Lagrange* interpolation. Since $y(x)$ fulfills the differential equation (4) we obtain:

$$\sum_{k=0}^{N} l_k''(x)y(x_k) + (R_N y)''(x) = -f(x_i, y(x_i)), \ i = 1, 2, ..., N-1.$$

Setting $y(x_k) = y_k$ and ignoring the rest, one obtains the nonlinear system:

$$\sum_{k=0}^{N} l_k''(x)y(x_k) = -f(x_i, y(x_i)), \ i = 1, 2, ..., N-1, \tag{10}$$

with unknowns $y_k, \ k = 1, ..., N-1$, here $y_0 = y(a) = \alpha$ and $y_N = y(b) = \beta$. The approximate solution (that is the collocation polynomial for (BVP) problem), is the *Lagrange* interpolation polynomial at nodes $\{x_k\}, \ k = 0, 1, 2, ...N$ :

$$y_N(x) = \sum_{k=0}^{N} l_k(x)y(x_k). \tag{11}$$

The nonlinear system (10) can be rewritten as:

$$AY_N = F(Y_N) + b_N$$

where:

$$A = [a_{ik}], \ a_{ik} = l_k''(x_i), \ k, i = 1, 2, ..., N-1,$$

$$F(Y_N) = \begin{bmatrix} -f(x_1, y_1) \\ -f(x_2, y_2) \\ \vdots \\ -f(x_{N-1}, y_{N-1}) \end{bmatrix}, b_N = \begin{bmatrix} -\alpha l_0''(x_1) - \beta l_N''(x_1) \\ -\alpha l_0''(x_2) - \beta l_N''(x_2) \\ \vdots \\ -\alpha l_0''(x_{N-1}) - \beta l_N''(x_{N-1}) \end{bmatrix}.$$

If the nodes $\{x_k\}, \ k = 0, 1, ...N$ are symmetric with respect of $(a+b)/2$, $A$ is centro-symmetric [6, for proof], so nonsingular. So we choose the nodes given by:

$$x_i = \frac{(b-a)\cos\frac{\pi i}{N} + b + a}{2}, \ i = 1, 2, ..., N, \tag{12}$$

i.e. the Chebyshev-*Lobatto* nodes. We introduce :

$$G(Y) = A^{-1}F(Y) + A^{-1}b_N. \tag{13}$$

To solve numerically (PVP) problem on $\Delta$ given by (9) we apply pseudo-spectral collocation method at points $[a, b]$ and a *Runge-Kutta* method to other points. To apply the *Runge-Kutta* method for the solution of two (IVP) on $(0, a]$ and $[b, +\infty)$ we need the derivatives $y'(a)$ and $y'(b)$, this can be computed by deriving the formula (11). In work [5], the authors prove the existence of unique solution of the system (10) which can be calculated by successive approximation method:

$$Y^{(N+1)} = G(Y^{(N)}), \ n \in N^*,$$

with $Y^{(0)}$ fixed and $G$ given by (13), also they estimate the error:

$$\|Y - Y_N\| \le \frac{\|A^{-1}\| \, \|R\|}{1 - \|A^{-1}\| L}$$

where

$$Y = [y(x_1), y(x_2), ..., y(x_{N-1})]^T ,$$

$y(x)$ is the exact solution of (BVP) problem,

$$Y_N = [y_1, y_2, ..., y_{N-1}]^T ,$$

$y_i$ are the values of approximated solution at $x_i$ computed by (12),

$$R = [-(R_N y)''(x_1), -(R_N y)''(x_2), ..., -(R_N y)''(x_{N-1})]^T ,$$

and $L$ is the *Lipschitz* constant. Combining these results with the stability and convergence of *Runge-Kutta* methods in [16, Theorem 2.3] the authors prove the convergence of this method and occurs:

$$|y_N(a) - y'(a)| = \mathbb{O}(h^k),$$
$$|y_N(b) - y'(b)| = \mathbb{O}(h^k),$$

and for each points $x_i$ in $\Delta$ giving by (9):

$$|y_N(x_i) - y_i| = \mathbb{O}(h^k), \ i = -q, ..., N + p.$$

If the *Runge-Kutta* method is stable and has the order $k$, then the final solution has the same accuracy.

## 4    MATLAB solver `bvp4c` and Runge-Kutta methods

MATLAB solver `bvp4c` is a strong solver based on collocation. It allows a flexible description of the ODEs, various kind of boundary conditions, parameters and options (jacobian, tolerances, vectorization and so on). It requires a guess solution. As a side effect it provides an approximation of the derivative of the solution. This allows us to combine `bvp4c` with an IVP solver.

## 5    Numerical examples

For the both methods we implemented the ideas in `MATLAB 2010a [15]`, for the first method our code use `Matlab Spline Toolbox`, the function `spcol` allows us to compute easily the collocation matrix and for (IVP) problems the solver `ode23tb` works fine (when the problem is stiff). To avoid the error propagation, we choose for (BVP) problem B-splines of order 4 (degree 3) or order 5 (degree 4), in this we implemented the function `polycalnlinRK`.

The second method was implemented in `MATLAB` using the functions `cebdif`, `cebint` and `cebdiff` contained in `dmsuite` and described in [21], we write the function `solvepolylocalceb` who solve the nonlinear system and call the *Runge-Kutta* solver `ode23tb`. The derivatives at $a$ and $b$ were computed by calling `cebdifft`. The third method following the idea given by [19].

In order to compare the costs (run-times) experimentally we use `Matlab` functions `tic` and `toc`.

- **The first example is the case where we know the exact solution**

  Consider the (BVP) problem:

$$\begin{cases} y''(x) + 2\pi^2 \exp(-y(x)) = 0, \ 0 < x < \infty, \\ y(1/10) = 2\ln\sin\pi/10, \\ y(9/10) = 2\ln\sin 9\pi/10 = 2\ln\sin\pi/10. \end{cases} \tag{14}$$

The exact solution of this problem is:

$$y(x) = 2\ln(\sin(\pi x)),$$

125

and we see that the exact solution is a periodic function of the period $T = 1$. Using step control algorithm [1] we determined that problem (14) has singularities in $x = 0$ and in $x = 1$. We have set the tolerance to $\varepsilon = 10^{-10}$, we took $N = 1025$ and maximum number of iterations $NMAX = 50$. The start solution are obtained using the *Lagrange* interpolation polynomial with nodes: $1/4, 5/24, 1/6$. The results we have obtained after 10 iterations, run times are:

| $Tolerance$ | $1st\ Method$ | $2nd\ Method$ | $3rd\ Method$ |
|:---:|:---:|:---:|:---:|
| $10^{-10}$ | 0.777144 | 7.204652 | 1.350691 |

The graph of approximate solution are presented in Figure 1, and the errors in semi-logarithmic scale for the first method in Figure 2(a), for the second in Figure 2(b) and for the third in Figure 2(c), respectively.
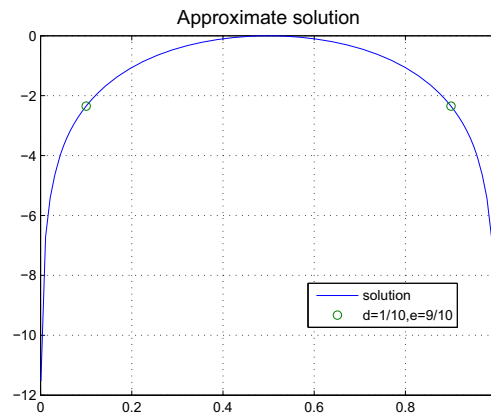


Figure 1: Approx-solution



(a) B-splines+Runge-Kutta    (b) C.C+Runge-Kutta    (c) bvp4c+Runge-Kutta
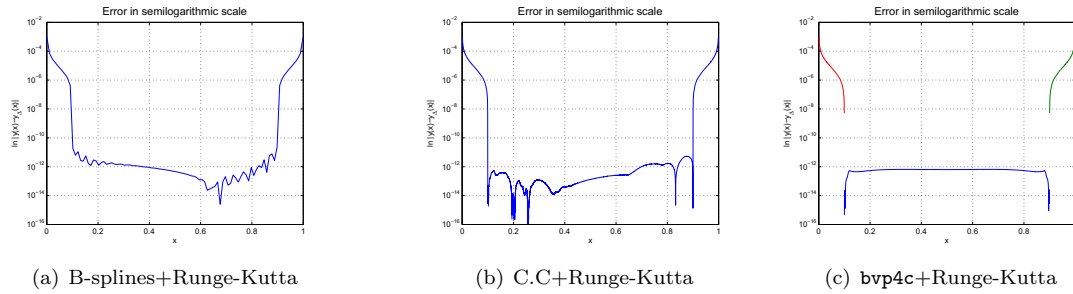
Figure 2: Errors

• **The second example is the case of unknown exact solution**

$$y''(x) = x^{-1/2}y^{3/2}, \tag{15}$$

with boundary conditions:

$$y(0) = +\infty, \quad y(\infty) = 0. \tag{16}$$

This (BVP) arises in a semiclassical description of the charge density in atoms of high atomic number. There are difficulties at both end points. These difficulties are discussed at length in *Davis* (1962) and in *Bender and Orszag* (1999). *Davis* discusses series solutions for:

$$y(x) \text{ as } x \to 0.$$

It is clear that there are fractional powers in the series. That is because, with $y(0) = 1$, ODE requires:

$$y''(x) \sim x^{-1/2} \text{ as } x \to 0,$$

and hence there be a term $\frac{4}{3}x^{3/2}$ in series for $y(x)$. Of course, there must also be lower-order terms so as to satisfy the boundary condition at $x = 0$. Bender and Orszag discuss the asymptotic behavior of $y(x)$, $x \to 0$. Verify that trying a solution of the form

$$y(x) \sim ax^{\alpha},$$

yields for the start solution:

$$y_0(x) = 144x^{-3}.$$

We use for inner points $a = 0.015$, and $b = 59$. The results we have obtained after 12 iterations, run times are:

| Tolerance | 1st Method | 2nd Method | 3rd Method |
|---|---|---|---|
| $10^{-10}$ | 0.493448 | 1.204652 | 0.838735 |

The graph of approximate nonlinear solution of *Fermi-Thomas* problem is presented in Figure 3.
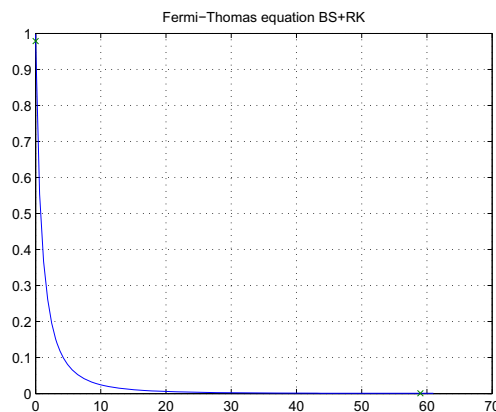


Figure 3: The charge density in atoms of high atomic number

127

# 6    Conclusions

The running time for B-spline collocation is the shortest, because its collocation matrix is banded. Tchebychev collocation has the longest time, since its collocation matrix is full. The `bvp4c` solver has an intermediary position, since it has a different type collocation. Nevertheless, further tests are necessary.

# References

[1] O.Agratini, I.Chiorean, G.Coman, R.Trîmbiţaş, *Analiză numerică şi Teoria Aproximării* , Presa Universitară Clujeană, Cluj-Napoca, 2002.

[2] U.Ascher, S.Pruess and R.D. Russel,    On spline basis selection for solving differential equations, *SIAM J. Numer. Anal. 20 nr:1*, 1983.

[3] U.Ascher, Solving boundary value problems with spline collocation code, Journal of computational physics, 34 , Nr:3, pp: 401-415, 1980.

[4] U. Ascher, R.M. Mattheij, and R.D. Russel, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, SIAM, 1997.

[5] F.A.Costabile, A.Napoli, A coolocation method for global approximation of general second order BVPs, *Computing Letters* 3, pp: 23-24, 2007.

[6] F.A.Costabile, F.Dell-Accio,   On the nonsingularity of a special class of centrosymmetric matrices arising in spectral method in BVPs, *Applied Mathematics and Computation*, 206, pp: 991-993, 2008.

[7] C.de Boor,   *A practical guide to splines*, Springer-Verlag, Berlin, Heidelberg, New-York, 1978.

[8] C.de Boor, R.Weiss, *Solveblock: A package for solving almost block diagonal linear systems, with applications on the numerical solution of odinary differential equations* , MCR#TSR 1625, Madison,Wisconsin, 1976.

[9] M.A. Diminescu, V. Nistorean,   *Gospodărirea durabilă a resurselor de apă,Conf. Internaţ. Energie-Mediu,* CIEM , Bucureşti, 2005.

[10] W.Gautschi,   *Numerical Analysis, An Introduction*, Birkhauser Verlag, Basel, 1997.

[11] L.Greengard, V.Rohlin   *On the Numerical Solution of Two-Point Boundary Value Problems*, *Comm.Pure Appl.Math.XLIV* , pp:419-452,1991.

[12] W.Heinrichs, Strong convergence estimates for pseudospectral methods, *Applications of Mathematics*, 37(6), pp:24-37, 1992.

[13] E.Houstis, A collocation method for systems of nonlinear ordinary differential equations, *J. Math. Anal. Appl.*, 62, pp:24-37, 1978.

[14] P. Henrici, *Discrete Variable Methods in Ordinary Differential Equations*, John Willey and Sons, New York, 1962.

[15] Mathlab 2010a, *Set of manuals,* , MathWorks, Inc., Natick, MA, 2010.

[16] Ion Păvăloiu, Daniel N.Pop, Radu T.Trîmbiţaş, Solution of polylocal problem with a pseodospectral method, *Annals of "Tiberiu Popoviciu" Seminar of functional equations, approximation and convexity*, vol 8, pp: 53-65, Cluj-Napoca, 2010.

[17] Daniel N.Pop, Radu T.Trîmbiţaş,   An approximation methods for second order nonlinear value polylocal problems using B-splines and Runge-Kutta metods, N.A.A.T. 2010, *Studia Universitas "Babes-Bolyai", Series Mathematica*, Vol.LVI, Number 2, pp:515-527, June 2011.

[18] R.D.Russel, L.F.Shampine,   A Collocation Method for Boundary Value Problems, *Numer. Math.*, 19, Springer-Verlag, pp: 1-28, 1972.

[19] L. F. Shampine, I. Gladwell, S. Thompson, Solving ODEs with MATLAB, Cambridge University Press, 2003.

[20] R.Stover,   Collocation methods for solving linear-algebraic boundary value problem, *Numer.Math.* 88, pp:771-795, 2001.

[21] J.A.C Weideman, S.C Reddy,   A MATLAB Differentiation Matrix Suite, *ACM Trans. on Math. Software*, 26, pp: 465-519, 2000.

Daniel N.Pop
Romanian-German University Sibiu
Faculty of economic engineering
in electric, energy, electronic
Calea Dumbravii street nr: 28-32
Romania
E-mail: *popdaniel31@yahoo.com*

Radu T. Trîmbiţaş
"Babeş-Bolyai" University Cluj-Napoca
Faculty of Mathematics and Computer
Science
Mihail Kogalniceanu street nr. 1
Romania
E-mail: *tradu@math.ubbcluj.ro*

# Modeling and Analysis for Social Media Network - Case Study: The Small World Type Network for Social Media Networks Analysis in PMML

**Ioan Pop**

**Abstract**

The networks of "small world" are exceptional models that can be adapted for modeling networks of social media. This paper proposes a model for social networking model which can be the basis of efficient analysis of these networks. At the end of this paper, we implement the model designed for analyzing social networks. Implementation of the model is done using PMML language (based on XML).

## 1 Models of social networks

Development potential of Web 2.0 and its synergies with the Semantic Web have facilitated the development of the Social Media. There have been many social networking platforms that favored the development of various social networks operating in various fields of activity of people around the world. A social network is a network of entities that have common interests. Entities in a social network that process information and knowledge are people, groups, organizations, computers, software entities (blogs, software agents, etc.).
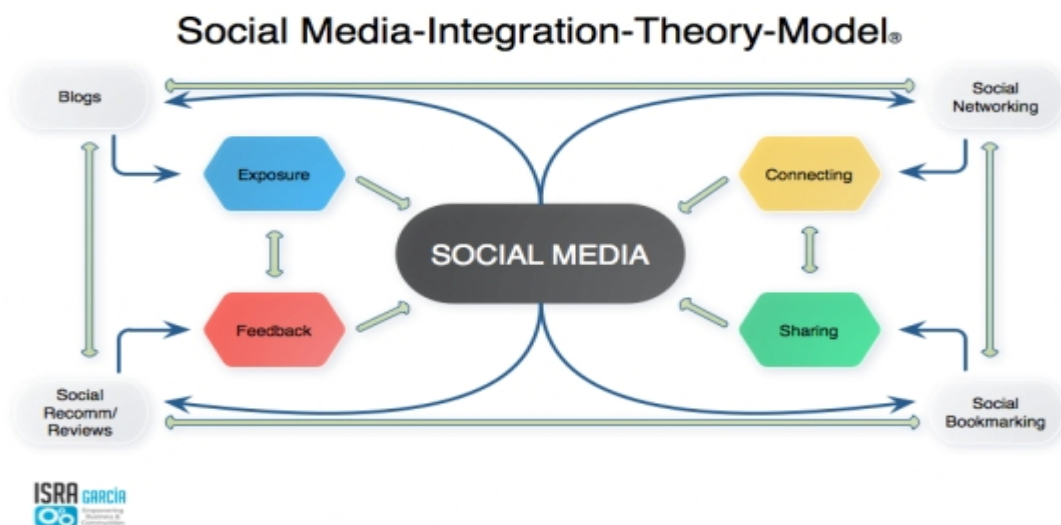


Figure 1.1. Representation of a theoretical model of integration of Social Media [1].

The development of social networking has led researchers to deal with the study and analysis of data stored on social networking platforms to extract knowledge through data mining. An interesting model for social media analysis is illustrated in Figure 1.1.

In social network analysis is used  to measure relationships and flows of information between entities that make up the network. Analysis of social networks Social Media enables integration approach. Social Media is seen as an interactive process that makes the exchange of information between network entities with extensive exposure of communication features, feeedback, commitment and sharing. A  .

## 1.1 From random graphs to model social networks

Graphs are used in practice to represent computer science communication networks, organizing information, computer equipment, flow calculation, etc. For example, a website link structure could be represented by a directed graph. Nodes are web pages available on that site and the existence of the directed arc from page A to page B is assured by the existence of a link A to B.
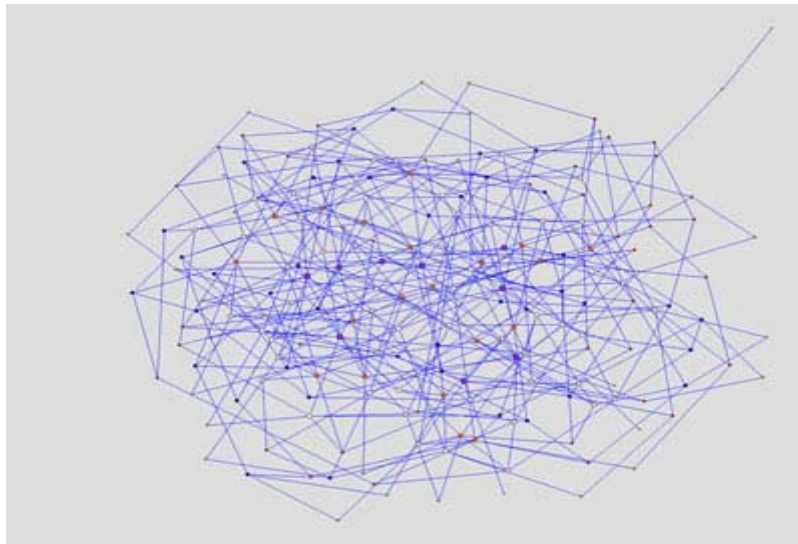


Figure 1.2. A web graph model. [7]

A graph has nodes and arcs. If connections between nodes are determined at random then the graph is called a random graph type. These graphs have special properties such as the probabilities associated with arcs (connections) allocated as contained in the interval [0,1]. An example is given in Figure 1.3.
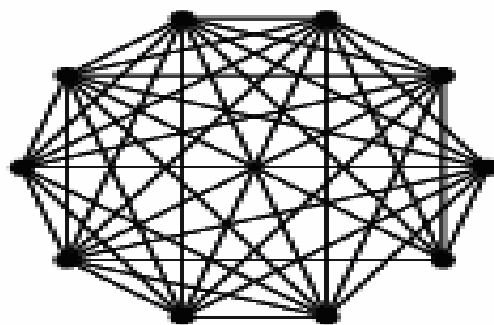


Figure1.3. Type random graph with 10 nodes and uniform distribution of connections [2].

For social network analysis theory of graph model is a powerful modeling tool that helps abstractions needed in this area of exploration. Among the different types of graphs the best type adopted were random graph. [6]. Random graph type is often used to model the web. This model accurately enough plays Web, but in our opinion, is very suitable for modeling social networks.

There are interesting studies regarding fuzzy graphs, a special type of random graphs, graphs which are the subject of the case study in Section 2. Internet Web space can be imagined as a fuzzy graph with the evolution in time. These theoretical models support the activities of the Social Media Study by instrument making analysis offered.

## 1.2 The power-law in modelling social networks

In general, power-law models scientific phenomena used in fields such as physics, computer science, linguistics, geophysics, sociology, economics. In particular, we refer to the shape of the small world network type.

Regarding the „distribution power law" (briefly called power-law) for random graphs, it has a special application. Power-law applies to a random variable associated with a network of "Small World" described below (section 1.3).

In recent works [3], power-law distributions of type were observed in various aspects of the web. Two ways of modeling work in the social networks are of particular interest to us:

1. First, power laws can be used to characterize the behavior of network users in two interrelated aspects:

a. Statistics of access to social network, which can be easily obtained from server logs (but caching effects) .

b. The power law shows the number of users enjoy access to pages from a particular site. This number is handled and controlled by trees of cache web proxies or clients [4].

2. Second, our context is immediately relevant for the distribution of weights on a tree network graph, ie the probability distribution of the number of connections that go from a full network node (out-degree) or the probability distribution of the number of connections received across the network node (in-degree). The in-degree and out-degree power law-abiding [5].

Power law has the form given by the equality

$$y = \alpha x^{\beta} + o\left(x^{\beta}\right) \qquad (1)$$

where $\alpha$ and $\beta$ are constant, and $\beta$ is usually called, the scaling exponent, and $o(x^{\beta})$ is a asymptotic function of $x^{\beta}$ called the "small $o$"[8]. If social networks, which can be considered special cases of subnets modeled as subgraphs of the web graph, the quantity $o(x^{\beta})$ can be neglected. Logarithms then equality (1) becomes

$$log\ y = \beta\ log\ x + log\ \alpha \qquad (2).$$

If we assume that social networks are *y* nodes, each with weight *x*, which satisfy condition (2). It is noted that retains power law form in itself, but re-scaling argument leads to the proportionality constant change. From this we conclude that the law-power frequency model random events that occur in social networks.

## 1.3 Models based on networks of "small world"

The idea of "small world" was first developed in Milgram's experiment. A "small world" is perceived to be a network of relationships between people which can be identified with each other in this "small world". This refers colloquially to "Six Degrees of Separation", and was the subject of considerable interest in the research community in recent years [9]. According to this concept worldwide, on

average, every person is known by any other person in a large community with information about six steps in relationship knowledge step by step.

Milgram's experiment based on that outlined by the "six degrees of separation" in a small world network type, the conceptualization of this network can become the extrapolation model for social networks of "social media".

It is believed that almost any pair of people in the world can be connected to each other through a short chain of intermediate acquaintances, with a typical length of about six people.

These interpretations have generated new modeling of social networks, which have revealed new features for these social networks.

The mathematical model is based on the notion of random graph associating at each node a weight or a degree of connectivity. The degree of connectivity is given by the probability associated with the event to connect a person with another person.

The most important property inherited from the networks of "small world" is that the graph diameter is relatively small compared to the overall size of the graph associated network. This feature is very important for social networks.

In mathematical terms, the social networks for finding the shortest path between the origin node and the network node of interest in a graph requires, in general $O(n)$ steps, which can be reduced to $O(n \log n)$ for a sparse graph. [10].

# 2. Implementation of the SWSNM model in the PMML

## 2.1 SWSNM model based on fuzzy graph with power law.

A fuzzy graph $G$ is a pair defined by $G=(V,F)$, with $V=\{v_i\}$, $F=(f_{ij})$, $0 \leq f_{ij} \leq 1$, where $V=\{v_i\}$ is a set of nodes, $f_{ij}$ is the fuzziness of the arc from the node vi to the node vj .[11] This fuzziness of the arc $(v_i, v_j)$ is given by the power law of the probability expressed in the formula (2) from section 1.2.

In figure 2.1 is illustrated a fuzzy graph $G=(V,F)$, where $V=\{v1, v2, v3, v4\}$ with 4 vertices ($y = 4$). Each node in the graph can be connected to another node through a one-way weighted arc with probability given by the power law (2). $F$ is a matrix with 4 rows and 4 columns whose elements are weights corresponding to the graph arcs formed between nodes.
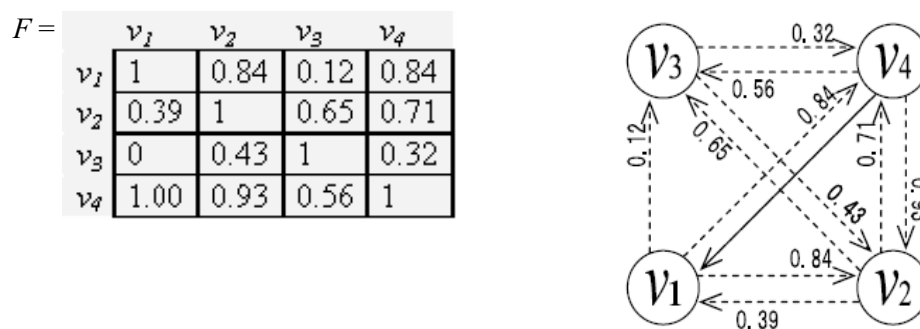


Figure 2.1. Fuzzy graf representation.

The model presented above can be generalized to a social network, which in turn can be seen as a network of the "small world". From this representation, the fuzzy graph $G$ and a social network, can then use a series of measurements of the network arising from the properties associated to the designed graph. The problem of data collection, i.e. discrete values of the weights associated with network connections, is resolved using a simple program that oversees the network of social statistics. Connections made between the attributes of social network members are obtained from server logs and retained in a database.

133

The attribute's values can be: connectivity of the graph, number of connected components, distribution of nodes per site, distribution of incoming and outgoing connections per site, average and maximal length of the shortest path between any two nodes (diameter), frequency of occurrence of the event connection, connection type, frequency of occurrence of a theme within a connection, frequency of occurrence of a theme within a subgraph of the graph attached to a social network, frequency of occurrence of a theme throughout the term social network, etc..

The model created based on this type of graph characterizing a social network is implemented in PMML

## 2.1 Implementing in PMML

PMML (Predictive Model Markup Language) is a standard maintained by "Data Mining Group Consortium" based on XML to create data mining models. The structure of any model is described as an XML schema with a specific namespace. A PMML document is an XML tree structure that can represent the model. SWSNM model can be represented in a PMML document where input parameters are weighted connectivity (probability values associated with power law graph model fuzzy). Input parameters (see Fig. 3.1) are represented in a matrix structure as described in Figure 3.2.

```xml
<xs:element name="Matrix">
  <xs:complexType>
    <xs:choice minOccurs="0">
      <xs:group ref="NUM-ARRAY" maxOccurs="unbounded" />
      <xs:element ref="MatCell" maxOccurs="unbounded" />
    </xs:choice>
    <xs:attribute name="kind" use="optional" default="any" >
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="diagonal"/>
          <xs:enumeration value="symmetric"/>
          <xs:enumeration value="any"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="nbRows" type="INT-NUMBER" use="optional"/>
    <xs:attribute name="nbCols" type="INT-NUMBER" use="optional"/>
    <xs:attribute name="diagDefault" type="REAL-NUMBER" use="optional"/>
    <xs:attribute name="offDiagDefault" type="REAL-NUMBER" use="optional"/>
  </xs:complexType>
</xs:element>

<xs:element name="MatCell">
  <xs:complexType >
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="row" type="INT-NUMBER" use="required" />
        <xs:attribute name="col" type="INT-NUMBER" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
```

Figure 3.2 Data structure stored in a matrix.

Input parameter values in the matrix are retained in PMML document as shown in Figure 3.3.

```xml
<Matrix nbRows="4" nbCols="4">
  <Array type="real">1.00 0.84 0.12 0.84</Array>
  <Array type="real">0.39 1.00 0.65 0.71</Array>
  <Array type="real">0.00 0.43 1.00 0.32</Array>
  <Array type="real">1.00 0.93 0.56 1.00</Array>
</Matrix>
```

Figure 3.3. Model input parameters.

When we set input parameters we use the XML schema in a PMML document. The scheme is called "MiningModel". This is coded as shown in Figure 3.4.

```
<xs:element name="MiningModel">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element ref="MiningSchema"/>
    <xs:element ref="Output" minOccurs="0"/>
    <xs:element ref="ModelStats" minOccurs="0"/>
    <xs:element ref="ModelExplanation" minOccurs="0"/>
    <xs:element ref="Targets" minOccurs="0"/>
    <xs:element ref="LocalTransformations" minOccurs="0" />
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="Regression"/>
      <xs:element ref="DecisionTree"/>
    </xs:choice>
    <xs:element ref="Segmentation" minOccurs="0"/>
    <xs:element ref="ModelVerification" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
   </xs:sequence>
   <xs:attribute name="modelName" type="xs:string" use="optional"/>
   <xs:attribute name="functionName" type="MINING-FUNCTION" use="required"/>
   <xs:attribute name="algorithmName" type="xs:string" use="optional"/>
  </xs:complexType>
</xs:element>
```

Figure 3.4. XML Scheme for „MiningModel".

Different connectivity analysis on a social network can be implemented based on the scheme of Fig. 3.4 "MiningModel" through elements of "segmentation". An element of "segmentation" contains several segments and model combination methods. Each segment contains an element of the predicate that specifies the conditions under which this segment is used. The "segmentation" is coded as shown in Figure 3.5.

```
<xs:element name="Segmentation">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element ref="LocalTransformations" minOccurs="0" />
    <xs:element ref="Segment" maxOccurs="unbounded"/>
   </xs:sequence>
   <xs:attribute name="multipleModelMethod" type="MULTIPLE-MODEL-METHOD" use="required"/>
  </xs:complexType>
</xs:element>

<xs:element name="Segment">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    <xs:group ref="PREDICATE"/>
    <xs:choice>
      <xs:element ref="ClusteringModel"/>
      <xs:element ref="GeneralRegressionModel"/>
      <xs:element ref="NaiveBayesModel"/>
      <xs:element ref="NeuralNetwork"/>
      <xs:element ref="RegressionModel"/>
      <xs:element ref="RuleSetModel"/>
      <xs:element ref="SupportVectorMachineModel"/>
      <xs:element ref="TreeModel"/>
      <xs:element ref="Extension"/>
```

```
      </xs:choice>
     </xs:sequence>
     <xs:attribute name="id" type="xs:string" use="optional"/>
     <xs:attribute name="weight" type="NUMBER" use="optional"/>
    </xs:complexType>
   </xs:element>
```

Figure 3.5. „Segmentation" element.

Segment element is used to combine other models as part of a whole. One way to combine several data processing models to analyze a social network is to specify the attribute "multipleModelMethod" in element segmentation. This method is coded as shown in Figure 3.6.

```
   <xs:simpleType name="MULTIPLE-MODEL-METHOD">
     <xs:restriction base="xs:string">
      <xs:enumeration value="majorityVote"/>
      <xs:enumeration value="weightedMajorityVote"/>
      <xs:enumeration value="average"/>
      <xs:enumeration value="weightedAverage"/>
      <xs:enumeration value="median"/>
      <xs:enumeration value="max"/>
      <xs:enumeration value="sum"/>
      <xs:enumeration value="selectFirst"/>
      <xs:enumeration value="selectAll"/>
     </xs:restriction>
   </xs:simpleType>
```

Figure 3.6. Elements with attribute analysis "multipleModelMethod".

To use the PMML model for distributed SWSNM results can be posted as a project within the DMG group [12].

# 3   Conclusions

1. Social networking features deserve to be investigated. These features may show us the functional properties of these networks related to robustness, trend research utility. A relevant modeling of social networks can ensure the functionality of the correlations that exist between content and connectivity. In addition, various thematic communities may become detectable by subconectivity patterns that can be highlighted by a cleverly designed model.
2. The SWSNM can become an effective framework for integrating tools of investigation and analysis of social networks. In this case the model can also lead to better algorithms for the exploitation of th network connections available.
3. The PMML platform provides a number of classes of data mining algorithms that can become useful tools for an efficient analysis of social networks.

# References

[1]  http://socialmediatoday.com/isra-garcia/278936/social-media-integration-theory-model
[2]  R. C. Read, R. J. Wilson, Atlas of Graphs. Oxford, England: Oxford University Press, 1998.
[3]  http://arxiv.org/abs/0706.1062.
[4]  B. A. Huberman, P. L. T.Pirolli, J. E Pitkow, R. M. Lukose, Strong regularities in World Wide Web surfing. Science 280, 1998, pp.95–97.
[5]  S. R. Kumar, P. Raghavan, S.Rajagopalan, D.Sivakumar, A.Tomkins, E. Upfal, Stochastic models for the Web graph, In Proc. 41st IEEE Ann. Symp. on the Foundations of Computer Science, 2000, pp. 57–65.
[6]  P. Erdos and A. Renyi, On the strength of connectedness of random graphs, Acta Math. Acad. Sci. Hungar. 12 (1961), pp.261-267.

[7] http://155.198.210.128/~time/networks/ Evolution of Page Popularity under Random Web Graph Models

[8] http://en.wikipedia.org/wiki/Big_O_notationÎn cachePagini similare

[9] S. Milgram, The small world problem. Psychology Today 1, 1967, pp. 61.

[10]T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, 2nd edn. Cambridge, MA: MIT Press, 2001.

[11]A. N. Gani, K. Radha, On Regular Fuzzy Graphs, Journal of Physical Sciences, Vol. 12, 2008, pp.33-40.

[12]http://www.dmg.org/

[13]J. Scott, Social Network Analysis: A Handbook, SAGE publication inc. California, scd ed. 2000, pp.8-33.

[14]R. Albert, H. Jeong and A.L. Barabasi, Diameter of the World-Wide Web, Nature 401, 1999, pp.130-131.

[15]http://xxx.lanl.gov/abs/cond-mat/9907038.

[16]W. Aiello, F. Chung, L. Lu, A Random Graph Model for Power Law Graphs, Experimental Math Journal, vol. 10, 2000, pp.53-66.

[17]K. Sameena, M. S. Sunitha, Fuzzy graphs in fuzzy neural networks, Proyecciones Journal of Mathematics, Vol. 28, No 3, December 2009, pp.239—252.

[18]http://www.eecs.harvard.edu/~michaelm/NEWWORK/postscripts/webgraph.pdf

[19]A.Z. Broder, S.R. Kumar, F. Maghoul, P. Raghavan, S.Rajagopalan, R. Stata, A. Tomkins, and J. Wiener, Graph structure in the web: experiments and models, Proc. 9th WWW Conf., 1999.

[20]R. Kosala, H. Blockeel, Web Mining Research: A Survey, ACM SIGKDD, July 2000. Vol. 2, Issue 1, 1-14.

IOAN POP
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Romania
E-mail: me.ioanpop@yahoo.com

# About Hash Function and Watermark Algorithms

## Crina Rațiu, Dominic Bucerzan, Mihaela Crăciun

### Abstract

In today's corporate world, images and documents travel widely and rapidly through email, across the Internet and mobile devices. Controlling and protecting sensitive or confidential documents and images has become very important. [12]

To protect the ownership rights on digital audio, image, video and all forms of media content, digital watermarking techniques can be used combined with cryptography and hash functions.

In this paper we focus on different algorithms of digital watermarking for image applications and on several types of watermarking attacks which aim at its robustness, its form and even at its removal. Also we propose a solution of digital watermarking completed with a hash function in order to increase the degree of security of the transferred data through today's uncertain environment. To sum up we present our view upon specific solutions in this field.

**Key words:** digital watermarking, hash function, cryptography

## 1 Introduction

Digital information can be copied and it is difficult to make the distinction between the original and the copied version. In order to protect the digital data against copying and forgery two complementary techniques have been developed: cryptography and digital marking. Cryptosystems may be used in order to protect digital data on its way from the sender to the receiver. The received and decrypted data is identical to the original one and it is no longer protected. Here digital watermarking can be a solution for ensuring the authenticity of digital data.

The techniques of digital watermarking as well as cryptography straighten each other in the process of sending and receiving digital information through today's unsecure networks.

Taking into consideration nowadays technology it is not difficult to apply a watermark to a digital data set, but in the same time there are many ways to remove it. In order to be able to detect when o digital watermark has been altered a "hash" function may be used considering that it has an important role in the identification of the content of a message sent through computer networks. [11]

Hash function may also be called "dispersion" function or "digest" function. These type of function changes a series of symbols of arbitrary lengths such as a password of 8 characters or a 1000 pages document, in a series of symbols with relatively short of fixed dimension. The gist of the idea is that when a difference occurs in the case of the inserted series the result changes automatically. [11]

Hash functions have an important role in cryptographic systems which protect communication channels. In the article "Find me a hash", Susan Landau, names this kind of function "tape function"-

the duct tape of cryptography, because it has so many usages: to prove a message is genuine, to ensure the integrity of the software, to create passwords which you can only use once, to create digital signatures and it has also the function of enabling many internet communication protocols.

The paper is organized as follows: in section 1 we made an introduction about the importance of assuring digital information security, section 2 contains the most used terms in cryptography especially hash functions, section 3 is reserved to watermarking algorithms, in section 4 we tackled upon different types of attacks which aim watermarking algorithms, we present our contribution to watermarking techniques in section 5 and some conclusions in section 6.

# 2 Cryptography and Hash Functions

The latest development of our society has lead to the increasing of the information volume sent through computer networks, which has accelerated the progress and the usage of cryptographic methods. The purpose of cryptography is to ensure through mathematical specific techniques, the fundamental characteristics of electronic data: confidentiality, integrity, authenticity and non-repudiation.

The hash functions, also known as one-way functions, are a fundamental class of primitive functions in modern cryptography, used mainly in digital signatures and in ensuring the integrity of digital data. The hash cryptographic functions compress electronic data set of arbitrary length into electronic data set of fixed length.

The characteristics of a hash cryptographic function [14]:
- it is a one way function which means it is difficult to reverse it.
- it can easily face collisions ( it is difficult to find messages which generate the same type of hash )

The hash function may be regarded as a surjective function (because there is a possibility, however remote, to find "n" messages for the same hash).

Not all the hash functions are proper to cryptography. An example of a hash function which cannot be regarded as a cryptographic function is a function which gives you the sum of a series of numbers. This function is not cryptographic, first of all because we can easily find two numbers for which the function may give us the same result ( for ex. 23 and 50 ) and secondly because it has no fixed length.

An example of daily used hash function is the algorithm used to determine the Control number of the Personal Numeric Code. The hash function which has been mostly used till recently is MD5. [14] MD5 algorithm has been made public because of possible adjustments and for a possible acceptance as a standard procedure. This algorithm is mainly designed for the application of digital signature, where a file must be safely compressed before being encrypted using a secret key through a cryptographic system with a public key.

# 3 Algorithms used for watermarking the images

## 3.1 Robust, space algorithm proposed by A. Tefas and I. Pitas

This algorithm works in space. This means that costly processing (time for calculations) is not needed for the inclusion of the watermark. Brightness values (L (x, y) $\in$ {1, .., L}) or the red, green and blue (r (x, y), g (x, y), b (x, y) - Red Green Blue) are directly manipulated [9].

Watermark W is a ternary image with pixel value {0, 1 or 2}. These values are generated using digital key K. Watermark insertion is made by modifying the pixel value of the original image: [13]

$$
P'(x,y) \quad = \quad
\begin{cases}
P(x,y), & \text{if } W(x,y)=0 \\
E1\ (P(x,y),\ IN\ (x,y)), & \text{if } W(x,y)=1 \quad \text{where:} \\
E2\ (P(x,y),\ IN\ (x,y)), & \text{if } W(x,y)=2
\end{cases}
$$

P - original image (greyscale), (x, y)-location of a pixel, P '- image with watermark, IN - image neighborhood, E1 and E2 - functions for the inclusion of a watermark defined as follows:

E1 (P, IN) = (1-a1) •IN(x, y) + a1 •P(x, y)

E2 (P, IN) = (1-a1) •IN(x, y) + a2 •P(x, y),      where a1 > 0, a2 < 0, scaling constant.

The multiplying by (1 - a1) in scaling IN's value is used to ensure that the value P' of the image with watermark will not overcome the maximum value for a representation of the image, 8 bits, corresponding to a white pixel.

The value of a neighborhood pixel is calculated as the arithmetic mean of its neighboring pixel values in the original image, for a given value of neighborhood radius r. For example, if r = 1, then the value is calculated as follows: [13]

$$I_N(x, y) = \frac{P(x + 1, y) + P(x + 1, y + 1) + P(x, y + 1)}{3}$$

## 3.2 Algorithm proposed by G. C. Langelaar

The watermark consists of a bit string. The algorithm manipulates the pixel brightness, in blocks of 8 x 8 pixels. Thus we get XY/64 possible locations (XY is the number of pixels in the image). No qualitative selection is made among the possible locations which are randomly selected. It creates a binary model, pseudo-random size 8 x 8 pixels, which will be used for each insertion: [6]

pat(x,y) ∈ {0, 1}, where 0 ≤ x, y < 8

To insert a watermark's bit in a block B = {l (x + xo, y + yo) where 0 ≤ x, y <8}, the first time the block is divided into two parts B0 and B1, using the model:

B0 = {l(x+xo, y+yo) where pat(x, y) =0}

B1 = {l(x+xo, y+yo) where pat(x, y) =1}

The mean brightness is calculated for both categories l0 and l1. The difference between the two values is the signature bit, where α> 0 is the threshold:

l0 – l1 > +α          if s =1

l0 – l1 < -α          if s =0

If this relationship does not occur naturally, we will decrease or increase pixel brightness value B1, until the required relationship appears.

