ISSN 1727–6209

TERNOPIL NATIONAL ECONOMIC UNIVERSITY
RESEARCH INSTITUTE OF INTELLIGENT COMPUTER SYSTEMS

ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
НАУКОВО-ДОСЛІДНИЙ ІНСТИТУТ ІНТЕЛЕКТУАЛЬНИХ
КОМП’ЮТЕРНИХ СИСТЕМ

International Journal of

Computing

Published since 2002

March 2012, Volume 11, Issue 1

Міжнародний науково-технічний журнал

Комп’ютинг

Видається з 2002 року

Березень 2012, Том 11, Випуск 1

Постановою президії ВАК України № 1-05/3 від 14 квітня 2010 року науково-технічний журнал “Комп'ютинг” віднесено до переліку наукових фахових видань України, в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора і кандидата технічних наук

Тернопіль
ТНЕУ
2012

International Journal of Computing

March 2012, Vol. 11, Issue 1

Research Institute of Intelligent Computer Systems,
Ternopil National Economic University

Registered by the Ministry of Justice of Ukraine.
Registration certificate of printed mass media – KB #17050-5820PR, 15.07.2010.

It’s published under resolution of the TNEU Scientific Council, protocol # 5, February 29, 2012

Editorial Board

Address of the Editorial Board

Research Institute of Intelligent Computer Systems
Ternopil National Economic University
3, Peremoga Square
Ternopil, 46020, Ukraine

Phone: +380 (352) 47-5050 ext. 12234
Fax: +380 (352) 47-5053 (24 hours)
computing@computingonline.net
www.computingonline.net

Міжнародний журнал “Комп’ютинг”

Березень 2012, том 11, випуск 1

Науково-дослідний інститут інтелектуальних комп’ютерних систем
Тернопільський національний економічний університет

Зареєстрований Міністерством юстиції України. Свідоцтво про державну реєстрацію друкованого засобу масової інформації – серія КВ №17050-5820ПР від 15.07.2010.

Друкується за постановою вченої ради ТНЕУ, протокол № 5 від 29 лютого 2012 року

Редакційна колегія

Адреса редакції :

НДІ інтелектуальних комп’ютерних систем
Тернопільський національний економічний університет
площа Перемоги, 3
Тернопіль, 46020, Україна

Тел.: 0 (352) 47-50-50 внутр. 12234
Факс: 0 (352) 47-50-53
computing@computingonline.net
www.computingonline.net

CONTENTS

ЗМІСТ

СОДЕРЖАНИЕ

EDITORIAL

“Pattern Recognition and Intelligent Processing”

Guest Editor: Janusz Zalewski

1. OVERVIEW

The papers included in this issue have been divided into two basic categories: those addressing specific algorithms in image and pattern recognition and processing, and those covering general aspects of these areas, which include two review papers: one on multicore processing and one on prognostics. Accordingly, the rest of this Introduction discusses briefly the contents of these papers – divided into two respective sections – and their significance.

2. SPECIFIC ALGORITHMS

V. Koval et al., in their paper titled “Method of Environment Reconstruction Using Epipolar Images”, discuss a method for 3D environment reconstruction that allows fusing stereo-images under restricted timing and other computational resources. As they justly state, “3D reconstruction is an important task of computer vision that consists in a computation of 3D model of an environment on the basis of its 2D images and is widely used in industry, robotics, computer graphics, augmented reality.” Thus, the importance of related research cannot be overestimated.

Modern computer vision systems for 3D reconstruction, which are built on the basis of time-of-flight principle, have several disadvantages, such as low image resolution, errors in a reconstruction of mirror-like surfaces, high cost and power consumption that limits the working time of a camera in autonomous mode. Therefore computer vision systems based on parallax principle are more frequently used. Using a stereocamera in conjunction with a 3D reconstruction method, the authors develop an algorithm for epipolar image fusion with processing time of 3D coordinates two times shorter than that of an existing algorithm.

The paper by Aleksandra Maksimova on “The Model of Data Presentation with Fuzzy Portraits for Pattern Recognition” deals with data presentation based on fuzzy portraits, which is a fuzzy modification of a classification problem. The fuzzy portraits are formed by integral characteristics of pattern classes and form the basis for the construction of a fuzzy classifier. It turns out that further division of some classes of images into clusters increases the quality of pattern recognition.

The algorithm was tested on “wine” problem from the University of California at Irvine Machine Learning Repository [1]. It gave acceptable results on model data with error rate not more than 2%.

This approach is useful for pattern recognition problems with great number of classes. It can be used for quality control in chemical and food industries. It is applicable not only to machine learning problems but can be also used in storing the results of data analysis in a knowledge base for expert systems.

Vladimir Podolskiy, in his paper titled “Dividing Outliers into Valuable and Noise Points”, considers a specific problem in clustering, where data points, which do not belong to specific clusters (known as outlier data) can still be used for the analysis of unusual system behavior. He presents a fuzzy based optimization approach for division of these outliers into those interesting and usable, and those no longer relevant, that is, noise.

Data have to be first divided into clusters, for which a DBSCAN method (Density-Based Spatial Clustering of Applications with Noise) is used. The algorithm is essentially based on finding the closest cluster and calculating respective distances. The practical application of the algorithm can be found in many fields using data classification, for example: handling research data, fraud detection, searching for minerals, etc.

In comparison to other classification approaches, such as neural networks classification, SVM-based classification, decision trees classifiers, some fuzzy based algorithms, this method’s advantages are simplicity and flexibility.

The next paper, titled “Face Recognition for Machine Readable Travel Documents”, by S. Tulyakov and R. Sadykhov, presents a frontal face recognition system able to handle large image databases with high processing speed and low detection and identification errors. Developing such technologies is important in a view of a wide range of potential applications, from biometric systems and border control, to access control systems, video surveillance, human-computer interfaces and content based image retrieval.

In this particular method, the recognition is performed with the use of eigenface approach and the eigenface ranking measure. According to the authors: “The proposed *the biggest face* approach along with regions that most likely contain eyes significantly increase the speed of feature detection.” Furthermore, variable image scale parameter contributes to the increase in detection speed and accuracy, critical in handling large images databases. The computational sequence is able to standardize faces to the same state regardless of initial rotation and size.

Facial Recognition Technology (FERET) database from the National Institute of Standards and Technology (NIST) [2] has been used for experiments with around ten thousand face images.

The paper on “Influence Learning for Multi-agent Systems Based on Reinforcement Learning”, by A. Kabysh, V. Golovko and A. Lipnickas, describes a multi-agent influence learning approach and reinforcement learning adaptation to it. It discusses the rationale for using *influence*, as an “active offer from one agent to another, which can be accepted or not; if influence is accepted, then it affects the agent’s behavior and (or) translates it into a new state”.

Agents’ influences are used to estimate learning error between them. The best influence is then rewarded via reinforcement learning. As the authors show, this learning principle supports positive-reward interactions between agents and does not require any additional information than standard reinforcement learning. The experiments prove that this technique ensures fast convergence, much faster than Joint-Action Learners (JAL).

The authors claim that “This approach helps to solve almost all the difficulties facing the multi-agent learning supportive without losing its simplicity”, stating additionally that the technique can be applied to almost any problem of multi-agent reinforcement learning. They are aware that for full evaluation comparisons have to be made with more sophisticated multi-agent reinforcement learning algorithms.

3. GENERAL ISSUES

Development of specific algorithms for pattern recognition, image processing, machine learning, etc., as discussed in papers presented in the previous section, although extremely important by itself, is only one side of the whole story. Essentially, an algorithm can be written on paper and assessed theoretically. However, one has to keep in mind that as an artifact each algorithm can be additionally viewed from two perspectives: a detailed perspective, which concerns its implementation, and a broader perspective, which concerns its theoretical base. What is critical for the use of an algorithm is the implementation. But one has to keep in mind what specific theoretical apparatus is behind the reasoning provided in an algorithm. For this reason, the researchers need not only computational resources, to run the implementations and evaluate their effectiveness, but also sound mathematical theories that provide the framework for mathematical description of algorithmic processes.

Four papers related to this general requirement are included in this issue. Two of them present research related strictly to enhancing the practical and theoretical infrastructure, and the other two are review papers addressing the infrastructure problems in a broader sense from the perspectives of the entire domain.

3.1. INFRASTRUCTURE ENHANCEMENT

S.M. Avakaw et al., in their paper titled “Multi-Agent Parallel Implementation of Photomask Simulation in Photolitography”, describe a framework for parallelization of an aerial image simulation in one specific application domain of machine vision systems. The parallelization of image processing is based on a graph model of a parallel algorithm. The algorithm is constructed in a visual form from individual computing operations, and then converted to XML representation to be interpreted by the multi-agent system running in MPI [3]. Such framework allows for flexible analysis of parallel algorithms and their adaptation to specific architectures of computing clusters.