## 3.3 Algorithm proposed by Ingemar J. Cox

This algorithm is considered a classic between the watermarking algorithms and was first introduced in 1997 [3]. The image is divided into two parts: the background and the object of interest of the owner. A watermark is then generated by a key. It consists of a series of random real numbers, which are distributed according to a Gaussian curve. Based on the reference image, the selected object is transformed into frequency domain using discrete cosine transformation.[13]

Marking can be described mathematically as follows: X = (x1, x2, ..., xn) where n is the watermark's length. The choice of length is based on the targeted extent of the watermark's spectrum. The bigger the length, the smaller the restrictions is to include the mark in the image. To incorporate the marking in the picture, a number of factors must be extracted using discrete cosine transformation applied to the result image. Consider the image denoted by P and coefficients C = (c1, c2, ..., cn), n being the length of the bookmark. C coefficients are chosen as the most important coefficients of discrete cosine transformation. Coefficients are modified by using three different formulas:[13]

ci' = ci + a • xi          (1)

ci' = ci • (1 + a • xi )    (2)

ci' = ci • (e a • xi )      (3)

Equation (1) is not exactly suitable if the "ci" values have a wide variety. If the deviation is 106 when summing with 100, it may be insufficient to create a watermark, but if the deviation is 10, the summing with 100 will distort the value unacceptably. Equations (2) and (3) are more robust to such differences in scale. Figure 2 presents schematically the main steps in the insertion technique of a watermark using the algorithm proposed by Ingemar J. Cox.
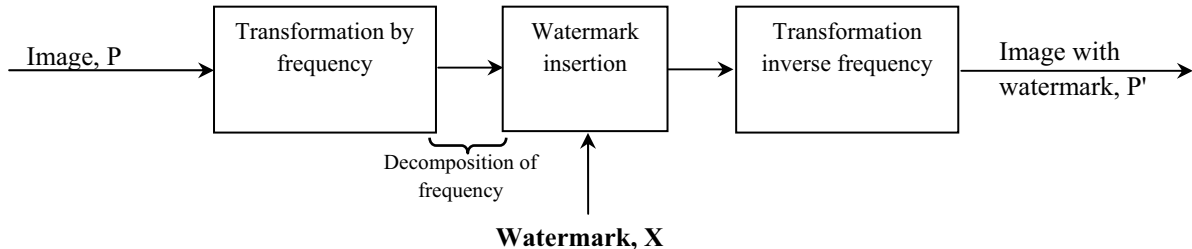


**Fig. 1** Description of Cox's additive algorithm [13]

Although watermarking techniques are developing constantly and become more complex, the attacks methods are also increasing. Below we present the main types of attacks that target the watermarking algorithms.

# 4   Attacks on watermark

Attacker's goal is to reduce the security of marking system, in other words to reduce the probability of extraction / detection of the original watermark, i.e. to increase the likelihood of extraction / detection of a watermark inserted in a signal that was not marked (false extraction).

## 4.1 Attacks on the watermark's robustness

These attacks change the image pixel values and can be classified into two categories: attacks based on signal processing algorithms and attacks with algorithmic characteristics. [2]

The first category includes attacks that use conventional methods of image processing: compression, filtering or scanning. Attacks using methods of video processing to modify images by adding noise are meant to weaken the strength of the watermark. There are currently several software programs that demonstrate the strength of attacks. One of them is UnZig. Image degradation in this case is reasonable, concluding that this attack is particularly effective.

*Compression* - the most common category of unintentional attack. This method removes insignificant items from a multimedia signal from the perceptual point of view.  Thus, by compression, perceptually similar signals come to be identical.

*Filtering* - marks the signal like an additive noise, which is a reasonable assumption for transparent marking spread spectrum type. Removing the marking is equivalent to a problem of disposal of a noise from an image, resulting in an estimation of the original image. Random Noise - introduces imperceptible distortions even at high signal per noise of 20 dB, but the impact on the detection of the watermark is insignificant.

*Attacks on cryptographic security* - cryptographic security refers to the integrity of the semantics marking and to the autentification to ensure confidentiality to robust marking. Attacks on cryptographic security are illegal. These kind of attacks have as target the cryptographic components from the transparent marking system. They use standard methods of cryptanalysis to attack the system: brute force search for the inserted message or key search.

Estimation attacks - in these cases, watermark or the host signal is estimated without knowing the secret key, but with information about the statistical of the watermark, and of the host signal. These attacks are applicable when the marking has been inserted into a redundant way, or when there are more marked copies available [10]. The re-modulation attack - with an estimated watermark an attacker can re-modulate the image: the estimated watermark (negative modulation) is extracted from the marked image. In the case of a correlation detector, this action cancels the positive correlation, provided the watermark is similar to the original. On the other hand, by extracting an amplified version of the estimated watermark, the detector using the correlation will fail to find the watermark in the marked image.

Attacks using *multiple marked copies* (multiple-copy) occur when the host signal is the same, and the marking is different. Essentially, these attacks are collision attacks [8]. There are two main approaches: signal processing and information encoding. In a collision attack, a coalition of pirates who have different versions of multimedia products, examine different copies in the hope of creating a new signal that is not tied to any one of them. There are several types of collision. One method consists in synchronizing the marked copies differently and mediating them, which is a simple example of linear collision. Another collision attack is called "copy-and-paste", in which attackers assemble parts cut from their copies, resulting in a new signal.

## 4.2 Attacks on Watermark's shape

The attacks on the presentation differ from the previous. The purpose is not to eliminate the watermark, but to change it so that a detector cannot find it. Examples of such attacks are: rotation, minimize and maximize, affine transformations in general.

*The mosaic attack* - divides the signal as if it is displayed as an entity, but marker detection is not possible. This is called a Stirmark attack. An example of such an attack was given by Petitcolas (Cambridge University Computer Laboratory). The attack is to divide an image into smaller parts. These parts will be assembled on a Web page using HTML. An intelligent agent responsible for searching the network for watermarked images will find only blocks of images which are too small to constitute a watermark. The attack does not cause a degradation of image quality because the pixel values are preserved [10].

*Geometric attacks* – try to desynchronize the detector in such a way that the detector cannot find the watermark. Examples of geometric attacks: translation, rotation, resizing (scaling) and their combinations; nonlinear image curving.

## 4.3 Attacks on the interpretation of the watermark's presence

This attack consists in the creation of another watermark by the cyber attacker and the insertion of that watermark into the image. The newly created Watermark will have the same intensity and shape with that of the owner. In this situation, a correct decision cannot be taken regarding the property of that image [11].

*Attack of ambiguity* - The attacker extracts its own watermark from the watermark signal, resulting in a pseudo-host signal, which when used in informational detection, will allow detection of the attackers watermark. Thus we are faced with the uncertainty regarding the identity of the copyright owner. To protect copyright, the markings must not be reversible. In other words, an attacker should not be able to extract a watermark from the original multimedia signal. A solution to this problem is to make the watermark addicted to the original signal by means of one-way function (no inverting). Copy Attack - estimates the marking from a marked signal and inserts it into another signal, called the target signal. This type of attack can be applied if the target signal can be produced without knowing a valid marking system for marking or key. Again, the marking signal depending on the original signal may be resistant to copy attack.

*Attack of the multiple marking* - the marked image will be marked again with another watermark algorithm. The question that rises is: which was the first watermark? A possible solution would be to generate time-stamped watermarks. Possible solutions against these types of

attacks are the establishment of some rules for the construction of the marking system, to combat the known ones, such as using non-invertible marks in the case of ambiguity attack.

## 4.4 Attacks by law

These types of attacks are of a completely different nature than those described above. Attacks by law may involve existing or future laws of copyright and ownership of digital images, data interpretations of laws, the owner and the credibility of the attacker, attacker's ability to spread doubts about a scheme of watermark in a courtroom and so on. One should also consider other factors such as financial strength of the owner or the attacker that the experts brought in as witnesses, the competence of judges involved in such trials.

Nowadays it is more popular to use compression with loses in the preparation of digital images via electronic data transmission. JPEG is most commonly used, but the wavelet methods are to replace TCD methods in the near future.

# 5 Contribution to watermarking techniques

As mentioned before, there are several methods to erase or replace a watermark from a digital image. In order to increase the security of electronic data against deleting or replacing, we drew some conclusions comparing the invisible watermarking algorithm proposed by A.Tefas and I. Pitas implemented in Delphi environment by A.Benczik and D. Bucerzan in "Protecția dreptului de autor asupra produselor multimedia. Tehnici de Watermarking" [10] and the solution presented by C.Ratiu and D.Bucerzan at SICOMAP2010 which proposes a digital watermark completed by a hash function implemented in JAVA [11].

In order to reduce this type of threats and to increase the security of transactions of electronic data, C. Ratiu and D. Bucerzan have designed an application which marks an image through a watermark technique which is secured by a hash function which uses the MD5 algorithm. In order to reach the above mentioned goals, JAVA has been chosen as the environment programming language, because it has been designed for a complex environment like the internet. [1]
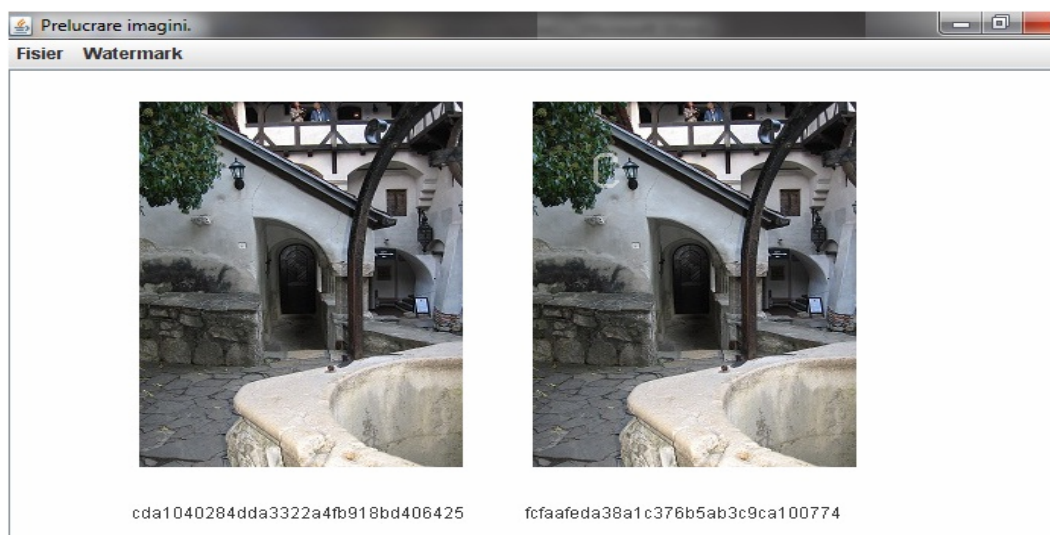


**Fig. 2** The loaded image – marked with double intensity watermark and the digest code obtained after having applied the hash algorithm [11]

As seen in figure 2, after loading an image, the hash cod is processed. The user can select which watermark to use from the main menu of the program. In the next step the image is marked

with the selected watermark. Finally the hash code for the watermarked image is processed. It can be observed that a different digest code is obtained.

To implement the Pitas algorithm, the Borland Delphi programming environment was chosen. Delphi is a software development environment based on components, facilitating rapid development of highly efficient applications based on Microsoft Windows and requires only minimal code writing. Delphi class library is a solution for the traditional Windows programming requirements, sparing us of complicated and often repetitive programming [4].

As seen in figure 3, taking advantage of the opportunities offered by the Delphi environment, we will use the application to see an image selected by the user. In the next step the user chooses a key needed to mark the image. To show that the two images are not identical, we create the difference image.
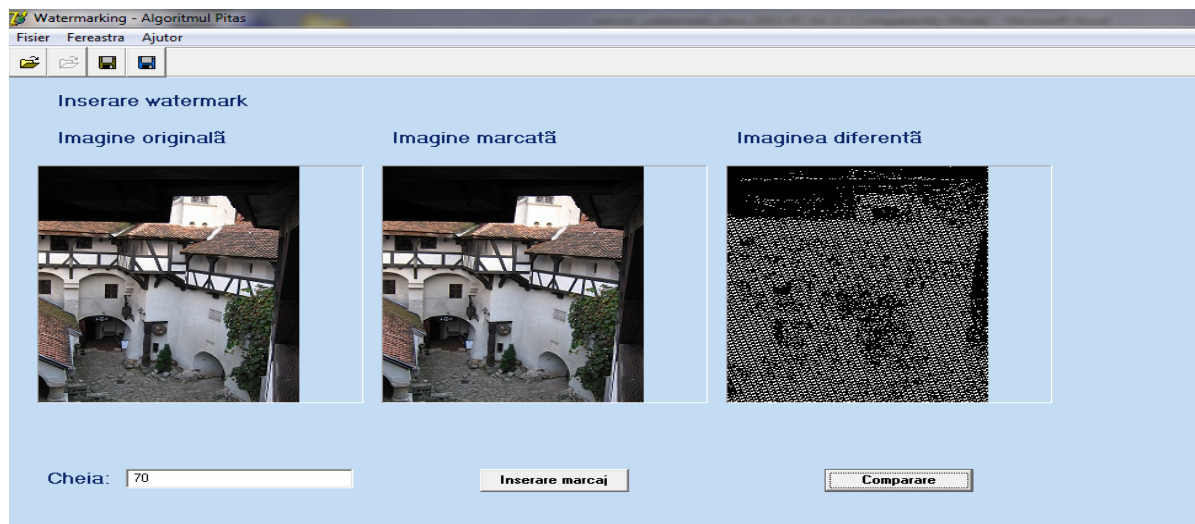


**Fig.3.** Pitas algorithm implemented in Delphi [10]

After comparing the results obtained from the two applications we can say that the Pitas algorithm implemented in Delphi is faster than the visible marking algorithm proposed in Java. This is due to the facilities of mending the images in Delphi environment such as the drawing canvas and its pixels property [10]. The main disadvantage of the application made in Delphi is that it can process only files of BMP type; the application implemented in Java can process different types of images.

An advantage of Java solution is the cancelation of the keys issue by applying a HASH function to the marked image. A disadvantage of Java solution is the fact that it has a longer execution time than the Pitas algorithm.

By using these techniques of perceptible or imperceptible marking, many cases in which the copyright is either violated or doubtful, can be clarified or even avoided. This technique proves to be useful if the multimedia document is used in medical applications, commercial transactions, evidence in court, etc. The electronic data must be first of all authenticated, and then a test will be made upon its integrity. Watermark can authenticate the electronic data.

## 6  Conclusions

In this paper we present a personal aproach to data security issues; presenting both a clasical solution of digital watermarking (A.Pitas, I.Tefas) as well as an original solution that combines digital marking with the safety given by the involvement of a HASH function.

Also we make a parallel between the presented algorithms analizing both strenght as well as vulnerable points; we also analyze the typology of possible attacks that aim digital watermarking so that a future study may approach the reduction of this kind of threats.

As written in the above rows, both algorithms that we have approached in our study have advantages and disadvantages. The Pitas algorithm is faster but the JAVA solution proposed by the authors is more secure. The JAVA solution uses a visible algorithm which does not have an artistical shape, the Pitas algorithm is an invisible one. We cannot say that either of them is a bad one or a good one. But we can see the good part of each and we can focus on improving or even eliminate the vulnerable points.

The algorithm proposed by the authors is far from being finalized but the performances reveled in this work recommend it for future development.

# References

[1] ***, http://cs.unitbv.ro/~costel/secupdfs/3_3_0_Valori%20Hash%20si%20IntegritateaDatel

[2] C. Chiru, *Contribuții la asigurarea securității software-ului*, PHD Theses, București, 2003

[3] I. J. Cox, J. Killian, F. T. Leighton and T. Shamoon, *Secure Spread Spectrum Watermarking for Multimedia*, IEEE Transactions on Image Processing, vol. 6, no. 12, December 1997, pp. 1673

[4] ***, Delphi User's Guide, *Delphi for Windows*, Borland International, 1995, available at: http://www.borland.com/resources/en/pdf/white_papers/delphi_2005_reviewers_guide.pdf

[5] B. Jellinek, *Invisible Watermarking of Digital Images for Copyright Protection*, Ph.Desis, Salzburg'00.

[6] G. C. Langelaar, R. L. Lagendijk and J. Biemond, *Robust Labeling Methods for Copy Protection of Images*, Proceedings of the SPIE Storage and Retrieval for Image and Video Databases V, volume 3022, San Jose, California, 1997

[7] P. Meerwald, *Digital Image Watermarking in the Wavelet Transform Domain*, Department of Scientific Computing, University of Salzburg, Austria, 2001

[8] C. Naforniță, *Creșterea securității rețelelor de comunicații de date prin autentificare bazată pe watermarking*, Research report, Universitatea Politehnica, Timișoara, 2006

[9] A. Tefas and I. Pitas, *Robust Spatial Image Watermarking Using Progressive Detection*, Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing (Vol. 3), 2001

[10] D. Bucerzan, A. Benczik, *Protecția dreptului de autor asupra produselor multimedia. Tehnici de Watermarking*, Dissertation report, Universitatea Aurel Vlaicu, Arad, 2008

[11] D.Bucerzan, C.Ratiu, *Aspects of a Watermark Solution*, Informatica Economica Journal, Vol.14, No. 4/2010

[12] ***, http://www.digitalwatermarkingalliance.org/app_docimage.asp, *Document and Image Security*, consultat la 14.07.2011

[13] D.Bucerzan, C.Ratiu, *Attacks on the Watermark Techniques*, 2-nd Symposium on Business Informatics in Central and Eastern Europe

CRINA RAȚIU
SC Daramec SRL

Loc. Șofronea, F.N., Jud. Arad
ROMANIA
E-mail: ratiu_anina@yahoo.com

DOMINIC BUCERZAN
Aurel Vlaicu University of Arad
Department of Mathematics-Informatics
310330 Arad, 2 Elena Drăgoi
ROMANIA
E-mail: dominic@bbcomputer.ro

MIHAELA CRĂCIUN
Aurel Vlaicu University of Arad
Department of Mathematics-Informatics
310330 Arad, 2 Elena Drăgoi
ROMANIA
E-mail: mihaeladacianacraciun@yahoo.com

# Playing with threads in Java 7

**Ernest Scheiber**

### Abstract

The new Java 7 version introduce the *fork-join* technique and the class *Phaser*. These techniques are compared with other tools introduced in earlier Java versions (`join, CountDownLatch, CyclicBarrier, ExecutorService`) in terms of the time to solve a simple embarrassing parallel test problem based on a synchronous algorithm for the successive approximation method.

In the same way a comparison is made between Java 6 and Java 7.

## 1    Introduction

The new Java 7 version introduce the *fork-join* technique that will allow recurrence definitions in parallel-concurrent programs as well as an implicit support for multi-core processors. There are some good tutorials for this Java technique [4],[3]. In [3] there is stated that, for the *fork-join* technique, the algorithmic speed increase linearly relative to the number of cores.

The class *Phaser* offers a flexible barrier synchronization technique based on phases associated to registered parties, i.e. threads.

We are interested to compare

- *fork-join* technique with `Phaser` and other tools introduced in earlier versions of Java (`join, CountDownLatch, CyclicBarrier, ExecutorService`) and

- the behavior of these tools between Java 6 and Java 7

in terms of the time to solve a simple embarrassing parallel test problem based on a synchronous algorithm for the successive approximation method.

## 2    A test problem

The Jacobi method to solve a linear system $Ax = b$, $A = (a_{i,j})_{1 \le i,j \le n} \in M_n(\mathbb{R})$, $b = (b_i)_{1 \le i \le n} \in \mathbb{R}^n$ is well known, [1]. For parallel computing, it is known a block version, too, [2]. We shall use the most simple version, with the iterations given by

$$u_i^{k+1} = \frac{1}{a_{i,i}}(b_i - \sum_{\substack{j=1 \\ j \ne i}}^{n} a_{i,j} u_j^k), \quad i \in \{1, \dots, n\},\ k \in \mathbb{N}.$$

There are known several convergence theorems. Our test data will satisfy the conditions of a convergence theorem. Starting with an arbitrary vector $u^0 \in \mathbb{R}^n$, the stopping rule will be

$$\|u^{k+1} - u^k\|_\infty < \varepsilon \qquad \text{or} \qquad k > k_{\max}, \tag{1}$$

where $\varepsilon$ is a given tolerance to be satisfied in given fixed number of iterations $k_{\max}$.

Each $u_i^{k+1}$ component will be computed within a thread. There is a synchronization problem: before starting a new iteration, all launched threads must finish their activities and the stopping test must be applied.

Our test data are defined in the class

```
public class Data{
    public double a[][]={{4,1,1},{1,4,1},{1,1,4}};
    public double b[]={9,12,15};
    public int n=a.length;
    public double x[]=new double[n];
    public double eps=1.0e-5;          // tollerance
    public int nmi=50;                 // maximum permissible number of iterations
    public int error;                  // error code
    public double y[]=new double[n];   // the solution
}
```

# 3   Java parallel-concurrent programming templates

The threads act in a pool. The most simple case considers the threads as elements of an array, collection, etc. Java 5 introduces specialized pools that implements the interface `java.util.concurrent.ExecutorService`. Also there is introduced the class `java.util.concurrent.CyclicBarrier` to synchronize the threads that have arrived at a specific point of their activities. The *fork-join* technique uses a specialized pool, `java.util.concurrent.ForkJoinPool`.

We have developed 6 classes to solve the linear system as it was described in the previous section:

1. The threads will be elements of an array and the required synchronization is obtain using the `join` method of the class `Thread`.

   The computation required by (1) is perform by the thread

```
class JacobiThread1 extends Thread{
    int myindex;

    public JacobiThread1(int myindex){
        this.myindex=myindex;
    }

    public void run(){
        double suma=0;
        for(int i=0;i<d.n;i++)
            if (i!=myindex)
                suma+=d.a[myindex][i]*d.x[i];
        d.y[myindex]=(d.b[myindex]-suma)/d.a[myindex][myindex];
    }
}
```

The instantiation of the threads, their starting and the synchronization is done by the method

```
void solve(){
    int ni=0;
    double nrm;
    JacobiThread1 t[]=new JacobiThread1[d.n];
    do{
        ni++;                              // A new iteration
        for(int i=0;i<d.n;i++){            // For any equation
            t[i]=new JacobiThread1(i);     // a new thread is instantiated
            t[i].start();                  // and is launched to work
        }
        try{                               // Synchronization
            for(int i=0;i<d.n;i++) t[i].join();  // on the condition that the threads
        }                                        // are terminated
        catch(InterruptedException e){}
        nrm=0;
        for(int i=0;i<d.n;i++){            // Stopping rule
            if(nrm<Math.abs(d.x[i]-d.y[i]))
                nrm=Math.abs(d.x[i]-d.y[i]);
            d.x[i]=d.y[i];
        }
    }
```

```
22      while ((nrm>=d.eps)&&(ni<d.nmi));
23      if (nrm<d.eps)
24        d.error=0;
25      else
26        d.error=1;
27    }
```

At each iteration there are created and started a new set of threads corresponding to the components of the new approximation to be computed.

2. Using the same thread code, the usage of the `java.util.concurrent.Phaser` class is based on a template presented in the *docs/api* of the Java 7 release. In this case the code of the *solve* method is

```
1  void solve(){
2    List<Runnable> tasks=new ArrayList<Runnable>(d.n);
3    for(int i=0;i<d.n;i++)
4      tasks.add(new JacobiThread1(i));
5    runTasks(tasks);
6  }
```

where the code based on the template is

```
1  static void runTasks(List<Runnable> tasks) {
2    int ni=0;
3    double nrm;
4    final Phaser phaser = new Phaser(1); //"1" to register self
5    do{
6      ni++;
7      for (final Runnable task : tasks) {
8        phaser.register();
9        new Thread() {
10          public void run() {
11            task.run();
12            phaser.arriveAndAwaitAdvance();
13          }
14        }.start();
15      }
16      try{Thread.sleep(5);}catch(InterruptedException e){}
17      nrm=0;
18      for(int i=0;i<d.n;i++){
19        if(nrm<Math.abs(d.x[i]-d.y[i])) nrm=Math.abs(d.x[i]-d.y[i]);
20        d.x[i]=d.y[i];
21      }
22      if (nrm<d.eps)
23        d.error=0;
24      else
25        d.error=1;
26    }
27    while((nrm>=d.eps)&&(ni<d.nmi));
28    phaser.arriveAndDeregister();
29  }
```

3. Very close to the previous template is as follows. The class `java.util.concurrent.CountDownLatch` introduced in Java 5 is used for synchronization. The code of the thread reproduced above has an additional line code (after the line 13)

```
countDownLatch.countDown();
```

In the solver method, at the beginning of each iteration it is instantiated an instance of the class `CountDownLatch`

```
countDownLatch=new CountDownLatch(d.n);
```

and the lines relating to the `join` method are replaced by

```
try{
  countDownLatch.await();
} catch (InterruptedException e){}
```

4. The synchronization will be done through a `CyclicBarrier` object. In this case the following thread is used to compute the attached component to each iterations:

```
class JacobiThread3 extends Thread{
    int myIndex;

  JacobiThread3(int index){
     myIndex=index;
  }

  public void run(){
     double s;
     while(!sfarsit){
         ni++;
         s=0;
         for(int i=0;i<d.n;i++){
             if(i!=myIndex){
                 s+=d.a[myIndex][i]*d.x[i];
             }
         }
         d.y[myIndex]=(d.b[myIndex]-s)/d.a[myIndex][myIndex];
         try{
             barrier.await();
         }
         catch(Exception e){}
     }
  }
}
```

The solver method is

```
public void solve(){
    StoppingRule test=new StoppingRule();
    barrier=new CyclicBarrier(d.n, test);
    for(int i=0;i<d.n;i++)
      (new JacobiThread3(i)).start();
    while(!sfarsit);
  }
```

As the name, the *StoppingRule* class is a thread required by the `CyclicBarrier` class containing the stopping rule:

```
class StoppingRule extends Thread{
    public void run(){
       double nrm=0,dif;
       for(int i=0;i<d.n;i++){
         dif=Math.abs(d.y[i]-d.x[i]);
         if(dif>nrm) nrm=dif;
         d.x[i]=d.y[i];
       }
       if((nrm<d.eps)||(ni/d.n>d.nmi))
          sfarsit=true;
       if(nrm<d.eps)
          d.error=0;
       else
          d.error=1;
    }
  }
```

5. Instead of the array containing the threads, a specialized thread pool is used. The pool implements the interface `java.util.concurrent.ExecutorService.` In this case the code of the solver method is

```
  public void solve(){
     StoppingRule test=new StoppingRule();
     barrier=new CyclicBarrier(d.n, test);
     ExecutorService executor=Executors.newFixedThreadPool(d.n);
     for(int i=0;i<d.n;i++){
       Runnable action=new JacobiThread2(i);
       executor.execute(action);
     }
     try{
       executor.shutdown();
       while(!executor.isTerminated());
     }
     catch(Exception e){}
  }
```

6. Finally, we use the *fork-join* technique. The threads pool is an instance of the class `java.util.concurrent.ForkJoinPool`. Instead of pure thread objects there are using descendants of the class `java.util.concurrent.ForkJoinTask`, especially `RecursiveAction` or `RecursiveTask`.

Because we do not have a recurrence formula, a trick is used. A fake `RecurentAction` class is introduced:

```java
static class JacobiTask extends RecursiveAction{
    private int index;

    JacobiTask(int index){
        this.index=index;
    }

    @Override
    protected void compute(){
        if(index==-1){
            for(int i=0;i<d.n;i++){
                JacobiTask action=new JacobiTask(i);
                action.fork();
            }
        }
        else{
            double s=0;
            for(int i=0;i<d.n;i++){
                if(i!=index){
                    s+=d.a[index][i]*d.x[i];
                }
            }
            d.y[index]=(d.b[index]-s)/d.a[index][index];
        }
    }
}
```

The invocation of the object *JacobiTask(-1)* is used to spawn recursively the needed tasks and to launch them asynchronously. The solver method is

```java
public void solve(){
    int processors=Runtime.getRuntime().availableProcessors();
    ForkJoinPool pool=new ForkJoinPool(processors);

    int ni=0;
    JacobiTask task=null;
    double nrm,dif;
    do{
        ni++;
        task=new JacobiTask(-1);
        pool.invoke(task);
        while(pool.isTerminated());
        nrm=0;
        for(int i=0;i<d.n;i++){
            dif=Math.abs(d.y[i]-d.x[i]);
            if(dif>nrm) nrm=dif;
            d.x[i]=d.y[i];
        }
        if(nrm<d.eps)
            d.error=0;
        else
            d.error=1;
    }
    while(nrm>=d.eps && ni<=d.nmi);
}
```

# 4   Results and conclusions

Each example is solved several times with the same data. For each run the beginning and ending time are obtained and hence the duration of the computation. Finally the average time is computed:

```
long averageTime=0,duration,beginTime,endTime;
for(int i=0;i<testsNumber;i++){
  beginTime=System.currentTimeMillis();
  //       Instantiates the main class
  //       Calls the method to solve the problem
```

```
    endTime=System.currentTimeMillis();
    duration=endTime-beginTime;
    averageTime+=duration;
}
duration/=testNumber;
```

In our experiences we have $testsNumber = 100$. The obtained computation time is only indicative because the time can't be measured with the same computer used to do the computation.

Instead of calling the `currentTimeMillis` method, the usage of the *perf4j* framework offers a more elaborate approach, but we will to keep the code as simple as possible.

On a 64 bit PC with an Intel CORE2 Duo CPU and Windows 7 Home Premium OS, we have run the classes using Java 6 and Java 7. The results of our computing experiences are given in the table

| Java distribution | Time(ms)/Example | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| jdk-6u27-windows-x64 | 16.70 | — | 21.37 | 62.71 | 5.78 | — |
| jdk-7-windows-x64 | 7.62 | 102.92 | 7.98 | 0.99 | 1.1 | 0.81 |

Of course the class based on the *Phaser* class (example 2) and the *fork-join* technique (example=6) can't be run using Java 6. Our conclusion is that the best results are obtained using the `ExecutorService` with *CyclicBarrier*. The discrepancy for *CyclicBarrier* technique between Java 6 and Java 7 is due to the need to introduce a waiting period after the launch of the threads (after the line 6 of the *solve* method). Java 7 doesn't require such a waiting time. A waiting period is needed by the usage of the *Phaser* class, too. We must also admit that `ExecutorService` and `CyclicBarrier` offer a simpler programming template and that the *fork-join* technique allows the use of recurrence formulas in parallel-concurrent programs.

The present Java 7 release is more efficient than the last Java 6 release. The relations between the older tools are unchanged.

# References

[1] D. Kincaid, W. Cheney, *Numerical Analysis Mathematics of Scientific Computing.* Brooks/Cole Publishing Company, Pacific Grove, California, 1991.

[2] J.M. Bahi, S. Contassot-Vivier, R. Couturier, *Parallel Iterative Algorithms. From Sequential to Grid Computing.* Chapman & Hall/CRC, Boca Raton, 2007.

[3] J. Ponge, Fork and Join: Java can Excel at Painless Parallel Programming Too! `http://www.oracle.com/technetwork/articles/java/fork-join-422606.html`.

[4] * * *, Java Fork/Join for Parallel Programming. `http://www.javacodegeeks.com/2011/02/java-forkjoin-parallel-programming.html`.

Scheiber Ernest
Transilvania University of Braşov
Department of Computer Science
Str. I. Maniu 50
ROMANIA
E-mail: *scheiber@unitbv.ro*

# On an approach for cubic Bézier interpolation

**Dana Simian, Corina Simian**

**Abstract**

The aim of this paper is to introduce an interpolation approach using cubic Bézier curves and surfaces which shape is controlled by means of two parameters. Implementation of the proposed method is realized using MATLAB.

# 1 Introduction

Bézier curves and surfaces have a wide applicability in Computer Aided Geometric Design (CAGD). In the last years, many studies were dedicated to obtain new classes of Bézier curves and surfaces suitable for the approximation process of various shapes. One direction of generalizations consists of replacing the Bernstein basis in the Bézier curves and surfaces parametric expression with a generalized Bernstein basis. Bernstein–Stancu polynomials, q-Bernstein polynomials, q-Bernstein–Stancu polynomials are examples of this kind of bases ([4], [6], [7], [8], [12]). The shape preserving properties of various operators which generalize the Bernstein operator are essential in defining new generalization of Bernstein basis which can be used in Bézier curves and surfaces theory. The quasi-Bézier curves and surfaces are based on a class of polynomial basis functions with n adjustable shape parameters ([5]). In [11], umbral calculus is used to generalize Bernstein polynomials and Bézier curves. Farin, introduced in [3] a class of 3D A- Bézier curves, defined by their degree, a vector v and a matrix M. These curves are used in the design of those parts of a car which are critical for its aesthetic appearance: parts of the hood, fender, or roof. An embedded Bézier shape parameterizations is constructed and employed in [1], to define multi-level optimum-shape algorithms.

In practice, a designer needs to fit a curve to digitized points. The aim of this paper is to study and implement an approach to design an interpolation cubic Bézier curve which passes through 4 digitizes points and depends on two parameters which control the curve's shape. Using tensor product method we define also an interpolation cubic Bézier surface.

The article is organized as follow: in the section 2 we present the parametric and matrix representation of Bézier curves and surfaces. In section 3 is introduces the proposed approach for obtaining interpolation cubic Bézier curves and surfaces. Section 4 is dedicated to the implementation and results' analysis. Conclusion and further directions of study can be found in section 5.

# 2 Bézier curves and surfaces

Bézier curves and surfaces are parametric curves and surfaces expressed in Bernstein basis using a set of control points as coefficients. The most used are the cubic Bézier curves and surfaces and we will referee to them in the following of the paper.

The equation of a cubic Bézier curves is:

$$c(t) = (x(t), y(t), z(t)) = \sum_{i=0}^{3} b_i \cdot B_i^3(t), t \in [0,1] , \tag{1}$$

$$B_i^3(t) = \binom{i}{3} t^i (1-t)^{3-i} \tag{2}$$

being the Bernstein polynomials of degree 3 and $b_i = (bx_i, by_i, bz_i)$ the control points which form the control polygon.

**Remark 1.** *A cubic Bézier curve is determined by its 4 control points.*

The properties that make the Bézier curves suitable for CAGD applications are affine invariance, (invariance under affine transformations), convex hull property (Bézier curve lies in the convex hull of the control points), endpoint interpolation and pseudo – local control (a change of one of the control point affects the Bézier curve only in the region of this control point). More details about Bézier curves properties can be found in the Farin comprehensive book [2].

A cubic Bézier surface is given by the equation:

$$s(u,v) = \sum_{i=0}^{3} \sum_{j=0}^{3} b_{ij} \cdot B_j^3(u) \cdot B_i^3(v), u,v \in [0,1] , \tag{3}$$

$b_{ij} = (bx_{ij}, by_{ij}, bz_{ij})$ being the control points of the surface and $B_j^3(u), B_i^3(v)$ the Bernstein polynomials given in (2).

**Remark 2.** *A Bézier surface is determined by its 16 control points*.

The properties enumerated for Bézier curves are also valid for Bézier surfaces.

For computational reasons it is very useful to transform the equations (1) and (3) in matrix form.

Matrix form of a Bézier curve:

$$c(t) = b \cdot B(t), \text{ with } c(t) \in \mathsf{M}_{3,1}, b \in \mathsf{M}_{3,4}, B(t) \in \mathsf{M}_{4,1}, t \in [0,1] \tag{4}$$

*c(t)* represents the column vector of the coordinates of a point from the Bézier curve corresponding to the value *t* of the parameter.

*b* is the matrix of control points coordinates. Each column correspond to a control point coordinates.

*B(t)* is the column vector of Bernstein polynomials computed for the value *t* of the parameter.

Matrix form of a Bézier surface:

$$s(u,v) = B'(v) \cdot b \cdot B(u) \tag{5}$$

*(5)* represents in fact 3 equations, one for each coordinate

$$x(u,v) = B'(v) \cdot bx \cdot B(u)$$

$$y(u,v) = B'(v) \cdot by \cdot B(u)$$

$$z(u,v) = B'(v) \cdot bz \cdot B(u)$$

$x(u,v), y(u,v), z(u,v) \in \mathsf{M}_{1,1} \equiv \mathbf{R}$ represent the coordinates of a point from the Bézier surface, corresponding to the values *u* and *v* of the parameters.

$bx, by, bz \in \mathsf{M}_{4,4}$ are matrixes containing the coordinates of the 16 control points of the surface.

*B(u), B(v)* are the column vectors of Bernstein polynomials computed for the values *u* and *s*.

We denoted by *M'* the transpose of the matrix *M*.

# 3. Bézier interpolation. Main results.

The approximation problem of more complicated curves has been solved by using piecewise Bézier curves satisfying different conditions in the junction points. From a practical point of view piecewise cubic Bézier curves of class $C^1$ and $G^1$ are preferred. Many interpolation problems as well as many solutions have been formulated: $C^1$ piecewise cubic Hermite interpolation, point-normal interpolation, F-Mill interpolation (see [2]), etc. The most complex design problems of curves and surfaces require techniques related to Bézier splines, B-splines and NURBS. A brief presentation of these concepts can be found in [10] and many mathematical and algorithmic details are included in [2].

Our aim is to present and to implement a method for solving an interpolation problem using a single Bézier curve dependent of two parameters and to generalize the method in the case of surface interpolation using tensor product method. We will show that the Bézier curve shape is strong influenced by the two parameters.

**Problem 1. (Interpolation using Bézier curve)**
*Being given 4 points on a curve C, find a cubic Bézier curve that passes through the given points. We suppose that two of the given points are the endpoints of the curve C.*

Let $P_i, i \in \{0,...,3\}$ be the interpolation points. They satisfy the equation of the Bézier curve for some values of the parameter $t$.

Taking into account the *Remark 1*, for solving the *Problem 1* it is sufficient to find the 4 control points.

From the endpoint interpolation property of the Bézier curve, we have obviously

$$c(0) = P_0 = b_0 \qquad (6)$$

and

$$c(1) = P_3 = b_3 \qquad (7)$$

Let $t_1 < t_2 \in (0,1)$ be the values of parameters for which $c(t_i)=P_i$, $i \in \{1,2\}$. Then

$$c(t_i) = (1-t_i)^3 b_0 + 3t_i(1-t_i)^2 b_1 + 3t_i^2(1-t_i) + t_i^3 b_3, i \in \{1,2\} \qquad (8)$$

The system (6), (7), (8) can be formulated as

$$A \cdot b = P \qquad (9)$$

with

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ (1-t_1)^3 & 3t_1(1-t_1)^2 & 3t_1^2(1-t_1) & t_1^3 \\ (1-t_2)^3 & 3t_2(1-t_2)^2 & 3t_2^2(1-t_2) & t_2^3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad (10)$$

and

$$b = \begin{pmatrix} bx_0 & by_0 & bz_0 \\ bx_1 & by_1 & bz_1 \\ bx_2 & by_2 & bz_2 \\ bx_3 & by_3 & bz_3 \end{pmatrix}; P = \begin{pmatrix} Px_0 & Py_0 & Pz_0 \\ Px_1 & Py_1 & Pz_1 \\ Px_2 & Py_2 & Pz_2 \\ Px_3 & Py_3 & Pz_3 \end{pmatrix}$$

We have

$$\det(A) = 9t_1 t_2 (1-t_1)(1-t_2)(t_2-t_1) \neq 0 \qquad (11)$$

The control points are given by

$$b = A^{-1} \cdot P \qquad (12)$$

From (11) we have that

$$\lim_{t_i \to 0} A(t_1,t_2) = \lim_{t_i \to 1} A(t_1,t_2) = \lim_{t_1 \to t_2} A(t_1,t_2) = 0 \qquad (13)$$

154

In practice, the points $P_i$ are usually obtained by measurements. A method which considers that the points are uniformly distributed and correspond to the values of parameters $t_1 = \dfrac{1}{3}, t_2 = \dfrac{2}{3}$ on the interpolation curve has been used before, but it is difficult to estimate exactly these points and for the complicated shape of the curve the method does not offer good results. The values $t_1$ and $t_2$ from our approach can be modified in order to obtain an enough complicated shape by using a single cubic Bézier curve. More, it is not necessary an expensive method to measure the coordinates of the desired interpolation points.

The conclusion is that it is of interest to implement relation (12) such that using a continuous variation of the two parameters to obtained the desired shape.

Relation (13) suggests us that exist real values $\varepsilon$, $\delta$ and $\lambda$ such that for

$$t_1 < \varepsilon, 1 - t_2 > \delta \text{ and } |t_1 - t_2| < \lambda \qquad (14)$$

the matrix A from (12) is close to a singular matrix. The parameters $t_i$, $i \in \{1,2\}$, satisfying the inequalities (14) with the corresponding values $\varepsilon$, $\delta$ and $\lambda$ small enough will be call *limit parameters* and we will refer to the inequalities from (14) as *limit inequalities*.

So it is interesting to study the dependence of the shape properties depending on the relative position of these two parameters.

The implementation details will be presented in section 4.

**Problem 2. (Interpolation using Bézier curve)**
*Being given 16 points on a surface S, find a cubic Bézier surface that passes through the given points.*

Let be $P_i, i \in \{0,...,15\}$ the interpolation points. The surface is generated using the tensor- product method. Let denote by $px, py, pz \in \mathsf{M}_{4,4}$ the matrixes containing the coordinate of the 16 interpolation points. In the tensor- product method we apply relation (12) first for the lines of the matrix px, py, pz and then for the columns of the new obtained matrix. The values t1, t2 were chosen the same for all the lines of the three matrixes. Another possibility is to choose different values $t_1^j, t_2^j$ for the lines $j \in \{1,2,3\}$. The observation regarding the limits values for the parameters $t_1$, $t_2$ remain the same as in the case of Bézier curve.

# 4   Implementation details and results

We implemented our proposed approach, for curves and surfaces interpolation, using MATLAB. The main reasons for which we choose MATLAB were:
- Capabilities to easy manipulate and operate with matrix structures
- Powerful function for graphical representation
- Possibility of integration of programs in a high quality Graphical User Interfaces
In [9] we presented a software system implemented in MATLAB for the shape design of punches used in deformation process and the analysis and comparison of the behaviour of different materials function of the desired shapes.


**4.1 Implementation of our approach for interpolation of curves.**

The computation of the Bézier curve points is made using the equation (4). The implemented function returns the coordinates of the control points. For the input data $P = \begin{pmatrix} 0 & 20 & 40 & 70 \\ 0 & 60 & 60 & 0 \end{pmatrix}$, the influence of the parameters $t_1$, $t_2$ is illustrated in the figures below:
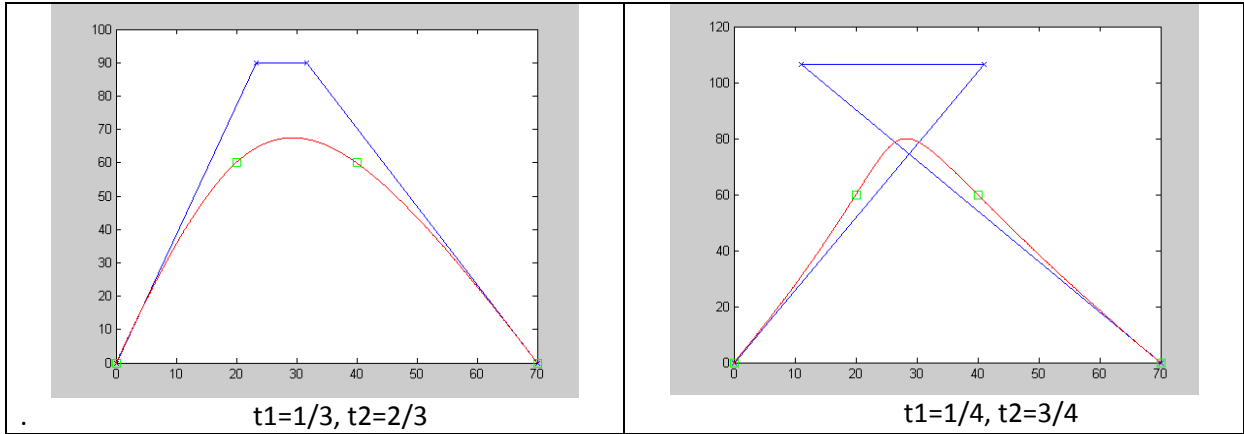
Fig.1 -  Bézier curves obtained for 0<t1<0.5<t2<1

The convexity of the two curves is different. The equidistant points lead to a curve without inflexion points.
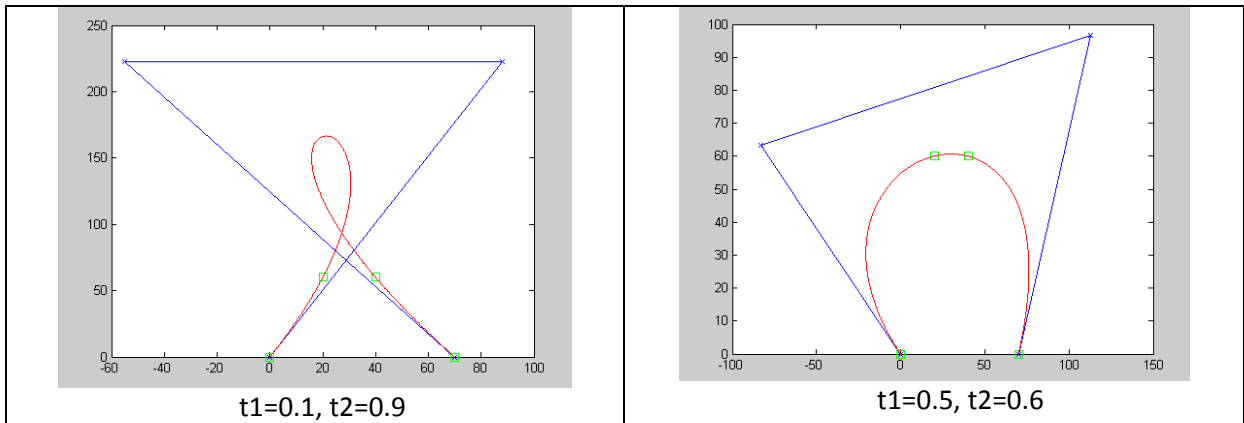


Fig. 2a  - Bézier curve with  t1=0.1, 1-t2=0.1      Fig. 2b -  Bézier curve with  t2-t1=0.1

In Fig. 2a and Fig. 2b the parameters $t_i$ satisfy the limit inequalities (13) with the values of $\varepsilon$, $\delta$ and $\lambda$ equal to 0.1.
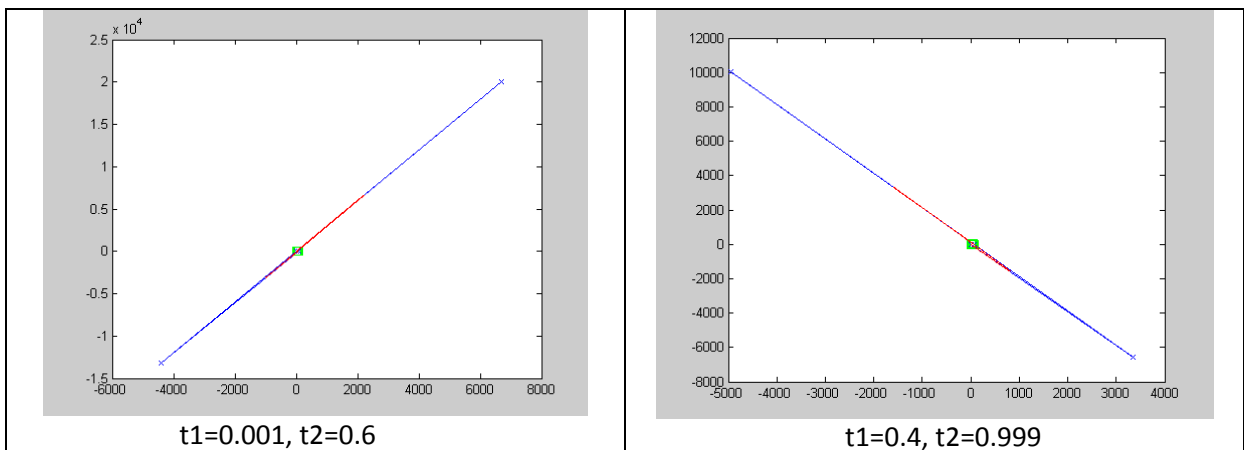


Fig. 3 – Bézier curves obtained for  limit parameters ε =0.001, δ=0.001

For a small value of the limit parameters $\varepsilon$ (left graphic) or $\delta$ (right grahic), the Bézier curves degenerate in a straight line.

t1=0.495, t2=0.501

t1=0.7, t2=0.701
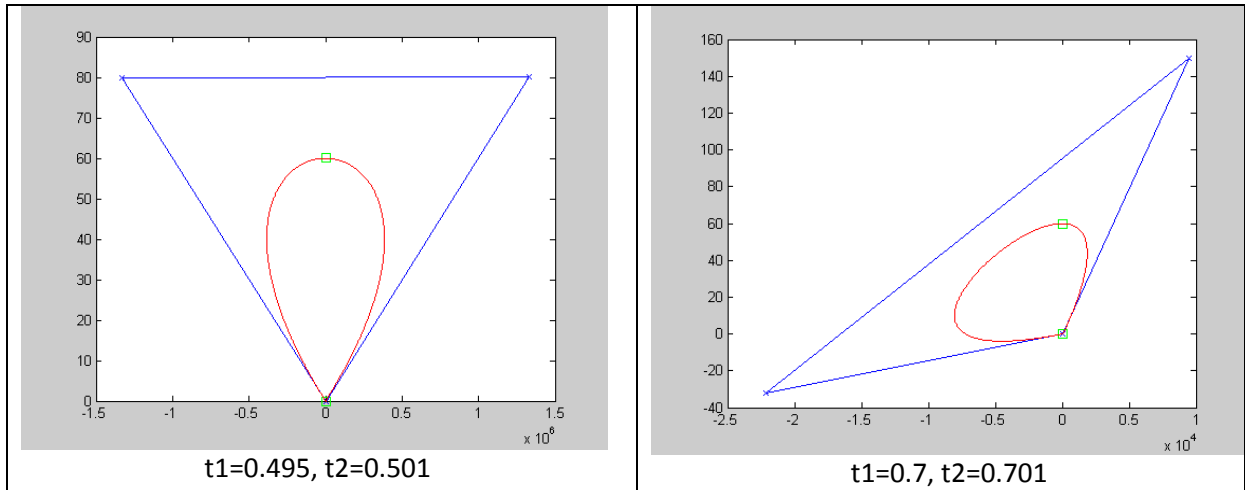
Fig. 4 – Bézier curves obtained for  limit parameter λ =0.001

For a small value of limit parameter λ, the control polygon reduces to a triangle i.e. the cubic Bézier curve reduces to a quadratic one.



t1=0.1, t2=0.3

t1=0.2, t2=0.3
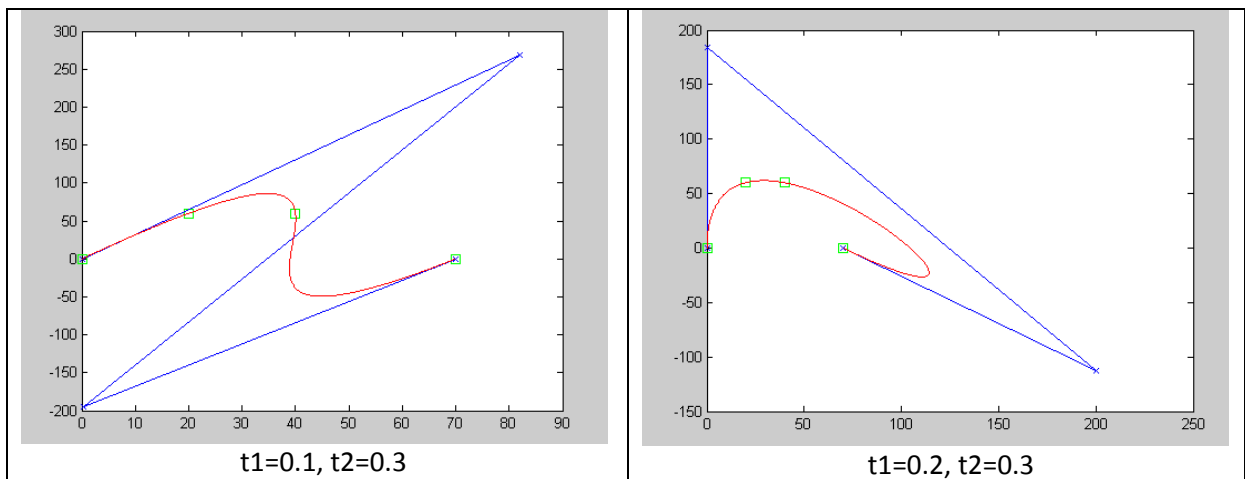
Fig. 5 - Bézier curves obtained for 0<t1<t2<0.5



t1=0.6, t2=0.8

t1=0.8, t2=0.9
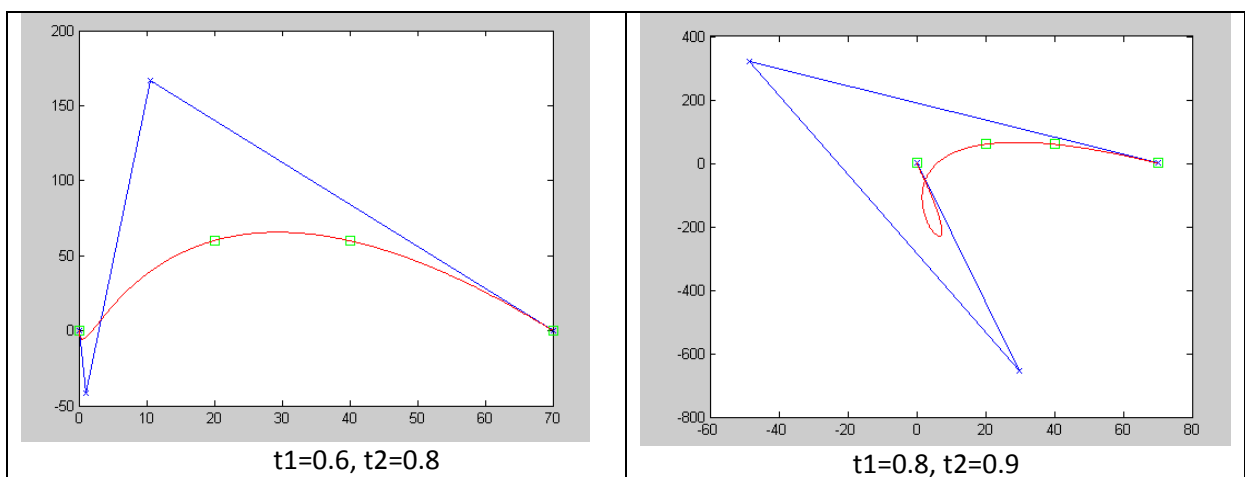
Fig. 6 -  Bézier curves obtained for  0.5<t1<t2<1

If the parameters are located in the same half of the interval (0, 1), more complicated shapes are obtained.

### 4.2 Implementation of our approach for interpolation of surfaces.

In the case of surfaces interpolation we implemented the tensor product method for the coordinates x, y and z, using the interpolation function defined for curve interpolation.



.            t1=1/3, t2=2/3                        t1=1/4, t2=3/4
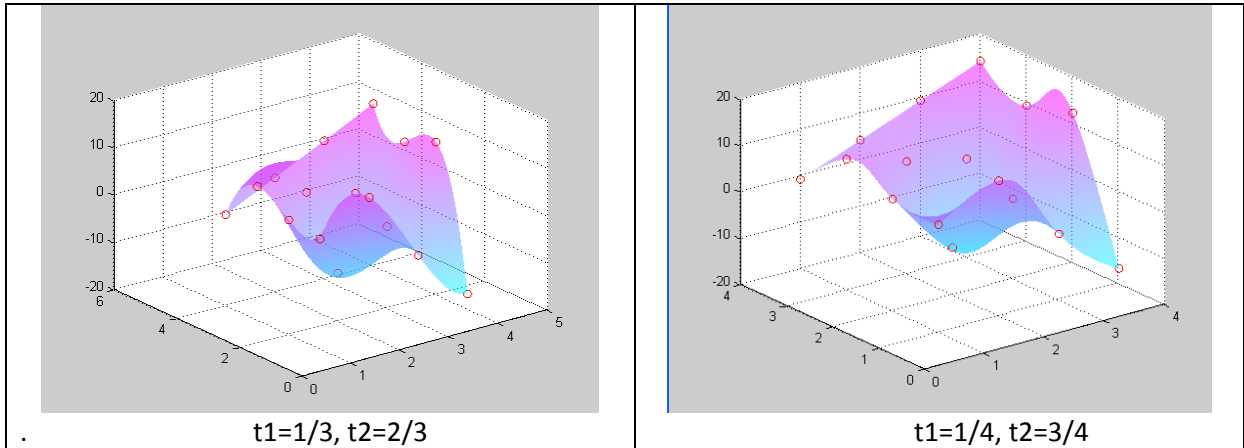
Fig.7 -  Bézier surfaces obtained for 0<t1<0.5<t2<1



t1=0.1, t2=0.9                         t1=0.5, t2=0.6
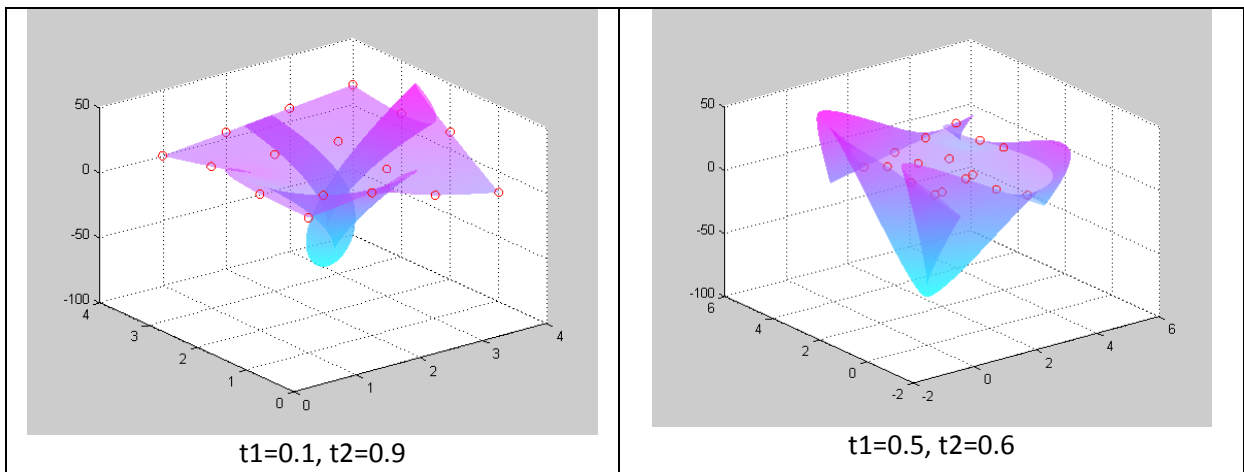
Fig. 8a - Bézier surface  for t1=0.1, 1-t2=0.1        Fig. 8b - Bézier surface   for t2-t1=0.1



t1=0.1, t2=0.2                          t1=0.8, t2=0.9
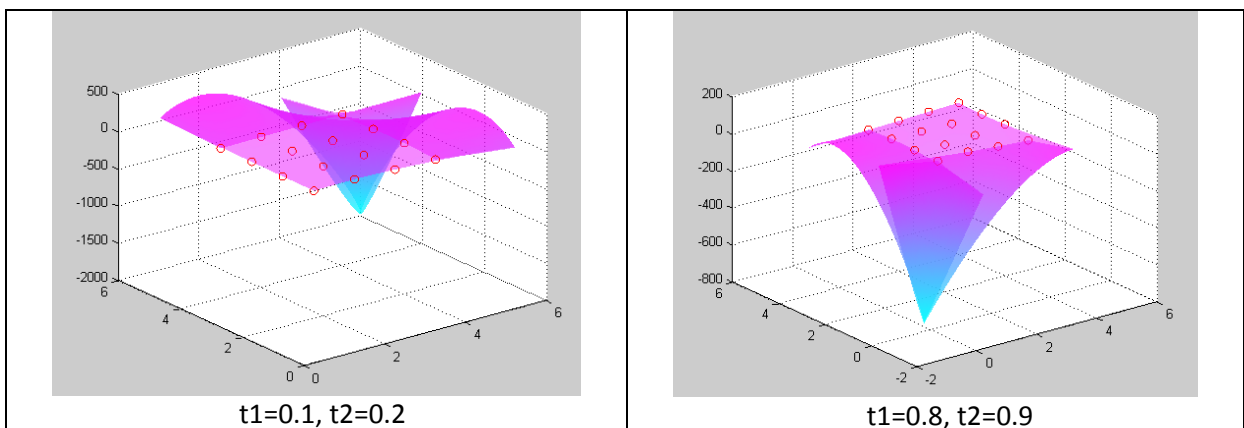
Fig. 9a  - Bézier surface for  0<t1<t2<0.5        Fig. 9b  - Bézier surface  for  0.5<t1<t2<1

The surfaces presented in the Fig. 7 – Fig. 9   were obtained for the input data

$$Px = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, \quad Py = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{pmatrix}, \quad Py = \begin{pmatrix} 3 & 9 & -6 & -17 \\ 4 & -10 & -3 & 12 \\ 8 & 4 & 1 & 9 \\ -1 & 4 & 9 & 14 \end{pmatrix}$$

# 5  Conclusions and further directions of study

In this paper we propose an approach which allows us to modify in an easy way the shape of a cubic Bézier interpolation curve and surface using two real parameters. The relative position of these parameters and their distance from 0 (for the first parameter) and respective 1 (for the second parameter) has a significant influence. An advantage of the method is that the interpolation points can be situated anywhere on the original curve or surface and the convexity properties of the interpolation curve or surface can be obtained by parameters modification. We can also use our cubic Bézier interpolation curves in order to obtain a piecewise Bézier curve of $G^1$ class.

A further direction of study is represented by the mathematical foundation of the limit cases of the parameters and mathematical study of the parameters' values –shape dependence.

# References

[1]  J.-A. Désidéri, *Hierarchical Optimum-Shape Algorithms, Using Embedded Bezier Parameterizations,* Numerical Methods for Scientific Computing Variational Problems and Applications, Y. Kuznetsov, P. Neittanmaki and O. Pironneau (Eds.) CIMNE, Barcelona, 1-12, 2003.

[2]  G. Farin, *Curves and Surfaces for Computer Aided Geometric Design,* Academic Press, Boston, 1996.

[3]  G. Farin*, Class A Bezier Curves,* Computer Aided Geometric Design 23, Elsevier Press, 573-581, 2006.

[4]  A.D. Gadjiev, A.M. Ghorbanalizadeh, *Approximation properties of new type Bernstein–Stancu polynomials of one and two variables,* Applied Mathematics and Computation 216, 890–901, 2010.

[5]  Xi-An Han, YiChen Ma, XiLi Huang, *A novel generalization of Bézier curve and surface, Journal of Computational and Applied Mathematics, 217, 180-193, 2008.*

[6]  G. Nowak,  *A de Casteljau Algorithm for q-Bernstein-Stancu Polynomials,* Hindawi Publishing Corporation Abstract and Applied Analysis, Volume 2011, Article ID 609431, 13 pages doi:10.1155/2011/609431.

[7]  Halil Oru, George M. Phillips, *q-Bernstein polynomials and Bézier curves,* Journal of Computational and Applied Mathematics*, 15, 1-12,* 2003.

[8]  G. M. Phillips, *A de Casteljau Algorithm for Generalized Bernstein Polynomials,* BIT 36:1, 232-236, 1996.

[9]  D. Simian, C. Simian, *Bézier Techniques Applied in a Complex Virtual Testing System Design,* Academic Journal of Academic Engineering, Vol. 8, Issue 2/2010, 75-81, 2010.

[10] D. E. Ulmet, *Swept Surfaces in Computer Aided Geometric Design,* Proceedings of International workshop  New Approaches, Algorithms and Advanced Computational Techniques in Approximation Theory and its Applications, "Lucian Blaga" University Press,  Sibiu, 19-31, 2007.

[11] R. Winkel, *Generalized Bernstein Polynomials and Bézier Curves: An Application of Umbral Calculus to Computer Aided Geometric Design, 1-25, 2001.*

[12] Lianying Yun, Xueyan Xiang, *On Shape-Preserving Properties and Simultaneous Approximation of Stancu Operator,* Analysis in Theory and Applications, Volume 24, Number 2, 195–204, 2008, DOI:10.1007/s10496-008-0195-0.

DANA SIMIAN                          CORINA SIMIAN
"Lucian Blaga" University            Zütich University
Department of Informatics            Institut for Mathematics
ROMANIA                              SWITZERLAND
E-mail: dana.simian@ulbsibiu.ro      E-mail: corina.simian@math.uzh.ch

# Supervised Approach to Learning Multivariate Linear Systems

**Luminiţa State, Iuliana Paraschiv-Munteanu**

### Abstract

We consider the proble   of developing a learning fro   data sche  e for the unknown input-output dependency of a syste   **S** of linear type in the sense that the $m$-di  ensional output of **S** results by co   bining in a linear way the effects of $n$ observable variables and the effects of several unobservable latent variables. The effects of the latent variables on the output is treated as additive noise, that is being given the observable vector $x$, the syste   co   putes the output $y = \beta^T \begin{pmatrix} 1 \\ x \end{pmatrix} + \varepsilon$, where $\beta$ is a $(n+1) \times m$   atrix and $\varepsilon$ is a $m$-di  ensional Gaussian variable. In the paper the   athe  atical argu  ents for the esti  ation sche  e based exclusively on a finite size set of observations is provided. We present an experi  ental evaluation of the quality of the resulted learning sche  e in order to establish conclusions concerning their accuracy and generalization capacities, the evaluation being perfor  ed in ter  s of   etric, probabilistic and infor  ational criterion functions.

## 1   Introduction

The tremendous growth in practical applications of machine learning over the past decade has been accompanied by a wide variety of important developments in the underlying algorithms and techniques that make use of concepts and results coming from several areas as mathematical statistics, computer science and engineering.

Since the main aim of machine learning is to obtain computer programs that are able to extract information from samples of data and as well as knowledge from the past experience and include them in the process of solving problems of high complexity, the research methodology in the field of machine learning is essentially based on a large class of concepts and results coming from mathematical statistics, neural and evolutionary computation, brain models, adaptive control theory and so on.

The aim of the research was to develop a model free learning methodology in order to predict a system behavior, conventionally denoted by **S** on the basis of finite size sequence of observations. In real life applications, the input data sequence are either obtained in a controlled way, that is the observer knows in advance the generating mechanism, or, in an uncontrolled way when the generating mechanism is ignored by the observer. We denote by **G** (Generator) the component that supplies a series of samples from an $n$-dimensional space loaded as inputs to us. The learning environment is assumed to be of supervised type, that is the output of **S** is available to the observer. The goal is to develop a learning component **L** on the basis of a finite size set of input-output observations, that is to infer the unknown the input-output dependency of **S** and use it for further predictions. The general scheme of our model is presented in Figure 1 ([3]).

The learning component **L** is responsible for a class of possible models for the unknown dependency corresponding to **S**. In other words, the learning component **L** implements a class of hypothesis (models) $\Omega$, such that to each particular hypothesis $\omega \in \Omega$ corresponds a function $\varphi_\omega : \mathcal{X} \to \mathcal{Y}$ defined on the space
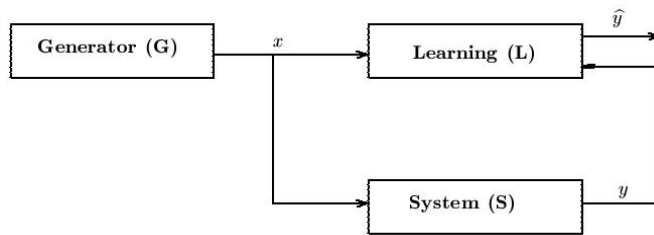
Figure 1: The learning environment.

of inputs $\mathcal{X}$ and taking values in the space of outputs $\mathcal{Y}$. For each particular input $x_0$, $\widehat{y}_0 = \varphi_\omega(x_0)$ is the estimate of the **S**'s output corresponding to $x_0$ in case of the model $\omega$. Being given a criterion function $\mathcal{C}$ that expresses numerically the fitness of each model with respect to the available evidence $E$, about **S**, the best model $\omega_0(E)$ is a solution of the optimization problem

$$\arg\left(\text{optimize}_{\omega\in\Omega}\,\mathcal{C}(\omega, E)\right). \tag{1}$$

In the case of supervised learning the available evidence $E$ is represented by a finite set of pairs $\{(x_i, y_i)\,,\, 1 \leq i \leq N\} \subset \mathcal{X} \times \mathcal{Y}$, where each $y_i$ is the actual output of **S** for the input $x_i$. If we assume that the unknown dependency is of deterministic type, that is the inputs and the outputs of **S** are functionally related a reasonable choice of the criterion function $\mathcal{C}$ is the arithmetic mean of the square errors, that is for each $\omega \in \Omega$,

$$\mathcal{C}(\omega, E) = \frac{1}{N}\sum_{i=1}^{N}\|y_i - \varphi_\omega(x_i)\|^2. \tag{2}$$

The optimization problem (1) becomes

$$\arg\left(\min_{\omega\in\Omega}\,\mathcal{C}(\omega, E)\right), \tag{3}$$

and its solutions are called the Minimum Square Errors (MSE) models computed on the basis of $\{(x_i, y_i)\,,\, 1 \leq i \leq N\}$ ([14]).

A more realistic approach is to accept that besides the inputs, the outputs of **S** are also influenced by a series of unknown number of unobservable factors referred as latent variables. In this case the cumulated effects of the latent variables can by only modelled in probabilistic terms, assuming some class of multivariate probability distributions, each hypothesis corresponding to a certain input-output dependency combined with a probabilistic model for the latent vector. For simplicity sake, we consider that the latent vector is a continuous random vector, that is to each hypothesis $\omega \in \Omega$ corresponds a conditional density function $f(\cdot|\cdot\,;\,\omega)$ ([9], [11]). Put in other words, for each $\omega \in \Omega$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$, $f(y|x\,;\,\omega)$ expresses 'the chance' of getting the output $y$ for the input $x$ in case of the model $\omega$. If the available evidence about **S** is $\{(x_i, y_i)\,,\, 1 \leq i \leq N\}$ then a reasonable choice of $\mathcal{C}(\omega, E)$ is the likelihood function. If we assume that the inputs $x_1, \ldots, x_N$ are independently generated by **G** then

$$\mathcal{C}(\omega, E) = \prod_{i=1}^{N} f(y_i|x_i\,;\,\omega), \tag{4}$$

and the optimization problem (1) becomes

$$\arg\left(\max_{\omega\in\Omega}\,\mathcal{C}(\omega, E)\right). \tag{5}$$

The solutions of (5) are the MLE (Maximum Likelihood Estimation) models computed on the basis of $\{(x_i, y_i)\,,\, 1 \leq i \leq N\}$.

## 2   A MLE-based approach in learning multivariate linear systems

In our work we assume that the inputs come from a $n$-dimensional real space and for each input $x$ the output $y$ of $\mathbf{S}$ is a tuple of $m$ real variables. It is also assumed that the output dependency on the input is of linear type, that is the influence of the input is given by $\beta^T x$ where $\beta$ is an unknown $(n+1) \times m$ matrix. The effects of the latent variables can by taken as noise additively superimposed on the influence of the inputs, that is the output of $\mathbf{S}$ is $y = \beta^T \begin{pmatrix} 1 \\ x \end{pmatrix} + \varepsilon$ where $\varepsilon$ is a $m$-dimensional Gaussian distributed random vector, $\varepsilon \sim \mathcal{N}(\mu, \Sigma)$. The parameters $\mu \in \mathbb{R}^m$ and $\Sigma \in \mathcal{M}_m(\mathbb{R})$, symmetric positive definite matrix, are unknown. Therefore, the dimension of the space of hypotheses $\Omega$ is $m(m + n + 2)$, and for each particular tuple of $m(m + n + 2)$ parameters, $\omega = (\beta, \mu, \Sigma)$ defines a model of $\mathbf{S}$. We denote by $f(y|x\,;\,\omega)$ the conditional density function on the output space corresponding to the model $\omega$.

Several intuitively justified criterion functions can be considered in order to quantitatively express the quality of each possible model with respect to available sequence of observations. For each criterion function, the identification of the "fittest" model reduces to solving an constrained/unconstrained optimization problem ([6]).

Let $\mathcal{S}_N = \{(x_i, y_i)\,,\, 1 \le i \le N\}$ be the sequence of input-output observations taken on $\mathbf{S}$. Being given the model $\omega = (\beta, \mu, \Sigma)$, the estimate $\widetilde{y}$ of the actual output of $\mathbf{S}$ being given the input $x$ is a random vector of density function

$$f(\widetilde{y}|x, \omega) = \frac{1}{\sqrt{(2\pi)^m |\Sigma|}} \exp\left\{ -\frac{1}{2} \left( \widetilde{y} - \beta^T z - \mu \right)^T \Sigma^{-1} \left( \widetilde{y} - \beta^T z - \mu \right) \right\},$$

where $z = \begin{pmatrix} 1 \\ x \end{pmatrix}$.

Consequently, the log-likelihood function is

$$l(\beta, \mu, \Sigma, \mathcal{S}_N) = -\frac{mN}{2}\ln(2\pi) - \frac{N}{2}\ln|\Sigma| - \frac{1}{2}\sum_{i=1}^{N} \left( (y_i - \mu)^T - z_i^T \beta \right) \Sigma^{-1} \left( (y_i - \mu) - \beta^T z_i \right),$$

and the best model from the point of view of maximum likelihood principle is a solution of the constrained optimization problem

$$\begin{cases} \min\limits_{\beta, \mu, \Sigma} (\Phi(\beta, \mu, \Sigma)) \\[2mm] \Sigma \in \mathcal{M}_m(\mathbb{R}) \text{ symmetric and positive defined,} \end{cases}$$

where

$$\Phi(\beta, \mu, \Sigma) = N\ln|\Sigma| + \sum_{i=1}^{N} \left( (y_i - \mu)^T - z_i^T \beta \right) \Sigma^{-1} \left( (y_i - \mu) - \beta^T z_i \right).$$

Let us denote

$$\begin{aligned} Y &= (y_1, \ldots, y_N) \in \mathcal{M}_{m \times N}(\mathbb{R}),\ Z = (z_1, \ldots, z_N) \in \mathcal{M}_{(n+1) \times N}(\mathbb{R}), \\ u &= (1, \ldots, 1)^T \in \mathbb{R}^N,\ A = \mathbf{I}_N - \frac{1}{N}uu^T. \end{aligned} \tag{6}$$

For any matrix $B$, we denote by $B^+$ the generalized inverse (Penrose pseudo-inverse) of $B$.

**Theorem 1** *The objective function $\Phi(\beta, \mu, \Sigma)$ has an unique critical point $(\beta_0, \mu_0, \Sigma_0)$ where*

$$\beta_0 = \left( Y(ZA)^+ \right)^T,\ \mu_0 = \frac{1}{N}\left( Yu - Y(ZA)^+ Zu \right),\ \Sigma_0 = \frac{1}{N}Y\left( A - (ZA)^+(ZA) \right)Y^T, \tag{7}$$

*and $\Sigma_0$ is a symmetric and positive semi-defined matrix.*

**Proof.** The generalized gradients of $\Phi(\beta, \mu, \Sigma)$ with respect to $\beta$, $\mu$, and $\Sigma$ respectively, are

$$\begin{cases} \nabla_\beta \, \Phi\left(\beta, \mu, \Sigma\right) = -\left(ZY^T - Zu\,\mu^T - ZZ^T\beta\right)\Sigma^{-1}, \\ \nabla_\mu \, \Phi\left(\beta, \mu, \Sigma\right) = -\Sigma^{-1}\left(Yu - N\mu - \beta^T Zu\right), \\ \nabla_\Sigma \, \Phi\left(\beta, \mu, \Sigma\right) = \Sigma^{-1}D\Sigma^{-1} - \dfrac{1}{2}\,\mathrm{diag}\left(\Sigma^{-1}D\Sigma^{-1}\right), \end{cases}$$

where

$$D = N\Sigma - YY^T + \left(\mu\left(Yu\right)^T + \left(Yu\right)\mu^T\right) + \left(\left(ZY^T\right)^T \beta + \beta^T ZY^T\right) - N\left(\mu\mu^T\right) -$$
$$\left(\mu\left(Zu\right)^T \beta + \beta^T\left(Zu\right)\mu^T\right) - \beta^T ZZ^T \beta\,.$$

From the system

$$\begin{cases} \nabla_\mu \, \Phi\left(\beta, \mu, \Sigma\right) = \mathbf{0}_m, \\ \nabla_\beta \, \Phi\left(\beta, \mu, \Sigma\right) = \mathbf{O}_{n+1,m}, \end{cases}$$

since $|\Sigma| \neq 0$, we get

$$\begin{cases} Yu - N\mu - \beta^T Zu = \mathbf{0}_m \\ ZY^T - Zu\,\mu^T - ZZ^T\beta = \mathbf{O}_{n+1,m}\,, \end{cases}$$

that is

$$\beta = \left(Y\left(ZA\right)^+\right)^T = \beta_0\,, \quad \mu = \frac{1}{N}\left(Yu - Y\left(ZA\right)^+ Zu\right) = \mu_0\,.$$

Replacing $\mu_0$, $\beta_0$ in the system

$$\nabla_\Sigma \, \Phi\left(\beta, \mu, \Sigma\right) = \mathbf{O}_m$$

we obtain

$$\Sigma^{-1}D\Sigma^{-1} - \frac{1}{2}\,\mathrm{diag}\left(\Sigma^{-1}D\Sigma^{-1}\right) = \mathbf{O}_m,$$

where $\mathrm{diag}\left(\Sigma^{-1}D\Sigma^{-1}\right) \in \mathcal{M}_m(\mathbf{R})$ is the diagonal matrix that retains only the entries placed on the main diagonal of $\Sigma^{-1}D\Sigma^{-1}$. Since $\Sigma$ is a positive definite matrix, we get $D = \mathbf{O}_m$ and consequently,

$$\Sigma = \frac{1}{N}\left(YY^T + \beta_0^T ZZ^T\beta_0 - YZ^T\beta_0 - \beta_0^T ZY^T +\right.$$
$$\left.\frac{1}{N}\left(\beta_0^T Zuu^T Y^T - \beta_0^T Zuu^T Z^T\beta_0 + Yuu^T Z^T\beta_0 - Yuu^T Y^T\right)\right) =$$

$$\frac{1}{N}\left(YAY^T + \beta_0^T ZAZ^T\beta_0 - YAZ^T\beta_0 - \beta_0^T ZAY^T\right).$$

Using the well-known properties of the Penrose pseudo-inverse, the expression of $\Sigma$ becomes

$$\Sigma = \frac{1}{N}\left(YAY^T + Y\left(ZA\right)^+ \left(ZA\right)AZ^T\left(Y\left(ZA\right)^+\right)^T - YAZ^T\left(Y\left(ZA\right)^+\right)^T - Y\left(ZA\right)^+ ZAY^T\right) =$$
$$\frac{1}{N}\left(YAY^T + Y\left(\left(ZA\right)^+\left(ZA\right)\right)^T\left(ZA\right)^T\left(\left(ZA\right)^+\right)^T Y^T - Y\left(\left(ZA\right)^+\left(ZA\right)\right)^T Y^T - Y\left(ZA\right)^+\left(ZA\right)Y^T\right) =$$
$$\frac{1}{N}Y\left(A + \left(\left(ZA\right)^+\left(ZA\right)\left(ZA\right)^+\left(ZA\right)\right)^T - \left(ZA\right)^+\left(ZA\right) - \left(ZA\right)^+\left(ZA\right)\right)Y^T =$$
$$\frac{1}{N}Y\left(A - \left(ZA\right)^+\left(ZA\right)\right)Y^T \overset{not}{=} \Sigma_0\,.$$

A further simplification can be obtained by noting that $B = A - \left(ZA\right)^+ ZA$ is a symmetric matrix and $B^2 = B$. Which finally yields to

$$\Sigma_0 = \frac{1}{N}YBB^T Y^T\,.$$

Obviously, the estimate $\Sigma_0$ is a symmetric and positive semi-defined matrix. $\square$

**Theorem 2** *Let $\beta_0$ and $\mu_0$ be given by Theorem 1. Then for any $(\beta, \mu, \Sigma)$ in the parameter space,*

$$l\left(\beta, \mu, \Sigma, \mathcal{S}_N\right) \leq l\left(\beta_0, \mu_0, \Sigma, \mathcal{S}_N\right). \tag{8}$$

**Proof.** Using (6) and $v^T v = \operatorname{tr}\left(vv^T\right)$ the expression of the log-likelihood function can be written as

$$l\left(\beta, \mu, \Sigma, \mathcal{S}_N\right) = -\frac{mN}{2}\ln(2\pi) - \frac{N}{2}\ln|\Sigma| - \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}\left(Y - \mu u^T - \beta^T Z\right)\left(Y - \mu u^T - \beta^T Z\right)^T\right).$$

Therefore

$$l\left(\beta_0, \mu_0, \Sigma, \mathcal{S}_N\right) = -\frac{mN}{2}\ln(2\pi) - \frac{N}{2}\ln|\Sigma| - \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}Y\left(A - (ZA)^+ ZA\right)\left(A - (ZA)^+ ZA\right)^T Y^T\right).$$

Using the relations $A = A^2 = A^T$ we get

$$\left(A - (ZA)^+ ZA\right)\left(A - (ZA)^+ ZA\right)^T =$$
$$A^2 - A(ZA)^+(ZA) - (ZA)^+(ZA)A + (ZA)^+(ZA)(ZA)^+(ZA) =$$
$$A - A(ZA)^+(ZA) - (ZA)^+(ZA) + (ZA)^+(ZA) = A - A(ZA)^+(ZA).$$

therefore the term $Y - \mu u^T - \beta^T Z$ becomes

$$Y - \mu u^T - \beta^T Z = Y\left(A - (ZA)^+(ZA)\right) + \frac{1}{N}\left(Yu - Y(ZA)^+ Zu - N\mu\right)u^T + \left(Y(ZA)^+ - \beta^T\right)Z =$$

$$Y\left(A - (ZA)^+(ZA)\right) + (\mu_0 - \mu)u^T + (\beta_0 - \beta)^T Z.$$

Consequently,

$$l\left(\beta, \mu, \Sigma, \mathcal{S}_N\right) = -\frac{mN}{2}\ln(2\pi) - \frac{N}{2}\ln|\Sigma| - \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}Y\left(A - (ZA)^+ ZA\right)\left(A - (ZA)^+ ZA\right)^T Y^T\right) -$$
$$\operatorname{tr}\left(\Sigma^{-1}Y\left(A - (ZA)^+ ZA\right)\left((\mu_0 - \mu)u^T\right)^T\right) - \operatorname{tr}\left(\Sigma^{-1}Y\left(A - (ZA)^+ ZA\right)\left((\beta_0 - \beta)^T Z\right)^T\right) -$$
$$\frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}(\mu_0 - \mu)u^T u(\mu_0 - \mu)^T\right) - \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}(\beta_0 - \beta)^T ZZ^T(\beta_0 - \beta)\right) = \tag{9}$$

$$l\left(\beta_0, \mu_0, \Sigma, \mathcal{S}_N\right) - \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}(\mu_0 - \mu)u^T u(\mu_0 - \mu)^T\right) - \frac{1}{2}\operatorname{tr}\left(\left(\Sigma^{-1}\beta_0 - \beta\right)^T ZZ^T(\beta_0 - \beta)\right) -$$
$$\operatorname{tr}\left(\Sigma^{-1}Y\left(\mathbf{I}_N - (ZA)^+ Z\right)Au(\mu_0 - \mu)^T\right) - \operatorname{tr}\left(\Sigma^{-1}Y\left(\mathbf{I}_N - (ZA)^+ Z\right)AZ^T(\beta_0 - \beta)\right).$$

Since

$$Au = \left(\mathbf{I}_N - \frac{1}{N}uu^T\right)u = u - \frac{1}{N}u\left(u^T u\right) = \mathbf{0}_N$$

we get

$$Y\left(\mathbf{I}_N - (ZA)^+ Z\right)Au(\mu_0 - \mu)^T = \mathbf{O}_m.$$

Also, using the properties of the Penrose pseudo-inverse, we get

$$\left(\mathbf{I}_N - (ZA)^+ Z\right)AZ^T = (ZA)^T - (ZA)^+ ZAA^T Z^T = (ZA)^T - \left((ZA)^+(ZA)\right)^T(ZA)^T =$$
$$(ZA)^T - \left((ZA)(ZA)^+(ZA)\right)^T = (ZA)^T - (ZA)^T = \mathbf{O}_{N,n+1},$$

that is $Y\left(\mathbf{I}_N - (ZA)^+ Z\right)AZ^T(\beta_0 - \beta) = \mathbf{O}_m$.