The experiments conducted for 20x20 μm topological structure showed the speedup of approx. 3 times for four processors, which is considered by the authors as significant in the processing of photomasks. Another set of experiments proved nearly linear scalability of the algorithm (measured as performance vs. data sizes).

Overall, such architecture based on the use of multi-agent systems can be easily adapted to many other application domains, making this concept a flexible framework for parallelization of machine vision.

In the paper titled “Algebraic Approach to Information Fusion in Ontology-Based Modeling Systems”, I. Aremieva, A. Zuenko and A. Fridman discuss possibilities of using algebraic methods (in particular, n-tuple algebra, NTA) to improve functioning of convenient ontology-based modeling systems. This new, NTA-based approach, according to the authors allows for storing and processing of both data and knowledge structures by similar techniques, which is crucial for combining data and knowledge bases in a single system.

In NTA, the domain ontology and factual information can be recorded as a number of multiplace relations, and queries can be expressed by operations similar to those in set theory, which facilitates reasoning by comparatively simple algorithms. Consequently, complex algorithms can be implemented relatively easily.

The authors suggest that algebraic approaches seem to be a complement of traditional formal approaches used in logic, and can help in unification of information representation and processing, and improving logical analysis. In a view of increasing the intelligence of databases, the NTA technique can be considered an extension of relational algebras to knowledge processing. The theory is illustrated by solving relatively simple tasks in chemical synthesis, with initial and goal substances being fixed.

3.2 DOMAIN REVIEWS

Robert Hiromoto in his paper titled “The Art and Science of GPU and Multi-core Programming” examines the programmatic issues that arise from the introduction of graphical processing units (GPUs) and multi-core computer systems. This topic is critically important in a view of parallelization as emphasized in papers on multi-agent systems discussed above, both by Kabysh et al. and by Avakaw et al. He argues that two essential principles of *spatial* and *temporal* locality provide sufficient metrics to guide programmers in designing and implementing efficient applications.

Summarized by the author in one paragraph: “The art of high performance programming is to take combinations of these principles and unravel the bottlenecks and latencies associated with the architecture for each manufacturer computer system, and develop appropriate coding and/or task scheduling schemes to mitigate or eliminate these latencies.” It’s easier said than done, so the author identifies several specific problems in programming such system.

For example, one critical problem in multi-core systems is sharing of the cache between multiple cores, which may have negative impact on performance due to the unintended data access congestion between cores. The challenge of massively parallel programming paradigm of the GPU requires that for the optimal performance all threads must be active at any given time and make uniform progress. Consequently, the programmer faces the selection of appropriate coding practices, task scheduling mechanisms, and resource allocations techniques to mitigate or eliminate the latencies and bottlenecks.

The last paper, “Use of Artificial Intelligence Techniques for Prognostics: New Application of Rough Sets”, by Zalewski and Wojcik, reviews the domain of prognostics and sets the context for the potential application of rough sets. The authors review selected AI techniques used in prognostics, for predicting faults and failures in technical systems and then propose the application of rough sets to build the system health prognostication model.

Among the methods reviewed one can find model-based algorithms, especially particle filters, and data driven algorithms, such as support vector machines. Rough sets are placed among data driven methods, because they rely heavily on the data set analysis. In this view, rough sets allow for accurate data mining and prediction, with such new tools as rough sets variance, rough sets covariance, time-sequenced data fusion model, and deterministic machine learning.

The proposed rough sets prognostic model considers only data sequenced by time. The time makes data mining deterministic in the domain of health degradations. This near-deterministic approach allows making predictions based on degradation patterns gathered automatically for different applications and at the time instants of failures. This is the value of the rough sets as compared to more traditional techniques based on statistical methods.

This approach uncovers large areas of applications for rough sets, including predictions of health problems, diagnostics, mission readiness evaluation, equipment maintenance, optimization of logistics operations, and others [4].

4. CONCLUSION

The selection of papers collected in this issue proves that the scientific domain of pattern recognition and the associated AI techniques are alive and well. New problems and challenges are addressed by researchers every day, and solutions to some of them found their way into this issue. It concerns both the individual algorithms and respective implementation and theoretical infrastructures.

Looking back at the contents of the papers included here, it is important to stress that the critical issue in this area is the public availability of data sets, on which respective algorithms can be tested [1-2], [4]. So is the availability of computing standards and public domain tools [3], which make the implementations of the algorithms easier and more effective.

Finally, this Guest Editor would like to thank the Editorial Board of the International Journal of Computing for providing him an opportunity to work on the enhanced versions of papers from the PRIP 2011 Conference and make them look together more consistently to form a representative sample of this field of research.

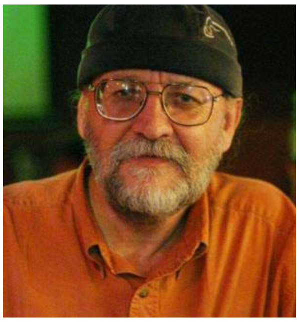
5. REFERENCES

[1] University of California at Irvine. Center for Machine Learning and Intelligent Systems. *Machine Learning Repository*. URL: http://archive.ics.uci.edu/ml/

[2] National Institute of Standards and Technology. *The Facial Recognition Technology (FERET) Database*. URL: http://www.itl.nist.gov/iad/humanid/feret/feret_master.html

[3] Argonne National Laboratory. Mathematics and Computer Science Division. *The Message Passing Interface (MPI) Standard*. URL: http://www.mcs.anl.gov/research/projects/mpi/

[4] NASA Ames Research Center. Intelligent Systems Division. Prognostics Center of Excellence. *Prognostics Data Repository*. URL: http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/



Prof. Janusz Zalewski
Deptartment of Software Engineering,
Florida Gulf Coast University
Fort Myers, FL 33965, USA
e-mail: zalewski@fgcu.edu,
http://www.fgcu.edu/zalewski/

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

METHOD OF ENVIRONMENT RECONSTRUCTION USING EPIPOLAR IMAGES

Vasyl Koval [1)], Oleh Adamiv[1)], Anatoly Sachenko [1)], Viktor Kapura [1)], Hubert Roth [2)]

[1)] Ternopil National Economic University, 3 Peremoga Square, 46020, Ternopil, Ukraine
vko@tneu.edu.ua, oad@tneu.edu.ua, vka@tneu.edu.ua, as@tneu.edu.ua, www.ics.tneu.edu.ua
[2)] University of Siegen, Hoelderlinstr 3, D-57068 Siegen, Germany
hubert.roth@uni-siegen.de, www.uni-siegen.de

Abstract: *In this paper the method for 3D Environment reconstruction usingepipolar images is presented. Method allows to fuse stereoimages within certain time depending on acceptable computational resources.*

Keywords: *3D Environment Reconstruction, Stereovision, Computer Vision, Epipolar Images.*

1. INTRODUCTION

The 3D reconstruction is an important task of computer vision that consists in a computation of 3D model of an environment on the basis of its 2D images and is widely used in industry, robotics, computer graphics, augmented reality.

Modern computer vision systems for 3D reconstruction on the basis of image set are based on Time-of-Flight Principle or parallax [2, 3, 18-20]. Computer vision systems based on Time-of-Flight Principle uses cameras with a source of modulated light and a specialized sensor [18-20]. Their disadvantages are low image resolution, errors in a reconstruction of mirror-like surfaces, expensive cost and big power consumption that limits the working time of a camera in autonomous mode. Therefore computer vision systems based on parallax are used broader. A stereocamera in conjunction with 3D reconstruction methods and algorithms is an example. There are many commercial models of stereocamera with expensive cost that depends on stereocamera specification.

Methods of stereoimage fusion are used for 3D reconstruction in a processing of stereoimages acquired from a stereocamera. Such methods search of corresponding points on a stereoimage with subsequent 3D reconstruction on the basis of a formed set of corresponding points and known parameters of the camera geometric model. Nowadays there are many methods of stereoimage fusion in particular:

- methods which use wavelet transform [9-11];
- methods which use interest point detectors and (or) contour segmentation for an initial image processing [4, 5, 8, 14, 16];
- methods which use contour segmentation with subsequent determination of straight lines and connections between them on images [1, 6, 7];
- methods which use Kalman filter [1, 12];
- methods which use dynamic programming for search of corresponding points on stereoimages [13, 15];
- methods which use neural networks [17].

Main disadvantage of existed methods is a computational complexity that limits their application in a real-time mode. Therefore a goal of this work is a development of method for stereoimage fusion with acceptable computational complexity.

2. DEVELOPED METHOD FOR FUSION OF EPIPOLAR IMAGES FOR 3D RECONSTRUCTION

The algorithm of the method for fusion of epipolar images presented on fig.1.