Taking into account these arguments we finally obtain,

$$l\left(\beta, \mu, \Sigma, \mathcal{S}_N\right) = l\left(\beta_0, \mu_0, \Sigma, \mathcal{S}_N\right) - \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}(\mu_0 - \mu)u^T u(\mu_0 - \mu)^T\right) - \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}(\beta_0 - \beta)^T ZZ^T(\beta_0 - \beta)\right).$$

Obviously, since $\Sigma$ a positive definite matrix,

$$\mathrm{tr}\left(\Sigma^{-1}\left(\mu_0-\mu\right)u^T u\left(\mu_0-\mu\right)^T\right) = N\,\mathrm{tr}\left(\left(\mu_0-\mu\right)^T\Sigma^{-1}\left(\mu_0-\mu\right)\right) = N\left(\mu_0-\mu\right)^T\Sigma^{-1}\left(\mu_0-\mu\right) \geq 0$$

and

$$\mathrm{tr}\left(\Sigma^{-1}\left(\beta_0-\beta\right)^T ZZ^T\left(\beta_0-\beta\right)\right) = \mathrm{tr}\left(Z^T(\beta_0-\beta)\Sigma^{-1}(\beta_0-\beta)^T Z\right) = \mathrm{tr}\left(\left((\beta_0-\beta)^T Z\right)^T \Sigma^{-1}(\beta_0-\beta)^T Z\right) \geq 0$$

that is

$$l\left(\beta,\mu,\Sigma,\mathcal{S}_N\right) \leq l\left(\beta_0,\mu_0,\Sigma,\mathcal{S}_N\right). \quad \square$$

**Remark.** Although, a long series of tests pointed out that the estimate $\Sigma_0$ given by Theorem 1 is a positive matrix the mathematical proof is still an open problem. Also, it is not known whether the unique critical point $(\beta_0,\mu_0,\Sigma_0)$ corresponds to the best model in the sense of the maximum likelihood principle.

An adaptive learning procedure can be obtained using the gradient ascent method applied to the log-likelihood criterion function. The search developed by the adaptive procedure in a $m(m+n+2)$-dimensional space aims to adjust the model parameters $\beta$, $\mu$, $\Sigma$ in order to maximize the log-likelihood function or, equivalently, to minimize $\Phi(\beta,\mu,\Sigma)$. The procedure should be implemented using a control parameter $\delta > 0$ and a stopping condition $\mathcal{C}(\delta)$ usually expressed in terms of the magnitude of the displacement in the parameter space due to the current iteration. In our tests $\mathcal{C}(\delta) = true$ if

$$\|\beta^{new}-\beta^{old}\| < \delta\,, \ \ \|\mu^{new}-\mu^{old}\| < \delta\,, \ \ \|\Sigma^{new}-\Sigma^{old}\| < \delta$$

where $\|\cdot\|$ is a conventionally norm, for instance Euclidian norm.

Also, the implementation of the procedure can be done using either a constant learning rate $\rho > 0$ or a decreasing sequence of positive learning rates $(\rho_k)$ that refines the search while the search advances.

Since during the search process the estimates of $\Sigma$ are not guaranteed to be invertible, the implementation procedure `MLE_gradient_ascent` uses approximations of the actual generalized gradients where the generalized inverse is used instead.

```
procedure MLE_gradient_ascent
Input:   {(x_1,y_1),…,(x_N,y_N)}
Initializations:  δ > 0, ρ > 0, β̃, μ̃, Σ̃,
```
$$Z = \begin{pmatrix} 1 \\ x_1 \end{pmatrix}, \ldots, \begin{pmatrix} 1 \\ x_N \end{pmatrix}, \ \ Y = (y_1,\ldots,y_N), \ \ u = (1,\ldots,1)^T$$
$$\beta^{old} \leftarrow \widetilde{\beta}, \ \ \mu^{old} \leftarrow \widetilde{\mu}, \ \ \Sigma^{old} \leftarrow \widetilde{\Sigma}$$
```
Compute  S = ZZ^T, Q = ZY^T, P = YY^T
```
$$Z_1 = Zu, \ \ Y_1 = Yu$$
```
repeat
```
$$\Sigma_1 \leftarrow \left(\Sigma^{old}\right)^+$$
$$\beta^{new} \leftarrow \beta^{old} + \rho\left(Q - Z_1\left(\mu^{old}\right)^T - S\beta^{old}\right)\Sigma_1$$
$$\mu^{new} \leftarrow \mu^{old} + \rho\Sigma_1\left(Y_1 - \left(\beta^{old}\right)^T Z_1 - N\mu^{old}\right)$$
$$D = N\Sigma^{old} - P + Y_1\left(\mu^{old}\right)^T + \left(Y_1\left(\mu^{old}\right)^T\right)^T + Q^T\beta^{old} + \left(Q^T\beta^{old}\right)^T -$$
$$N\mu^{old}\left(\mu^{old}\right)^T - \mu^{old}(Z_1)^T\beta^{old} - \left(\mu^{old}(Z_1)^T\beta^{old}\right)^T - \left(\beta^{old}\right)^T S\beta^{old}$$
$$\Sigma^{new} \leftarrow \Sigma^{old} + \rho\left(-\Sigma_1 D\Sigma_1 + \frac{1}{2}\,\mathrm{diag}\left(\Sigma_1 D\Sigma_1\right)\right)$$
```
        evaluate C(δ)
```
$$\beta^{old} \leftarrow \beta^{new}, \ \mu^{old} \leftarrow \mu^{new}, \ \Sigma^{old} \leftarrow \Sigma^{new}$$
```
until C(δ)
Output:  β^new, μ^new, Σ^new.
```

## 3   Experimental analysis

Because the model $(\beta_0, \mu_0, \Sigma_0)$ given by (7) is not theoretical guaranteed as the best model from the point of view of maximum likelihood principle, we have performed a long series of tests aiming to derive conclusions concerning the performance of the proposed method on experimental way. The test examples $x_i$'s were random generated from $n$-dimensional Gaussian repartition $\mathcal{N}(\mu_1, \Sigma_1)$. The target responses $y_i$'s were computed as $y_i = \widetilde{\beta}^T x_i + \varepsilon$ for given $\beta$ and $\varepsilon$ randomly generated from known Gaussian repartition $\mathcal{N}\left(\widetilde{\mu}, \widetilde{\Sigma}\right)$.

According to the previous arguments the expression of conditional density function on the output space corresponding to each example $x_i$ being given the model $\omega = (\beta, \mu, \Sigma)$ is

$$f\left(y|x_i, \omega\right) = \frac{1}{\sqrt{(2\pi)^m |\Sigma|}} \exp\left\{-\frac{1}{2}\left(y - \beta^T z_i - \mu\right)^T \Sigma^{-1} \left(y - \beta^T z_i - \mu\right)\right\},$$

where $z_i = \begin{pmatrix} 1 \\ x_i \end{pmatrix}$, therefore the most likely output predicted value is $y_i' = \beta^T z_i + \mu$.

In order to evaluate the quality of the resulted model we considered three indicators to evaluate the overall error.

The first indicator evaluates the overall mean error of miss-prediction for the given set of example $\{(x_i, y_i) \,|\, 1 \le i \le N\}$ corresponding to each possible model $\omega$

$$error\_1 = \frac{1}{N} \sum_{i=1}^{N} \left(1 - f\left(y_i|x_i, \omega\right)\right). \tag{10}$$

The second indicator is a mean error computed in terms of the actual responses and the most likely predicted values,

$$error\_2 = \frac{1}{N} \sum_{i=1}^{N} \|y_i - y_i'\|^2 = \frac{1}{N} \sum_{i=1}^{N} \|y_i - \beta^T z_i - \mu\|^2. \tag{11}$$

The third measure, of informational type, aims to evaluate the informational correlation between the input and the computed output corresponding to each model, and it is expressed in terms of the empirical mutual information. Since $x_1, \ldots, x_N$ are randomly generated $\mathcal{N}(\mu_1, \Sigma_1)$, the probability distribution $\widetilde{p} = (\widetilde{p}(x_1), \ldots, \widetilde{p}(x_N))$ characterizes the collection of examples,

$$\widetilde{p}(x_j) = \frac{p(x_j)}{\sum\limits_{i=1}^{N} p(x_i)},$$

where $p(x_j) = \dfrac{1}{\sqrt{(2\pi)^n |\Sigma_1|}} \exp\left\{-\dfrac{1}{2}(x_j - \mu_1)^T \Sigma_1^{-1} (x_j - \mu_1)\right\}$, $1 \le j \le N$.

The empirical entropy of the input samples $x_1, \ldots, x_N$ is given by the Shannon entropy corresponding to $\widetilde{p}$,

$$H(\widetilde{p}) = -\sum_{i=1}^{N} \widetilde{p}(x_i) \ln \widetilde{p}(x_i). \tag{12}$$

Using the transition probabilities

$$\widetilde{p}(y_j | x_i, \omega) = \frac{f(y_j | x_i, \omega)}{\sum\limits_{k=1}^{N} f(y_k | x_i, \omega)}, \quad 1 \le i, j \le N,$$

we define the probability distribution $\widetilde{q} = (\widetilde{q}(y_1), \ldots, \widetilde{q}(y_N))$ on the set of target responses by $\widetilde{q}(y_j) = \sum\limits_{i=1}^{N} \widetilde{p}(x_i) \widetilde{p}(y_j | x_i, \omega)$, $1 \le j \le N$, and let

$$H(\widetilde{q}) = -\sum_{i=1}^{N} \widetilde{q}(y_i) \ln \widetilde{q}(y_i), \tag{13}$$

be the empirical Shannon entropy of the set of **S**'s outputs.

Using the well-known expression of the mutual information ([4]), the empirical mutual information can be defined as

$$\mathcal{I}\left(\mathcal{S}_N\right) = H\left(\widetilde{q}\right) - \sum_{i=1}^{N}\sum_{j=1}^{N} \widetilde{p}\left(x_i\right)\widetilde{p}\left(y_j \mid x_i, \omega\right)\ln\widetilde{p}\left(y_j \mid x_i, \omega\right). \tag{14}$$

We evaluated the performance of the model $\omega_0 = (\beta_0, \mu_0, \Sigma_0)$ given by (7) for different sizes for learning samples $\mathcal{S}_N$.

**Test 1.** The settings are $n = 2$, $m = 1$, $\mu_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$, $\Sigma_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $\widetilde{\beta} = \begin{pmatrix} 0 \\ 2 \\ 4 \end{pmatrix}$, $\widetilde{\mu} = 0.25$ and

$\widetilde{\Sigma} = 1$. Some of the results for data of different sizes $N$, corresponding to the quasi-optimal model $\omega_0$ computed for each data set are summarized in Table 1$a$.

Table 1: *Model evaluation in case $n = 2$, $m = 1$ and for different volumes of: a) learning data; b) test samples.*

| | a | | | | | | b | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $N$ | error_1 | error_2 | $H(\widetilde{p})$ | $H(\widetilde{q})$ | $\mathcal{I}(\mathcal{S}_N)$ | $N_{test}$ | error_1 | error_2 | $H(\widetilde{p})$ | $H(\widetilde{q})$ | $\mathcal{I}(\mathcal{S}_N)$ |
| 10 | 0.603 | 0.533 | 2.214 | 2.205 | 1.335 | 15 | 0.706 | 1.149 | 2.458 | 2.582 | 1.346 |
| 20 | 0.702 | 0.802 | 2.767 | 2.796 | 1.304 | 30 | 0.710 | 1.257 | 3.243 | 3.276 | 1.367 |
| 50 | 0.711 | 0.892 | 3.768 | 3.831 | 1.226 | 50 | 0.683 | 1.039 | 3.752 | 3.829 | 1.573 |
| 100 | 0.712 | 0.950 | 4.477 | 4.524 | 1.233 | 70 | 0.650 | 1.136 | 4.146 | 4.172 | 1.573 |
| 200 | 0.716 | 0.959 | 5.098 | 5.188 | 1.221 | 90 | 0.707 | 1.292 | 4.309 | 4.400 | 1.637 |
| 300 | 0.721 | 1.019 | 5.510 | 5.590 | 1.106 | 100 | 0.709 | 1.276 | 4.376 | 4.483 | 1.662 |
| 400 | 0.713 | 0.953 | 5.822 | 5.897 | 1.155 | 200 | 0.715 | 1.361 | 5.124 | 5.235 | 1.616 |
| 500 | 0.725 | 1.025 | 6.023 | 6.113 | 1.150 | 300 | 0.719 | 1.375 | 5.547 | 5.633 | 1.579 |

Several tests aimed to establish conclusions concerning the generalization capacity of the quasi-optimal model. For instance, in case of a training sequence of volume $N = 100$, the computed quasi-optimal model is $\omega_0 = (\beta_0, \mu_0, \Sigma_0)$ is

$$\beta_0 = \begin{pmatrix} 0.000 \\ 2.614 \\ 4.077 \end{pmatrix} \quad \mu_0 = -0.218, \quad \Sigma_0 = 0.457,$$

the results are summarized in table 1$b$ and the variations of the input, output empirical entropies and the empirical mutual information as functions of the test sample sizes are depicted in Figure 2$b$.

**Test 2.** The settings are $n = 2$, $m = 2$, $\mu_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$, $\Sigma_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $\widetilde{\beta} = \begin{pmatrix} 0 & 0 \\ 2 & 1 \\ 4 & 5 \end{pmatrix}$, $\widetilde{\mu} = \begin{pmatrix} 0.25 \\ 0.25 \end{pmatrix}$

and $\widetilde{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. Some of the results for data of different sizes $N$, corresponding to the quasi-optimal model $\omega_0$ computed for each data set are summarized in Table 2$a$.

Several tests aimed to establish conclusions concerning the generalization capacity of the quasi-optimal model. For instance, in case of a training sequence of volume $N = 100$, the computed quasi-optimal model is $\omega_0 = (\beta_0, \mu_0, \Sigma_0)$ is

$$\beta_0 = \begin{pmatrix} -0.000 & -0.000 \\ 1.872 & 1.012 \\ 4.134 & 5.047 \end{pmatrix} \quad \mu_0 = \begin{pmatrix} 0.274 \\ 0.104 \end{pmatrix}, \quad \Sigma_0 = \begin{pmatrix} 0.677 & -0.007 \\ -0.007 & 0.899 \end{pmatrix},$$

the results are summarized in table 2$b$ and the variations of the input, output empirical entropies and the empirical mutual information as functions of the test sample sizes are depicted in Figure 3$b$.
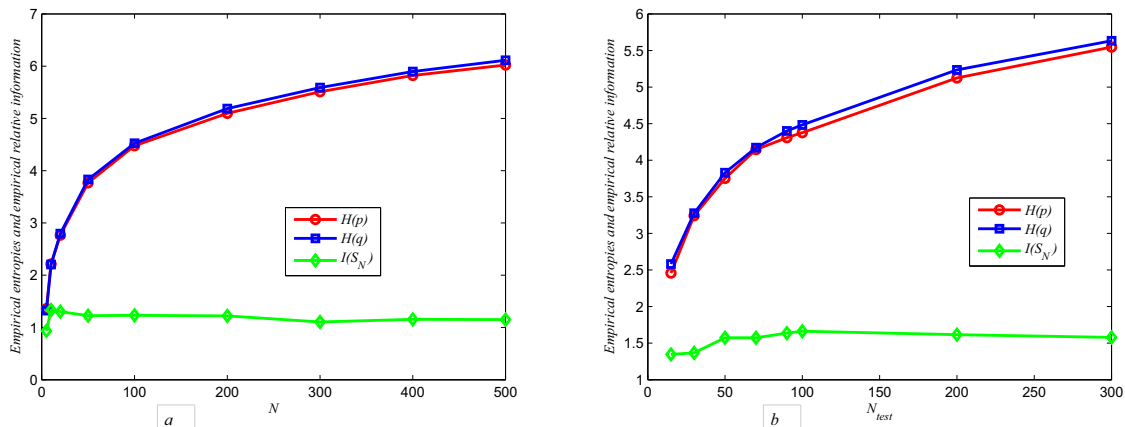
Figure 2: *The variation of the empirical input/output entropies and empirical mutual information: a) as function of $N$, volume of learning data; b) for $N = 100$ and different volumes of test samples.*

Table 2: *Model evaluation in case $n = 2$, $m = 2$ and for different volumes of: a) learning data; b) test samples.*

| | | a | | | | | | b | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $N$ | error_1 | error_2 | $H(\widetilde{p})$ | $H(\widetilde{q})$ | $\mathcal{I}(\mathcal{S}_N)$ | $N_{test}$ | error_1 | error_2 | $H(\widetilde{p})$ | $H(\widetilde{q})$ | $\mathcal{I}(\mathcal{S}_N)$ |
| 15 | 0.887 | 1.436 | 2.500 | 2.483 | 1.610 | 15 | 0.917 | 2.154 | 2.542 | 2.311 | 1.277 |
| 20 | 0.877 | 1.282 | 2.816 | 2.720 | 1.521 | 30 | 0.912 | 1.874 | 3.190 | 3.116 | 1.763 |
| 50 | 0.892 | 1.473 | 3.748 | 3.675 | 1.810 | 50 | 0.890 | 1.438 | 3.764 | 3.699 | 1.627 |
| 100 | 0.926 | 2.030 | 4.441 | 4.369 | 1.583 | 70 | 0.910 | 2.216 | 4.042 | 3.889 | 1.772 |
| 200 | 0.908 | 1.774 | 5.090 | 5.065 | 1.704 | 90 | 0.907 | 1.881 | 4.349 | 4.277 | 1.769 |
| 300 | 0.923 | 2.072 | 5.513 | 5.495 | 1.638 | 100 | 0.912 | 2.014 | 4.432 | 4.292 | 1.783 |
| 400 | 0.919 | 1.982 | 5.810 | 5.768 | 1.661 | 200 | 0.906 | 1.938 | 5.098 | 5.062 | 1.698 |
| 500 | 0.918 | 1.954 | 6.014 | 5.987 | 1.763 | 300 | 0.902 | 1.860 | 5.512 | 5.457 | 1.822 |

## 4 Conclusions

The aim of the report research was to develop a model free learning methodology in order to predict a system behavior, conventionally denoted by **S** on the basis of finite size sequence of observations. The learning environment is assumed to be of supervised type, that is a finite size sample of pairs of input-output values is available to the observer. The goal was to develop a learning component on the basis on the input-output sample, that is to infer the unknown the input-output dependency of **S** and use it for further predictions. In Section 2, the input-output dependency of **S** is modeled in terms of a Gaussian repartition of the output space. The model parameters are adjusted on the basis of the available sequence of observations using the Maximum Likelihood Principle. We managed to compute explicitly the quasi-optimal model $\omega_0$ (Theorem 1) and establish its 'almost' optimality (Theorem 2).

The performance evaluation of the quasi-optimal model is evaluated in Section 3 aiming to establish conclusions concerning the quality of the predictions about **S** computed by the learning component. The performance is evaluated in terms of three indicators, on the basis of the available sequence of observations and new test samples. The tests confirm that the proposed methodology assures fast learning of good quality, the predicted values being 'close' to the actual values in case of learning samples as well as in case of new test data. A series of further developments, still in progress, aiming to extend and refine the proposed methodology by taking into account non-linear models are going to be published in the near future.
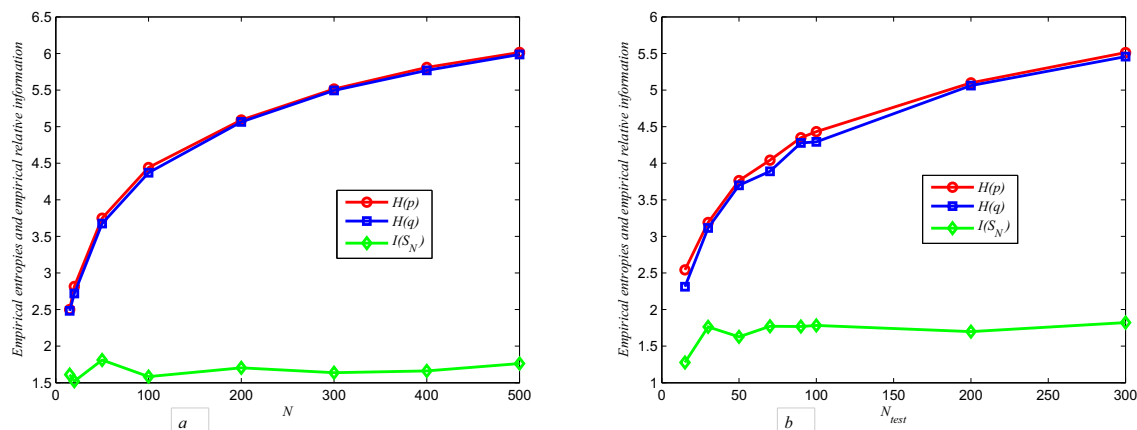
Figure 3: *The variation of the empirical input/output entropies and empirical mutual information: a) as function of N, volume of learning data; b) for N = 100 and different volumes of test samples.*

# References

[1] E. Alpaydin, *Introduction to Machine Learning*, The MIt Press, Massachusetts, 2010.

[2] C. Bishop, *Pattern Recognition and Machine Learning*, Springer Verlag, 2006.

[3] V. Cherkassky, F. Mulier, *Learning from Data Concepts, Theory, and Methods*, Second Edition, John Wiley & Sons, Inc., 2007.

[4] T.M. Cover, J.A. Thomas, *Elements of information theory (Second Edition)*, John Wiley & Sons, Inc., 2006.

[5] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning - Data mining, Inference and Prediction (Second Edition)*, Springer-Verlag, 2009.

[6] A.J. Izenman, *Modern Multivariate Statistical Techniques: Regressian, Classification and Manifold Learning*, Springer, 2008.

[7] A.K. Jain, R. Duin, J. Mao, Statistical Pattern Recognition: A Review, *IEEE Transactions On Pattern Analysis and Machine Intelligence* 22(1): 4–37, 2000.

[8] S. Marsland, *Machine Learning: An Algorithmic Perspective*, CRC Press, Taylor & Francis Group, Boca Raton - London - New York, 2009.

[9] K.E. Muller, P.W. Stewart, *Linear Model Theory - Univariate, Multivariate and Mixed Models*, Wiley-Interscience, 2006.

[10] I. Paraschiv-Munteanu, Theoretical approach in performance evaluation of a classsification system, *University of Pitesti - Scientific Bulletin, Serie Mathematics and Computer Science*, 14: 139–150, 2008.

[11] A.C. Rencher, *Methods of Multivariate Analysis*, Wiley-Interscience, 2002.

[12] L. State, I. Paraschiv-Munteanu, *Introducere în teoria statistică a recunoaşterii formelor*, Editura Universităţii din Piteşti, Romania, 2009.

[13] S. Sharma, *Applied Multivariate Techniques*, John Wiley & Sons, Inc., 1996.

[14] K. Takeaki, K. Hiroshi, *Generalized Least Squares*, John Wiley & Sons, Inc., 2004.

Luminiţa State
University of Piteşti
Faculty of Mathematics and Computer Science
1 Târgu din Vale St., Piteşti 110040
ROMANIA
E-mail: *lstate@clicknet.ro*

Iuliana Paraschiv-Munteanu
University of Bucharest
Faculty of Mathematics and Computer Science
14 Academiei St., Bucharest 010014
ROMANIA
E-mail: *pmiulia@fmi.unibuc.ro*

# Considerations about the implementation of an ATL model checker

**Laura Florentina Stoica, Florin Stoica**

**Abstract**

The problem of model checking is to verify if a finite-state abstraction of a reactive system satisfies a temporal-logic specification. A Computation Tree Logic (CTL) specification is interpreted over Kripke structures, which provide a model for the computations of a closed system (the behaviour is completely determined by the state of the system). In order to capture compositions of open systems, we present an extension of CTL, the alternating-time temporal logic (ATL), which is interpreted over game structures. In this paper we will show how our original ANTLR-based model checker for CTL can be modified to check an ATL specification, using a data structure suitable for multi-graph representation of a concurrent game structure.

## 1 Introduction

Model checking is a technology often used for the automated system verification. The model checking algorithms are currently used as verification techniques implemented in varied programming environments. The verified system can be a physical system or a real-time concurrent program. The behavior of a closed system can be described by the Kripke model. The Kripke models are based on the states and use the SMV (Symbolic Model Verifier) technique. The SMV model checking takes as input the model and formula then check whether or not the formula is satisfied or not by the model. [1]

A Computation Tree Logic (CTL) model is defined as a directed graph and its semantics is interpreted over a Kripke structure. A Kripke model $M$ over a set of atomic propositions, denoted by AP, is a triple $M=\langle S, Rel, P:S\rightarrow 2^{AP} \rangle$ where $S$ is a finite set of states, $Rel\subseteq S\times S$ is a transition relation, and $P:S\rightarrow 2^{AP}$ is a function that assigns each state with a set of atomic propositions from AP.

CTL model checker is branching-time logic, meaning that its formulas are interpreted over all paths beginning in a given state of the Kripke structure.

For each state from graph $M$ there is a successor and a path composed by a sequence of some states.

Details about formal definitions for syntax and semantics of a CTL model checker can be found in paper [3].

A Kripke structure offers a natural model for the computations of a ***closed system***, whose behavior is completely determined by the state of the system. The compositional modeling and design of reactive systems requires each component to be viewed as an *open system*.

An ***open system*** is a system that interacts with its environment and whose behaviour depends on the state of the system as well as the behaviour of the environment. In order to construct models suitable for open systems, was defined the Alternating-time Temporal Logic (ATL) [5].

ATL extends CTL by replacing the path quantifier's    and    by cooperation modalities $\langle\!\langle A\rangle\!\rangle$, where A is a team of agents. A formula $\langle\!\langle A\rangle\!\rangle$ φ expresses that the team A has a collective strategy to enforce φ.

ATL is a branching-time temporal logic that naturally describes computations of multi-agent system and multiplayer games. It offers selective quantification over program-paths that are possible outcomes of games [5]. ATL uses alternating-time formulas to construct model-checkers in order to address problems such as receptiveness, realizability, and controllability.

Over structures without fairness constraints, the model-checking complexity of ATL is linear in the size of the game structure and length of the formula, and the symbolic model-checking algorithm for CTL extends with few modifications to ATL.

From a formal point of view, implementation of a CTL model checker will be equated with implementation of an algebraic compiler which can be defined using $\Sigma – algebras$ and $\Sigma– heterogenic$ $homomorphism$ as $C:L_s{\rightarrow}L_t$, where $L_s$ is the source language and $L_t$ is the target language. The source language $L_s$ is CTL, and the target language $L_t$ is a language which describes the set of nodes from the model $M$ where a CTL formulas $f$ is satisfied. The algebraic compiler $C$ translates formula $f$ of the CTL model to set of nodes $S'$ over which formula $f$ is satisfied. That is, $C(f)=S'$ where $S'=\{s \in S|\ (M,s) \vDash f\}$.

The CTL language is defined as a $\Sigma – language$ [2]. The operator scheme $\Sigma_{ctl}$ is defined as a triple $\langle S_{ctl}, O_{ctl}, \sigma_{ctl}\rangle$ where set $S_{ctl}$ contains the representations of the CTL formulas, $O_{ctl} = \{ T, \perp, \neg, \wedge, \vee, \rightarrow,$ $AX, EX, AU, EU, EF, AF, EG, AG\}$ is the set of operators,  and the $\sigma_{ctl}:O_{ctl}{\rightarrow}S_{ctl}^{*}{\times}S_{ctl}$ is a function which defines the signature of the operators [2]. The *CTL* model checker can be defined as the $\Sigma_{ctl}$ - language given in the form $L_{ctl}=\langle Sem_{ctl},\ Syn_{ctl},\ L_{ctl}:Sem_{ctl}{\rightarrow}Syn_{ctl}\rangle$ where $Syn_{ctl}$ is the word algebra of the operator scheme $\Sigma_{ctl}$ generated by the operations from $O_{ctl}$ and a finite set of variables, representing atomic propositions, denoted by *AP*. $Sem_{ctl}$ represents *CTL* semantic algebra defined over the set of *CTL* formulas which are satisfied  by the CTL model $M$. $L_{ctl}$ is a mapping which associates the set of satisfied formulas from $Sem_{ctl}$ to CTL expressions from $Sin_{ctl}$ which satisfy these formulas. [4]

In the same way we can define the ATL language. The operator scheme $\Sigma_{atl}$ is defined as a triple $\langle S_{atl}, O_{atl}, \sigma_{atl}\rangle$ where set $S_{atl}$  contains the representations of the ATL formulas, $O_{atl} = \{\neg, \vee, \wedge, \rightarrow, \Diamond,$ $\circ, \square, U\}$ is the set of operators,  and the $\sigma_{atl}:O_{atl}{\rightarrow}S_{atl}^{*}{\times}S_{atl}$ is a function which defines the signature of the operators. The $\Diamond$ ('future'), $\circ$ ('next'), $\square$ ('always'), and U ('until') are temporal operators. The *ATL* model checker can be defined as the $\Sigma_{atl}$ - language given in the form $L_{atl}=\langle Sem_{atl},\ Syn_{atl},$ $L_{atl}:Sem_{atl}{\rightarrow}Syn_{atl}\rangle$ where $Syn_{atl}$ is the word algebra of the operator scheme $\Sigma_{atl}$ generated by the operations from $O_{atl}$ and a finite set of variables, representing atomic propositions, denoted by *AP*. $Sem_{atl}$ represents *ATL* semantic algebra defined over the set of *ATL* formulas which are satisfied  by the ATL model $M$. $L_{atl}$ is a mapping which associates the set of satisfied formulas from $Sem_{atl}$ to ATL expressions from $Sin_{atl}$ which satisfy these formulas.

Having well-defined the ATL language, implementation of a ATL model checker will be equated with an algebraic compiler which translates a formula $f$ of the ATL model to set of nodes $Q'$ over which formula $f$ is satisfied.

The Kripke structure is a natural "common-denominator" model for ***closed systems***, independent of the high-level description of a system (given by example as a product of state machines).

In analogy, the natural "common-denominator" model for compositions of ***open systems*** is the *concurrent game structure.*

Thus, unlike CTL which is interpreted over Kripke structures, the ATL is interpreted over game structures. In order to capture compositions of *open systems*, we consider multi players games in which the set of players represents different component of the system and the environment.

The remainder of this paper is organized as follows. In section 2 we present a formal definition of concurrent game structures. In section 3 is presented the ATL logic with its syntax and semantics. A Java implementation of ATL model checker based on ANTLR is described in section 4. Conclusions are presented in section 5.

# 2 Concurrent game structures

Concurrent game structures can be used to model compositions of open systems. Unlike in Kripke structures, in a concurrent game structure, the environment is involved a state transition. The environment is modelled by a set of agents. Each agent may perform some actions and at least one action is available to the agent at each state.

## 2.1 Definition

A *concurrent game structure* is a tuple $S=\langle k,Q,\prod,\pi,d,\delta\rangle$ with the following components [5]:

- A natural number $k \geq 1$ of *players*. We identify the players with numbers $1,\ldots,k$.
- A finite set $Q$ of *states*
- A finite set $\prod$ of *propositions* (also called *observables*)
- For each state $q \in Q$, a set $\pi(q) \subseteq \prod$ of propositions true at q. The function $\pi$ is called *labeling* (or *observation*) *function*.
- For each player $a \in \{1,\ldots,k\}$ and each state $q \in Q$, we identify the moves of player *a* at state q with the numbers $1,\ldots, d_a(q)$, where $d_a(q) \geq 1$ represents the number of available moves. For each state $q \in Q$, a *move vector* at *q* is a tuple $\langle j_1,\ldots,j_k \rangle$ such that $1 \leq j_a \leq d_a(q)$ for each player *a*. Given a state $q \in Q$, we write $D(q)$ for the set $\{1,\ldots, d_1(q)\} \times \ldots \times \{1,\ldots, d_k(q)\}$ of moves vector. The function D is called ***move function***.
- For each state $q \in Q$ and each move vector $\langle j_1,\ldots,j_k \rangle \in D(q)$, $\delta(q,j_1,\ldots,j_k) \in Q$ represents the state that results from state q if every player $a \in \{1,\ldots,k\}$ choose move $j_a$. The function $\delta$ is called ***transition function***.

The number of states of the structure *S* is $n = |Q|$. The *number of transitions* of *S* is $m = \Sigma_{q \in Q} d1(q) \times \ldots \times dk(q)$, that is, the total number of elements in the move function D. Note that unlike in Kripke structures, the number of transitions is not bounded by $n^2$. For a fixed alphabet $\Pi$ of propositions, the size of *S* is $O(m)$. [5]

We refer to a computation starting at state *q* as a *q-computation*. For a computation $\lambda$ and a position $i \geq 0$, we use $\lambda[i]$, $\lambda[0,i]$, and $\lambda[i,\infty]$ to denote the *i*-th state of $\lambda$, the finite prefix $q_0, q_1,\ldots,q_i$ of $\lambda$, and the infinite suffix $q_i, q_{i+1} \ldots$ of $\lambda$, respectively. [5]

## 2.2 Example of a concurrent game structure

Consider a system with two processes, *Px* and *Py*. The process *Px* assigns values to the Boolean variable *x*. When *x=false*, then *Px* can leave the value of *x* unchanged or change it in *true*. When *x=true*, then *Px* leaves the value of *x* unchanged. The process *Py* assigns values to the Boolean variable *y*, in the same way as the process *Px*.
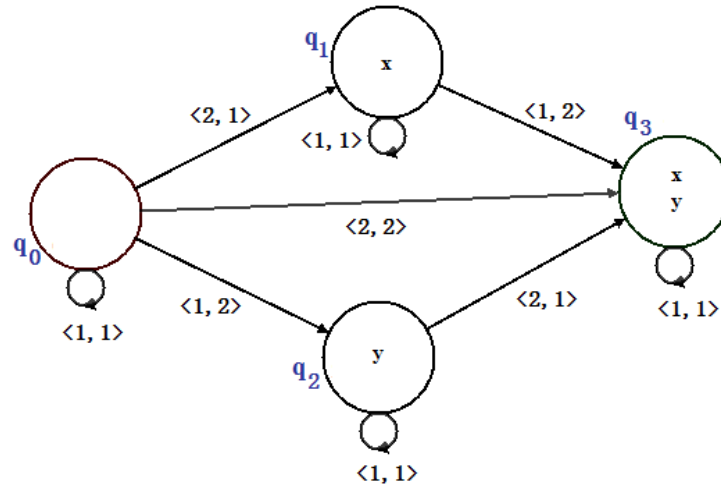
Fig. 1: An example of a concurrent game structure ($S_{xy}$)

We model the synchronous composition of two processes by the $S_{xy}$ concurrent game structure, where $S_{xy}=\langle k,Q,\prod,\pi,d,\delta\rangle$

- k=2 (player 1 represents process *Px*, player 2 represents process *Py*)
- Q={$q_0,q_1,q_2,q_3$} – $q_0$ means x=y=false,
  $q_1$ means x=true and y=false, etc.
- $\prod$={x,y}
- $\pi(q_0)=\varnothing$, $\pi(q_1)$={x}, $\pi(q_2)$={y}, $\pi(q_3)$={x,y}
- $d_1(q_0)$= $d_1(q_2)$=2 (means in state $q_0$ and $q_2$ move 1 of player 1 leave the value of x unchanged, and move two changes the value of x)

$d_1(q_1)$= $d_1(q_3)$=1 (means in state $q_1$ and $q_3$ player 1 has only one move, namely, to leave the value of x unchanged)

$d_2(q_0)$= $d_2(q_1)$=2, $d_2(q_2)$= $d_2(q_3)$=1

- state $q_0$ has four successors: $\delta(q_0,1,1)$= $q_0$, $\delta(q_0,1,2)$=$q_2$, $\delta(q_0,2,1)$=$q_1$, $\delta(q_0,2,2)$=$q_3$

A concurrent game is played on a state space. Every player chooses a move. The combination of choices determines a transition from the current state to a successor state.

The game structure presented in figure 1 is classified as a Moore synchronous game structure. That is, the state is partitioned according to the players. In each step, every player updates its own component of the state independently of the other players. The Moore subclass of concurrent game structures captures various notions of synchronous interaction between open systems.

# 3   ATL Logic

Alternating-time Temporal Logic (ATL) is a branching-time temporal logic that naturally describes computations of multi-agent system and multiplayer games. It offers selective quantification over program-paths that are possible outcomes of games [5].

## 3.1. ATL syntax

The temporal logic ATL is defined with respect to a finite set $\prod$ of propositions and a finite set $\Sigma$={1,…,k} of players.

An ATL formula is one of the following:

(S1) p, for propositions p$\in\prod$

(S2) ¬ φ or $\varphi_1\vee\varphi_2$, where φ, $\varphi_1$ and $\varphi_2$ are ATL formulas

(S3) $\langle\langle A \rangle\rangle \circ \varphi$, $\langle\langle A \rangle\rangle \square \varphi$, or $\langle\langle A \rangle\rangle \varphi_1 U \varphi_2$, where $A \subseteq \Sigma$ is a set of players, and $\varphi$, $\varphi_1$ and $\varphi_2$ are ATL formulas.

The operator $\langle\langle \rangle\rangle$ is a path quantifier, and $\circ$ ('next'), $\square$ ('always'), and U ('until') are temporal operators. The logic ATL is similar to the branching time temporal logic CTL, only that path quantifiers are parameterized by sets of players. Sometimes we write $\langle\langle a_1,\ldots,a_k \rangle\rangle$ instead of $\langle\langle \{a_1,\ldots,a_k\} \rangle\rangle$, and $\langle\langle \rangle\rangle$ instead of $\langle\langle \varnothing \rangle\rangle$. Additional Boolean connectives are defined from $\neg$ and $\vee$ in the usual manner. Similar to CTL, we write $\langle\langle A \rangle\rangle \Diamond\varphi$ for $\langle\langle A \rangle\rangle$ true U $\varphi$.

## 3.2. ATL semantics

Consider a game structure $S=\langle k,Q,\prod,\pi,d,\delta \rangle$.
$\Sigma=\{1,\ldots,k\}$ denote the set of players.

A *strategy* for player $a\in\Sigma$ is a function $f_a$ that maps every nonempty finite state sequence $\lambda\in Q^+$ to a natural number such that if the last state of $\lambda$ is q, then $f_a(\lambda) \leq d_a(q)$. Thus, the strategy $f_a$ determines for every finite prefix $\lambda$ of a computation a move $f_a(\lambda)$ for player a. Each strategy $f_a$ for player *a* induces a set of computations that player *a* can enforce.

Given a state $q\in Q$, a set $A\subseteq\{1,\ldots,k\}$ of players, and a set $F_A=\{ f_a \mid a\in A\}$ of strategies, one for each player in A, we define the *outcomes of $F_A$ from q* to be the set *out(q, $F_A$)* of q-computations that the players in A enforce when they follow the strategies in $F_A$;

A computation $\lambda=q_0,q_1,q_2,\ldots$ is in *out(q,$F_A$)* if $q_0=q$ and for all positions $i\geq 0$, there is a move vector $\langle j_1,\ldots,j_k \rangle \in D(q_i)$ such that

- $j_a=f_a(\lambda[0,i])$ for all players $a\in A$, and
- $\delta(q_i, j_1,\ldots,j_k)= q_{i+1}$

Formal definition of ATL semantics is to consider a game structure $S=\langle k,Q,\prod,\pi,d,\delta \rangle$. We write $S, q\vDash\varphi$ to indicate that the state q satisfies the formula $\varphi$ in the structure S. When S is clear from the context, we omit it and write $q\vDash\varphi$.