The first step of methods for stereoimage fusion on the basis of parallax is an initial image processing with aim of an improvement of image quality, a noise reduction and an image segmentation for detection of certain type of features on it (contours, interest points, etc) and a computation of descriptors for subsequent processing. The search of correspondences for segmented image elements using different types of a correlation of image areas that contains these elements and (or) descriptors with epipolar constraints performs on the next step.

The 3D reconstruction on the basis of found correspondences is the last step.

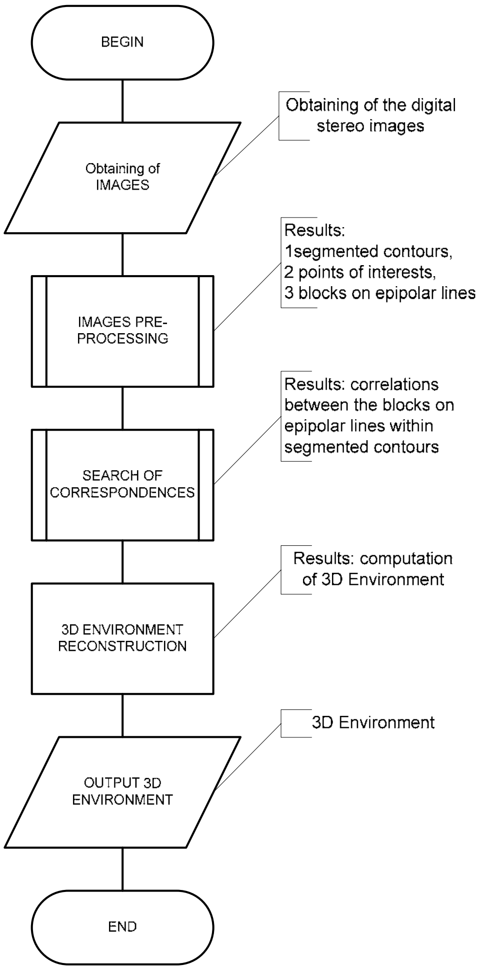

BEGIN
Obtaining of IMAGES
Obtaining of the digital stereo images
IMAGES PRE-PROCESSING
Results:
1segmented contours,
2 points of interests,
3 blocks on epipolar lines
SEARCH OF CORRESPONDENCES
Results: correlations between the blocks on epipolar lines within segmented contours
3D ENVIRONMENT RECONSTRUCTION
Results: computation of 3D Environment
OUTPUT 3D ENVIRONMENT
3D Environment
END

Fig. 1 – Flowchart for epipolar image fusion

The developed method is based on three main algorithms:

- stereoimage formation (image prepre-processing block on fig. 1);
- stereoimage fusion (search of correspondence block on fig. 1);
- computation of 3D coordinates.

The correlation is used as a similarity measure of image areas for correspondence point matching on stereoimage:

$$k = \frac{\sum_{m}\sum_{n}\left(I_{mn}-\bar{I}\right)\left(J_{mn}-\bar{J}\right)}{\sqrt{\left(\sum_{m}\sum_{n}\left(I_{mn}-\bar{I}\right)^2\right)\left(\sum_{m}\sum_{n}\left(J_{mn}-\bar{J}\right)^2\right)}}, \qquad (1)$$

where I_{mn} (J_{mn}) – pixel value in position (m,n) of image area I (J), and $\bar{I}$ ($\bar{J}$) – mean value of pixels in image area I (J).

The geometry interpretation of the developed algorithm of *stereoimage formation* is presented on fig 2 and the flowchart of it is presented on fig. 3. The main steps of developed *algorithm of stereoimage formation* are:

1) Camera calibration. Computed camera projection matrixes P, P' and fundamental matrix F are results of it.
2) Computation of homogenous coordinates of epipol e on image plane of projection matrix P:
   2.1) Computation of homogenous coordinates of camera centre C' of camera P' with constraint $P'C'=0$ using singular value decomposition (SVD);
   2.2) Computation of homogenous vector e, as $e=PC'$;
3) Define initial point a with coordinates $[x \quad y \quad 1]^T$ on image plane, which lies on first epipolar line;
4) From 1 to n, n – defined number of epipolar lines:

Compute epipolar line $l_e = e \times a'$ through epipol e and point a' with coordinates $[x \quad y+(j-1)*o \quad 1]^T$ on image plane, o – shift in pixels and $o=(y\_\max - y_a)/n$, y_a – coordinate of point a on axis y and $y\_\max$ – defined maximum coordinate of point a' on image axis y;

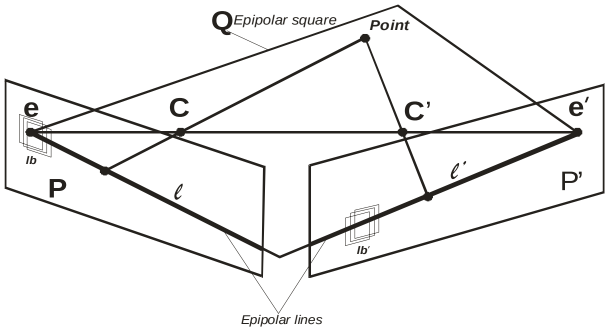


Fig. 2 – Geometry interpretation of algorithm for stereoimage formation during image preprocessing

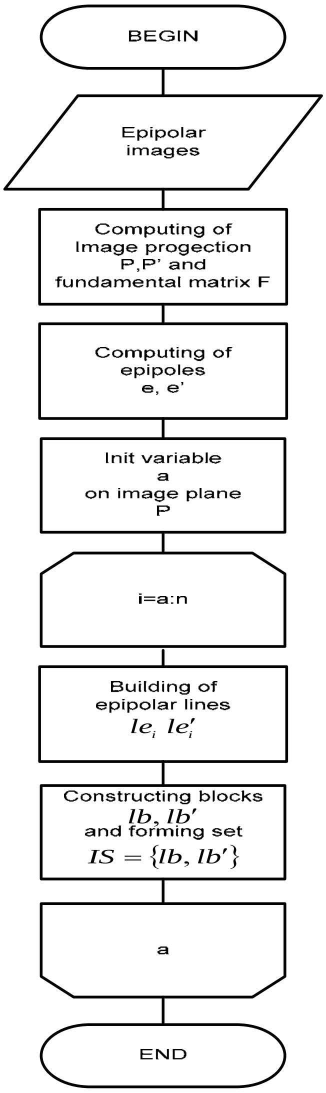

BEGIN
Epipolar images
Computing of Image progection P,P' and fundamental matrix F
Computing of epipoles e, e'
Init variable a on image plane P
i=a:n
Building of epipolar lines le_i le_i'
Constructing blocks Ib, Ib' and forming set $IS = \{Ib, Ib'\}$
a
END

Fig. 3 – Flowchart of the developed algorithm of stereoimage formation

5) Computation of corresponding epipolar line $l_e' = F[e]_\times l_e$ for epipolar line l_e on image plane of P';
6) Formation of image blocks Ib and Ib', which contains points of epipolar lines l_e, l_e' and adjacent points from $y_l + d$ to $y_l - d$ on image axis y, and adding these blocks to set of image blocks IS. Go to step 4.

The results of algorithm for stereoimage formation is given on fig. 4.

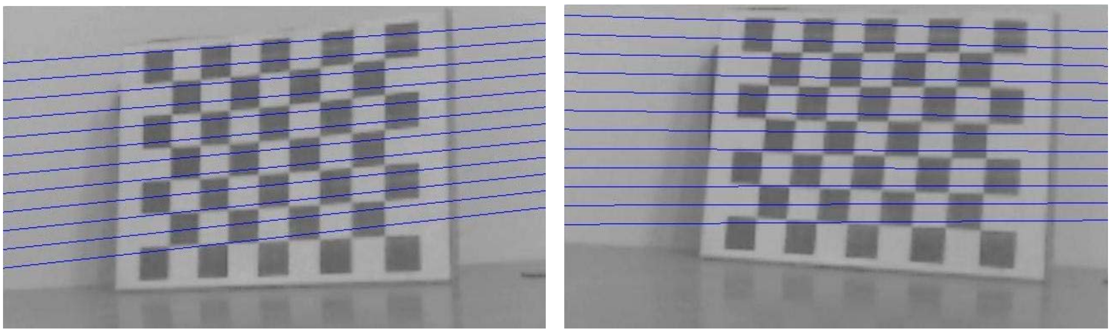
The next task for environment reconstruction is to search the correspondences (see fig. 1) between the set of blocks from the previous algorithm. This task is reached by using of the developed *algorithm of stereoimage fusion* (see the flowchart on fig.5). The algorithm of stereoimage fusion contains the following steps:

1) For each block Ib_i and Ib_i' of formed stereoimage IS, where $i = 1 \ldots n$, and n – defined number of epipolar lines:
2) Detect interest points on each block Ib_i and Ib_i' using developed method of interest point detection.
3) Determine matrix FPs (FPs') that contains coordinate vectors of interest points on epipolar lines of image blocks Ib_i (Ib_i') in columns.
4) For each column FPs_t of matrix FPs:
   - 4.1) Compute similarity measure of image area with size $m \times m$ and center in FPs_t and image area with size $m \times m$ and center in each interest point in image block FPs' using (1).
   - 4.2) Corresponding point is determined as point with minimal value $\min\limits_t (k_t)$ of similarity measures with subsequent deleting of column of matrix FPs' that contains coordinates of corresponding point.
   - 4.3) Add coordinates of corresponding points to set Cp.
   - 4.4) Go to step 4
5) Compute 3D coordinates of 3D point for each pair of corresponding image points using developed algorithm.

Main steps of developed *algorithm for computation of 3D coordinates* are*:*

1) For each pair of coordinate vectors of corresponding points x_i and x_i' of set Cp on image planes of cameras P and P':
   - 1.1) Compute 3D coordinate vector $X^+ = P^+ x_i$, $X^+ = X^+ / W$, W – fourth component of homogenous vector X^+, P^+ – pseudoinverse matrix of camera projection matrix P (P^+ is computed while camera calibration).
   - 1.2) Compute coordinate vector x^+ for projection of 3D point X^+ on image plane of camera P' using $x^+ = P'X^+$.
   - 1.3) Solve system of linear equations $Aa = x_i'$, $A = [x^+ \quad e]$, e is coordinate vector of epipol on image plane of camera P' and $a = [\lambda_1 \quad \lambda_2]^T$.

Compute 3D coordinates of point X_i using $X_i = \lambda_1 X^+ + \lambda_2 C$, $X_i = X_i / W_i$, W_i – fourth component of homogenous vector X_i.



a) b)

Fig. 4 – Results of epipolar lines construction using image preprocessing: a) left image; b) right image

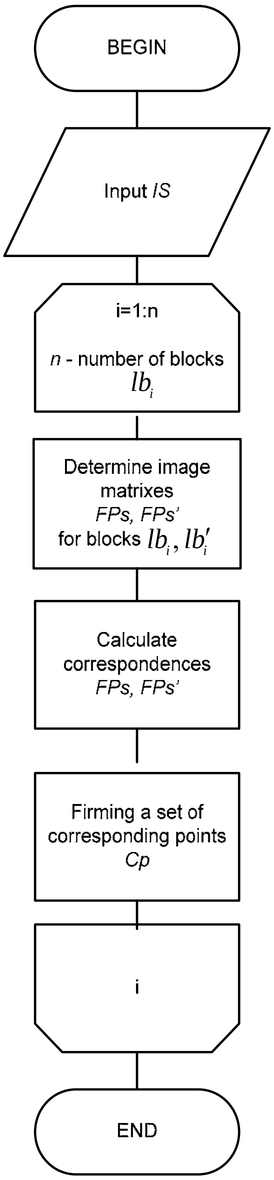


Fig. 5 – Flowchart of the developed algorithm for stereoimage fusion

3. EXPERIMENTS

The time of stereoimage fusion that depends on threshold value T and number of epipolar lines n is shown in Table 1. The time of stereoimage fusion can be controlled by number of epipolar lines n, threshold of interest point detection T and reduced using computer with more computational resources than one used during experiments (CPU Celeron 1.1 Ghz and 512 RAM) in Matlab environment and implementation on programming language C\C++ and (or) CUDA. We can conclude from Table 1: if value of T is more than 6 pixels and number of epipolar lines is not more than 18 then a method has an acceptable performance with mentioned computational resources.

Table 1. Time of Stereoimage Fusion

Number of epipolar lines	Mean time of fusion in seconds
$T = 3.5$ pixels	
$n = 6$	300,721
$n = 18$	1473,808
$T = 7$ pixels	
$n = 6$	3,162
$n = 18$	89,895
$n = 36$	139,425
$n = 72$	366,091
$T = 10$ pixels	
$n = 6$	1,057
$n = 18$	33,936
$n = 36$	54,537
$n = 72$	127,211

The dependence of the time of 3D coordinates computation from a number of epipolar lines for existed algorithm of triangulation using SVD (line with markers „+”) and developed one (line with markers „o”) is shown on Fig. 6. Mean error of 3D coordinates computation was equal to 20.0352 mm for existed algorithm and 58.6495 mm for developed one. We may conclude from graph on Fig. 6 that time of 3D coordinates computation for a developed algorithm in less in two times than for existed one.

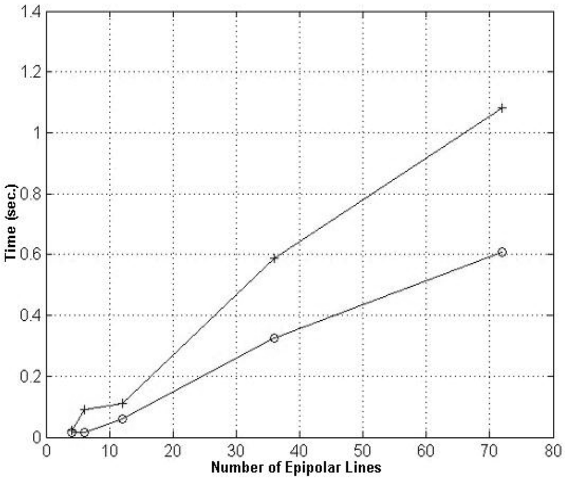


Fig. 6 – The dependence of the time of 3D coordinates computation from a number of epipolar lines

4. CONCLUSIONS AND FUTURE WORK

Authors propose a method which allows to fuse stereoimages within certain time depending on acceptable computational resources. On its base the algorithm was developed providing processing time of 3D coordinates in two times less than existed one.

The implementation of proposed method in CUDA and a development of the interest point descriptor will be objectives of future work.

5. ACKNOWLEDGMENT

This work has carried out within the Ukrainian-German joint research project “Development of Stereovision Methods and Devices for Autonomous Navigation of Mobile Robots”, supported by Ukrainian and German government both.

6. REFERENCES

[1] Ayache N., *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception*, The MIT Press; Cambridge Massachusetts, London England. – 1991.

[2] Hartley R., Zisserman A., *Multiple-View Geometry in Computer Vision*, 2nd edition; Cambridge university press. – 2003.

[3] Marr D., Poggio T., A computational theory of human stereo vision, *Proc. R. Soc. Lond.*, London, (1979), pp. 301-328.

[4] Kanazawa Y., Uemura K., Wide baseline matching using triplet vector descriptor, *Proc. 17th British Machine Vision Conf.*, Edinburgh, U.K., (I) (2006), pp. 267-276.

[5] Ming J., Xianlin H., A lunar terrain reconstruction method using long base-line stereo vision, *Proc. of the 26th Chinese Control Conference*, Hunan, China, (2007), pp. 488-492.

[6] Sekhavat S., Kamangar F., Geometric feature-based matching in stereo images, *Proceedings of the Information, Decision and Control.* – 1999.

[7] Folsom T.C., Non-pixel robot stereo, *Proc. of the IEEE Symposium on Computational Intelligence and Signal Processing* (CIISP 2007). – 2007.

[8] Moallem P., Fast edge-based stereo matching algorithm based on search space reduction, *IEICE Transaction Information and Systems*, Madrid, Spain (E-85) 11 (2006), pp. 243-247.

[9] Chen T., Klarquist W., Bovik A., Stereo vision using Gabor wavelets, *Proc. IEEE International Conf. on Systems, Man, and Cybernetics*, (1994), pp. 55-60.

[10] Chen T., Bovik A., Super B., Multiscale stereopsis via Gabor filter phase response. – 1994.

[11] Sarkar I., Bansal M., A wavelet-based multiresolution approach to solve the stereo correspondence problem using mutual information, *IEEE transaction on systems, man. and cybernetics*, (37) 4 (2007).

[12] Yi J., Oh J., Recursive resolving algorithm for multiple stereo and motion matches, Image and Vision Computing, Vol. 15.-1997, pp. 181-196.

[13] Torr P., Criminisi A., Dense stereo using pivoted dynamic programming, *Image and Vision Computing*, (22) (2004), pp. 795-806.

[14] Maciel J., Costeira J., Robust point correspondence by concave minimization, *Image and Vision Computing*, (20) (2002), pp. 683-690.

[15] Zhang Y., Gerbrands J., Method for matching general stereo planar curves, *Image and Vision Computing*, (13) (1995).

[16] Shi J., Tomasi C., Good features to track, *IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, 1994.

[17] Wang J., Hsiao C., Stereo matching by neural network that uses sobel feature data, *IEEE International Conference on Neural Networks*, vol. 3, 3-6 June 1996, pp. 1801-1806.