The satisfaction relation $\vDash$ is defined, for all states q of S inductively as follows:

- $q\vDash p$, for propositions $p\in\prod$, iff $p\in\pi(q)$
- $q\vDash\neg\varphi$ iff $q\nvDash\varphi$
- $q\vDash\varphi_1\vee\varphi_2$ iff $q\vDash\varphi_1$ or $q\vDash\varphi_2$
- $q\vDash\langle\langle A \rangle\rangle \circ \varphi$ iff there exists a set $F_A$ of strategies, one for each player in A, such that for all computations $\lambda\in out(q, F_A)$, we have $\lambda[1] \vDash \varphi$
- $q\vDash\langle\langle A \rangle\rangle \square \varphi$ iff there exists a set $F_A$ of strategies, one for each player in A, such that for all computations $\lambda\in out(q, F_A)$, and all positions $i\geq 0$, we have $\lambda[i] \vDash \varphi$
- $q\vDash\langle\langle A \rangle\rangle \varphi_1 U \varphi_2$ iff there exists a set $F_A$ of strategies, one for each player in A, such that for all computations $\lambda\in out(q, F_A)$, there exists a position $i\geq 0$ such that $\lambda[i] \vDash \varphi_2$ and for all positions $0\leq j\leq i$, we have $\lambda[j] \vDash \varphi_1$

ATL can naturally express properties of open system [5]. Properties is the absence of deadlocks, where deadlock state is one in which a thread, say *t*, is permanently blocked from accessing a critical section.

In the following is described this requirement using the CTL formula, respectively ATL formula.

The CTL formula only asserts that it is always possible for all threads to cooperate so that *t* can eventually read and write ("collaborative possibility")

$$\forall\square\ (\exists\ \Diamond\ \text{read} \wedge \exists\ \Diamond\ \text{write})$$

The ATL formula guarantees execution of the critical section by the thread *t*, *no matter what the other threads in the system do* ("adversarial possibility")

$$\forall\square\ (\langle\langle t \rangle\rangle\Diamond\text{read} \wedge \langle\langle t \rangle\rangle\Diamond\text{write})$$

The path quantifiers *A*, *E* of CTL can be expressed in ATL with ⟨⟨∅⟩⟩, ⟨⟨Σ⟩⟩ respectively. As a consequence, the CTL duality axioms can be rewritten in ATL, and become validities in the basic semantics: ¬⟨⟨Σ⟩⟩ ◊ φ ↔ ⟨⟨∅⟩⟩□¬ φ, ¬⟨⟨∅⟩⟩ ◊ φ ↔ ⟨⟨Σ⟩⟩□¬ φ, where the *Σ∈{1,...,k}* describe the set of agents.

# 4 An ANTLR – Java implementation of ATL model checker

## 4.1 ATL Symbolic Model Checking

Model checking is a technology often used for the automated system verification.
- The model checking algorithms are currently used as verification techniques implemented in varied programming environments.
- The verified system can be a physical system or a real-time concurrent program.
- A model checking tool can be used to verify if a given system satisfies a temporal logic formula.
- The model checking problem for ATL: given a game structure $S = \langle k, Q, \prod, \pi, d, \delta \rangle$ and an ATL formula φ the task is to find the set of states in Q that satisfy φ.

In order to solve the ATL model checking problem we designed and implemented an algebraic compiler denoted with $\mathcal{C}$.

## 4.2 ATL algorithm

The algebraic compiler $\mathcal{C}$ translates formula φ of the ATL model to set of nodes $Q'$ over which formula φ is satisfied. That is, $\mathcal{C}(\varphi) = Q'$ where $Q' = \{q \in Q | \ q \vDash \varphi\}$.

The implementation of the algebraic compiler $\mathcal{C}$ is made in two steps.
- First, we need a syntactic parser to verify the syntactic correctness of a formula φ.
- Then, we should deal with the semantics of the ATL language, respectively with the implementation of the ATL operators from the set {¬, ∨, ∧, →, ◊, ○, □, U}.

For implementation of the algebraic compiler we choose the ANTLR (*Another Tool for Language Recognition*). ANTLR is a compiler generator which takes as input a grammar - an exact description of the source language, and generates a recognizer for the language defined by the grammar.

The algebraic compiler $\mathcal{C}$ implements the following ATL symbolic model checking algorithm:

```
Function Eval_A(φ) as set of states ⊆ Q
    case φ=p:
            return [p];
    case φ= ¬θ:
            return Q\Eval_A(θ);
    case φ=θ₁∨θ₂:
            return Eval_A(θ₁)∪ Eval_A(θ₂);
      case φ=θ₁∧θ₂:
            return Eval_A(θ₁) ∩ Eval_A(θ₂) );
      case φ=θ₁→θ₂:
            return ( Q\Eval_A(θ₁) ) ∪ Eval_A(θ₂);
    case φ =⟨⟨A⟩⟩○θ:
            return Pre(A,Eval_A(θ));
    case φ =⟨⟨A⟩⟩□θ:
        ρ:=Q; τ:= Eval_A(θ); τ₀:= τ;
        while ρ ⊈ τ do
                ρ := τ;
            τ:=Pre(A, ρ)∩τ₀;
```

```
                           wend
               return  ρ;
    case φ = ⟨⟨A⟩⟩  θ₁  U  θ₂:
               ρ:=∅;  τ:= Eval_A(θ₂);  τ₀:= Eval_A(θ₁);
                while ρ ⊈ τ do
                  ρ := ρ ∪ τ;
            τ:=Pre(A,  ρ)∩τ₀;
                         wend
               return  ρ;
End Function
```

The Pre(A,ρ) function, where A⊆Σ and ρ⊆Q, returns the set of states q such that from q, the players in A can cooperate and enforce the next state to be in ρ.

Pre(A,ρ) contains state q∈Q if for every player a ∈A there exists a move $j_a \in \{1,...,d_a(q)\}$ such that for all players b∈Σ\A whatever are their moves we have $\delta(q, j_1,..., j_k) \in \rho$

In order to translate a formula φ of an ATL model to the set of nodes $Q'$ over which formula φ is satisfied, is necessary the attachment of specific actions to grammatical constructions within specification grammar of ATL.

The actions are written in target language of the generated parser (in our case, Java). These actions are incorporated in source code of the parser and are activated whenever the parser recognizes a valid syntactic construction in the translated ATL formula. In case of the algebraic compiler $\mathcal{C}$, the actions define the semantics of the ATL model checker, i.e., the implementation of the *ATL* operators.

The model checker generated by ANTLR from our specification grammar of ATL, takes as input the concurrent game structure S and formula φ, and provides as output the set $Q'=\{q \in Q| \ q \vDash \varphi\}$ – the set of states where the formula φ is satisfied.

The corresponding *action* included in the ANTLR grammar of *ATL* language for implementing the □ operator is:

```
'<<A>> #' f=formula
{
HashSet r = new HashSet(all_SetS);
HashSet p = $f.set;
while (!p.containsAll(r))
{
    r = new HashSet(p);
    p = Pre(r);
    p.retainAll($f.set);
}
$set = r;
trace("atlFormula",4);
printSet("<<A>>#"+$f.text,r);
}
```

Fig. 2: ANTLR implementation of □ operator

For ATL operator □, in ANTLR we use the # symbol.

In our implementation the all_Set is Q, and means all the state from model. The *formula* represents a term from a production of the ATL grammar and *p, r, f* variables are sets used in internal implementation of the algebraic compiler.

The Pre(r) is a function that returns the set of states *p* such that from *p*, the players in A can cooperate and enforce the next state to be in *r*.

The code from figure 2 represents the implementation of the □ ATL operator which is described in the symbolic model checking algorithm as:

```
Function Eval_A(φ) as set of states ⊆ Q
    …
    case φ =⟨⟨A⟩⟩□θ:
        ρ:=Q; τ:= Eval_A(θ); τ_0:= τ;
         while ρ ⊈ τ do
                ρ := τ;
            τ:=Pre(A, ρ)∩τ_0;
                      wend
              return ρ;
    …
endfunction
```

Fig. 3 The □ ATL operator from the model checking algorithm

In figure 4 is represented the algebraic compiler implementation process, based on our specification grammar of ATL language.



Fig. 4 Algebraic compiler implementation
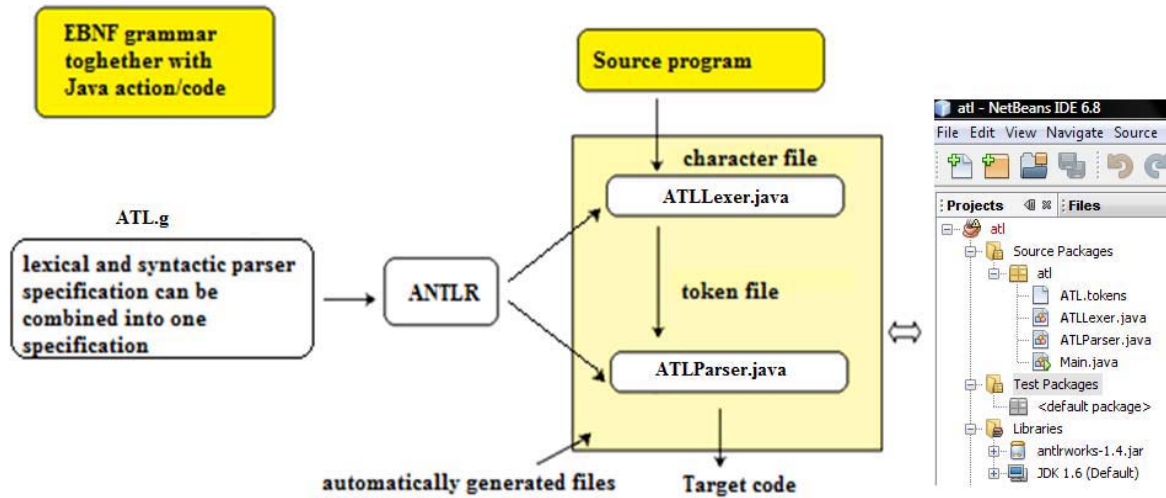
For verification of formula $\varphi = <<A>> \; \square \; (x \vee y)$ we can use the ANTLR debugging facility to visualize the Abstract Syntactic Tree (AST), presented in the figure 5.
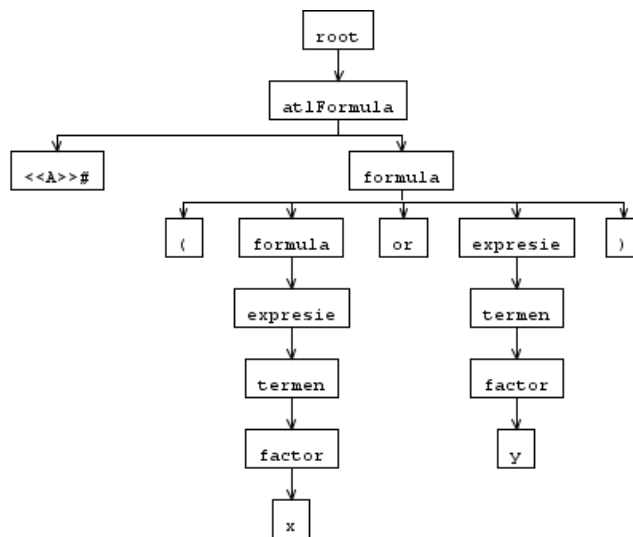


Fig. 5: Abstract Syntactic Tree (AST)

The AST is decorated with actions automatically executed when the parser recognizes syntactic components of formula φ. These actions implement the algebraic compiler $\mathcal{C}$. The output of $\mathcal{C}$ is presented in figure 6.

Given the ATL formula $\varphi = <<A>> \circ (x \wedge y)$ for game structure from figure 1 with A={2}, the output of the model checker is Q' = {1,3}. From state $q_1$ if player 2 chooses the move 2 the next state is $q_3$ whatever is the move selected by the player 1. From the state $q_2$ for the move 1 of the player 2, the player 1 can choose the move 1. Thus the game remains in state $q_2$. For that reason the state $2 \notin Q'$.



Fig. 6: The output of compiler $\mathcal{C}$ for ATL formula $<<A>> \square (x \vee y)$

Fig. 7: Output of the model checker for ATL formula $<<A>> \circ (x \wedge y)$

## 5 Conclusion

In this article we built a CTL model checking tool, based on robust technologies (Java, ANTLR)

As a great facility we mention the capability of interactive debugging / visualization of the execution of the symbolic model checking algorithm.

The ATL algebraic compiler based on Java code generated by ANTLR using an original ATL grammar provides error-handling for eventual lexical/syntax errors in formula to be translated.

## References

[1] Laura F. Cacovean, Florin Stoica, Dana Simian, *A New Model Checking Tool*, Proceedings of the European Computing Conference (ECC '11), Paris, France, April 28-30, 2011, pp. 358-364, ISBN: 978-960-474-297-4, ISSN: 2222-7342

[2] L. Cacovean, F. Stoica, *Algebraic Specification Implementation for CTL Model Checker Using ANTLR Tools*, 2008 WSEAS International Conferences, Computers and Simulation in Modern Science - Volume II, Bucharest, Romania, 2008, pp. 45-50

[3] M. Huth, M. Ryan, *Logic in Computer Science: Modelling and Reasoning about Systems*, Cambridge University Press, 2000.

[4] L. F. Cacovean, *Using CTL Model Checker for Verification of Domain Application Systems*, Proceedings of the 11th WSEAS International Conference on EC, Iaşi, Romania, 2010, pp. 262-267.

[5] R. Alur, T. A. Henzinger , O. Kupferman, *Alternating-Time Temporal Logic*, Journal of the ACM, Vol. 49, No. 5, September 2002, pp. 672–713

LAURA FLORENTINA STOICA
Department of Computer Science
"Lucian Blaga" University of Sibiu
Str. Dr. Ion Ratiu 5-7, 550012, Sibiu
ROMANIA
E-mail: laura.cacovean@ulbsibiu.ro

FLORIN STOICA
Department of Computer Science
"Lucian Blaga" University of Sibiu
Str. Dr. Ion Ratiu 5-7, 550012, Sibiu
ROMANIA
E-mail: florin.stoica@ulbsibiu.ro

# A technique for constructing training sets in data stream mining: kSiEved Window Training Set

**Sabina Surdu**

### Abstract

One of the challenges that data mining is facing today is applying its techniques to dynamic data, *i.e.* data streams. Data streams are produced over time by data sources and systems that process them run continuous queries in a perpetual manner, in order to produce streamed results. New data management and query processing paradigms emerged in this context. Applying data mining algorithms on data streams requires adapting the classical methods of data mining to the novel processing paradigms. One of the main challenges in data stream processing is represented by limited system resources. Our goal is to provide a resource-aware technique that constructs training sets for a data stream mining algorithm, namely the kSiEved Window Training Set technique.

## 1  Introduction

Data mining has been around for a while, in a great number of fields, ranging from science to business domains. The technology enables hidden patterns discovery and extraction from large data sets and predicting future values for data elements based on some known information. But the data on which one might apply data mining is not always static. One of the current challenges in the field is to perform data mining on rapid, continuous streams of data.

Consider the case of an astronomic application, that receives data from a large number of celestial bodies, which emanate radio waves. Processing such data could aim at discovering similar planets in neighbouring parts of the galaxy. A surveillance system continuously processes streams of information from multiple monitoring devices, to detect unauthorized access. ATM transactions also generate streams of continuous data, recording clients operations, to detect fraudulent activity. Monitoring clickstreams in electronic commerce can reveal various patterns in web sites usage, which can lead to better design solutions, serving both the customer and the business.

In an increasing number of domains, continuous data processing has become a stringent requirement. Novel paradigms are developed for data management and query processing in this context. Although stream processing technology has yet to reach maturity, data mining requirements are already formulated for streaming applications, as the above examples show.

In this paper we introduce a novel, resource-aware technique that builds training sets for a data stream mining algorithm (the design of such an algorithm is out of the scope of our paper). The kSiEved Window Training Set (kSEWT) technique constructs training records by sifting streams in a parameterized manner, depending on the value of the sieve parameter k and specified error bounds.

Our objective in this paper is two-fold. Apart from introducing kSEWT, we also provide a high-level description of stream processing models and paradigms. We motivate the need to perform data mining on continuous data. We highlight the issues encountered when processing data streams, which represent core challenges in data stream mining.

The paper is structured as follows. In section 2 we discuss basic aspects concerning data mining. Section 3 takes a more detailed look at data stream processing. Exemplifying with specific Data Stream Management Systems, we show the main difficulties and challenges encountered in designing such systems, highlighting problems that need to be tackled when applying data mining techniques. In Section 4 we present our kSiEved Window Training Set technique. We provide an informal description as well as the formalized kSEWT model. Section 5 discusses the kSEWT strategy and presents experimental results. Section 6 provides future research directions. Section 7 concludes on the results of our work.

## 2 Data mining

### 2.1 Data proliferation

We are moving through the Information Age at a constantly increasing pace. The quantities of produced information expand on a daily basis. [14] identifies *data overload* as one of the main challenges raised by the digital era. Nowadays databases are growing in two dimensions: number of fields and number of objects they contain. Moreover, recently emerged data streams are infinite.

Data proliferation is a result of both computer hardware and software advancements. The cost of storing large volumes of data is perpetually decreasing, whereas storage capacities are increasing. Processing computing power is also advancing on a daily basis. Constant improvements in transmission speed enable larger quantities of data to be transported over networks. Fast progress in data capture also plays an important role, as input data can come from a generous palette of devices: sensors, ATMs, RFID tags, hand-held devices, GPS receivers, etc.

We are heading towards the Internet of Things, an ubiquitous network society composed of interconnected devices, that provide useful information to the user, whatever his or her location [23]. The interconnectivity of smart, communicating devices enables data flow paradigms that haven't been encountered before. [16] describes a pervasive environment, where one can query classical data (like data from databases), data streams and functionalities in a consistent, generalized and declarative framework. Such an enviromnent, queryable in an abstract, easily read by humans manner, will lead to even greater quantities of data than those we need to cope with today. Scientific databases (*e.g.* radio astronomy databases) are expected to reach exascale sizes in the years to come [9]. [19] results show that in 2007 the world's storage capacity was 290 exabytes (1 exabyte = $10^{18}$ bytes), the humankind communicated 2 zettabytes (1 zettabyte = $10^{21}$ bytes) and the computing power of general-purpose computers was 6.4 exa instructions per second.

### 2.2 Motivation

In this context, organizations, research communities and other entities have started to base their decisions chiefly on the data they collect [18]. In some cases, raw data needs to be mined in order to extract useful information for a decision maker.

Take for instance the case of a chocolate company that sells three times as much chocolate Santa Clauses during Christmas than it usually does. The company will naturally supply the retailers with increased quantities of chocolate Santa Clauses right before Christmas. Further data analysis can reveal that 85% of the people who buy chocolate Santa Clauses also buy chocolate ornaments for the tree. These patterns are called association rules with chocolate Santa Clause as antecedent and chocolate ornament as consequent, having a confidence factor of 85% [3]. The chocolate company can maximize its revenue during Christmas, by supplying increased quantities of both chocolate Santa Clauses and ornaments. This technique is usually referred to as market basket analysis. Another tactic than can be employed is arranging products that are part of the same association rule next to each other on shelves.

In a financial context, a bank may want to predict a new customer's probability to default a loan, based on existing information it has on its current customers. A monitoring application in radio astronomy could aim at automatically classifying the monitored objects, based on radio waves values. Marketing campaigns could target new customers that are more likely to respond to the offers, instead of randomly choosing potential clients. And the examples can go on.

In the past analysis was performed by human specialists. But on data sets with millions of records and hundreds or thousands of fields, direct hands-on analysis is not possible any more. The process of extracting useful information needs to be performed by computer, in an automated manner.

## 2.3 Data mining

Data mining is the automated analysis of large data sets in order to find unsuspected relationships and to summarize the data in novel, understandable and useful ways [17].

[14] emphasizes the fact that data mining is a step in the Knowledge Discovery in Databases (KDD) process. KDD refers to the whole process of mapping low-level, raw data into high-level information. Apart from data mining, KDD also include steps like data collection, cleaning, integration, selection and transformation, and model evaluation and validation.

Numerous techniques can be applied in the data mining step. [22] enumerates, among the most important ones: artificial neural networks, decision trees, genetic algorithms, nearest neighbour method and rule induction.

Data mining can be driven by two types of goals: verification and discovery [14]. Verification restricts the data mining task to the process of verifying previously formulated user hypotheses. Discovery allows data mining to find unknown patterns in data. Furthermore, discovery goals can be classified as either descriptive (find patterns that describe data) or predictive (find patterns that predict future behaviour).

New challenges arise when considering data streams as input to data mining algorithms. Another dimension is to be considered in the analysis process: time. Consider a stream that contains records representing customers transactions. And let a transactional database represent, at each time instant, a snapshot of the stream (*i.e.* a large window). If the temporal variables are not taken into account, a mining algorithm can provide association rules such as $BuyProductA \implies BuyProductB$. However, if time is also considered, association rules like these can be discovered: $BuyProductA \implies BuyProductB$ jumps to 70% at the weekend, during 6pm to 10pm [24]. These temporal rules allow retailers to make more efficient, time-aware promotion strategies.

# 3 Data stream processing

## 3.1 Data streams and continuous queries

Data streams are sequences of values produced in a continuous fashion by data sources. In our paper we consider streams of structured data, which contain elements or records, similar to those found in relational databases. From now on we will use the terms record, tuple, element and event interchangeably, to denote the basic unit of data that arrives on a stream. Consider the example of Saturn's radio emissions. The planet is the data source that provides a stream of radio data. Figure 1 depicts three tuples that arrive on the stream displaying Saturn's emissions, one every 30 seconds (the values are normalized).

In a data stream, every element has a timestamp, which represents the time (logical or physical) when that element arrived on the stream. If a patient in a hospital has a monitoring device attached, which tracks the blood pressure, one can tell the value of the monitored parameter at any time. Traditional data from classical databases doesn't encompass the notion of time. This apparently minor add-on has a significant influence on the data processing paradigm.

Applications that handle data streams are called monitoring applications, because they usually scan or monitor a number of data streams. These applications are best served by dedicated systems that process data streams, called Data Stream Management Systems (DSMSs). The main feature of DSMSs that distinguishes them from traditional Database Management Systems (DBMSs) is the fact that they run continuous queries over input data streams. Queries in traditional DBMSs are executed once against stored data relations, in a finite amount of time (these are called one-shot queries [5]). Queries in DSMSs are executed continuously against transient input data streams. Experiments in [6] show that a DSMS outperforms a DBMS when processing high load data streams and executing both continuous and one-shot queries.
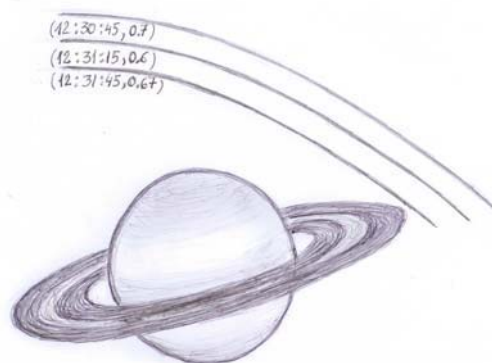
Figure 1: Stream of Saturn's radio emissions

The results of continuous queries are provided in real time, with a specified low latency requirement. For instance, in a financial application, one might want to be alerted whenever a certain stock exceeds a threshold. This kind of answer needs to be produced immediately, it is useless if it is delivered later in the future. Moreover, data is usually processed in a one-pass manner, *i.e.* each tuple that arrives on the stream is processed and then discarded. On the fly processing is a chief trait of DSMSs. Some history of input data can be stored, but in the end it is deleted.

The temporal dimension must be taken into account by any data stream mining algorithm. Furthermore, training sets can be constructed using continuous queries.

## 3.2   Data Stream Management Systems

We describe the main challenges in processing data streams with three DSMSs, that we consider to be highly relevant in this field.

Aurora is a DSMS developed at Brandeis University, Brown University and M.I.T [1], that allows users to express queries via a visual programming language. Queries are created by dragging operators on a GUI and appear in the form of network diagrams, where boxes represent operations and arcs represent data streams.

STREAM is a general purpose DSMS developed at Stanford [5]. STREAM's continuous queries are written in an SQL-like declarative interface, using a specialized query language: CQL (Continuous Query Language). Both STREAM and Aurora are centralized DSMSs.

Borealis is a second generation stream processing engine, which inherits its stream functionality from Aurora [2]. It expresses queries using the boxes-and-arrows paradigm from Aurora.

For each of the described systems, a data stream is a sequence of elements produced by a data source, where each element is attached to a timestamp [7]. This stream is append-only for Aurora and STREAM (an element once arrived on a stream cannot be deleted or updated). Borealis however encompasses a useful mechanism of revision processing, which means previously arrived tuples on a stream can be corrected (and so can the results of queries that take into account such data) [4].

As we can see, up to this point there are no standards for a data model (revision processing is not present in all systems) or a query language over data streams. One of the main difficulties for applying data mining in this context is coping with different data models and different ways of specifying continuous queries.

## 3.3 Improving performance

One way to improve performance in DSMSs is by scheduling query operators in a way that reduces memory and CPU usage. Aurora schedules operators using a train superbox technique [13]. Multiple tuples are pushed as long trains through multiple boxes, in order to reduce box call overhead, improve box optimization and avoid the cost of going to disk. STREAMs scheduling approach is adaptive. If the stream tuple rate is uniform, the system uses a FIFO approach, which aims at minimizing state. If the arrival tuple rate is bursty, STREAM uses its Chain algorithm [8]. Taking into account Quality of Service values, Borealis can schedule operators in a way that maximizes the chances of obtaining important values results [2].

In order to cope with high input data rates, the DSMSs sometimes perform load shedding, *i.e.* they drop tuples, saving both memory and CPU usage. In this manner, even if the data rate is bursty, the DSMS can still provide answers to its continuous queries in a timely fashion. Aurora and Borealis perform load shedding taking into consideration QoS values. Each output value is associated with a QoS function which indicates the quality of the output (in terms of latency, value, etc). Aurora performs two types of load shedding: random and semantic [1]. STREAM performs load shedding by dropping input tuples or minimizing state when the system is overloaded.

Aurora and Borealis base their load shedding and operator scheduling decisions on QoS information, a notion we don't encounter in STREAM. A performant data mining algorithm should also take into consideration different approaches to operator scheduling and load shedding, in order to improve performance, by reducing resource usage.

In this section we provided a general overview of data stream processing aspects. In [21] we give a more detailed presentation of data streams and DSMSs.

# 4 kSiEved Window Training Set technique

## 4.1 Mining data streams in a resource-constrained setting

We identify one of the core difficulties in data stream mining to be resource usage. Data mining on multiple, high load data streams poses problems concerning memory usage as well as CPU time.

In [20] we addressed the problem of resource usage in a continuous setting, by assessing the sizing window effect over continuous queries results. We will borrow some specifications from this model in the kSEWT technique.

[15] describes data-based subset oriented data stream mining solutions as techniques that choose a subset of the input data stream to be subject to data analysis. Examples include sampling (probabilistic choice of a data tuple to be processed or not), load shedding (dropping fractions of input data streams) and sketching (vertically project the data stream, by omitting some fields in its schema). We propose a data-based subset oriented approach to analysing data streams. The basic components of such a technique are represented by the data stream training sets, constructed in an efficient manner, using load shedding mechanisms.

## 4.2 Motivation

Stream data rates can be very high. [11] enumerates several requirements that must be met by a stream mining algorithm, among which we are interested in two: the algorithm must use a limited amount of memory and work in a limited amount of time. Under heavy load input streams, it is difficult to meet these expectations [12]. Hence we will build training sets that are to be supplied to an algorithm in a resource-aware manner, using queries over sliding windows.

With existing computational and memory capabilities, it is physically impossible to compute a continuous query on an entire stream of data, since the stream is infinite. Therefore queries must take into account portions of the streams, *i.e.* sliding windows. A sliding window represents a finite sequence of data from a stream. As new elements arrive on the stream, the window slides over them, continuously refreshing itself (oldest elements are removed from the window as new elements get inserted into the window). The result of executing a query against an entire stream of data would be the ideal result.

When running the query against a sliding window, one obtains an approximation of the ideal result. A sliding window has a fixed size, expressed as number of time instants (time-based sliding window) or number of elements it contains (tuple-based sliding window) [5]. In the rest of the paper we will consider time-based sliding windows.

In time series analysis, a common solution for predicting future values is to use the nearest neighbour technique [10]. If a stream contains temporal stock prices, one approach is to construct training records encompassing 10 consecutive stock prices (the first 9 values represent the predictor, whilst the 10th element is the prediction value). Training records built in this manner are in fact sliding windows.

But even these approximate answers can be computationally difficult to compute. Consider for instance a training set that is the result of a multiple attribute join between three 60 seconds sliding windows (training records) over three data streams, each of which has a data rate of over 1000 tuples / second. Submitting training sets like these to a data mining algorithm would put a great pressure on resource usage.

To address the problem of constructing training sets over streams with high input data rates in a limited resource setting, we designed the kSiEved Window Training Set technique.

## 4.3   Informal description

We provide an informal description of our technique, based on an example from astronomy. We consider that the result of a query on a sliding window is the correct result. The correct result is good enough for our requirements and is an approximation of the ideal result, which could be obtained if we could execute the query against the entire stream.

Consider a data stream that gets tuples from multiple radio telescopes, which monitor different portions of the observable universe. A data mining algorithm could aim at discovering hidden patterns in the stream, indicating for instance the "traffic" of celestial bodies in the monitored areas. An end user might be interested in predicting the number of meteorites passing by whenever a given event occurs. In such a context one would use a specific aggregation class of queries, that mainly performs counting operations.

We intend to build training sets in a way that is dependent on the end user requirements. If the mining algorithm mainly uses aggregation queries that perform certain tasks, we integrate this class of queries in the process of building resource-aware training sets. We do not limit the goal of data mining to only encompass verification. The end user doesn't formulate a rigid hypothesis that needs to be verified. Instead, he or she simply provides a narrowing of the hypotheses characteristics that can be found in the mining process.

We are provided with an error bound, *i.e.* if the error of the approximation of a correct result doesn't exceed this value, then the approximation is good. We superpose sieves with decreasing number of holes over a sliding window. The first sieve has the same number of holes as the number of time instants in the sliding window. The sieve sifts the window's elements, generating a sieved window, that contains only the elements in the initial sliding window that passed through the sieve's holes. The result of executing a continuous query against this sieved window is, at any time instant, the correct result. Subsequently we choose another sieve, with a lower number of holes. When superposing this sieve against the window, it sifts windows elements having positions in the window corresponding to the sieve's holes. We constantly decrease the number of holes in the sieve. We compute queries over each of the chosen sieved windows, thus obtaining approximate answers.

We apply these steps at every time instant. Subsequently we assess the correctness of our results with respect to the error bound. We find the sieve for which the error of the approximation is just below the error bound. That sieve will provide us with training records for a training set. We build training sets composed of a variable number of training records. The number of training sets and the number of training records in a training set both depend on the queries used in the process.

## 4.4 kSEWT model

### 4.4.1 Notion of time

In order to cope with temporal data, we adopt a slightly modified version of the discrete time model described in [16]. $t_i \in T, i \in \mathbb{N}^*$ represents a time instant and we have that: $\forall i, j \in \mathbb{N}^*, i < j \Rightarrow t_i < t_j$.

$T$ is an infinite ordered set of time instants, which goes from past to future instants. For simplicity we are going to choose the set of positive natural numbers to represent it: $T = \mathbb{N}^* = \{1, ..., +\infty\}$.

### 4.4.2 Data model

We borrowed basic notations for our model from [16]: $A$ represents the countable infinite set of attribute names and $D$ is the countable infinite set of constants.

A stream $S$ has a schema (a named set of attributes): $schema(S) \subset A$. A tuple $s$ on stream $S$ is an element of $D^{|schema(S)|}$.

We formally define a stream $S$ as follows:

$$S = \{(s,t)|s \in D^{|schema(S)|}, t \in T\}. \tag{1}$$

We formally define a sliding window over stream S:

$$SW(S, [t_i, t_j]) = \{(s,t) \mid (s,t) \in S, t \in [t_i, t_j]\}. \tag{2}$$

We will use $SW_{ij}(S)$ as an equivalent notation to $SW(S, [t_i, t_j])$, to simplify our equations. $t_i$ is the starting point of the window and $t_j$ is the endpoint of the window. A window slides over its stream with one time instant. When the stream $S$ is obvious, we will denote a sliding window over it by $SW_{ij}$.

We define the size of a sliding window to be the number of time instants it contains: $Size(SW_{ij}(S)) = t_j - t_i + 1$. Since this definition is trivial, when referring to the size of a given sliding window, we will denote it by $\sigma$.

We first established these definitions of the stream, sliding window and sliding window size in [20].

### 4.4.3 kSiEved Window

We denote by $SW(S)$ the set of all sliding windows over stream $S$. In the following, we consider a sliding window $SW_{ij}$ over stream $S$, of size $\sigma$, therefore we omit $S$ in our definitions, to provide a clearer focus on the sieved window.

We define a function that gives the position of a timestamp $t$ inside its window ($t \in [t_i, t_j]$):

$$position: T \times SW(S) \to \mathbb{N}^*, position(t, SW_{ij}) = t - (i - 1). \tag{3}$$

If one applies this function to each timestamp in the window, starting from the earliest, one obtains the set of positions $\{1, 2, ..., \sigma\}$. We will call this set $Positions$. A window with size $\sigma$ has $\sigma$ timestamps and function $position$ maps each of these timestamps to their corresponding ordered positions in the set $Positions$:

$$Positions = position(T_{ij}, SW_{ij}), \text{where } T_{ij} = [t_i, t_j], \text{ for a given window } SW_{ij}. \tag{4}$$

In order to apply a sieve over the given window, we define a function that extracts a subset of positions from $Positions$, in a parameterized manner:

$$kposition: \mathcal{P}(\mathbb{N}^*) \times \mathbb{N}^* \to \mathcal{P}(\mathbb{N}^*), kposition(Positions, k) = \{kp \in Positions \mid kp \equiv 1 \pmod{k}\}. \tag{5}$$

Given a parameter $k$, we choose the holes in our sieve to be represented by all values from $Positions$ which belong to the congruence class of 1 modulo $k$. We denote by $kPositions$ the subset from $Positions$ obtained when applying function $kposition$ with parameter $k$.

We define a kSiEved Window over the sliding window, by a value $k$ ($kPositions$ is computed for window $SW_{ij}$, as described above):

$$SEW(SW_{ij}, k) = \{(s, t) \mid (s, t) \in SW_{ij}, position(t, SW_{ij}) \in kPositions\}. \tag{6}$$

We denote this window by $SEW_{ij}(k)$. This window has the same temporal boundaries $t_i$ and $t_j$ as the sliding window it was computed from. It however only contains elements whose positions are filtered by the $kposition$ function.

### 4.4.4 Continuous queries

Let $Q$ be a continuous query. We denote the result of evaluating $Q$ at time $t$ against a sliding window by $Q(SW_{it}, t)$ (similar to the notation we used in [20]). This is a correct result. We denote the result of evaluating $Q$ at time $t$ against a kSiEved Window by $Q(SEW_{it}(k), t)$. This is an approximation of the correct result.

We fill formally express the notion of query equivalence. We say that $Q(SW_{it}, t)$ is equivalent to $Q(SEW_{it}(k), t)$ at time $t$ and write:

$$Q(SW_{it}, t) \equiv Q(SEW_{it}(k), t) \text{ wrt } \epsilon \iff distance(Q(SW_{it}, t), Q(SEW_{it}(k), t)) \leq \epsilon. \tag{7}$$

In the following we will define our *distance* function.

### 4.4.5 Aggregated value distance

We will consider the result of $Q$ at time $t$ is an aggregated value. As a future direction, we also want to define a distance function that takes into account collection results. For the time being, we will only consider the case when the queries issue aggregated results.

Consider $t_c$ to be the current timestamp.

Let $R_{sw}$ be the result of executing $Q$ against window $SW_{ic}$ at time $t_c$:

$$R_{sw} = Q(SW_{ic}, t_c), R_{sw} \in \mathbb{R}. \tag{8}$$

Consider a kSiEved Window, computed from the sliding window, $SEW_{ic}(k)$. Let $R_{sew}$ be the result of executing $Q$ against $SEW$:

$$R_{sew} = Q(SEW_{ic}(k), t_c), R_{sew} \in \mathbb{R}. \tag{9}$$

Similar to [20], we define the distance function between two aggregated value results as follows:

$$distance : \mathbb{R} \times \mathbb{R} \to \mathbb{R}, distance(R_{sw}, R_{sew}) = |R_{sw} - R_{sew}|. \tag{10}$$

## 5 kSEWT strategy

The purpose of our work is to construct a training set encompassing training records in a manner that considers the limited available resources. We save both memory and CPU time by sifting windows elements through sieves. As the number of elements to be processed decreases, both memory and CPU time are saved. We require less memory and less CPU time to store and to process respectively a smaller number of elements. Some general directions are borrowed from our paper [20], but in this case we describe an algorithmic process that is applied during data mining training sets construction.

Let $H$ be a set of hypotheses constraints specified by an end user. $H$ doesn't establish formulated hypotheses, but some limitations on the models that are to be discovered in the mining process. For our astronomical example, $H$ contains an element that simply states that the user is interested in finding

celestial traffic correlations and / or predictions, using aggregated count queries. So we will construct our training sets with a query that asks: "How many objects were seen by all the telescopes in a given timeframe?".

Let Q be this query. Remember that $t_c \in T$ is the current timestamp and consider $t_i \in T$ to be a timestamp that marks a window's starting point in time. Also note that $c$ actually denotes timestamp $t_c$, for simplicity, where necessary. As $t_c$ slides on the time axis, its value changes. We denote by $CrtTS$ the finite set of all values from $T$ that $t_c$, the current timestamp, will take in our experiments. Subsequently:

1. Choose a threshold, $\delta$. An approximate answer should not differ by more than $\delta$ from the correct answer. This means the query that provides the correct answer is equivalent to the query that provides the approximate answer wrt $\delta$ (according to (7)).

2. Compute the correct result of the query executed against a sliding window of size $\sigma$:

$$R_{sw_c} = Q(SW_{ic}, t_c) \tag{11}$$

3. Construct sieves over the sliding window. Start with k = 1 and increase k by one unit until it reaches $\sigma$. Compute results of Q over kSiEved Windows $SEW_{ic}(k)$:

$$R_{sew_{c_k}} = Q(SEW_{ic}(k), t_c) \tag{12}$$

and compute distances between the correct result and the result of the query against each of the kSiEved Windows:

$$distance_{c_k} = \mid R_{sw_c} - R_{sewc_k} \mid \tag{13}$$

Note: in our experiments, we will normalize the distance function results, so that a distance will take values from the interval [0,1].

4. Perform steps 2 and 3 for all current timestamps $t_c$ from $CrtTS$ (as the current timestamp changes).

5. Once $t_c$ has taken all the values from $CrtTS$, compute the average of distance results over time, for every k value parameter:

$$AvgDistance(k) = \frac{\sum\limits_{c \in CrtTS} d_{c_k}}{|CrtTS|} \tag{14}$$

6. Choose the k value for which $AvgDistance(k)$ doesn't exceed $\delta$, such that $AvgDistance(k+1)$ exceeds $\delta$. This k will provide the kSiEved Window Training Set, that is the set of all kSiEved Windows obtained in the experiment. These windows are the training records that compose the actual training set, that is to be submitted to a generic stream mining algorithm.

The results from step 5 can be plotted on a two-dimensional graph, that will describe how the average distance from the correct result evolves as parameter k increases.

## 5.1 Experimental results

We conducted our experiments on random data with uniform distribution. The simplified data schema looks like this: (TimeStamp, TelescopeID, IN/OUT), where IN/OUT represents an object entering into or exiting from a monitored area. We considered windows of fixed size $\sigma = 60$ time instants. For such a window, the function that gives the cardinality of $kPositions$ is:

$$NoKPositions : \mathbb{N}^* \to \mathbb{N}^*, NoKPositions(k) = \begin{cases} \sigma/k & \text{if } \sigma \mod k = 0 \\ [\sigma/k] + 1 & \text{if } \sigma \mod k \neq 0 \end{cases}$$

This function gives the number of holes in a sieve that is to be applied on a sliding window of size $\sigma$, given parameter k. Figure 2 displays the evolution of the number of holes in the sieve as the k parameter increases from 1 to 60, according to our function $NoKPositions$.
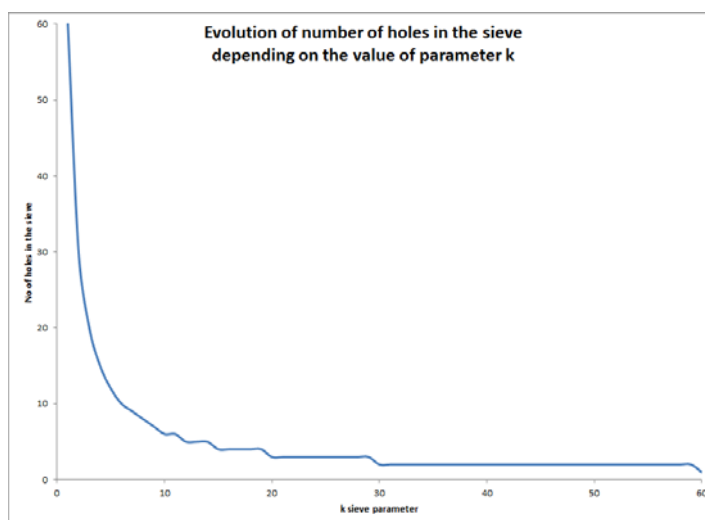
188

Figure 2: The number of holes in the sieve, depending on k

We followed the strategy described above in our experiments. We plotted the results from step 5 on the graph in Figure 3. If we choose $\delta = 0.5$ from this graph it follows we can apply sieves with parameter k = 2 when constructing kSEWT. This means we are dropping half of the input tuples, which translates to significant resource saving. Different data distributions could lead to greater values for k (we leave this as future work).
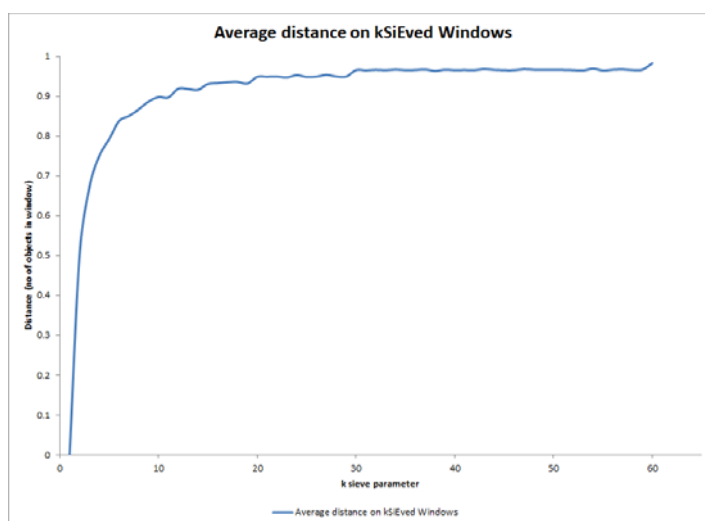


Figure 3: Average distance on kSiEved Windows

# 6    Future research directions

We are working on a language that allows a rigorous specification of the hypotheses constraints set $H$. Having such a formalization will allow us to define quantifiable relationships between the mining algorithm goals and the construction of the training sets, via specific continuous queries.