[18] http://www.fotonic.com/

[19] http://www.vicon.com/

[20] http://www.pmdtec.com/

__Vasyl Koval__ was born in Ternopil in 1975. In 1998 he received B.Sc. in "Management Information Systems" at Ternopil Academy of National Economy, Institute of Computer Information Technologies. In 1999 he obtained M.Sc. in "Economic Cybernetics" at Ternopil Academy of National Economy. In 2004 he obtained Ph. D. degree.

Now he works as Vice-dean and Associated Professor at Faculty of Computer Information Technologies, Ternopil National Economic University.

His research area includes: Robotic Systems; Distributed Systems; Sensor Fusion Techniques; Sensor Fusion Algorithms; Data Acquisition Systems; Stereo Vision; Stereo Matching; Image Processing; Artificial Intelligence; Neural Networks; Robot Navigation Systems; Sensor Systems, etc.

__Oleh Adamiv__ graduated Ternopil Academy of National Economy in 2000 with speciality "Information Systems in Management", 2001 – Master in Economical Cybernetics, 2007 – PhD degree.

At the moment he is a cheaf of International Information Department at Faculty of Computer Information Technologies, Ternopil National Economic University.

Areas of scientific interests includes: Artificial Intelligent, Neural Networks, Robotics, Autonomous Control, Robot Navigation.

__Anatoly Sachenko__ is Professor and Head of the Department of Information Computing Systems and Control and Research advisor of the Research Institute for Intelligent Computer Systems, Ternopil National Economic University. He earned his B.Eng. Degree in Electrical Engineering at L'viv Polytechnic Institute in1968 and his PhD Degree in Electrical Engineering at L'viv Physics and Mechanics Institute in 1978 and his Doctor of Technical Sciences Degree in Electrical and Computer Engineering at Leningrad Electrotechnic Institute in 1988. Since 1991 he has been Honored Inventor of Ukraine, since 1993 he has been IEEE Senior Member.

His main Areas of Research Interest are Implementation of Artificial Neural Network, Distributed System and Network, Parallel Computing, Intelligent Controllers for Automated and Robotics Systems. He has published over 450 papers in areas above.

__Viktor Kapura__ graduated Ternopil Academy of National Economy in 2005 as bachelor of Computer Engineering, and Ternopil State Economy University as specialist in 2006 in Computer Systems and Networks.

How he works at Research Institute of Intelligent Computing Systems.

Research interests: projective geometry, 3D reconstraction methods, digital image processing.

__Hubert Roth__ was studying "Electrical Engineering" at University Karlsruhe, Germany, from 1974 until 1979 with emphasis in "Control Engineering and Computer Science". Until 1984 he was working as a Scientific Co-worker at the Institute for Automatic Control Engineering at the University Karlsruhe and got his doctoral degree in Electrical Engineering. From 1984 until 1988 he was employed as a System Design Engineer at Dornier System, Spacecraft Control Department, in Friedrichshafen, Germany. After this industrial internship he went back to academia and worked as a Professor for Control and Sensor Systems at the University of Applied Sciences, Ravensburg-Weingarten, Germany, for 12 years. In the year 2000 he became head of the Institute for Automatic Control Engineering in the faculty of Electrical Engineering and Computer Science at the University of Siegen, Germany. In parallel he is Vice Chair of the Centre for Sensor Systems (ZESS) at the University of Siegen and is head of the working group "Automation, Mechatronics and Medical Engineering" in this centre. Prof. Roth is Vice Chair of the Technical Committee on Computers and Telematics of the "International Federation of Automatic Control" (IFAC).

The main teaching interests of Prof. Roth are in the fields of Control Theory, Robotics, Mechatronic Systems and Attitude Control for Space Applications. His research interests are focused on Mobile Robotics, Aero-space applications and Telecontrol as well as environmental exploration.

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

THE MODEL OF DATA PRESENTATION WITH FUZZY PORTRAITS FOR PATTERN RECOGNITION

Aleksandra Maksimova

Institute of Applied Mathematics and Mechanics of National Academy of Sciences of Ukraine,
74, Roza Luksemburg st., Donetsk, Ukraine
maximova.alexandra@mail.ru,
http://iamm.ac.donetsk.ua/en/employees/e934/

Abstract: *This paper deals with a data presentation model based on fuzzy portraits. The fuzzy portraits are formed by integral characteristics of pattern classes. It is the basis for fuzzy classifier construction. It is determined that further division of some classes of images into clusters increases the quality of pattern recognition algorithm. The main idea of fuzzy clustering for fuzzy portraits creating and problem of adequate fuzzy partition choice is considered. The paper provides the stages of fuzzy production knowledge base construction on the basis of fuzzy portraits. The local validity measure for fuzzy portrait is defined. The problem of identification in chemical and food industries is considered as an application of this approach.*

Keywords: *Pattern recognition, fuzzy logic, clustering, data mining, fuzzy classifier.*

1. INTRODUCTION

Pattern recognition problems very often include fuzzy aspects. According to Lotfi Zadeh, the transition between classes is more gradual than intermittent. This statement is confirmed by practice of problem solving. In many empirical fields the standard approach that seeks a boundary between several classes gives no satisfactory results. There are different forms of fuzziness, described in [1]. We shall consider the following manifestations of fuzziness: diffusiveness, vagueness and relative position of classes.

In classical pattern recognition problem statement classes of images have to be strongly divided. The separating surface can be a hyperplane or have a rather complicated form. A large group of pattern recognition algorithms is based on estimates computation algorithms (ECO) concept proposed by Yu. I. Zhuravlev [2]. The basic principle of calculating if an object belongs to a class is calculating its closeness to the master object of owner class and farness to the master objects of other classes.

If classes of images have intersections the algorithm for making decisions about whether an object belongs to the class or not plays a big role. Previously, an algorithm based on fuzzy portraits of classes of images was proposed [3]. These fuzzy portraits describe all objects belonging to a certain class of images in general. Fuzzy portraits accumulate the information about elements of the class such as frequency of object popularity, degree of objects “concentration” and number of objects in every class. The suggestion is to use these portraits for representing the results of algorithm in the form of fuzzy sets. This approach to a certain degree solves the problem of ambiguous results in the case of intersecting classes. Information about an object’s distance from other classes has a substantially smaller significance in this situation and is used only for adjustment of fuzzy sets for aims of quality improvement. However the internal structure of classes is also taken into account. The classes of images can be further divided into groups of clusters that can have intersections too. Thus initial fuzzy portraits are expanded through additional structuring.

The results of this research were presented at the Pattern Recognition and Image Processing Conference in Belarus, Minsk and partially presented in [4].

2. FEATURES OF MODIFICATION OF PATTERN RECOGNITION PROBLEM

The following problem statement will be considered. It is a fuzzy modification of classification problem. Let us suppose that there is a set of training samples

$Y = \{(x^{(i)}, v^{(i)}) \mid x^{(i)} \in X, v^{(i)} \in V, i = 1,\dots,n\}$, where $x^{(i)} \in X \subset R^m$ are vectors in the n-dimensional space of images, $V = \{v_i \mid i = 1,\dots,k\}$ – set of classes of images. Pairs $(x^{(i)}, v^{(i)})$ set the correspondence between object $x^{(i)}$ and pattern class $v^{(i)}$. It is necessary to produce the algorithm of identification $a_Y(x) = \tilde{y}$. Vector $x \in X$ is the input of the algorithm, $X \supseteq X$ – the set of possible input data that depends on the problem domain. Generally, a fuzzy information vector $\tilde{y}$ describing membership x of pattern classes is obtained.

The training samples set can be incomplete, i.e. it may not contain all classes of images. The classes of images may overlap which is a manifestation of a form of fuzziness – diffusiveness of classes of images. For every object that is an input for the algorithm it is necessary to determine its membership degree in every pattern class.

Let us consider a qualitative attribute of an object called “source”. A pair of “sources” can produce a pair of images $\overline{x_1}$ and $\overline{x_2} : \overline{x_1} = \overline{x_2}, \overline{x_1} \in v_i, \overline{x_2} \in v_j, v_i \neq v_j$, i.e. patterns with same attribute values belong to different classes in the initial training set. In a situation when this attribute is a part of classification goal some classes will be nonseparable and their intersection can be very large, to a point where one class is absorbed by another. In such cases when the researcher fails to find a new attribute that would allow to divide the classes, the problem becomes unsolvable.

Such problems are common in quality control laboratories. In these cases a “source” is a product manufacturer that needs to be identified. Many problems in chemical and food industries can be reduced to this modification of the pattern recognition problem.

For such training samples a recognition algorithm based on the use of fuzzy portraits was proposed in [3]. The algorithm’s output is a fuzzy information vector set $\tilde{y} = (\mu_1,\dots,\mu_i,\dots,\mu_k)$, where μ_i is object’s membership degree for pattern class V_i, k is number of classes. Algorithm creates a knowledge base of fuzzy productions where every pattern class has its own rule of inference. The fuzzy inference algorithm works with knowledge base.

We consider a modification of the proposed algorithm. The analysis of empirical data often shows that there are subclasses (i.e. clusters) present in pattern classes. That is why it is useful to perform clustering for objects related to class to obtain a more precise fuzzy portrait.

3. ANALYSIS OF INITIAL SET

A necessary step in solving problems of pattern recognition is a preliminary analysis of initial data which is often called exploratory analysis of data. The basic aims of this analysis were formulated by J. Tukey:

- maximum penetration of data;
- the basic structures detection;
- selection of the more important attributes;
- discovering deviations and anomalies;
- testing the basic hypotheses;
- the construction of the initial model.

The proposed approach allows efficiently detecting basic structure of classes of images and constructing data presentation model as a result.

The study of initial data structure allowed to choose an algorithm for solving the problem. Generally every pattern recognition algorithm works well with certain data structure. For example, linear classifiers were created for linearly separable classes. The EM-algorithm [5] works with intersecting classes and implies the existence of a hidden variable. There is a number of algorithms covering similar situation that produce the result in the form of a fuzzy set.

In practice there are samples with one or more classes that consist of several clusters. Let us consider a case of strongly intersecting clusters that have pronounced centers (typical points) but have no clear border between them. It is possible, for example, to get this kind of data by changing some parameters of industrial technological process. A number of attributes changes which creates new clusters. It is reasonable to use fuzzy clustering approach [1,6] for intra-class structure analysis.

A criterion for the quality of the partitions on the original set of clusters is the determining parameter. Moreover, a solution that is optimal from a mathematical point of view does not always meet your goals. The current paper deals with one possible strategy of intra-class structure study with clustering methods.

Let a coverage Y_S be defined on the set of images X, that is a part of training samples set Y. Let $P = \{P_1,\dots,P_q,\dots,P_m\}$ be a named set of attributes, where m is the number of attributes. The attributes are defined on sets of values $X_{P_q} \subset X$, $P_q \in P$. Similarly to the class V_j, let us denote the set of images of training samples corresponding to this class as $V_j = \left\{ \overline{x_j^{(i_j)}} \mid i_j = 1,\dots,t_j \right\}$, $j = 1,\dots,k$, where k is the number of classes represented in training sample, t_j is the number of objects in the class with number j, and where $\overline{x_j^{(i_j)}} = (x_{j1}^{(i_j)},...,x_{jq}^{(i_j)},...,x_{jm}^{(i_j)}), q = 1,...,m$. So the coverage is $Y_S = \{V_1,...,V_j,...,V_k\}$.

Let the set $D_j^q = pr_{P_q} V_j = \left\{ x_{jq}^{(i_j)} \mid i_j = 1,...,t_j \right\}$ be a projection of class with number j on attribute P_q.

The basic principle of discriminant analysis is “the effect of essential multidimensionality”. According to this principle it is necessary to take into account the information about all attributes and to solve the problem in the m-dimensional space [7]. There are at least two reasons to avoid multiple dimensions. Firstly, data with small training samples gives us a sparse set of points which most pattern recognition algorithms do not process correctly. In this case a clustering by one or several attributes may be not worse than general clustering. Secondly, there are difficulties in substantiating the selection of distance measure which is used in multidimensional clustering.

Thus, we have two strategies for finding clusters. The first one is to perform clustering in the m-dimensional space taking into account the previously described shortcomings of this approach. The second approach is to perform fuzzy clustering for each projection $D_j^q, q = \overline{1,m}, j = \overline{1,k}$.

Let us consider the general scheme of data analysis and then details of both strategies of intra-class clustering.

Most commonly fuzzy clustering algorithms produce a fuzzy partition or fuzzy coverage as a result. Let us consider a family of fuzzy sets $R(X) = \{A^l \mid l = 1,...,c\}$, where $A_l = \{(x, \mu_l(x)) \mid \mu_l(x) \in [0,1], x \in X\}$ is fuzzy set and c is the number of clusters described by fuzzy set A_l. $R(X)$ is a fuzzy c-partition when the following condition is satisfied:

$$\sum_{l=1}^{c} \mu_l(x_i) = 1, i = 1,...,n, \mu_l \in [0,1]. \qquad (1)$$

$R(X)$ is a fuzzy coverage when the condition of fuzzy coverage is satisfied:

$$\sum_{l=1}^{c} \mu_l(x_i) \geq 1, i = 1,...,n, \mu_l \in [0,1]. \qquad (2)$$

The fuzzy clustering procedure is performed for every set V_j, $j = 1,...,k$ giving a fuzzy partition or a fuzzy coverage $R(V_j)$ as a result. It establishes a correspondence between partition Y_S and set of fuzzy partitions or coverages $R_S = \{R_j(X) \mid j = 1,...,k\}$.

The result of such analysis of training sample is the transformed data that is used in the algorithm and it is described by the following data model:

$$M_S = \{(V, R(V)) \mid V \in Y_S, R \in R_S\}. \qquad (3)$$

Thus we receive special data structure where a fuzzy coverage corresponds to every set $V \in Y_S$.

The major stages of data presentation model creating are the following:

1. The initial learning set Y is prepared based on real data from applied problem. The set of attributes P is defined for current problem.
2. The set of classes of images V is defined and initial set Y_S is constructed.
3. The intra-class clustering is carried out for all classes of images and fuzzy partition R_s is constructed.
4. The model of data presentation model with fuzzy portraits M_S is constructed on the basis of Y_S and R_S.
5. The quality of model M_S is tested. If the quality of the model M_S does not satisfy the purposes of pattern recognition, then the adjustment of the model by machine learning methods is carried out.

It should be noted that the number of clusters for every class of images V_j is unknown in this situation and it is the reason for using unsupervised clustering algorithms [8]. The quality estimation for the fuzzy partition or the fuzzy coverage $R(V_j)$ is carried out from the pattern recognition point of view. Thus we have to use clustering algorithms with super goal.

The described approach gives as a result an integral description of class of images that takes intra-class structure into account. The presence of the number of clusters in a class may be due to technological process specifics. For example, it can be a material change in the composition of substance in the manufacturing process that does not impact the stated quality but generates a homogeneous subgroup of images for the class, describing the substance.

4. DATA REPRESENTATION THROUGH FUZZY PORTRAITS OF CLASSES OF IMAGES AND FUZZY CLASSIFIER GENERATION

A number of pattern recognition algorithms uses the information about pattern classes structure. In these algorithms the distance function is used to calculate the typical points or build probability distributions of points in classes [9].

It is suggested to present the proposed data structure M_S as fuzzy portraits.

Let us introduce certain notions before giving the definition of a fuzzy portrait. Let the quintuple of objects $\{name(P_q),T_q,X_{P_q},G,M\}$ be called the linguistic variable L_q by the attribute P_q, where $name(P_q)$ denotes the name of linguistic variable, $T_q=\{\mathcal{M}_q^{(w)}\in[0,1]\mid w=1,\dots,r\}$ is term-set of linguistic variable values, w is the cluster number, r is the total number of clusters for all pattern classes. Syntactical rule G that generates variable names is trivial in this case, because all terms are atomic and it simply gives terms the name of a class or a subclass. Semantic rule M is represented as an algorithm for forming term-set membership functions.

Let *the fuzzy portrait* S_j of class of images V_j be the fuzzy area in m-dimensional space that is described by the three objects $(L,Rule,F)$ where L is the set of linguistic variables $\{L_q\mid q=1,m\}$, m is the number of attributes, $T_q=\{\mathcal{M}_q^i(x)\mid i=1,l\}$is the set of terms, where l corresponds to number of clusters R_j to current class V_j, R is set of rules and $F=\{F_{AND},F_{OR}\}$ are the pair of operations AND and OR correspondingly:

$$(\{L_q\mid q=1,\dots,m\},\{Rule_q\mid q=1,\dots,l\},\{F_{AND},F_{OR}\}). \quad (4)$$

The inference rule $Rule_q$ is the following:

$$IF\ \mathcal{M}_q^1(x)\ is\ A_q\ AND\ \dots AND\ \mathcal{M}_q^m(x)\ is\ A_q\ THAN\ V_j\ , \quad (5)$$

where A_q is the defined by $R(V_j)$ cluster. The number of such inference rules corresponds to the number of clusters detected by cluster analysis and can be different for different classes of images.

Let an operator F_{OR} be the function of taking the maximum then fuzzy portrait S_j can be determined by function:

$$F_{S_j}=\max_{i=1}^{l}(F_{AND}(\mathcal{M}_i^1(x^{(j)}),\dots,\mathcal{M}_i^m(x^{(j)}))) \quad (6)$$

The value of F_{S_j} is always in $[0,1]$ and if $F_{S_j}(\bar{x})=1$ then $\bar{x}$ belongs to a class of images V_j and if $F_{S_j}(\bar{x})=0$ then it does not.