We intend to quantify resource consumption, in terms of memory and CPU usage. The kSEWT

technique will also take into account, apart from the error bound, a resource consumption bound. This will allow the solution to work on different hardware and software platforms, in a highly configurable manner.

In constructing training sets, the technique currently uses a distance function that measures the error of an answer with respect to expanded versions of the training records, *i.e.* the sliding windows. A future goal is to define an improved version of the distance function. This version of the function will take into account models provided from the mining algorithm the training sets are to be submitted to. This way we will greatly improve the accuracy of our results.

# 7    Conclusion

Data stream processing is still in its infancy. Standards for data models and query processing have yet to be defined, as the field has a long way until it reaches maturity. In this context, data mining tasks must be applied in a heterogeneous environment, where streams are represented in different ways, depending on the system and queries are specified and executed in different manners. We proposed a resource-aware, generalized approach to constructing training sets for a generic data stream mining algorithm, the kSiEved Window Training Set technique. We highlighted the main difficulties we encountered and our solutions to some of these challenges. We also outlined future research directions generated by our work.

# References

[1] D.J. Abadi, D. Carney, U. Cetintemel, M. Cherniack, C. Convey, S. Lee, M. Stonebraker, N. Tatbul, S. Zdonik, Aurora: a new model and architecture for data stream management. *The VLDB Journal*, vol. 12, n° 2, p. 120-139, 2003.

[2] D.J. Abadi, Y. Ahmad, M. Balazinska, U. Cetintemel, M. Cherniack, J.H. Hwang, W. Lindner, A.S. Maskey, A. Rasin, E. Ryvkina, N. Tatbul, Y. Xing, S. Zdonik, The Design of the Borealis Stream Processing Engine. *Proceedings of the 2005 Conference on Innovative Data Systems Research (CIDR)*, p. 277-289, 2005.

[3] R. Agrawal, T. Imielinski, A. Swami, Mining Association Rules between Sets of Items in Large Databases. *Proceedings of the 1993 ACM SIGMOD Conference*, p. 207-216, 1993.

[4] Y. Ahmad, B. Berg, U. Cetintemel, M. Humphrey, J.H. Hwang, A. Jhingran, A. Maskey, O. Papaemmanouil, A. Rasin, N. Tatbul, W. Xing, Y. Xing, S. Zdonik, Distributed Operation in the Borealis Stream Processing Engine (demo). *ACM SIGMOD Conference*, p. 882-884, 2005.

[5] A. Arasu, B. Babcock, S. Babu, J. Cieslewicz, M. Datar, K. Ito, R. Motwani, U. Srivastava, J. Widom, STREAM: The Stanford Data Stream Management System. *Technical Report*, Stanford InfoLab, 2004.

[6] A. Arasu, M. Cherniack, E. Galvez, D. Maier, A.S. Maskey, E. Ryvkina, M. Stonebraker, R. Tibbetts, Linear Road: A Stream Data Management Benchmark. *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB '04)*, p. 480-491, 2004.

[7] B. Babcock, S. Babu, M. Datar, R. Motwani, J. Widom, Models and Issues in Data Stream Systems. *Proceedings of the 21st ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (PODS '02)*, p.1-16, 2002.

[8] B. Babcock, S. Babu, M. Datar, R. Motwani, Chain: Operator Scheduling for Memory Minimization in Data Stream Systems. *Proceedings of Special Interest Group on Management of Data Conference 2003 (SIGMOD'03)*, p. 253-264, 2003.

[9] J. Becla, K.T. Lim, D.L. Wang, Report from the 4th Workshop on Extremely Large Databases. *Data Science Journal*, vol. 9, pp. MR1-MR8, 2011.

[10] A. Berson, S. Smith, K. Thearling, Building Data Mining Applications for CRM. *McGraw-Hill Companies*, 1999.

[11] A. Bifet, G. Holmes, R. Kirkby, B. Pfahringer, Moa Data Stream Mining. A Practical Approach. *Center for Open Software Innovation, The University of Waikato*, 2011.

[12] A. Bifet, G. Holmes, B. Pfahringer, P. Kranen, H. Kremer, T. Jansen, T. Seidl, Moa: Massive Online Analysis, a Framework for Stream Classification and Clustering. *JMLR: Workshop and Conference Proceedings*, vol. 11, p. 44-50, 2010.

[13] D. Carney, U. Cetintemel, A. Rasin, S. Zdonik, M. Cherniack, M. Stonebraker, Operator Scheduling in a Data Stream Manager. *Proceedings of the 29th International Conference on Very Large Data Bases (VLDB '03)*, p. 838-849, 2003.

[14] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, vol. 17, n° 3, p. 37-54, 1996.

[15] M.M. Gaber, A. Zaslavsky, S. Krishnaswamy, Mining Data Streams: A Review. *ACM SIGMOD Record*, vol. 34, n° 2, p.18-26, 2005.

[16] Y. Gripay, *A Declarative Approach for Pervasive Environments: Model and Implementation*. Ph.D. Thesis, Institut National des Sciences Appliquées de Lyon, 2009.

[17] D. Hand, H. Mannila, P. Smyth, *Principles of Data Mining*. The MIT Press, Cambridge, MA, 2001.

[18] H. H.O. Nasereddin, Stream Data Mining. *International Journal of Web Applications*, vol. 1, n° 4, 2009.

[19] M. Hilbert, P. Lopez, The World's Technological Capacity to Store, Communicate, and Compute Information. *Science*, vol. 332, n° 6025, p. 60-65, 2011.

[20] S. Surdu, V. M. Scuturici, Addressing resource usage in stream processing systems: sizing window effect. *The International Database Engineering and Applications Symposium ACM International Conference Proceeding Series*, p. 247-248, 2011.

[21] S. Surdu, Data stream management systems: a response to large scale scientific data requirements. *Annals of the University of Craiova, Mathematics and Computer Science Series*, vol. 38, n° 3, p. 66-75, 2011.

[22] K. Thearling, An Introduction to Data Mining. *URL: http://www.thearling.com/text/dmwhite/dmwhite.htm* .

[23] International Telecommunication Union, The Internet of Things. *ITU Internet Reports*, 2005.

[24] Q. Zhao, S. S. Bhowmick, Sequential Pattern Mining: A Survey. *Technical Report, CAIS, Nanyang Technological University, Singapore*, 2003.

Sabina Surdu, Ph.D. Student
Babes-Bolyai University
Faculty of Mathematics and Computer Science
Str. Mihail Kogalniceanu nr. 1, RO-400084, Cluj-Napoca
ROMANIA
E-mail: *surdusabina@yahoo.com*

# From Bernstein Polynomials to Lagrange Interpolation

**Gancho Tachev**

### Abstract

For a given continuous function $f(x)$ on $[0,1]$ we construct sequence of algebraic polynomials based on Bernstein approximation. We prove that the limit of this sequence is the Lagrange interpolation polynomial of degree $n$. Application to the representation of polynomial curves will be given.

## 1 Introduction

For any continuous function $f \in C[0,1]$ the classical $n-$th degree Bernstein polynomial at the point $x \in [0,1]$ is given by

$$B_n(f;x) = \sum_{k=0}^{n} f(\frac{k}{n}) \cdot p_{n,k}(x), \tag{1.1}$$

where the Bernstein basis polynomials $p_{n,k}$ are defined as

$$p_{n,k}(x) = \binom{n}{k} x^k (1-x)^{n-k}. \tag{1.2}$$

Setting

$$\bar{F} := \begin{pmatrix} f(0) \\ f(\frac{1}{n}) \\ \vdots \\ f(1) \end{pmatrix} = (f(0), f(\frac{1}{n}), \ldots, f(1))^t, \tag{1.3}$$

and

$$\bar{b}_n(x) = (p_{n,0}(x), p_{n,1}(x), \ldots, p_{n,n}(x)) \tag{1.4}$$

we may rewrite (1.1) in matrix representation by

$$B_n(f,x) = \bar{b}_n(x) * \bar{F}, \tag{1.5}$$

where in the remainder of the paper $*$ always denotes matrix multiplication. The Lagrange interpolation polynomial (Lagrange interpolant) of degree $n$ is given by

$$L_n(f,x) = \sum_{k=0}^{n} f(\frac{k}{n}) \cdot l_{n,k}(x), \tag{1.6}$$

where the Lagrange basis polynomials $l_{n,k}$ are defined as

$$l_{n,k}(x_i) = \delta_{i,k} = \begin{cases} 1, & i = k \\ 0, & i \neq k \end{cases}, \ 0 \leq i, k \leq n. \tag{1.7}$$

Setting

$$\bar{l}_n(x) = (l_{n,0}(x), l_{n,1}(x), \ldots, l_{n,n}(x)) \tag{1.8}$$

we may rewrite (1.6) by

$$L_n(f, x) = \bar{l}_n(x) * \bar{F}. \tag{1.9}$$

The essential role in our further considerations plays the following $(n+1) \times (n+1)$ matrix $T$, whose elements $[t_{k,j}^{(n)}]$, $k, j = 0, \ldots, n$ are the values of Bernstein basis polynomials $\{p_{n,i}(x)\}_{i=0}^n$ at the knots $0, \frac{1}{n}, \frac{2}{n}, \ldots, 1$:

$$T := \begin{pmatrix} p_{n,0}(0) & p_{n,1}(0) & p_{n,2}(0) & \ldots & p_{n,n}(0) \\ p_{n,0}(\frac{1}{n}) & p_{n,1}(\frac{1}{n}) & p_{n,2}(\frac{1}{n}) & \ldots & p_{n,n}(\frac{1}{n}) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ p_{n,0}(1) & p_{n,1}(1) & p_{n,2}(1) & \ldots & p_{n,n}(1) \end{pmatrix} \tag{1.10}$$

This matrix was studied and applied in many different areas of analysis, numerical methods, computer aided geometric design etc. For example in [1] a new fast method was introduced, to approximate the value of a definite integral of $f \in C[0,1]$. This method gives considerable better results for a broad class of sufficiently differentiable functions, if compared with other known quadrature rules like Simpson rule, composite trapezoid rule etc. The author has not observed that the limiting operator defined in [1] by $J_n(f)$ is actually the Lagrange interpolant of degree $n$, defined in (1.6). The algorithm developed in [1] is the main motivation for us to write this paper. Other application of the matrix $T$ is given in [4] to construct recursive subdivision algorithm for polynomial curves-one of the basic tools in computer aided geometric design (CAGD). In [4] it was shown, that the control polygons produced by recursive subdivision always converge to their original curve. We may continue with many other applications of the matrix $T$, but these two examples are enough to show its significance and application to solve different problems. In our paper we consider only the values of a given function $f$ at equidistant knots $\frac{k}{n}$, $k = 0, 1, \ldots, n$. This may be extended to arbitrary set of knots $0 \le t_0 < t_1 < \ldots < t_n \le 1$ as it was studied in [4]. Our main result states the following

**Theorem 1** *If $f$ is any bounded function, defined on the interval $[0,1]$, then for all $n = 1, 2, \ldots$ and all $x \in [0,1]$ we have*

$$\bar{l}_n(x) * \bar{F} = \bar{b}_n(x) * T^{-1} * \bar{F}, \tag{1.11}$$

*where $T^{-1}$ is the inverse matrix of $T$.*

In Section 2 we establish some properties of $T$, we construct a sequence of algebraic polynomials $\{\tilde{B}_m\}_{m=1}^\infty$ of degree $n$ and show that its limit is the Lagrange interpolant $L_n(f)$. The proof of Theorem 1 is based on this observation. In Section 3 we show some applications and corollaries of our main result.

## 2   Proof of Theorem 1

We establish some properties of the matrix $T$.

**Lemma 1** *The matrix $T$ is nonsingular.*

**Proof:** If we suppose the contrary, then the columns are linear dependent, i.e. there are constants $\lambda_0, \lambda_1, \ldots, \lambda_n$, at least one of which different from 0, such that

$$h_n(x) := \lambda_0 \cdot p_{n,0}(x) + \lambda_1 \cdot p_{n,1}(x) + \cdots + \lambda_n \cdot p_{n,n}(x) = 0$$

holds for $x = 0, \frac{1}{n}, \frac{2}{n}, \ldots, 1$. Hence $h_n(x) \equiv 0$ for all $x \in [0,1]$. It follows that polynomials $\{p_{n,i}(x)\}_{i=0}^n$ should be linear dependent, but this is not possible, because they build basis in the space of all algebraic polynomials of degree $\le n$. Therefore $P$ is a nonsingular matrix. $\square$

The matrix $T$ was used also in [2] to establish the eigenstructure of the classical Bernstein operator. An useful property of $T$ is that $T$ is a positive definite -see Proposition 5.1 in [5]. Using this fact the following was proved in Lemma 4.3 in [1], which we formulate here as

**Lemma 2** *For $i = 0, 1, \ldots, n$ we have $0 < \lambda_i(T) \le 1$, where $\lambda_i(T)$ are the eigenvalues of $T$.*

Lemma 2 implies that $\rho(I - T) < 1$ where $\rho(I - T)$ is the spectral radius of the matrix $I - T$, with $I$-the identity matrix. In [2] it was established that the eigenvalues of the operator $B_n$ are given by

$$\lambda_k^{(n)} := \frac{n!}{(n-k)!n^k}, \ k = 0, 1, \ldots, n.$$

Therefore

$$1 = \lambda_0^{(n)} = \lambda_1^{(n)} > \lambda_2^{(n)} > \ldots > \lambda_n^{(n)} > 0.$$

If $f_k(x)$ is the eigenfunction of $B_n$ corresponding to $\lambda_k^{(n)}$ then

$$B_n(f_k, x) = \lambda_k^{(n)} \cdot f_k(x). \tag{2.1}$$

Consequently we arrive at

**Lemma 3** *The eigenvalues of $T$ choincides with the eigenvalues of $B_n$.*

**Proof:** In the representation (2.1) we set $x = 0, \frac{1}{n}, \frac{2}{n}, \ldots, 1$ and obtain

$$T * \bar{f}_k = \lambda_k^{(n)} \cdot I * \bar{f}_k,$$

where $\bar{f}_k = [f_k(0), f_k(\frac{1}{n}), \ldots, f_k(1)]^t$. The last equation implies that

$$\det\left(T - \lambda_k^{(n)} \cdot I\right) = 0,$$

i.e. $\lambda_k^{(n)}$ is an eigenvalue of $T$. $\qquad\square$

The next statement is well known from the theory of linear algebra, namely

**Lemma 4** *If $A$ is a square matrix with $\rho(A) < 1$ then the matrix $I - A$ is invertible and we have*

$$(I - A)^{-1} = I + A + A^2 + \ldots.$$

If we set $A := I - T$ in the last formula we obtain

**Lemma 5** *The matrix $T$ is invertible and we have*

$$T^{-1} = I + (I - T) + (I - T)^2 + \ldots = \prod_{m=0}^{\infty} (I - T)^m. \tag{2.2}$$

Lemma 5 gives another proof of the property of $T$ formulated in Lemma 1 and in addition the useful representation of the inverse matrix $T^{-1}$. We end our study of $T$ with the following observation (see Lemma 2.2 in [4]):

**Lemma 6** *All rows of $T$ and $T^{-1}$ sum to 1.*

Let us calculate $T$ explicitly for $n = 1, 2, 3$

$$n = 1 \ T = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \ T^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$n = 2 \quad T = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & 1 \end{pmatrix}, \ T^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{2} & 2 & -\frac{1}{2} \\ 0 & 0 & 1 \end{pmatrix}$$

$$n = 3 \quad T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{8}{27} & \frac{4}{9} & \frac{2}{9} & \frac{1}{27} \\ \frac{1}{27} & \frac{2}{9} & \frac{4}{9} & \frac{8}{27} \\ 0 & 0 & 0 & 1 \end{pmatrix}, \ T^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{5}{6} & 3 & -\frac{3}{2} & \frac{1}{3} \\ \frac{1}{3} & -\frac{3}{2} & 3 & -\frac{5}{6} \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Let $f$ be arbitrary bounded function defined on $[0,1]$ and let $x \in [0,1]$-be fixed. We may write recursively

$$
\begin{aligned}
f(x) &= B_n(f,x) + r_1(x) \\
r_1(x) &= B_n(r_1,x) + r_2(x) \\
r_2(x) &= B_n(r_2,x) + r_3(x) \\
&\cdots \\
r_m(x) &= B_n(r_m,x) + r_{m+1}(x).
\end{aligned}
\tag{2.3}
$$

It is easy to verify, that by the method introduced in (2.3) we construct a sequence of $n$-th degree algebraic polynomials $\{\tilde{B}_m(f,x)\}_{m=0}^{\infty}$ defined by

$$
\tilde{B}_m(f,x) = \bar{b}_n(x) * \left[ \sum_{i=0}^{m} (I-T)^i \right] * \bar{F},
\tag{2.4}
$$

which satisfy

$$
f(x) = \tilde{B}_m(f,x) + r_{m+1}(x), \ m = 0,1,2,\ldots
\tag{2.5}
$$

It is easy to observe that

$$
\begin{aligned}
\tilde{B}_0(f,x) &= B_n(f,x) \\
\tilde{B}_1(f,x) &= B_n(f,x) + \bar{b}_n(x) * (I-T) * \bar{F} = \\
&= B_n(f,x) + B_n(f,x) - B_n \circ B_n(f,x) = \\
&= B_n(f,x) + B_n(f,x) - B_n^2(f,x) = \\
&= B_n(f,x) - B_n(f - B_nf, x),
\end{aligned}
\tag{2.6}
$$

where $B_n^k := B_n \circ B_n \circ \ldots \circ B_n \ -k$ times is the $k$-th iterates of the Bernstein operator. By the definition of the polynomial sequence $\{\tilde{B}_m\}_{m=0}^{\infty}$ in (2.4) we may consider $\tilde{B}_m$ as linear operator $\tilde{B}_m : C[0,1] \to C[0,1]$. It is clear from (2.6) that $\tilde{B}_m$ are not positive operators, like the Bernstein operator $B_n$. On the other hand they interpolate the function $f$ at the ends and it is natural to expect that $\tilde{B}_m f$ approximates $f$ at the point $x$ better than $B_n f$. For example if $f = e_2 : x \to x^2$, $x \in [0,1]$ it is known that (see Ch. 10 in [3])

$$
\begin{aligned}
\tilde{B}_0(e_2,x) &= B_n(e_2,x) = e_2(x) + \frac{x(1-x)}{n}, \\
\tilde{B}_1(e_2,x) &= e_2(x) + \frac{x(1-x)}{n^2}, \ldots \\
\tilde{B}_m(e_2,x) &= e_2(x) + \frac{x(1-x)}{n^{m+1}}.
\end{aligned}
\tag{2.7}
$$

It is clear from (2.7) that the error of approximation of $e_2(x)$ by $\tilde{B}_m(e_2,x)$ is essentially smaller than by $B_n(e_2,x)$. It is known that the optimal rate of approximation for $B_n$ is $O(\frac{1}{n})$, $n \to \infty$ and if $f(x) - B_n(f,x) = o_x(\frac{1}{n})$, $n \to \infty$, then $f$ is linear function -see Theorem 5.1 in Ch. 10 in [3]. It is easy to observe that for all $m \geq 0$, $\tilde{B}_m$ preserves linear functions. So we may conclude that the loss of positivity is compensate for by better degree of approximation. Consequently from (2.1) and (2.4) we get

$$
\tilde{B}_\infty(f,x) = \bar{b}_n(x) * T^{-1} * \bar{F}.
\tag{2.8}
$$

**Lemma 7** *The sequence of polynomials $\{\tilde{B}_m f\}_{m=0}^{\infty}$ uniformly tends to its limiting operator $\tilde{B}_\infty f$ over the interval $[0,1]$ when $m \to \infty$.*

**Proof:** The representations (2.4) and (2.8) imply

$$
\tilde{B}_\infty(f,x) - \tilde{B}_m(f,x) = \bar{b}_n(x) * \left[ \sum_{i=m+1}^{\infty} (I-T)^i \right] * \bar{F}.
\tag{2.9}
$$

By Lemma 5 we know that the power series (2.2) is convergent and this implies that the matrix $[\sum_{i=m+1}^{\infty} (I-T)^i] \to 0$ when $m \to \infty$. This completes the proof. $\qquad\square$

By elementary calculations we obtain that if $n = 2$ the limiting operator $\tilde{B}_\infty f$ preserves the monomial functions $e_i$, $i = 0, 1, 2$ and if $n = 3$ then $\tilde{B}_\infty f$ preserves all $e_i$, $i = 0, 1, 2, 3$. This could be generalized for arbitrary natural number $n$. Theorem 1 implies that the operator $\tilde{B}_\infty f$ reproduces all algebraic polynomials of degree not greater than $n$.

**Proof of Theorem 1**

For a given function $f$ let us denote by $\bar{B}_n f$ the vector

$$\bar{B}_n f := [B_n(f, 0), B_n(f, \frac{1}{n}), \ldots, B_n(f, 1)]^t. \tag{2.10}$$

It is clear that from (1.3) and the definition of $T$ we get

$$\bar{B}_n f = T * \bar{F}. \tag{2.11}$$

In a similar way if we denote by

$$\bar{B}_\infty := [\tilde{B}_\infty(f, 0), \tilde{B}_\infty(f, \frac{1}{n}), \ldots, \tilde{B}_\infty(f, 1)]^t$$

then (2.8) and (2.11) imply

$$\bar{B}_\infty = T * T^{-1} * \bar{F} = I * \bar{F} = \bar{F}, \tag{2.12}$$

that is the $n$-th degree algebraic polynomial $\tilde{B}_\infty(f, x)$ interpolates the function $f$ at the knots $0, \frac{1}{n}, \frac{2}{n}, \ldots, 1$. On the other hand such an algebraic polynomial is uniquely defined and this is $n$-th degree Lagrange interpolant $L_n(f, x)$ for a given function $f$. Therefore we arrive at

$$\tilde{B}_\infty(f, x) \equiv L_n(f, x).$$

The proof of Theorem 1 is completed. $\qquad\qquad\square$

# 3  Applications

**A. Approximation properties**

If $f \in C^{n+1}[0, 1]$ then the remainder for Lagrange interpolant can be represented as

$$f(x) = L_n(f, x) + \frac{(x - 0)(x - \frac{1}{n}) \cdots (x - 1)}{(n + 1)!} \cdot f^{(n+1)}(), \tag{3.1}$$

for some $\in (0, 1)$. This implies

**Theorem 2** *If $f \in C^{n+1}$ and $\|f^{(n+1)}\| \le M_{n+1}$, where $\|\cdot\|$ denotes sup norm, then*

$$\|f - L_n f\| \le \frac{M_{n+1}}{(n + 1)!} \to 0, \, n \to \infty. \tag{3.2}$$

Also using (2.9) and Lemma 3 we can easily prove

**Theorem 3** *For $f \in C[0, 1]$ the following holds true*

$$\|\tilde{B}_\infty f - \tilde{B}_m f\| \le \|f\| \cdot (1 - \lambda_n^{(n)})^{m+1}. \tag{3.3}$$

*Therefore*

$$\lim_{m \to \infty} \|\tilde{B}_\infty f - \tilde{B}_m f\| = 0.$$

196

**B. Applications in CAGD**

Let us consider the vector-valued parametric-defined function $f : [0,1] \to R^d$, $d \geq 2$. If $f$ is $n-$th degree algebraic polynomial, then as usual we associate with $f(x)$ the $d$-dimensional polynomial curve $\vec{C}(x)$, namely

$$\vec{C}(x) = f(x), \, x \in [0,1].$$

It is known from CAGD that any polynomial curve $\vec{C}(x)$ may be represented in its Bernstein-Bézier form as

$$\vec{C}(x) = \bar{b}_n(x) * \bar{P}, \tag{3.4}$$

where

$$\bar{P} := [P_0, P_1, \ldots, P_n]^t, \, P_i \in R^d, \, 0 \leq i \leq n,$$

denotes the control polygon of the curve and $P_i$ are the control points of $\vec{C}(x)$. From Theorem 1 we immediately get

**Corollary 1** *For any polynomial curve $\vec{C}(x)$, $x \in [0,1]$ the control points in its Bernstein-Bézier representation can be computed by*

$$\bar{P} = T^{-1} * \bar{F}, \tag{3.5}$$

*where $\bar{F}$ is defined in* (1.3) *and consists of points, lying on the curve.*

Our next statement is inverse to the previous one. If now the vector of control points $\bar{P}$ is given, then we can easily compute the coordinates of the points $\vec{C}(\frac{k}{n})$, $0 \leq k \leq n$. Hence

**Corollary 2** *Let the curve $\vec{C}$ be defined by* (3.4). *Then the vector $\bar{F}$ can be computed by*

$$\bar{F} = T * \bar{P}. \tag{3.6}$$

In [4] a subdivision algorithm was introduced, such that the control polygons, obtained in each step, uniformly tend to the curve. Similar statement follows immediately from Theorem 1 and the sequence of polynomials $\tilde{B}_m(f,x)$ defined in (2.4)

**Corollary 3** *The sequence of control polygons of the curves, associated with the polynomials $\tilde{B}_m(f,x)$ uniformly w.r.t. $x \in [0,1]$ tends to the control polygon of the curve $\vec{C}(x) = f(x)$.*

# References

[1] M.Amirfakhrian, Some Numerical Integration Methods Based on Bernstein Polynomials, Int. J. of Comp. Math., 88(2011),no.6, 1294-1304.

[2] S.Cooper and S.Waldron, The Eigenstructure of the Bernstein Operator, J. Approx. Th., 105(2000), 133-165.

[3] R.A.DeVore and G.G.Lorentz, Constructive Approximation. Berlin: Springer 1993.

[4] R.N.Goldman and T.D.Derose, Recursive subdivision without the convex hull property, Computer Aided Geometric Design, 3(1986), 247-265.

[5] A.Pinkus, Spectral properties of totally positive kernels and matrices, in "Total Positivity and Its Applications" (M. Gasca and C. A. Micchelli, Eds.), pp. 477511, Kluwer Academic, Dordrecht, 1996.

Gancho Tachev
University of Architecture
Dept. of Mathematics
1 Hr.Smirnenski, blvd., Sofia, 1046
BULGARIA
E-mail: *gtt_fte@uacg.bg*

# Ant colony optimization pheromone correction strategies

**Milan Tuba** [1]

### Abstract

Ant colony optimization (ACO) algorithm was recently successfully used to find suboptimal solutions to many hard optimization problems. There are few versions of the algorithm that improve its performance. The main problem is that algorithm can be trapped into local minima unable to escape. We implemented a hybridization for the ACO algorithm that proved to be efficient in avoiding that problem. The difference is that, while all other methods for avoiding local minima widen the search randomly, our algorithm excludes undesirable pars of the solution. We tested our method on four different graph optimization problems where in all cases it improved results.

## 1  Introduction

Many real-life problems can be represented as some kind of optimization problem, often an untractable one. Nature inspired metaheuristics have been used recently to find suboptimal solutions to hard optimization problems by simulating various natural phenomena. Swarm intelligence algorithms are a class of nature inspired algorithms that try to mimic collective intelligence of colonies of ants, bees, birds etc. By trying to simulate implicit intelligence of these swarms we talk about bee colony food finding or ant colony path finding but in essence, in all these diverse mimicking, we do two things. We exploit good found solutions, but also go to unknown less promising places in order to avoid being trapped in local minima. The successfulness of any algorithm is determined by proper balance between exploitation and exploration. This paper examines ant colony pheromone correction strategies which change exploitation and exploration behavior of the original algorithm and application of these strategies to some combinatorial problems.

## 2  Ant colony optimization (ACO) algorithm

The ant colony optimization (ACO) is a relatively new meta-heuristic for solving combinatorial problems. ACO is, like genetic algorithms (GA), population based. It was first used for the travelling salesman problem (TSP) by M. Dorigo [1] with very good results.

The basic idea of ACO is to imitate the behavior of ants in a colony while gathering food. Each ant starts from the nest and walks toward food. It moves until an intersection where it decides which path to take; in the beginning it seems to be a random choice, but after some time the majority of ants are using the optimal path (Figure 1). This happens because the colony works as a group and not just as individual ants and this is achieved by using pheromone. Each ant deposits pheromone while walking, which marks the route taken. The amount of pheromone indicates the usage of a certain route. Pheromone trail evaporates as time passes. Due to this a shorter path will have more of pheromone

because it will have less time to evaporate before it is deposited again. The colony behaves intelligent because each ant chooses paths that have more pheromone.
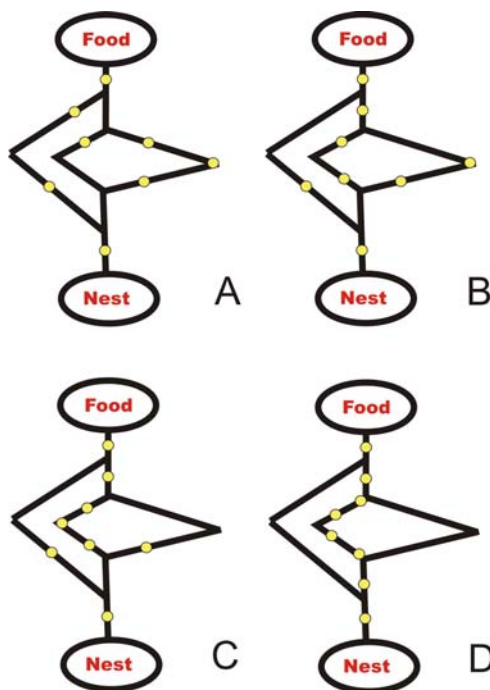


Figure 1: : Ant colony behavior over time

There are many different ways of converting the presented behavior into a computational system. We outline the one presented by Marco Dorigo and Luca Maria Gambardella [2], with small modifications,

$$
s = \begin{cases} \arg\max\limits_{u \notin M_k} \left\{ {\tau_{rs}}^{\alpha} {\eta_{rs}}^{\beta} \right\} & , q \leq q_0 \\ S & , q > q_0 \end{cases} \tag{1}
$$

$$
p_{rs}^{k} = \begin{cases} \dfrac{{\tau_{rs}}^{\alpha} {\eta_{rs}}^{\beta}}{\sum\limits_{u \notin M_k} {\tau_{ru}}^{\alpha} {\eta_{ru}}^{\beta}} & , s \notin M_k \\ 0 & , s \in M_k \end{cases} \tag{2}
$$

Equations 1 and 2 give the probabilistic decision method that artificial ant $k$, currently at node $r$, after visiting nodes in $M_k$ uses for choosing the next node $s$.

- $q$ is a random variable chosen uniformly from $[0, 1]$

- $q_0$ is a predefined parameter that gives us a balance between exploitation (use of known good paths, $q <= q_0$) and exploration (search for new paths, $q > q_0$)

- In the case of exploitation, the next node is selected by the highest value of $S$, which gives the value of desirability of an edge depending on the amount of pheromone and its length. In Equations (1) and (2), $\tau_{rs}$ is the value corresponding to the amount of pheromone deposited on edge connecting $r$ and $s$, and $\eta_{rs}$ is the length of $rs$ which is used as a heuristic.

- $\alpha$ and $\beta$ are predefined parameters that specify the influence of the pheromone and the heuristic, respectively.

- In the case of exploration the next node is chosen at random with a probability distribution given by Equation (2), where $p_{rs}$ is the probability of choosing edge $rs$.

The pheromone trail is created using two types of updates. Global update is used to reward good paths, or in other words more pheromone should be deposited on better paths. This is achieved by using the following formula

$$\tau_{ij} = (1 - \gamma)\tau_{ij} + \gamma\Delta\tau^k \qquad , \forall ij \in B^k \qquad (3)$$

$B_k$ is the set of all edges in the path ant $k$ used, $\Delta\tau^k$ is the quality of that solution, and $\gamma$ is a predefined constant.

The local updating is used to avoid creation of a very strong edge used by all ants, and it emulates pheromone evaporation. Every time an edge is chosen by an ant it loses some pheromone by the following formula where $\delta$ is a predefined constant.

$$\tau_{ij} = (1 - \delta)\tau_{ij} + \delta\tau_0 \qquad (4)$$

# 3 Variations of the ACO algorithm

There are different methods of improving ACO like certain types of hybridizations. Standard hybridizations are the combination of the basic algorithm with a local search [3], [4], [5] or some other algorithm. Combining ACO with genetic algorithms (GA) has resulted in algorithms that gave better results than separate use of these methods on a wide range of different problems [6], [7]. These hybridizations are effective in increasing the efficiency of ACO, but are often complicated for implementation. The complexity of their implementation is due to the fact that to separate algorithms need to be developed: one for the ACO and another for the local search or for the genetic algorithm. Also, it has been shown that hybridization may prevent the ACO from finding the optimal solution [8]. The other method of improving the performance of ACO is the use of different variations of the basic algorithm. On TSP different variations of ACO gave different quality of results, and no variation can be considered the best [9].

Variations of the basic ACO like elitist ant colony, rank based ant colony system, min-max ant system (MMAS) have been developed to improve the performance on TSP [10]. Similar variations have been used on other problems solved by ACO. All these variations have the problem of becoming trapped in local optima. An interesting approach, named the minimum pheromone threshold strategy (MPTS), was proposed for the quadratic assignment problems to solve the stagnation problem [11].

Ant System (AS), is the most basic implementation of ACO; in this version of the algorithm all ants are equal and leave pheromone. It is defined with Equations 5 and 4. AntS is the set of all the solutions created by ants in the current step of the algorithm.

$$\forall i \in \bigcup_{V_i' \in AntS} V_i'$$

$$\tau_i = \sum_{V_k' \in AntS} \frac{1}{\sum_{j \in V_k'} w(j)} \qquad (5)$$

Reinforced Ant System (RAS), which is the same as Ant system, except that the global best solution is reinforced each iteration.

$$\forall i \in V_{gb}' \cup \bigcup_{V_i' \in Ants} V_i'$$

$$\tau_i = \frac{1}{\sum_{j \in V_{gb}'} w(j)} + \sum_{V_k' \in AntS} \frac{1}{\sum_{j \in V_k'} w(j)} \qquad (6)$$

In some variations of this method the iteration best solution is also reinforced each iteration. With this approach the basic AS is made to be slightly greedier. It is defined with Equations 6 and 4.

201

Elitist Ant System (EAS) is the version of the ACO algorithm where individual ants do not automatically leave pheromone. In each iteration step of the colony, or in other words when all the ant complete their solutions, only the global best solution will be used to update the pheromone trail. In this way, the search is even more centralized around the global best solution. It is defined with Equations 7 and 4.

$$\tau_i = \frac{1}{\sum\limits_{j \in V'_{gb}} w(j)}, \quad \forall i \in V'_{gb} \tag{7}$$

MinMax Ant System (MMAS) is the same as the Elitist Ant colony System, but with an extra constraint that all pheromone values are bounded. We adopt the formulas presented in article [12] in which max is calculated dynamically as new best solutions are found by Equation 8, and min is calculated at the beginning of calculations by Equation 9. Parameter $avg$ is the average number of vertexes that are possible to be chosen, $pbest$ is the possibility of the best overall solution being found and $\tau_0$ is the initial value of the pheromone trail calculated as the quality measure of the greedy algorithm solution.

$$\tau_{max} = \frac{1}{(1-p)}\tau_{gb} \tag{8}$$

$$\tau_{min} = \frac{\tau_0(1 - \sqrt[n]{p_{best}})}{(avg - 1)\sqrt[n]{p_{best}}} \tag{9}$$

This variation has two effects that improve the effectiveness of EAS. First, the pheromone trial will not become very strong on some good vertexes and making them a part of almost all newly created solutions. By giving a lower bound to the pheromone trail the potential problem of certain parts of the solution being totally excluded from the search due to very weak values of pheromone is avoided.

Rank Based Ant Colony System (RANKAS) [13] is a modification in which besides the quality we also use the rank (R) of found solutions. Rank is defined by the quality of the solution compared to solutions found by other ants in the same iteration. It is defined with Equations 4 and 10:

$$BRank = \{V | (R(V) < RK)\hat{} (V \in AntS)\}$$

$$\forall i \in V'_{gb} \cup \bigcup_{V'_i \in BRank} V'_i$$

$$\tau_i = \frac{1}{\sum\limits_{j \in V'_{gb}} w(j)} + \sum_{V'_k \in BRank} \frac{(RK - R(V))}{RK} \frac{1}{\sum\limits_{j \in V'_k} w(j} \tag{10}$$

In the implementation of this algorithm, it is important how many best solutions will be taken into account when updating the pheromone trail. In Equation 10 parameter $RK$ is used to define the number of best ranked ants which will affect the trail. This parameter is user defined and it is very important for the effectiveness of this algorithm. In its extreme cases when $RK$ is equal to 0 $RANKAS$ is equivalent to EAS.

## 4    Avoiding stagnation in the ACO algorithm

The ACO meta-heuristic algorithm has several different variations which are used of which elitist ant system (EAS) and Min-max ant system (MMAS) are used most commonly.

EAS increases the efficiency of the basic ACO by making its search more greedy; this is done through intensifying the search near the global best solution. In this version only the global best solution (or in some variations only the iteration best solution) deposits the pheromone. This has the weakness of some edges becoming very strong and becoming a part of almost all solutions, while others becoming very weak and being chosen very rarely.

Min-max ant system (MMAS) [12] is an improvement of the EAS that tries to solve this problem. The improvement is done by adding an extra constraint that all pheromone values are bounded, $\tau_i \in [\tau_{\min}, \tau_{\max}]$. There are two effects of this. First, the pheromone trial will not become too strong on good edges and second, the other potential parts of the solution will not be totally excluded from the search due to very weak values of the pheromone. This approach improves the results of the EAS, but the stagnation still happens. Since $\tau_{min}$ has to be set to a low value, once the trail of a vertex reaches $\tau_{min}$ it becomes chosen very rarely. The search is never intensified near that point unless it becomes a part of the global best solution. The basic solution for this problem is to add some criteria for search stagnation and if stagnation has occurred to reset the pheromone trail to initial values [14].

Minimum pheromone threshold strategy (MPTS) is another approach to solving the early stagnation problem whose effectiveness has been shown on the quadratic assignment problems [11]. The idea behind MPTS it to use minimum threshold value $\tau_{mt}$ that is bounded $\tau_{min} < \tau_{mt} < \tau_{max}$. In the beginning of the algorithm it is set to some initial value and then adjusted during the search, depending on the performance. Threshold $\tau_{mt}$ is used for updating the pheromone trail. When the search is performed, values in the pheromone trail $\tau_i$ are compared to the $\tau_{mt}$ and if $\tau_i$ is lower than $\tau_{mt}$, than $\tau_i$ is changed to $\tau_{max}$. Thus the MPTS avoids reinitialization of the pheromone trail and explores the solution search space more systematically. No loss of information occurs related to the pheromone trail reset, while the good properties of the MMAS are preserved.

The idea of our hybridization is to use the information about the best-found solution to perform corrections on the pheromone trail. We introduce the concept of suspicion that some edges of the best found tour are not good. We try to direct the search to new areas of solution space that are less suspicious which means with less undesirable properties. Directing the search is done by greatly decreasing the pheromone trail values at suspicious edges.

Before we explain in detail this concept we wish to point out important differences between the MPTS for the basic MMAS and our hybridization. In the MMAS the increasing of diversification is done non-selectively to the whole search space, first by adding $\tau_{min}$ to prevent total exclusion of some vertexes from the search and, in the case of stagnation, by resetting the pheromone trail to initial values. When MPTS is added to the MMAS the reinitialization of the trail is avoided by testing at each iteration the pheromone values for all vertexes and if they have dropped below $\tau_{mt}$ they are set to $\tau_{max}$. This way some edges are added to the small group of edges that are frequently chosen by ants. A big drawback of this approach is that added edges are chosen just for having low values of $\tau_i$ which is a relatively random process. As a result, edges that do not belong to good solutions are often reintroduced in the search. Once the pheromone value for an edge is increased, it will take a long time for it to be removed from the intensively tested group. In our hybridization however, we do not add edges to the "popular" group but rather remove edges with suspicious properties, making the group smaller. In the following iterations ants will first select edges from the popular set, and when none are left, edges with better properties. This way we direct to new search areas that are less suspicious which means with less undesirable properties.