Fig. 1 shows an example of fuzzy portrait visualization for class of images in two-dimensional space with two attributes P_1 and P_2. The points of initial set for this class are shown on the picture. Bright areas correspond to large membership degrees and white areas represent $F_{S_j}(\bar{x})=1$. The areas of fuzzy portrait with less intensity represent class members with admixtures: $0<F_{S_j}(\bar{x})<1$.

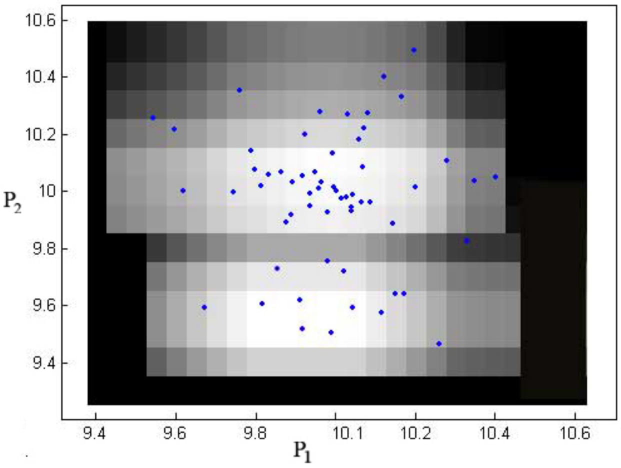


Fig.1 – Example of fuzzy portrait

Let *the degree of fuzzy portrait* S_j be the number of clusters that the class V_j consists of. Fig. 1 shows the second degree fuzzy portrait.

Let us denote the local validity measure for fuzzy portraits with:

$$F_{V_j}=\frac{1}{t_j}\sum_{i=1}^{t_j}F_{S_j}(\bar{x}_i) \quad (7)$$

where t_j is the number of initial set elements of class V_j. This measure estimates the measure of correspondence of the fuzzy portrait to initial set. The maximum value for this measure is one. It is an ideal situation that is never reached on real data taking into account the presence of noise.

The choice of operation F_{AND} and membership function for linguistic variables greatly influence the form of fuzzy portrait area. Some examples of operation AND are given in Table 1.

Table 1. –Function for AND operation

F_1	$F(\bar{x}) = \mu_1(x) \cdot \ldots \cdot \mu_m(x)$	(8)
F_3	$F(\bar{x}) = \min(\mu_1(x),\ldots,\mu_m(x))$	(9)
F_4	$F(x_1,x_2) = \max(\mu_1(x) + \mu_2(x) - 1,0)$	(10)

In [3] the following function has been proposed:

$$F_2(x_1,x_2,\ldots,x_m) = \log_2((x_1+1)\cdot\ldots\cdot(x_m+1))/m \quad (11)$$

where $x_i \in [0,1]$, $F_2(\bar{x}) \in [0,1]$.

The group of fuzzy inference rules describing fuzzy portraits represents a fuzzy classifier. This type of classifiers was used for pattern recognition purposes recently [10,11]. The fuzzy classifiers produce a fuzzy information vector $\tilde{y}$ as a result. The hardering procedure is used to get crisp answer. The class with maximum membership value is chosen.

The fuzzy classifiers are applied to pattern recognition and data analysis problems and each fuzzy inference rule has linguistic interpretation. That is why the mechanism of knowledge base generation can be used as an instrument for data analysis.

The advantage of this approach is having a meaningful interpretation of the result which is not the case with neural networks, for example.

5. INTRA-CLASS CLUSTERING BASED ON BATCH POINT-PROTOTYPE APPROACH

A popular approach for fuzzy clustering is fuzzy c-means model from batch point-prototype clustering models [6, p 16.] that aims to minimize the following functional:

$$J_m(A,C) = \sum_{i=1}^{l}\sum_{k=1}^{n}\mu_{ik}^{m}L_{ik}^{2}, \quad (12)$$

where A is a fuzzy partition like (1), $C = (c_1,\ldots,c_l)$ is a set of points prototypes, $m \geq 1$ is a degree of fuzzification, $L_{ik} = \|x_k - c_i\|$□is a distance.

The output of fuzzy c-means algorithm (fcm) is fuzzy c-partition like (1). Let us call this algorithm fcm-classic.

Let us call the following variation of this algorithm an fcm-projection algorithm. The attribute P_q is fixed and fcm is performed on $D_j^q = pr_{P_q}V_j$ with the output $R(X) = \{A^l \mid l = 1,\ldots,c\}$, where A_l is a vector determining membership for every element from D_j^q to cluster l. Multi-dimensional cluster is then reconstructed.

Since fcm works with a fixed number of clusters, which is unknown, it is proposed to make fuzzy portraits of different degrees and choose the best one by the criterion (7). For the same value of criterion for different number of clusters the fuzzy portrait with smaller degree is selected.

The major stages of fuzzy portraits generation algorithm for class of images V_j is given below. The parameter C of algorithm is the current degree of fuzzy portrait.

1. C=1 is fixed and fuzzy portrait of the first degree is made with fcm-classic algorithm. $F_{V_j}^{1,0}$ is calculated by (7).
2. The degree of fuzzy portrait C is incremented by one point.
3. The fuzzy portraits of C-degree are made for every D_q^j, $q = 1,\ldots,m$. $F_{s_j}^{cq}$ is calculated by (7).
4. The best fuzzy portrait is chosen:

$$F_{s_j}^{c} = \max_{q=0}^{m}(F_{S_j}^{cq}). \quad (12)$$

5. The dynamic of $F_{s_j}^{c}$ changes with C increasing is estimated. If the quality of fuzzy portraits is decreasing with C increasing then the algorithm is stopped, else steps 3 to 5 are repeated.

It is necessary to use a modification of sliding method [12] to create the membership function that takes weights of initial set elements from c-partition R_j into account.

6. ILLUSTRATIVE EXAMPLE

Let us consider the algorithm's performance on wine classification problem from UCI repository [13]. We have to solve problem of wine identification on the basis of chemical analysis data (like alcohol, malic acid, nonflavonoid phenol concentration, color intensity and other). It is a known set of data of chemical analysis of wines grown in the same region in Italy but derived from three different cultivars.

The problem using the approach described in part 5 of this paper based on fcm approach.

Fuzzy portraits of degrees 1 through 4 were created with different types of operations F_{AND} : F_1, F_2, F_3, F_4 .Table 2 shows values for F_{V_1} for different types of F_{AND} and different number of clusters created by fcm-classic.

Table 2. – Class V_1 (m-dimensional clustering)

clusters	F_2	F_1	F_3	F_4
1	0.79	0.58	0.65	0.54
2	**0.84**	0.67	0.72	0.65
3	0.82	0.62	0.67	0.60
4	0.69	0.43	0.47	0.39

Table 3 shows values for F_{V_1} for different types of F_{AND} and different number of clusters created by fcm-projection algorithm.

Table 3. – Class V_1 - (clustering by projection)

Clusters	F_2	F_1
1	--	--
2	0.84;**0.85**	0.65;0.68
3	0.8;0.82	0.61;0.63
4	0.7;8.1	0.37;0.62

From experiment results the best fuzzy portraits with maximum of local validity measure were selected.

Moreover, it was ascertained that quality of pattern recognition depends directly on the fuzzy portraits quality calculated by (7). It was demonstrated that clustering by projection increased the quality of pattern recognition and decreased computational complexity for this particular type of problems where clustering is due to variations in the technological process, i.e. to the change only in some attributes.

The algorithm without clustering proposed in produced about 8% of errors on the model data with distinct interpretation of result. After clustering of every class of training sample two clusters inside the first class were obtained and the modified algorithm improved the recognition rate to 96%.

7. THE AREA OF APPLICATION FOR THE CURRENT APPROACH. REMARKS AND RECOMMENDATIONS

The main idea of current approach is finding qualitative fuzzy portraits that give integral characteristic of a class of images. This approach allows to make fuzzy classifiers with great generalization ability. This approach is used for problems of identification and control of quality in chemistry and foods industries. Results of organoleptic analysis and data from chemistry devices are used as attributes. Such problems often have the property of a priori inseparability and large noisiness.

To improve pattern recognition quality we have to use different types of fuzzy clustering algorithms, make extra transformation of initial set and find bad points. [8] uses clustering algorithms to find bad points.

In pattern recognition problems it is often sufficient to have the answer: “This image does not belong to this class”. In such cases it is enough to use only those rules from the knowledge base that relate to particular classes to receive the necessary answer. This approach is important for the situation when the system can not answer the question “What class does this image belong to?” either because of the intersection of several classes, or, in a situation when this class of images is not described in the knowledge base.

In addition, we present an algorithm of fuzzy portraits creation on the basis of possibilistic clustering D-AFC(c) algorithm which has the following major stages.