# 5   Suspicious elements exclusion pheromone correction strategy

Here we describe our new type of hybridization for the ACO and implement it for the TSP. We improve the ACO with a strategy for leaving local optima i.e. avoiding stagnation in search for the best solution. This method is based on correcting the pheromone trail used in the ACO. We calculate this correction based on the properties of the best-found solution so far. The basic idea of this correction is to lower the possibility of edges with high level of undesirability to belong to the optimal solution. We do not claim that our method gives the best results on TSP compared to all other developed algorithms, but we show that our strategy improves results acquired by the ACO and that it is simple to add the method to the existing algorithms.

## 5.1   Application to the TSP

When the ACO algorithm gets trapped in local optima for TSP, in many cases it was obvious from visual observation which corrections should be made, or more precisely what should not appear in the shortest

path (Fig.2). There where two simple criteria used on the edges belonging to the best found tour: very long edges and intersecting edges are very unlikely to be a part of the optimal path. The next step was to find a way to, without of major corrections to the ACO algorithm, remove them from the ants search path. The solution was significantly lowering the amount of pheromone on randomly selected highly suspicious edges belonging to the best path and letting the colony resume its search.
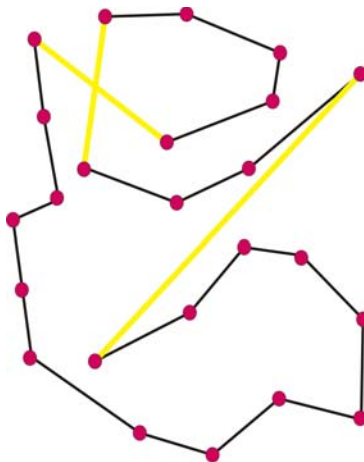


Figure 2: : Example of local optima found with ACO for TSP with suspicious edges on path

We have divided the correction into two parts: one considering the edge lengths, and the second that is connected to intersecting ones. All intersecting edges would be considered as highly suspicious and the pheromone trail would be corrected on them. Edges for which pheromone trail correction will be applied due to their length is defined in the following way. First we define a heuristic for suspicion Eq. 11

$$Sus(rs) = length(rs) \tag{11}$$

The next step is defining the probabilities of edges being selected for pheromone correction

$$p_{rs}(selected) = \frac{RK - RankSus(rs)}{RK} \tag{12}$$

In Eq. (12) instead of using the value of $Sus$ for edges, we used $RankSus$ which represents their rank by suspicion. $RK$ is the maximal number of edges that are considered for correction. The final step is to lower the pheromone trail for the selected edges:

$$\begin{aligned} \forall rs \in Selected \\ \tau_{rs} = \delta\tau_{rs} \end{aligned} \tag{13}$$

The use of suspicion defined in Eq. (11) is not fully effective because the same group of edges would be repetitively selected until a better tour was found. Because of this we introduce an improved suspicion criteria:

$$CorSus(rs) = Sus(rs) * ExSusepect(rs) \tag{14}$$

The improvement consists of tracking which edges have already been selected and preferring the selection of new edges. To do this, a new array $ExSuspect$ is introduced with elements initially set to 1. If edge $rs$ is selected, the following correction is done:

$$\begin{aligned} 0 < \lambda < 1 \\ ExSuspect(rs) = ExSuspect(rs) * \lambda \end{aligned} \tag{15}$$

If a new best set is found, the values of *ExSuspect* are reset to 1. A more complex suspicion criteria could have been used, that would better analyze the properties of the best found solution but we wished to show that even with a relatively simple heuristic improvements can be archived.

The last step that is needed to completely describe this hybridization is to define some stagnation criteria. We used the following: stagnation occurred if there was no change to the global best solution in at least $n$ iterations. If these criteria are satisfied, we apply previously defined correction algorithm to the pheromone trail. The pseudo-code for an iteration step for the improved ACO is:

```
Reset Solution for all Ants

 while  (! AllAntsFinished)
   For All Ants
     If(AntNotFinished)
       begin
         add new edge AB to solution
         based on probability
         local update rule for A
       End
     End for
   End while



 Compute Global Update

 If(Stagnation)and(UsingSuspisionImprovemnet)
   Use SuspisionCorrectionMethod
```

Experimental results show that our pheromone correction strategy for removing undesirable elements improves ACO algorithm for the TSP without any increase in computational complexity.

## 5.2   Other applications

Methods similar to one described in the previous section for the TSP were developed for three other graph problems: the minimum weight vertex cover problem, the minimum weight dominant set and the minimum connected dominant set problem. They are very similar and all three proved to be very efficient. For some problems we obtained significantly better results [15] than previous algorithms. The hybridization, although simple, has to be implemented as a module to the software system. That was easy since we developed a software framework [16] for the ACO that allows exactly that: easy incorporation of new module. A framework is a special kind of software library, that is similar to an application program interface (API) in the class of packages, that makes possible faster development of applications. However, while an API consists of a set of functions that user calls, a framework consists of a hierarchy of abstract classes. The user only defines suitable derived classes that implement the virtual functions of the abstract classes. Frameworks are characterized by using the inverse control mechanism for the communication with the user code: the functions of the framework call the user-defined functions and not the other way round. The framework thus provides full control structures for the invariant part of the algorithms and the user only supplies the problem-specific details. Such environment was suitable to implement our new pheromone correction strategy as a module that can be called from the framework.

## 6   Conclusion

We developed a pheromone correction strategy that is simple and universally applicable. It helps ACO algorithm escape being trapped in local minima, but in a more efficient way than other known methods.

While other used methods add new (usually random) elements to the currently investigated part of the solution space, our hybridization removes undesirable elements from that space, allowing more desirable elements to be included. Tests show that, without increasing computational complexity of the ACO algorithm, better results are achieved. Future development will include implementation in the software framework of our hybridization for other graph problems.

# References

[1] Dorigo M, Maniezzo V: Ant Colony system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics - Part B*, Vol. 26, No.1, pp. 29-41, 1996.

[2] M. Dorigo and L.M. Gambardella: Ant Colony System : A Cooperative Learning Approach to the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation*, vol 1, no 1, pp 53-66, 1997.

[3] Y.J Feng and Z.R. Feng: Ant colony system hybridization with simulated annealing for flow-shop scheduling problems. *WSEAS Transaction on Business and Economics*, Vol. 1, No. 1, pp. 133-138, 2004.

[4] Qi Chengming: An ant colony algorithm with stochastic local search for the vrp. *IEEE Computer Society, 3rd International Conference on Innovative Computing Information and Control*, pp. 464-468, 2008.

[5] J. Levine, F. Ducatelle: Ant colony optimization and local search for bin packing and cutting stock problems *Journal of the Operational Research Society*, pp. 705-716, 2004.

[6] Z.-J. Lee, S.-F. Su, C.-C. Chuang, K.-H. Liu: Genetic algorithm with ant colony optimization (ga-aco) for multiple sequence alignment *Applied Soft Computing* pp. 55-78 2008.

[7] Hong-hao Zuo, Fan-lun Xiong: Time Ant Colony Algorithm with Genetic Algorithms. *IEEE International Conference on Information Acquisition*, pp. 1057-1061, 2006.

[8] Frank Neumann, Dirk Sudholt, and Carsten Witt: Rigorous analyses for the combination of ant colony optimization and local search. In *Ant Colony Optimization and Swarm Intelligence, LNCS, Springer-Verlag* Vol. 5217, pp. 132-143, Berlin, Heidelberg, 2008.

[9] D.Asmar ,A. Elshamli, S. Areibi: A Comparative Assessment of ACO Algorithms Within a TSP Environment. *In DCDIS: 4th International Conference on Engineering Applications and Computational Algorithms*, pp. 462-467, 2005.

[10] Thomas Stützle and Marco Dorigo. Aco algorithms for the traveling salesman problem. In *Evolutionary Algorithms in Engineering and Computer Science: Recent Advances in Genetic Algorithms, Evolution Strategies, Evolutionary Programming, Genetic Programming and Industrial Applications, K Miettinen, P Niettaanmaki, M M Makela and J Periaux, editors*, pp. 500, Willey, 1999.

[11] Kuan Y. Wong and Phen C. See: A new minimum pheromone threshold strategy (mpts) for maxmin ant system. *Applied Soft Computing*, Vol. 9(3), pp. 882-888, 2009.

[12] T. Stützle and H.H. Hoos: MAX MIN Ant System. *Future Generation Computer Systems*, Vol. 16, pp. 889-914, 2000.

[13] B. Bullnheimer, R. F. Hartl, and C Strauss: A new rank-based version of the ant system: a computational study. *Central European Journal for Operations Research and Economics*, Vol. 7 No. 1, pp. 25-38, 1999.

[14] T. Stützle, H. Hoos: Improvements on the ant system: Introducing the max min ant system. In *Third International Conference on Artificial Neural Networks and Genetic Algorithms, Springer Verlag, University of East Anglia, Norwich, UK*, pp. 245249, 1998.

[15] Jovanovic R, Tuba M. An ant colony optimization algorithm with improved pheromone correction strategy for the minimum weight vertex cover problem. *Applied Soft Computing*, Vol. 11, No. 8, pp. 5360-5366, 2011.

[16] R. Jovanovic, M. Tuba, D. Simian: An object-oriented framework with corresponding graphical user interface for developing ant colony optimization based algorithms. *Wseas Transactions on Computers,* Vol. 7, No. 12, pp. 1948-1957, 2008.

Milan Tuba
Megatrend University Belgrade
Faculty of Computer Science
Bulevar umetnosti 29, N. Belgrade
SERBIA
E-mail: *tuba@ieee.org*

# "Least Significant Bit" method in steganography

**Gabriel TUDORICĂ, Paul STÂNEA, Daniel HUNYADI**

## Abstract

The purpose of this paper is to present the benefits of steganography and emphasize popular ways it can be applied to. We will focus on a digital steganography technique, using Bitmap files as carrier files for our hidden messages, thus hiding it in plain sight. Even though the picture can be seen by others, only the sender and the intended recipient will actually be able to get the messages.

Finally, we built an application using C# capable of applying this steganography technique, and allowing the user to embed hidden messages in 24 bit Bitmap files. Additionally, we implemented a chat-like environment using 24 bit Bitmap files to send the hidden data.

## 1. Introduction

We built our steganography application on the .NET Framework, using C# and it was designed to allow any user to embed hidden messages into 24 bit Bitmap files.

Using a substitution method called Least Significant Bit, this feature was made possible. This substitution method has a drawback though; namely the amount of text that can be hidden is limited to the size of the bitmap.

Basically, our application loads any 24 bit bitmap and using the Least Significant Bit method, it calculates the maximum number of bits (in characters) one can embed into the file. The substitution method replaces the last bit of every byte in the bitmap file with the bits from our binary transformed message. Clearly, our message must not exceed the maximum number of bits available for this process. Thus, we built a limiter which provides graphical feedback to the user.

After the message has been entered, for security reasons, the user must enter a password to protect the file for the decoding process.

After the password has been set, the application saves the new bitmap file with the embedded message and password and the new file can be sent to the intended recipient. The differences between the original file and the one with the embedded message are so small, that no one will notice unless they can compare it to the original.

In most cases, depending on the bitmap size and hidden message length, there were no visual and/or size differences between the two.

For the decoding part, the recipient will load the file into the application, fill in the password field, and the hidden message will be revealed.

Since there are other applications capable of doing the same thing, we thought of a way to significantly improve ours. While most steganography applications use one technique to embed messages, we implemented another one, namely the generation technique.

Two out of three techniques have some serious drawbacks!

More precisely, the insertion method adds harmless bits to an executable file for example, increasing the carrier file size depending on the embedded message length, thus arousing suspicion.

The big drawback of the substitution method is that the maximum available size for our hidden message is determined by the size of our bitmap file, which can cause problems.

Since the two above mentioned methods are not 100% trustworthy, we used a third one, the generation method, which generates a large enough image to carry the entered message. This method requires the message to be entered first, then it calculates the size of the text in bits and finally generates a large enough 24 bit bitmap file capable of embedding the message. The resulting bitmap file size will be directly dependent to the message length and its width and height are chosen so that the resulting file is large enough to withhold the message and not too much larger then it is needed. Since this is directly linked to the message length, we won't have 2 images of the same width or height.

Furthermore, we added another great feature to our application using the generation method. We created a chat-like environment where two users can chat using secure hidden messages found in 24 bit bitmap files using a FTP server. Once the messages are created, a picture carrying the message will be generated and uploaded to a FTP server, where the other instance of the application scans the server for new images.

When found, it downloads it locally, loads it, decodes it and displays the message. After the message is displayed, both the remote and local files are deleted. We did not protect the images with a password because we wanted to speed up the whole process and make it as close as a chat-like application as we could. And since the carrier files will stay for just about a few seconds on the server, the file does not present a security risk.

The main goal we managed to achieve here is that the whole conversation, viewed from outside, is camouflaged into a trivial file upload on a FTP Server, which is great!

# 2. Steganography

Steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and the intended recipient, suspects the existence of the message. It's a form of security through obscurity.

The main advantage of using steganography over cryptography alone, is that the messages do not attract attention on themselves, furthermore the messages are double protected. [1][2][3]

Diagram (1) provides a very generic description of the pieces of the steganographic process:



(1)
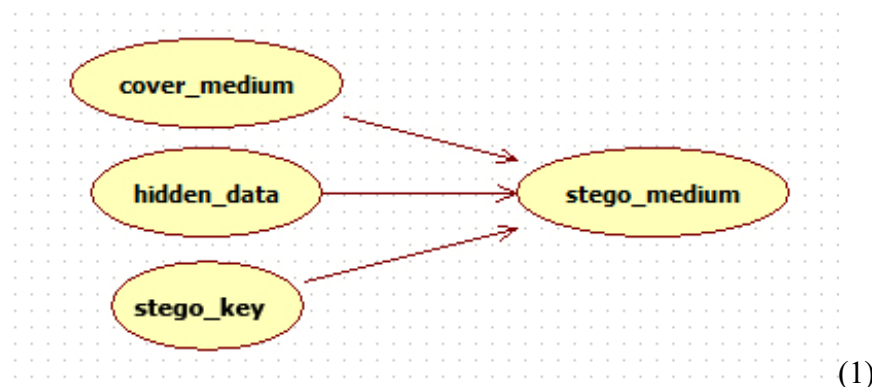
Cover_medium is the file in which we will hide the hidden_data (the carrier file), which may also be encrypted using the stego_key. The resultant file is the stego_medium, which will be the same file type as the cover_medium and it will be perfectly usable. The cover_medium are typically image or sound files. Our application will focus on image files.[2][3]

History confirms what we already know: the best place to hide messages is in plain sight. An example of ancient steganography use is that of Histiaeus, who shaved the head of his most trusted slave and tattooed a message on it. After his hair had grown, the message was hidden. His purpose was to revolt against the Persians. [1][2]

Another useful purpose of steganography is the so-called *digital watermarking*. A watermark, historically, is the replication of an image, logo, or text on paper stock so that the source of the document can be at least partially authenticated. A digital watermark can accomplish the same function.

A graphic artist, for example, might post on a website sample images with an embedded signature (known only by him), so that he can later prove his ownership in case others attempt to portray his work as their own. [3]

## 2.1 Techniques

### 2.2 Physical steganography

Steganography has been and it is used today in many forms. The ways of hiding data nowadays are endless, but the most popular include:[4][5]

- Messages tattooed/written on messenger's body;
- Messages written in secret inks;
- Messages written below the postage stamps on envelopes.

### 2.2.1 Digital steganography

Steganography evolved in 1985 with the advent of the personal computer. Since then, there have been built over 800 steganography applications recognized by official institutions around the world. Some common digital steganography methods are: [4][5]

- Hiding messages in the lowest bits of noisy images or sound files;
- Hiding data in encrypted data or within random data;
- Adding harmless bits to executable files;
- Embedding photos in video material;

### 2.2.2 Network steganography

Network steganography uses communication protocols' control elements and their basic functionality, making the whole process even harder to detect or remove. An example of network steganography method is **Steganophony** - the concealment of messages in Voice-over-IP conversations, e.g. intentionally sending corrupted packets that the receiver would ignore by default; [4][5]

### 2.2.3 Printed steganography

Nowadays much of the steganography employed today is quite high-tech, but steganography's goal is to mask the existence of a message. A message can be concealed by traditional means and produce a ciphertext.

A popular and almost obvious method is called a null cipher. In this type of steganography, one would decode the hidden message by taking the first or other fixed letter from each word and create new words. Other forms include deliberately making mistakes, using different fonts or other hard to notice symbols. [4][5]

Consider this cablegram that might have been sent by a journalist/spy from the U.S. to Europe during World War I:

*PRESIDENT'S EMBARGO RULING SHOULD HAVE IMMEDIATE NOTICE. GRAVE SITUATION AFFECTING INTERNATIONAL LAW. STATEMENT FORESHADOWS RUIN OF MANY NEUTRALS. YELLOW JOURNALS UNIFYING NATIONAL EXCITEMENT IMMENSELY.*

The first letters of each word form the character string: *PERSHINGSAILSFROMNYJUNEI*. A little imagination and some spaces yields the real message: *PERSHING SAILS FROM NY JUNE I.*

# 3. Algorithms and Techniques

There are three different techniques you can use to hide information in a cover file:
   a) Injection (or insertion);
   b) Substitution
   c) Generation

## 3.1 Injection or insertion

Using this technique, the data is stored in section of a file that is ignored by the application that processes it. For example in unused header bits or adding harmless bytes to a file leaving it perfectly usable.

The more data you add, the larger the file gets, and this can raise suspicion. This is the main drawback of this method.[1][4]

## 3.2 Substitution

Using this approach, the least significant bits of information are replaced with desired data, which will be reproduced on decoding.

The main advantage of this method is that the file size does not change, but the file can be affected by quality degradation, which in some cases may corrupt files. Another flaw is that the available amount of data is limited by the number of insignificant bits in the cover file. [1][4]

## 3.3 Generation

Unlike injection and substitution, this technique doesn't require an existing file, it generates it just to embed the message, which is better than the other two mentioned methods because it's not being attached to another file to suspiciously increase the file size, it has no limitation and the result is an original file and therefore immune to comparison tests. [1][4]

# 4. The Least Significant Bit (LSB) method

The Least Significant Bit (LSB) method is the most common substitution technique, which basically replaces the least significant bytes in a file to conceal data, which is extracted at the decoding process. It's especially effective on lossless image files, such as 24 bit Bitmap files.

The method takes the binary representation of any form of data and overwrites the last bit of every byte in an image.

As an example, we will take a 3x1 pixel 24 bit bitmap, each pixel color is represented on one byte so our picture bitmap will have a total of 9 bytes. We have the following RGB encoding:

*11010101   10001101   01001001*

*11010110   10001111   01001010*

*11011111   10010000   01001011*


Now suppose we want hide the following 9 bits of data: 101101101. If we overlay these 9 bits over the original file using the LSB method, we end up with the following new RGB encoding, which visually is not too different. The bits in **bold** have been changed.


*11010101   1000110**0**   01001001*

*1101011**1**   1000111**0**   0100101**1***

*11011111   10010000   01001011*


We managed to hide 9 bits of information by only changing 4 bits or about 50% of the lest significant bits.[2]

Similar methods can be applied to lower color depth image files, but the changes would be too dramatic and therefore obvious. On the other hand, grayscale images provide an excellent cover file for the LSB method.

 It is worth mentioning that *steganalysis* is the art of detecting and breaking steganography. A form o analysis is to carefully examine the color palette of an image, which very often has a unique binary encoding for each individual color. If steganography is used, the same color will end up having more binary encodings, which would suggest the use of steganography in the specific file.

But if we wanted to check for hidden information in images, which files should we analyze? Suppose I decide to send a message hidden in a picture that I will post on a website, where others can see it, but only certain people know that it contains hidden information. I can also post more images and only some of them would have hidden data so the potential steganalyst would get confused. [2][5]

The quantity of potential cover files make steganalysis a Herculean task, and we will exploit this very spot of steganalysis in our application.

# 5. MessageHider Application

MessageHider is a steganography application we developed using .NET Framework and C#. Compared with other software products that use only one steganography technique, our application uses two: substitution and generation. Our application has two modes: normal and chat mode. In the following, we will present the application in detail.

The Graphic Interface is basic, and allows the user to enter the two modes: normal and chat mode.
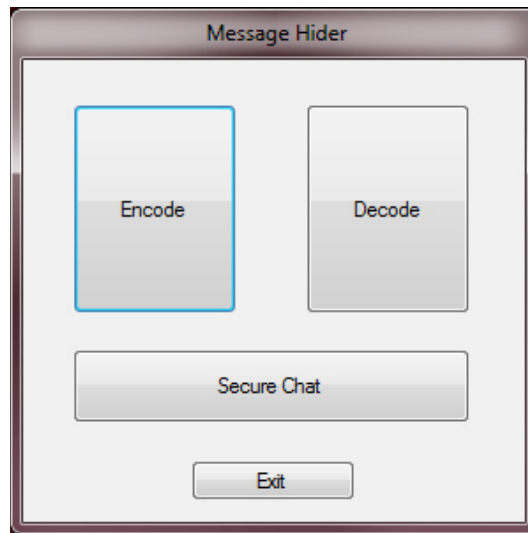
Main screen (2)

## 5.1 Normal mode

The normal mode, provides the basic functionality, using a substitution algorithm, more precisely the Least Significant Bit method. We will use 24 bit bitmaps as stego files, because of their lossless compression. Normal mode provides features for encoding and decoding hidden messages.

### 5.1.1   Encoding

To encode a hidden message, the user needs to first select the source file (24 bit bitmap). Once the bitmap is loaded, our application will estimate the maximum number of characters the user can hide in the selected file. As we have seen, the Least Significant Bit method, replaces the last bit of every byte in the file, with the bits for our hidden message. To get the maximum number of characters a user can hide we will use the following formula (ImageWith, ImageHeight are expressed in pixels):

$$AvailableCharacters \approx \text{CEILING}\left(\frac{1}{16}\left(\frac{ImageWidth \cdot BitsPerPixel}{8} ImageHeight + BmpHeader\right)\right)$$

(3)

In other words, for a 1024x768 bmp image we will have 147459 characters.

$$AvailableCharacters \approx \text{CEILING} \left( \frac{1}{16} \left( \frac{50 \cdot 24}{8} \, 50 + 54 \right) \right) \approx 473 \text{ characters}$$

(4)

When a bitmap is being loaded, the user will have a visual feedback for the maximum characters available as the message is being typed and will also limit the maximum number of characters he enters to the maximum available characters calculated before minus 32. The 32 characters we subtracted are reserved for the password, which can be by up to 30 characters, the two characters left being used as a delimiter from the original message on which the password will be concatenated.

After the message was entered and the password set, the Least Significant method will be called and the stego file will be generated.
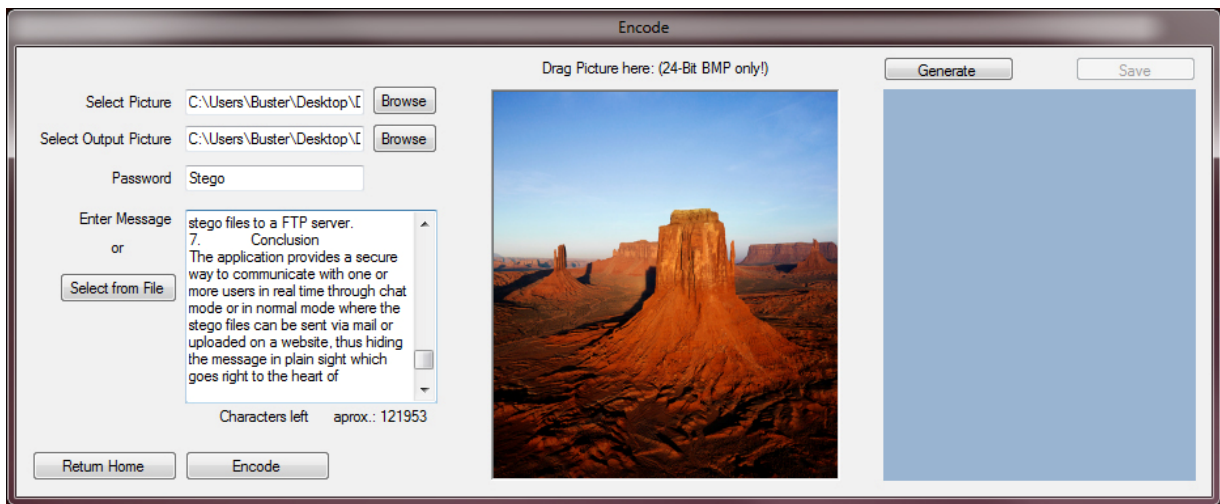


Encoding 1024x768 bmp (5)

The form the user will have to fill in has the following fields:
- Select picture – the path to the carrier image in which we will hide our message. The image will be  a 24 bit bitmap;
- Select output picture – the path to the output picture;
- Password – the password required for the decoding of the message;
- Message – the message we want to hide;

We also provided a drag and drop box so that the user can drag the bitmap in the box and the form will autocomplete its path. We can also load text from a file to speed up the process.

We actually managed to hide our entire article (without the images) in a single 1024x768 standard Windows 7 image which we converted to BMP format. Our article used only 18% of the available characters we could hide in the picture, having another 121,953 characters left.

We would also like to point out the fact that the size of the image did not change, both the orginal and the stego file had 2,359,352 bytes. Also, from a visual perspective, both files are identical.
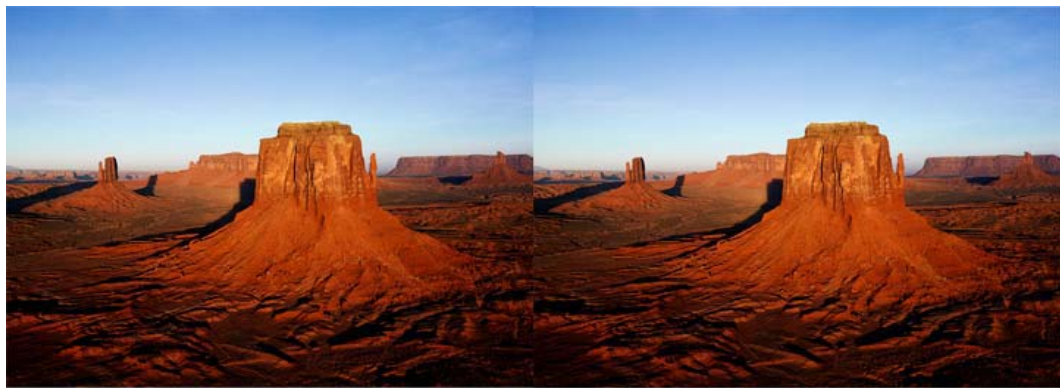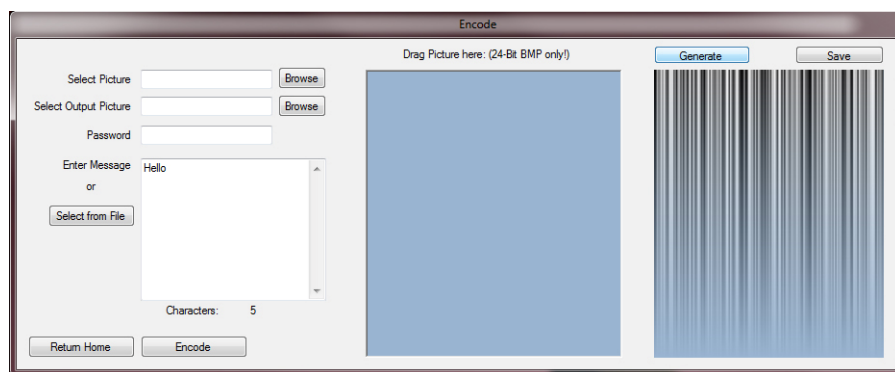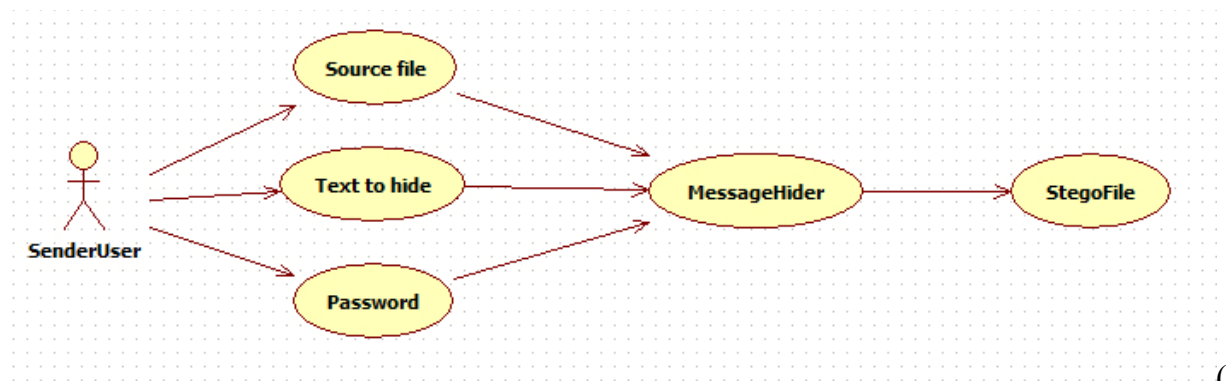
Image comparison (6)

Last, but not least we added a bitmap generation module so that the user could generate a large enough bitmap to withhold the entered message. The resulting bitmap will be black and white with lots of noise.



Encoding with generated b&w noisy bitmap (7)

The whole process can be visualized using diagram (8):



(8)

215

We can conclude that diagram (8), is the extended version of the diagram (1) we presented earlier.

### 5.1.2 Decoding

Firstly, the user will load the stego file into the application. Once this is done, the Least Significant Bit method will extract the last bit from every byte of the stego file and thus recreate the original binary message. After a conversion from byte to string, we get the original message. The message also has the password concatenated and delimited by the || characters.

The password is extracted from the string and stored for it to be checked in the next step, where the user is required to fill in the password field. If the password is correct, the original message sent will be revealed to the user.
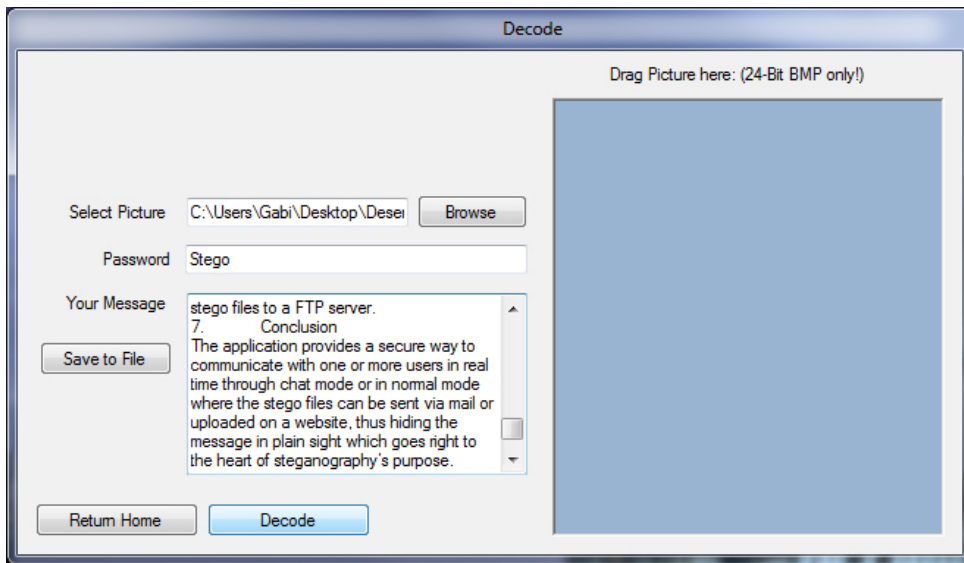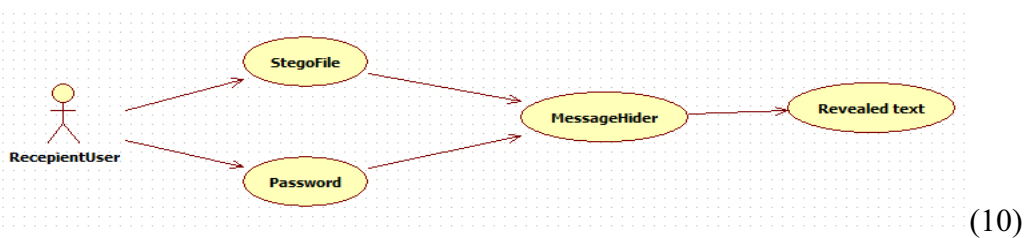


Decoding (9)

The decoding form has the following fields:
- Select picture – the path to the bitmap picture we want to extract the message from, which, as before, it can be obtained if the user drags and drops the file in the drop box;
- Password – the password we protected our message with;
- Your message – the area where the original message will be revealed if the password is correct.

The whole process can be visualized using diagram (10):



(10)

As diagram (10) shows, the decoding process is the inverse of the encoding process.

## 5.2 Chat mode

Chat mode uses the generation steganography technique. Basically, the user first types the message he wishes to hide and a large enough bitmap capable of embedding the message will be generated as a stegofile. The message will be protected by a generic password. Once the message was embedded, the stegofile will be named using a convention based on the nickname the user chose (which he previously filled in) and a timestamp. Then, the stegofile will be uploaded to an FTP server, using the credentials filled in by the user from the form, and the message he wrote will be displayed in the textfield.

The other user will connect on the same FTP server and the application will scan for the existing files that do not start with the current user's nickname based on the convention we talked about earlier. If found, the bitmaps will be downloaded from the FTP server, decoded using the same generic password and the message will be displayed in the textfield along with the other user's nickname. The local bitmaps and the ones on the FTP server will be erased as soon as the the bitamp is decoded.
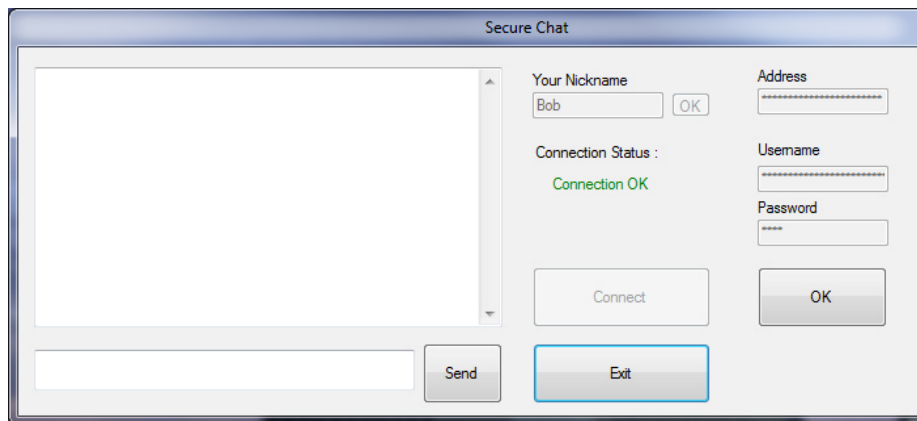


Secure chat mode (11)

The main advantage is that the whole chat can be seen as a trivial uploading and deleting procedures.

Once in chat mode, the user will have to first assign a nickname which will be used to differentiate the message sender. Once the user fills in the nickname field, he will fill in the credentials in order to connect to the FTP Server and lastly press the Connect button. A visual feedback will notify the user about the connection status.

Once connected to the FTP Server, the user can begin to exchange messages with the other user. Using the message textbox, the user can type in the message he wishes to send. When he presses the send button, a couple of processes are initiating.

Firstly, we need to generate a picture large enough for our message to fit in. The width of the image will be generated by a random number between 100-300 pixels, all we have to calculate now is the height. Based on (3), we conclude that the ImageHeight is:

$$\text{ImageHeight} \approx \text{CEILING}\left( 128 \, \frac{\text{MessageLength}}{\text{ImageWidth} \cdot \text{BitsPerPixel}} \right)$$

(12)

Now that we have the ImageWidth and ImageHeight, we can generate the carrier image. We just added noise to the image in order not to leave it blank, but random graphics can be generated or even fractals. Actually we have this in mind for future developments.

Once the image is generated, we apply the Least Significant Bit method, previously described in the normal mode – encoding part. After the message is embedded, we upload the bitmap file to the FTP Server and rename it as [nickname--unixTimestamp.bmp].

By using this naming convention we can easily determine the message sender, and sort the files/messages in chronological order because of the unixTimestamp. Once the image has been uploaded, it is deleted from the local path.



Bob--1316905935.bmp – zoomed in 291x1 pixel bitmap generated for the Hi, Alice! message sent by Bob (13)

The message we embedded in the file will also be displayed in the chat window, having the user's nickname as a prefix.
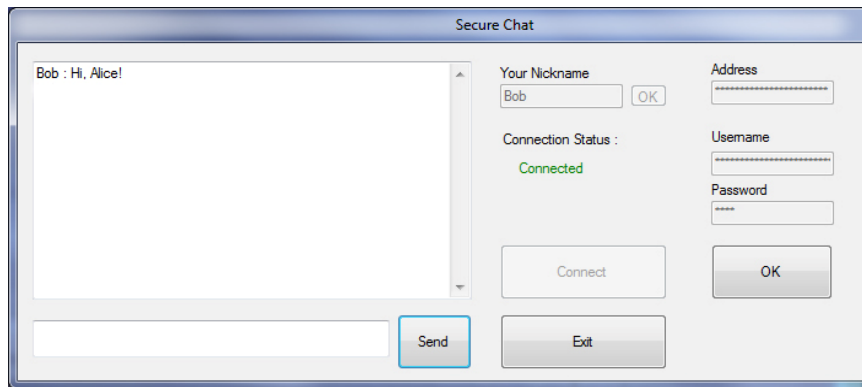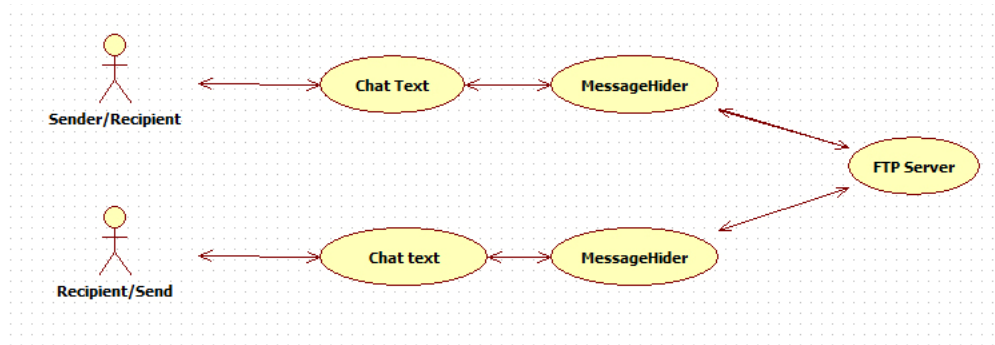


Secure chat mode – Bob (14)

Once the file is decoded, both the remote and local files are deleted. When the user closes the chat mode, the FTP connection will close as well.
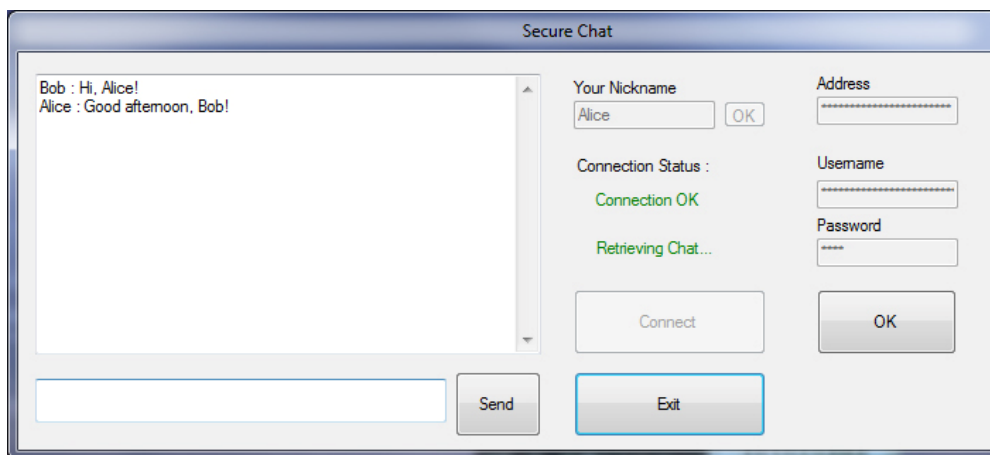


Secure chat mode – Alice (15)

Visually the whole process can be visualized using diagram (16):



(16)

As diagram (16) shows, our application in chat mode is similar to a native chat application, only instead of using sockets and a chat server, our application uploads stego files to a FTP server.

# 6. Conclusion

The application provides a secure way to communicate with one or more users in real time through chat mode or in normal mode where the stego files can be sent via mail or uploaded on a website, thus hiding the message in plain sight which goes right to the heart of steganography's purpose.

Of course, there is always room for improvements. For future implementations we thought of encrypting the message using strong cryptography algorithms and embed the ciphertext in the bmp files, thus creating a double protection. Another feature we thought of, was to implement methods capable of embedding the messages in other file types (other image types or even sound or video files).

# 7. References

[1] http://en.wikipedia.org/wiki/Steganography
[2] http://www.garykessler.net/library/steganography.html
[3] http://www.garykessler.net/library/fsc_stego.html
[4] http://www.infosyssec.com/infosyssec/Steganography/techniques.htm
[5] http://en.wikipedia.org/wiki/Least_significant_bit
[6] http://en.wikipedia.org/wiki/BMP_file_format
[7] http://www.fastgraph.com/help/bmp_header_format.html

Gabriel TUDORICĂ
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Sibiu, Dr. Ioan Ra   iu St. No. 5 - 7
ROMÂNIA
E-mail: office@eyenetworks.ro

Paul STÂNEA
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Sibiu, Dr. Ioan Ra   iu St. No. 5 - 7
ROMÂNIA
E-mail: psb77black@yahoo.com

Daniel HUNYADI
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Sibiu, Dr. Ioan Ra   iu St. No. 5 - 7
ROMÂNIA
E-mail: daniel.hunyadi@ulbsibiu.ro

# Estimation of the selectivity factor for a set of queries

## Letiţia Velcescu, Laurenţiu Vasile

### Abstract

In this paper, we study the selectivity factor, extending this concept to the case of a queries set $Q_1, ..., Q_n$. These queries are performed on the tables of a transactional database. So, they are supposed to be updated dynamically and, consequently, the selectivity factor associated to each query may vary in different moments. Because the selectivity factor has an important role in database optimization, it is necessary to be able to estimate it. We propose an algorithm for the estimation of the generalized selectivity factor, the concept we introduce, and also a hybrid estimator for it.

## 1    Introduction

Because the volume of information is growing continuously and fastly, the information processing and the quick access to it remain important issues in the information technology. In this paper, we present some factors which influence the execution of a query in a database management system (DBMS), such as the storage and access methods, and the selectivity factor of the queries.

The DBMSs include optimizers, which are software modules responsible with the adequate choice of a query's execution plan. Also, the DBMSs contain tools which allow the database administrator to tune the database such that it works appropriately to the specific of the applications. A more refined concept which appeared in this context is the one of database auto-administration (Aouiche et al 2003). This concept supposes the existence of some modules which build up the structures necessary to the optimum execution of an application.

Generally, the optimizer chooses the execution plan according to the query type and the available statistics generated by the system.

The optimization of the SQL code is also important because it has a significant impact over the resources and, consequently, on the database server's performances (Connolly, Begg, 2002).

The selectivity factor is presented in the second part of the paper. The third part is dedicated to the generalization of this concept, associated to one query so far, to the one associated to a query set. This part includes the estimation problems we deal with in this context.

## 2    The selectivity factor

A query's selectivity factor is extremely useful in the performance evaluation studies and in the record linkage problems. So, it is interesting that this factor could be estimated. In the recent research, the stochastic modelling was used in order to estimate this factor in the databases with uncertain information. The selectivity concept was used in the analysis of the performances, of the files organization, in the physical design of the databases (selection of indexes, partitions or attribute groups), as well as in the queries optimization. From this latter point of view, the definition of a statistic profile was taken in

consideration, including the information stored or estimated in the database in order to optimize it. Such a statistic profile may include the number of tuples in a relation, the number of distinct values in each domain, the average number of records in a data block etc. Based on these statistics, the optimizor should determine: the cost of the individual operations specific to the relational databases, the new statistic profiles of the relations derived from operations, the cost of a sequence of operations.

**Definition 1** (Delobel, Adiba, 1982) *Given a relation $R$, the selectivity factor of a condition $C$, denoted by $Sf(C)$, is the probability that a record in $R$ satisfies the condition $C$.*

# 3 The generalization of the selectivity factor for a queries set

In this part, we propose to extend the concept of selectivity factor associated to a query, to the notion of selectivity factor associated to a queries set $Q_1, ..., Q_l$. The framework in which we will place this concept supposes that the tables of the database are updated dynamically. So, at distinct moments of time, the selectivity factor associated to each query is different.

Suppose that the query $Q_i$ has the selectivity factor $p_i = \frac{\alpha_i}{k}$, where $k$ is the number (constant) of lines of the table, and $\alpha_i$ is the number of lines selected by the query $Q_i$. In this case, the selectivity factor of the queries set can be estimated by the variable:

$$\bar{p} = \frac{1}{l}\sum_i p_i = \frac{1}{l}\sum_i \frac{\alpha_i}{k} = \frac{1}{lk}\sum_i \alpha_i \tag{1}$$

The mean of this variable is $E(\bar{p}) = p$ and it represents the (theoretical) selectivity factor of the queries set, where $\bar{p}$ is an unbiased estimation for $p$. The dispersion of this variable is $\frac{p(1-p)}{l} \leqslant \frac{1}{4l}$, and the standard deviation is $\sigma \leqslant \frac{1}{2\sqrt{l}}$.

From the Chebyshev's inequality, we get:

$$P(|\bar{p} - p| < t\sigma) \geq 1 - \frac{1}{t^2} \tag{2}$$

We denote $1 - \frac{1}{t^2} = 1 - \delta$, and from here it results that $t_\delta = \frac{1}{\sqrt{\delta}}$.

We denote $t\sigma = \varepsilon \simeq 0.01$. For such a value ($\varepsilon$ small), we obtain a big value $1 - \delta$, so $\delta$ is small.

From the preceding notations, it implies that:

$$t_\delta \cdot \frac{1}{2\sqrt{l}} \leq 0.01 \tag{3}$$

Therefore, in order to be able to estimate the selectivity factor in the way described previously, the number $l$ of queries must satisfy the condition $l \geq \frac{t_\delta^2}{4 \cdot (0.01)^2}$.

Supposing that the tables are updated dynamically, we consider that the number of lines of the tables at the moments of different queries is $k_i$, uniformily distributed. We suppose that the minimum, respectively maximum values of this variable are known: $m \leq k_i \leq s$. Then, by means of the simulation methods of the uniform variable (Vaduva, 2004), we obtain an estimation of the generalized selectivity factor for a queries set through the algorithm AEGSF presented below (Algorithm 1).

On the basis of the values $p_1, ..., p_N$, generated at the step 6 of this algorithm, the selectivity factor of the queries set can be estimated by $\bar{p}$.

---

**Algorithm 1** AEGSF (Algorithm for the estimation of the generalized selectivity factor)

---

**Require:** The value of $t_\delta$.

**Ensure:** The value of $\bar{p}$.

1:   $N_0 \leftarrow \left[ \dfrac{t_\delta^2}{4 \cdot (0.01)^2} \right] + 1$

2:   Choose $N \geq N_0$

3:   **for** $i = 1$ TO $N$ **do**

4:      Generate a random value $U \in (0, 1)$

5:      Determine $k_i = m + [(s - m) \cdot U] + 1$

6:      Calculate $p_i = \dfrac{\alpha_i}{k_i}$

7:   **end for**

8:   **return**   $\bar{p} = \frac{1}{N} \sum\limits_i p_i$

---

## 3.1   Estimation of the selectivity factor

In many cases, in the frame of the statistic profiles used in the previous research, it was supposed that the tuples of the relations are identically distributed, relative to the values of an attribute, and the values of different attributes are independent. This supposition is not realistic, in many environments in which databases have an important role, because the values of the attributes can be imprecise and there can be dependencies between attributes. Because of this reason, the preoccupations concerning the estimation of this factor concerned parametric, non-parametric methods, and also the maximum entropy principle (Christodoulakis, 1989).

A subsequent classification of the methods of estimation of the selectivity factor used traditionally concerned the following estimator types (Ling, Sun, 1999):

- based on selection, which determine the selectivity factor only on the basis of the information at runtime, without using the information collected previously;

- parametric and table based, which use only the information collected previously, ignoring the on-line information.

The disadvantage of the first class of estimators consists in the insufficient use of the available information, whereas the second class leads to an imprecise estimation in an environment with frequent updates. There have been proposed hybrid estimators, which weight the two types enumerated above and whose results have been validated in practice.

## 3.2   Hybrid estimator for the generalized selectivity factor

We will present a hybrid estimator (Ling, Sun, 1999) for the selectivity factor of a query and we will extend the results for the case of the selectivity factor associated to a queries set, introduced before.

In the case of a single query, we consider $f$ the characteristic function of a selection predicate, and be $x_i$ a tuple. The function $f$ can be defined in the following way:

$$y_i = f(x_i) = \begin{cases} 1, \text{ if } x_i \text{ satisfies the selection predicate} \\ 0, \text{ otherwise} \end{cases} \tag{4}$$

In the approach based on selection, the main technique consists in choosing randomly, repeatedly, a tuple in the table on the basis of the query's predicate, followed by the realization of the inference about the real selectivity, using the estimated selectivity obtained from the sample data. Thus, one can realize the inference according to which an approximation of the real selectivity $p$ is:

$$\hat{p}_n = \frac{1}{n} \cdot \sum_{i=1}^{n} y_{r_i} = \frac{1}{n} \cdot \sum_{i=1}^{n} f(x_{r_i}) \tag{5}$$

222

where $n$ represents the sample size, $k$ is the total number of tuples and the index $r_i$ is a random integer, between 1 and $k$. Consequently, the average total number of tuples which satisfy the selection is $k \cdot \hat{p}_n$.

A hybrid estimator $\breve{p}_n$ of the selectivity factor is given by a linear combination between the estimated selectivity $\hat{p}_n$ and the estimated selectivity $\tilde{p}$, obtained by a parametric estimator or by a table based estimator:

$$\breve{p}_n = t \cdot \hat{p}_n + (1 - t) \cdot \tilde{p} \tag{6}$$

where $t$ is a parameter in the interval $[0, 1]$.

In order to validate an estimator, the mean-squared error ($mse$) is used to quantify the estimators performances:

$$mse = E(\bar{p} - p)^2 = \frac{1}{n} \cdot \sum_{i=1}^{n} (\bar{p}_i - p)^2 \tag{7}$$

where $\bar{p}_i$ is the individual selectivity estimated by an estimator, $p$ is the total, real selectivity, depending on the given query, and $n$ is the sample size.

The value $mse$ of an estimator represents the accuracy of its estimation, as well as its safety. The smaller the $mse$ value, the better the estimator is. For a method based on selection, the $mse$ value is $\frac{p \cdot (1 - p)}{n}$. For an estimator which uses a parametric or table based method, the $mse$ value is $(\tilde{p} - p)^2$, and $\tilde{p}$ remains unchanged for the given query, until the parametric or table based estimator is recomputed using the updated information.

The different values of the parameter $t$ represent different weights of the two estimators. In the extreme cases $t = 1$ or $t = 0$, the hybrid model reduces to a selection based estimator, respectively to a parametric or a table based one. The existence of an optimum value of the parameter $t$ and the calculation of this value have been determined in the following theorem.

**Theorem 1** (Ling and Sun, 1999) *The optimum value of $t$, denoted by $t_n^*$, is given by the formula:*

$$t_n^* = \frac{(\tilde{p} - p)^2}{p \cdot (1 - p)/n + (\tilde{p} - p)^2} \tag{8}$$

*The mse value for the hybrid estimator corresponding to the optimum parameter $t_n^*$ is smaller than each of the two estimators when $0 < p < 1$ and $p \neq \tilde{p}$, meaning that:*

$$E(\breve{p}_n^* - p)^2 < \min\{\frac{p \cdot (1 - p)}{n}, (\tilde{p} - p)^2\} \tag{9}$$

*where $\breve{p}_n^*$ is $\breve{p}_n$ for $t = t_n^*$.*

We can apply the hybrid estimators in the case proposed previously, of several queries, therefore of the generalized selectivity factor. We know that $p_1, ... p_q$ are the selectivity factors associated to the $q$ queries. Consider $f_i$ the characteristic functions of the predicates associated to each of the $q$ queries and be $x_{ij}$ a tuple in the query $i$. We consider a selection associated to each query.

The functions $f_i$ are given by the formula:

$$y_{ij} = f_i(x_{ij}) = \begin{cases} 1, \text{ if } x_{ij} \text{ satisfies the selection predicate} \\ \quad \text{ of the query } i \\ 0, \text{ otherwise} \end{cases} \tag{10}$$

Be $n_i$ the sizes of the samples associated to the $q$ queries. Then, an approximation of the selectivity of the query $i$ is given by:

$$\hat{p}_{n_i} = \frac{1}{n_i} \cdot \sum_{j=1}^{n_i} y_{ir_j} = \frac{1}{n_i} \cdot \sum_{j=1}^{n_i} f_i(x_{ir_j}) \tag{11}$$

where $r_j$ is a random integer, between 1 and $n_i$.

The approximate number of tuples of the result of the query $i$ will be $k_i \cdot \hat{p}_{n_i}$, where $k_i$ is the total number of tuples of the relation in the query $i$.

Every query has an associated hybrid estimator:

$$\breve{p}_{n_i} = t_i \cdot \hat{p}_{n_i} + (1 - t_i) \cdot \tilde{p}_i \tag{12}$$

For the queries set, we will have:

$$\bar{p} = \frac{1}{N} \cdot \sum_i \breve{p}_{n_i} \tag{13}$$

where $N$ was determined previously.

---

**Algorithm 2** AEGSFHE (Algorithm for the estimation of the generalized selectivity, using hybrid estimators)

---

**Require:** The value of $t_\delta$.
**Ensure:** The value of $\bar{p}$.

1: $N_0 \leftarrow \left[ \dfrac{t_\delta^2}{4 \cdot (0.01)^2} \right] + 1$
2: Choose $N \geq N_0$
3: **for** $i = 1$ TO $N$ **do**
4:     Generate a random value $U \in (0, 1)$
5:     Determine $n_i = m + [(n - m) \cdot U] + 1$
6:     Calculate $\breve{p}_{n_i} = t_i \cdot \hat{p}_{n_i} + (1 - t_i) \cdot \tilde{p}_i$
7: **end for**
8: **return** $\bar{p} = \frac{1}{N} \sum_i \breve{p}_{n_i}$

---

Thus, the algorithm AEGSF becomes AEGSFHE.

On the basis of the values $\breve{p}_{n_1}, ..., \breve{p}_{n_N}$ generated at step 6 of the algorithm, the selectivity factor of the queries set can be estimated by $\bar{p}$ which is computed in the last step of the algorithm.

# 4   Conclusions

This paper refered some problems related to the query optimization in database management systems. We mentioned some factors which interfere in this process, such as some physical parameters of the system on the access to data in the databases. We studied the queries' selectivity factor, which has an important role in database optimization. We proposed the generalization of this concept for a set of queries. The framework in which this concept becomes useful is the one in which the tables of the database update dynamically, so that the selectivity associated to a query may vary. We proposed an algorithm for estimation of the generalized selectivity factor, which uses simulation methods of the uniform univariate.

The methods for the estimation of the selectivity factor proposed so far concerned sample based estimators, parametric and table based estimators. Each of these classes present some disadvantages, reason for which the research in this field proposed the introduction of hybrid estimators, which weight these types and whose results have been validated in practice.

For the concept of generalized selectivity factor that we introduced, we also proposed a hybrid estimator formalized as an algorithm.

# References

[1] Aouiche, K., Darmont, J., Gruenwald, L., *Frequent itemsets mining for database auto-administration*, In Proceedings of the Seventh International Database Engineering and Applications Symposium (IDEAS03), 2003, p. 98-103.

[2] Christodoulakis, S., *On the estimation and use of selectivities in database performance evaluation*, Research Report CS 89-24, Department of Computer Science, University of Waterloo, Canada, 1989 (http://www.cs.uwaterloo.ca/research/tr/1989/CS-89-24.pdf).

[3] Connolly, T.M., Begg, C.E., *Database Systems: A Practical Approach to Design, Implementation and Management*, $3^{rd}$ edition, Addison-Wesley, 2002.

[4] Delobel, C., Adiba, M., *Bases de donnes et systmes relationnels*, Dunod, 1982.

[5] Feng, J., Qian, Q., Liao, Y., Li, G., Ta, N., *DMT: A Flexible and Versatile Selectivity Estimation Approach for Graph Query*, WAIM 2005, LNCS 3739, Springer, 2005, p. 663  669.

[6] Lim L., Wang M., Padmanabhan S., Vitter J., Parr R., *XPathLearner: An On-Line Self-Tuning Markov Histogram for XML Path Selectivity Estimation*, in Proceedings of the 28th VLDB Conference, 2002, p. 442-453.

[7] Ling, Y., Sun, W., *A Hybrid Estimator for Selectivity Estimation*, IEEE Transactions on Knowledge and Data Engineering, vol. 11, no. 2, IEEE Educational Activities Department, 1999, p. 338-354.

[8] Vaduva, I., *Modele de simulare*, Editura Universitatii din Bucuresti, 2004.

Letitia Velcescu
University of Bucharest
Faculty of Mathematics and Informatics
14 Academiei, 010014 Bucharest
ROMANIA
E-mail: *letitia@fmi.unibuc.ro*

Laurentiu Vasile
University of Bucharest
Faculty of Mathematics and Informatics
14 Academiei, 010014 Bucharest
ROMANIA
E-mail: *vsl@fmi.unibuc.ro*

# Some of the aspects of decision design in development of the intelligent wire casting machine

**Sergiu Zaporojan, Constantin Plotnic, Igor Calmicov**

**Abstract**

The process of microwire casting can be one of the methods of nanotechnology and advanced materials. The objective of this paper is to discuss the problem of decision design in the development of an intelligent machine for casting of microwire. The paper presents the decision making structure and describes its elements for microwire production based on direct casting from the melt. The most important element of the decision making structure is given by the drop model. According to this, important details of the model are discussed. The results presented here are intended to be used in the decision support system design for building of the intelligent casting machine.

# 1 Introduction

It is a fact, that innovative technologies are the engines of a competitive economics. Obviously, industrial technologies are very important for such economics. It is well-known, that most of modern technological processes are automated. But there are industrial applications where human presence is essential given the complexity of the technologies. For instance, in continuous casting of glass-coated microwires ([3]) it is not possible to dispense with manual intervention because of high complexity of the casting process. Such technologies require online recognition of the process and online decision making. In this context, it is important the development of advanced techniques in order to optimize industrial information systems with human in the loop. Such a system is concerned with all personnel, equipment, software, processes and knowledge to provide data for decision making structure. This structure should provide how the system will evolve, how far the best performance is, what actions should be undertaken.

The paper discusses the problem of decision design in the development of the intelligent wire casting machine. Some of the aspects of online analysis and decision making problem in continuous casting of glass-coated microwires are discussed in ([4]). Our purpose is to develop some of the preliminary results presented in ([4]). Next section describes the structure of decision making for the process of microwire production based on direct casting from the melt. The most important element of the decision making structure is given by the drop model. According to this, the last section develops the subject of the drop model.

## 2   Decision design problem in production of microwires

First of all, let us briefly describe the fabrication of glass-coated microwires. Glass-coated microwires are manufactured by means of a modified Taylor-Ulitovsky process based on direct casting from the melt ([3]). A rod of the alloy of desired composition is put into a glass tube and placed within a high frequency inductor heater. The alloy is heated up to its melting point, forming a droplet. While the metal melts, the portion of the glass tube adjacent to the melting metal softens, enveloping the metal droplet. A glass capillary is then drawn from the softened glass portion and wound on a rotating bobbin. At suitable drawing conditions, the molten metal fills the glass capillary and a microwire is thus formed where the metal core is completely coated by a glass shell. The casting process is carried out at a temperature that will melt the alloy and soften the glass tube. The final microwire structure is formed by water-cooling to obtain a metallic core in amorphous or non crystalline state. After passing through the cooling water, the microwire comes to spool of a receiving mechanism.

In the area of microwire fabrication a lot of human experience and knowledge has been accumulated. We consider that this fact should be used to optimize the production of microwires at the system level. The process of decision making represents one of the main problems in continuous casting of glass-coated microwires. In the casting of glass-coated microwires, the human operator plays the role of decision maker. The figure 1 illustrates the structure of decision making for the process of microwire production.
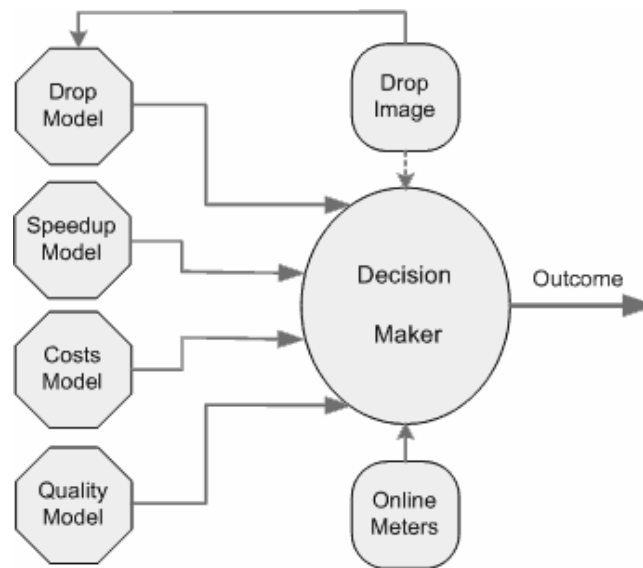


**Figure 1.** *Structure of decision making.*

As it can be seen from the figure 1, the structure of decision making consists of the following elements:

- Drop model.
- Speed up model.
- Costs model.
- Quality model.
- Drop image sensor.
- Online meters.
- Decision maker.

The process of decision making requires us to analyze a decision according to quality, costs, and speed up of wire production. The quality model must take into account the quality of the alloy and

glass tube, the quality of the online meters (sensors) and other measurement equipment. Besides, a coefficient of repeatability of the measurement results must be introduced into the quality model. The costs must include the alloy, glass tube, measurement equipment, and human costs. The speed up model reflects the speed up of the microwire production. It is referred to a set of values for the rod, glass tube, and spool speed.

It is evident that the quality of final product, the costs of production, and the speed up of production represent evaluative criteria in microwire production. On the other hand, all these factors are interconnected. As it was mention above, the human operator plays the role of decision maker. Therefore, the objective of the operator is to evaluate online the quality of wire (its parameters, such as the diameter) and follow a course of action to satisfy the above criteria.

It is necessary to say that a highly skilled operator can make good online decisions over the casting process only using the information captured by his eyes. The information is with respect to color and shape of the drop. Therefore, the machine vision techniques should be quite suitable for the process of wire casting. Machine vision ([2]) is successfully used today in industrial applications. This approach has become a vital component in the design of advanced systems because it provides a means of maintaining control of quality during manufacture. For this reason, the structure on the figure 2 contains a drop image sensor. The last one is connected as the input to the model of the drop, which represents one of the most important elements of decision making structure. Such a model must play a major role in the decision support system design for building of the intelligent casting machine for microwire production. Next section develops the subject of the drop model.

# 3  Drop model

The main ideas of the drop model presented here were developed and described in ([5]). Our purpose was to construct a drop model, which will be useful to explain and predict behavior within the process of casting. Farther, it will show in detail that the interpolation polynomial in the Lagrange form can serve the purpose.

More precisely, we have to find a function that describes the shape of the drop. It should be mentioned that the drop is quite symmetrical under normal working conditions. Moreover, it was established from the analysis of a lot of experimental data that the upper part of the drop can be ignored. Therefore, it is enough to consider the right/left shape of the drop due to its symmetry.

It was observed that during casting process the shape of the drop is changing. Besides, because of both the geometry of inductive heater and some other factors the shape of the drop may vary from one casting machine to other.

So, we are looking for a model that will take into account current features of the casting process and will approximate online the working shape of the drop. On the other hand, such a model should allow us to estimate the geometry of the metal-filled capillary and predict the diameter of microwire. In order to meet these requirements, we decided to extract a given set of data points from the shape at a time. Then, the current shape must be interpolated. In this way we can obtain the function, which approximates online the shape. This function may offer us the information about the geometry of capillary.

Let us consider the interpolation polynomial in the Lagrange form.

Given a fixed interval $I \in R$ and a set of ($p$+1) interpolation points $x_0 < x_1 < x_2 < ... < x_p$ on $I$, a function can be defined $f : I \rightarrow R$. To interpolate the function $f$, we define the values $y_i$ as $y_i = f(x_i)$, for $0 \leq i \leq p$.

The points $y_i$ are the values of interpolation. We must use the unique interpolation polynomial of degree $P \leq p$, which verifies $P(x_i)=y_i$, for $0 \leq i \leq p$ and $f(x) \cong P(x)$ for any $x \in I$.

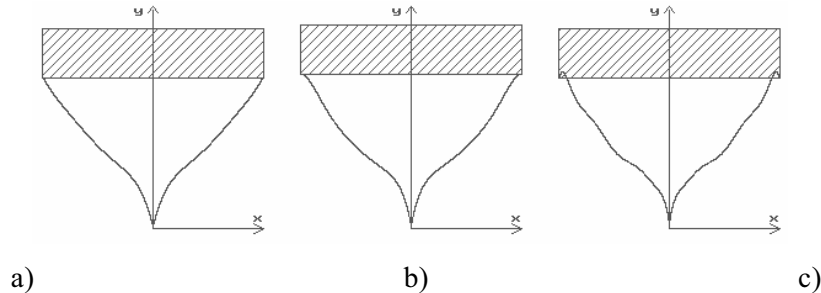Let $L_j$, $0 \leq j \leq p$, be the Lagrange basis polynomials

$$L_j(x) = \frac{(x-x_0)...(x-x_{j-1})(x-x_{j+1})...(x-x_p)}{(x_j-x_0)...(x_j-x_{j-1})(x_j-x_{j+1})...(x_j-x_p)}. \tag{1}$$

Then, the interpolation polynomial in the Lagrange $P(x)$, associated to the set of data points is

$$P(x) = \sum_{j=0}^{p} y_j L_j(x). \tag{2}$$

It should be noted that the Lagrange basis polynomials $L_j$ given by formula (1) depend only on the interpolation points selected, and do not require any auxiliary restrictions. This fact represents an advantage in using the interpolation polynomial in the Lagrange form (2) to approximate the shape of the drop.

To obtain the information about the optimum number of points to be used by the interpolation polynomial in the Lagrange, a comparative experimental analysis was carried out. During this experimental study we have established that when seven interpolation points are extracted from the shape a satisfactory approximation of the droplet shape is occurring. The figure 2 presents the experimental result of the interpolation when six (figure 2, a), seven (figure 2, b), and nine (figure 2, c) interpolation points are extracted online from the shape of the drop during casting.



a)                                    b)                                    c)

**Figure 2.** *Comparative result of the shape interpolation.*

Given a set of seven ($p = 6$) online data points $(x_0,y_0)$, $(x_1,y_1)$,...,$(x_6,y_6)$ where no two $x_j$ are the same, the Lagrange basis polynomials can be written. Then, the Lagrange basis polynomials can be easily calculated by means of the recursive functions. The figure 3 shows the result of the online approximation of the drop shape, when seven interpolation points are extracted.
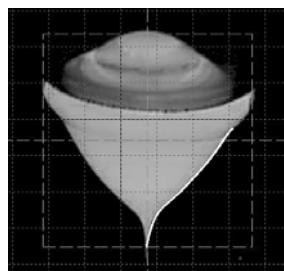


**Figure 3.** *Online approximation of the drop shape.*

As it can be seen from the figure 3, the drop shape follows a curve line. The curve line changes from being convex to concave. The interpolated online curve (see the white line on figure 3) does the same and follows the drop shape. The analysis of a lot of experimental data confirms the latter. We can conclude that the approximation based on the interpolation polynomial in the Lagrange form may be used for the construction and using the drop model in the decision making structure, which was described in the previous section.

Another important question now is how can we extract the information about the geometry of the metal-filled capillary? A possible way is illustrated in figure 4.

In our opinion, at first, it is necessary to determine the inflection point (IP) on the interpolated online curve. In order to calculate the location of the inflection point, it is necessary to find the second order derivative of the interpolation polynomial in the Lagrange form.

It can be shown that the second derivative takes the form:

$$P''(x) = \sum_{j=0}^{p} \left[ \frac{y_j}{\prod\limits_{i=0; i\neq j}^{p}(x_j - x_i)} \sum_{\substack{i_1=0 \\ i_1 \neq j}}^{p} \sum_{\substack{i_2=0 \\ i_2 \neq j \\ i_2 \neq i_1}}^{p} \prod_{\substack{i=0 \\ i\neq j \\ i\neq i_1 \\ i\neq i_2}}^{p}(x - x_i) \right]. \tag{3}$$

The second order derivative (3) equals zero in the inflection point. Hence, its location can be calculated online.
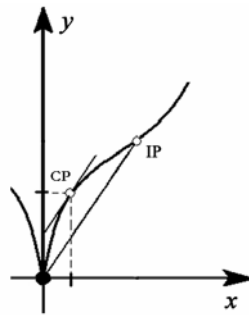


**Figure 4.** *Operating point of the metal-filled capillary.*

After that, another point may be determined. We call it the capillary or operating point (CP on figure 4). The latter should be very important because it seems to be the entry point of the metal-filled capillary. It is easy to observe that the point CP may be located by means of well-known Lagrange's formula:

$$f'(x_0) = \frac{f(b) - f(a)}{b - a} \ , \ x_0 \in (a, b). \tag{4}$$

In order to determine the location of the capillary point, the interval [0, *IP*] must be considered (see figure 4).

The above model must allow us to estimate online the geometry of capillary at each time during casting. To obtain online information about the drop, its image must be processed in real-time to track the geometrical and color features of the drop. For these reason, a software application was developed. To capture and process the image the application uses the DirectX libraries. This software application provide us with the following online information: geometrical characteristics and position of the drop during casting, the color histogram, YUV/RGB components, and the color map of the drop.

The process of the drop identification, involves image segmentation and edge detection. Segmentation involves separating the image into regions. Our application uses thresholding to determine the region of the drop area by identifying of four boundary points. Then, edge detection is performed. There are many ways to perform edge detection. We tried different algorithms to detect the drop edge. Our experiments showed that the Canny algorithm ([1]) provides the best results.

The last step is dealing with the approximation of the drop shape and the localization of the capillary (operating) point, according to the model presented above. In this context, the interpolation points are calculated as follows.

At first, a lot of points are extracted from the drop shape. Those points are integer numbers. Then a set of five reference points is determined based on extracted points, acording to the expression:

$$P_r = k_1 \bullet P_{j-2} + k_2 \bullet P_{j-1} + k_3 \bullet P_j + k_4 \bullet P_{j+1} + k_5 \bullet P_{j+2}, \tag{5}$$

where $j = 5(r-1)+3$, for $1 \le r \le 5$, and $k_1 = k_5 = 1$, $k_2 = k_4 = 2$, $k_3 = 3$. The expression (5) and the values of coefficients were obtained experimentally. It should be noted, that the coordinates of these five references are numbers in floating point format. Given the set of five reference points as well as two points at the ends of the drop shape (white circles in the figure 5), the Lagrange basis polynomials can be written and calculated.

Then the point of inflection and the capillary point location are determined by means of relations (3) and (4) respectively.
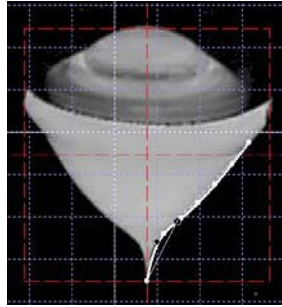


**Figure 5.** *Online calculation of the capillary point location.*

Figure 5 shows the calculated online location for the inflection point and capillary point (black points in the figure 5) during casting.

As it was mentioned in the introduction, we aim the development of advanced techniques for the optimization of industrial information systems with human in the loop. The proposed approach and methods must help the human operator in taking optimal online decisions during microwire casting.

In conclusion, we want to mention that the results presented above are intended to be used in the decision support system design for the microwire casting machines in order to provide data and knowledge for decision making.

# References

[1] J. Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 8*(6), 679-714, 1986.
[2] E. R. Davies, *Machine vision. Theory, algorithms, practicalities*, Morgan Kaufmann, 2004.
[3] V. Larin, A. Torcunov, A. Zhukov, J. González, M. Vazquez, L. Panina. Preparation and properties of glass-coated microwires. *Journal of Magnetism and Magnetic Materials, 249*(1-2), 39-45, 2002.
[4] S. Zaporojan. Online analysis and decision making for the process of microwire casting. *Advances in analysis and decision making for complex and uncertain systems,* Vol.1, 20-24, 2011. Published by the International Institute for Advanced Studies in Systems Research and Cybernetics, Baden-Baden, Germany.
[5] S. Zaporojan, C. Plotnic, I. Calmicov, V. Larin. A knowledge-based approach for microwire casting plant control. In: J. Jozefczyk and D. Orski (Eds). *Knowledge-Based Intelligent System Advancements: systemic and cybernetic approaches,* Information Science Reference, Hershey New York, 2010.

Sergiu Zaporojan
Technical University of Moldova
Computer Science Department
168, Stefan cel Mare str., Chisinau, 2004
MOLDOVA Republic of
E-mail: zaporojan_s@yahoo.com

Constantin Plotnic
Technical University of Moldova
Computer Science Department
168, Stefan cel Mare str., Chisinau, 2004
MOLDOVA Republic of
E-mail: pcpvir13@rambler.ru

Igor Calmicov
Technical University of Moldova
Computer Science Department
168, Stefan cel Mare str., Chisinau, 2004
MOLDOVA Republic of
E-mail: igorioc@bk.ru

# List of authors

**ALEXIEV Kiril**
Institute of Information and Communication Technologies
Mathematical Methods for Sensor Information Processing
Sofia, 25A Acad. G. Bonchev Str.
BULGARIA
E-mail: *alexiev@bas.bg*

**BOCHEVA Nadejda**
Institute of Neurobiology
Sensory Neurobiology
Sofia, 25 Acad. G. Bonchev Str.
BULGARIA
E-mail: *nadya@percept.bas.bg*

**BRUC Natalie**
Institute of Mathematics and Computer Science of ASM
5, Academiei street, Chisinau,
REPUBLIC OF MOLDOVA, MD 2028
E-mail: *nataliebruc@yahoo.com*

**BUCERZAN Dominic**
Aurel Vlaicu University of Arad
Department of Mathematics-Informatics
310330 Arad, 2 Elena Drăgoi
ROMANIA
E-mail*: dominic@bbcomputer.ro*

**BURTSEVA Liudmila**
Institute of Mathematics and Computer
Science of the Academy of Sciences of Moldova
Programming Systems Laboratory
5, Academiei street, Chisinau, MD 2028
REPUBLIC OF MOLDOVA
E-mail: *burtseva@math.md*

**CALMICOV Igor**
Technical University of Moldova
Computer Science Department
168, Stefan cel Mare str., Chisinau, 2004
REPUBLIC OF MOLDOVA
E-mail: *igorioc@bk.ru*

**COJOCARU Svetlana**
Institute of Mathematics and Computer
Science of the Academy of Sciences of Moldova
Programming Systems Laboratory
5, Academiei street, Chisinau, MD 2028
REPUBLIC OF MOLDOVA
E-mail: *svetlana.cojocaru@math.md*

**CRĂCIUN Mihaela**
Aurel Vlaicu University of Arad
Department of Mathematics-Informatics
310330 Arad, 2 Elena Drăgoi
ROMANIA
E-mail*: mihaeladacianacraciun@yahoo.com*

**DIMITROV Stefan**

University of Sofia "St. Kliment Ohridski"
FMI, Computing Systems Dept.
5, J. Bourchier, Str., Sofiq
BULGARIA
E-mail: *stefan@ucc.uni-sofia.bg*

**FARCAŞ Anca**

Babes-Bolyai University
Faculty of Mathematics and Computer Science
Kogalniceanu Street, No.1, 400084 Cluj-Napoca
ROMANIA
E-mail*: anca.farcas@ubbcluj.ro*

**GAO Xuefeng**

University College Cork
Dept. of Computer Science
WGB, UCC, Western Rd., Cork
IRELAND
E-mail*: xfg1@cs.ucc.ie*

**GAINDRIC Constantin**

Institute of Mathematics and Computer
Science of the Academy of Sciences of Moldova
Programming Systems Laboratory
5, Academiei street, Chisinau, MD 2028
REPUBLIC OF MOLDOVA
E-mail: *gaindric@math.md*

**GROZA Adrian**

Technical University of Cluj-Napoca
Computer Science Department
Memorandumului 28, Cluj-Napoca
ROMANIA
E-mail: *adrian.groza@cs.utcluj.ro*

**HUNYADI Daniel**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *daniel.hunyadi@ulbsibiu.ro*

**LALOV Petko**

University of Mining and Geology "St.Ivan Rilski"
Head of Dept. of Informatics
Students' Town, Sofia
BULGARIA
E-mail: *petko@mgu.bg*

**LAZĂR Valentina**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *lyah_valy@yahoo.com*

**MAGARIU Galina**

Institute of Mathematics and Computer Science of ASM
5, Academiei street, Chisinau,
REPUBLIC OF MOLDOVA, MD 2028
E-mail: *gmagariu@math.md*

**MAN Diana-Ofelia**

Babes-Bolyai University
Department of Computer Science
400084 Cluj-Napoca
ROMANIA
E-mail: *mandiana77@yahoo.com*

**MANIU Constantin**

Spiru Haret University of Brasov
Faculty of Management
ROMANIA
E-mail: *costelmaniu@yahoo.com*

**MANIU Ionela**

"Lucian Blaga" University of Sibiu,
Faculty of Sciences
ROMANIA
E-mail: *mocanionela@yahoo.com*

**MECHNO Raluca**

Technical University of Cluj-Napoca
Computer Science Department
Memorandumului 28, Cluj-Napoca
ROMANIA
E-mail: *raluca.mechno@student.utcluj.ro*

**MUSAN Mircea**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *musanmircea@yahoo.com*

**NUT Gabriela**

Babes-Bolyai University, Cluj-Napoca
Faculty of Mathematics and Computer Science
1, Mihail Kogalniceanu str., Cluj Napoca, 400084
ROMANIA
E-mail: *djantai@yahoo.com*

**PARASCHIV-MUNTEANU Iuliana**

University of Bucharest
Faculty of Mathematics and Computer Science
14 Academiei St., Bucharest 010014
ROMANIA
E-mail: *pmiulia@fmi.unibuc.ro*

**PITIC Elena Alina**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *alinap29@yahoo.com*

**PLOTNIC Constantin**

Technical University of Moldova
Computer Science Department
168, Stefan cel Mare str., Chisinau, 2004
REPUBLIC OF MOLDOVA
E-mail: *pcpvir13@rambler.ru*

**POP Daniel Nicolae**

Romanian-German University Sibiu
ROMANIA
E-mail: *danielnicolaepop@yahoo.com*

**POP Ioan**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *me.ioanpop@yahoo.com*

**POPCOVA Olga**

Institute of Mathematics and Computer
Science of the Academy of Sciences of Moldova
5, Academiei street, Chisinau, MD 2028
REPUBLIC OF MOLDOVA
E-mail: *oleapopcova@yahoo.com*

**RAȚIU Crina**

SC Daramec SRL
Loc. Șofronea, F.N., Jud. Arad
ROMANIA
E-mail: *ratiu_anina@yahoo.com*

**RĂULEA Cristina**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *cristina.raulea@ulbsibiu.ro*

**SCHEIBER Ernest**

Transilvania University of Bra_sov
Department of Computer Science
Str. I. Maniu 50
ROMANIA
E-mail*: scheiber@unitbv.ro*

**SECRIERU Iulian**

Institute of Mathematics and Computer
Science of the Academy of Sciences of Moldova
5, Academiei street, Chisinau, MD 2028
REPUBLIC OF MOLDOVA
E-mail: *secrieru@math.md*

**SIMIAN Corina**

Zurich University
Instiute of Mathematics
SWITZERLAND
E-mail: *corina.simian@math.uzh.ch*

**SIMIAN Dana**

„Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *dana.simian@ulbsibiu.ro*

**STÂNEA Paul**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMÂNIA
E-mail: *psb77black@yahoo.com*

**STATE Luminita**

University of Pitesti
Faculty of Mathematics and Computer Science
1 Targu din Vale St., Pitesti 110040
ROMANIA
E-mail: *lstate@clicknet.ro*

235

**STEFANOV Simeon**

Institute of Neurobiology
Sensory Neurobiology
Sofia, 25 Acad. G. Bonchev Str.
BULGARIA
E-mail: *simeon_st@percept.bas.bg*

**STOICA Florin**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *florin.stoica@ulbsibiu.ro*

**STOICA Laura Florentina**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail: *laura.cacovean@ulbsibiu.ro*

**SURDU Sabina**

Babes-Bolyai University
Faculty of Mathematics and Computer Science
Str. Mihail Kogalniceanu nr. 1, RO-400084, Cluj-Napoca
ROMANIA
E-mail: *surdusabina@yahoo.com*

**TABIRCA Sabin**

University College Cork
Dept. of Computer Science
WGB, UCC, Western Rd., Cork
IRELAND
E-mail: *s.tabirca@cs.ucc.ie*

**TACHEV Gancho**

University of Architecture
Dept. of Mathematics
1 Hr.Smirnenski, blvd., Sofia, 1046
BULGARIA
E-mail*: fte@uacg.bg*

**TRÎMBIȚAȘ Radu Tiberiu**

"Babeş-Boyai" University Cluj-Napoca
ROMANIA
E-mail: *tradu@math.ubbcluj.ro*

**TUBA Milan**

Megatrend University of Belgrade
SERBIA
E-mail: *tubamilan@ptt.rs*

**TUDORICĂ Gabriel**

"Lucian Blaga" University of Sibiu
Faculty of Sciences
ROMANIA
E-mail*: office@eyenetworks.ro*

**VASILE Laurentiu**

University of Bucharest
Faculty of Mathematics and Informatics
14 Academiei, 010014 Bucharest
ROMANIA
E-mail: *vsl@fmi.unibuc.ro*

**VELCESCU Letitia**       University of Bucharest
Faculty of Mathematics and Informatics
14 Academiei, 010014 Buchares
ROMANIA
E-mail: *letitia@fmi.unibuc.ro*

**VERLAN Tatiana**       Institute of Mathematics and Computer Science of ASM
5, Academiei street, Chisinau,
REPUBLIC OF MOLDOVA, MD 2028
E-mail: *tverlan@math.md*

**ZAPOROJAN Sergiu**       Technical University of Moldova
Computer Science Department
168, Stefan cel Mare str., Chisinau, 2004
REPUBLIC OF MOLDOVA
E-mail: *zaporojan_s@yahoo.com*