1. Fuzzy portraits for every class of training sample are created without clustering. A semantic rule based on the frequency of object occurrence for every attribute is used for term-set forming [14].

2. The possibilistic clustering of pattern classes is conducted using D-AFC(c) algorithm of fuzzy clustering for every attribute q which produces fuzzy coverage $R_j^q(X)$ for every class j . The best fuzzy coverage $R_j(X)$ is determined through solving the equation $R_j(X) = \arg\max_{\forall R_j^q(X)} F(R_j^q(X))$, where $F(R_j^q(X))$ is the criterion of fuzzy allotment quality:

$$F(R_j^q(X)) = \sum_{w=1}^{c_j^q} \frac{1}{n_w} \sum_{i=1}^{n} \mu_w^q(x_i) - \alpha \cdot c_j^q , \qquad (13)$$

where n_w is the number of elements on the fuzzy cluster w. The D-AFC(c) algorithm is described in [8].

3. If there are adequate fuzzy coverages they are used to adjust fuzzy portraits.

The fuzzy inference is carried out using the generated knowledge base of fuzzy productions. For every fuzzy portrait S_j one fuzzy inference rule is formed in case of no clusters or several fuzzy inference rules for every cluster in case of successful clustering. The linguistic variables values are calculated with fuzzy portraits terms $T_q : \mu_q^{(w)}(\bar{x}) = \mu_q^{(w)}(x_q)$ on fuzzification stage of fuzzy inference algorithm. The aggregation stage is carried out using F_2 (6).

On it a defuzzification procedure can be carried out to obtain a distinct result if need be.

The discrete fuzzy set $\tilde{y}$ defined on the set of classes of images is the result of fuzzy inference algorithm. On it hardering operation can be carried out to obtain a distinct result if need be.

8. CONCLUSION

In conclusion let us note certain features of the proposed algorithm. The main idea of this paper is data presentation model by fuzzy portraits making for pattern recognition problem decision.

This approach is useful for pattern recognition problems with great number of classes. The result of the recognition procedure is given in the form of a fuzzy set which, in many cases, is more natural for the expert, who makes a final decision on whether the pattern belongs to a class and therefore can be easily integrated into decision-making support system.

The independent consideration of individual attributes on different stages of the algorithm allows to avoid preliminary normalization of training samples and introducing distance measure which is usually done when processing multidimensional data. That is why it is proposed to extract main information about dissimilarity of classes component-wise using projection on attributes

The proposed variation of the algorithm using the fuzzy portraits created on the basis of individual attributes can be expanded further through using not only conventional one-dimensional projections, but also two-, three-dimensional etc. projections of original images as basic data subsets. Ideology of fuzzy portraits construction will be preserved. It is quite natural to call the one-dimensional variation of the proposed algorithm a first-order algorithm and the variation that takes into consideration not more than m-dimensional projections a k-order algorithm.

The approach described allows to not only solve machine learning problem but to store the results of data analysis procedure in a knowledge base. This method may be used for automatic context expansion in expert systems [14].

The algorithm gave good results on model data with error rate not more than 2% (hardering procedure is used). Algorithm was tested on “wine” problem from UCI data repository [15] with the error rate of 4%.

This approach has already been briefly discussed in [4], but the current article provides a more detailed presentation of the proposed concept.

9. REFERENCES

[1] D.A. Viattchenin, *Fuzzy Methods of Automatic Classification,* Technoprint, Minsk, 2004. p. 219 (in Russian)

[2] Yu.I. Zhuravlev, An algebraic approach to the solution of pattern recognition and identification problems, *Probl. Kibernet,* (33) (1978), pp. 5-68. (in Russian).

[3] V.A. Kozlovskii, A.Ju. Maksimova, Decision of pattern recognition problem with fuzzy portraits of classes, *Artificial Intelligence,* (4) (2010), pp. 221-228. (in Russian)

[4] V.A. Kozlovskii, A.Yu. Maksimova, Algorithm of pattern recognition with intra-class clustering, *Proceedings of 11th International Conference “Pattern Recognition and Information Processing”, Minsk, Belarus* 18-20 May 2011, pp. 54-57.

[5] M. Schlesinger, V. Hlavac, *Ten Lectures on Statistical and Structural Pattern Recognition,* Springer, 2002, p. 544.

[6] J.C. Bezdek, J.M. Keller, R. Krishnapuram, N.R. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing,* Springer Science, New York, p. 776.

[7] S.A. Aivasian, V.M. Buchstaber, *Applied statistics. Classification and reduction of dimensionality,* Moscow, 1989, p. 608. (in Russian)

[8] F. Klawonn, R. Kurse and H. Timm, *Fuzzy Shell Cluster Analysis.* [http://public.fh-wolfenbuettel.de/~klawonn/Papers/klawonnetaludine97.pdf]. University of Magdeburg. Magdeburg, Germany. pp. 1-15.

[9] M. Friedman, A. Kandel, *Introduction to Pattern Recognition: Statistical, Structural, Neural and Fuzzy Logic Approaches,* World Scientific Publishing Company. Singapore. 1999.

[10] H. Ishibuchi, T. Nakashima, M. Nii, *Classification and Modeling with Linguistic Information Granules: Advanced Approaches*

to Linguistic Data Mining, Springer, 2004. p. 318.

[11] H. Ishibuchi, K. Nozaki and H. Tanaka, Distributed representation of fuzzy rules and its application to pattern classification, *Fuzzy Sets and Systems*, (52) (1992), pp. 21-32

[12] V.A. Kozlovskii, A.Yu. Maksimova, Pattern recognition algorithm based on uzzy approach, *Artifical Intelligent,* (4) (2008), pp. 594-599 (in Russian).

[13] A. Frank, A. Asuncion, *UCI Machine Learning Repository* [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science. 2012.

[14] D.A. Viattchenin, A direct algorithm of possibilistic clustering with partial supervision, *Journal of Automation, Mobile Robotics and Intelligent Systems,* (3) 1 (2007), pp. 29-38.

[15] A.Yu. Maksimova, O.O. Varlamov, Mivar expert system for pattern recognition on the basis of fuzzy classification and data domains modeling with automatic context expantions, *Izvestia YuFU. Technics,* (12) 125 (2011), pp. 77-87.

***Aleksandra Maksimova**, received MSc degree (2004) of Computer Science from Donetsk National University. She is currently a PhD candidate with the Institute of Applied Mathematics and Mechanics. Primary research interests are pattern recognition, data mining, fuzzy clustering.*

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

DIVIDING OUTLIERS INTO VALUABLE AND NOISE POINTS

Vladimir E. Podolskiy

Computer Systems and Networks Department, Bauman Moscow State Technical University,
5, vtoraya Baumanskaya st., 105005, Moscow, RUSSIA, v.e.podolskiy@gmail.com, http://iu6.bmstu.ru

Abstract: *A great number of different clustering algorithms exists in computer science. These algorithms solve the task of dividing data set into clusters. Data points which were not included into one of these clusters are called 'outliers'. But such data points can be used for the discovery of unusual behavior of the analyzed systems. In this article we present a novel fuzzy based optimization approach for division these outliers into two classes: interesting (usable for solving the problem) outliers and noise.*

Keywords: *fuzzy sets, outlier analysis, data classification.*

1. INTRODUCTION

In most of the computer science papers notions of noise and outliers are used interchangeably. But when we talk about some kind of data that is very rare among all the data acquired in some research or observation we must take into consideration that real-life observations are based on physical detectors. Thus we may have a few incorrect data points among all the data acquired. Modern algorithms do not make any difference between these erroneous points (or noise) and valuable data [1]. These algorithms are either used to find all the outliers and throw them out or used to find all the outliers and process them.

In the following paper the problem of distinguishing valuable and rare data points from noise points is considered. In order to solve the task posed in the next section a set of different techniques briefly described in section 3 was used. Sections 4 and 5 describe an approach and an algorithm to solve the problem posed. Section 6 shows the results of algorithm's work on one small example. In section 7 some information on comparison is given. Conclusions about possible further developments in this field are made in section 8.

2. PROBLEM DESCRIPTION

Clusterization is a process of dividing data into groups named 'clusters'. One must distinguish clusterization from classification as classification requires knowledge about classes in which data must be divided (e.g. taxonomy classes, political parties). Each data set can be divided into two parts. The first one is the data that can be clusterized. The second one consists of the data points which cannot be put into any of clusters by the means of methods used. Currently existing algorithms were primarily developed to divide data set into clusters or to detect outliers. In this paper we address the problem of handling such outliers and dividing them into two classes. The following terms are needed for our approach description.

Term 1. Interesting point is an outlier which differs from the rest of the data in clusters and can be useful for the person whose primary goal is to detect the facts that do not fall upon restrictions of the laws which describe data clusters.

Interesting points form a class of interesting points, i.e. one of those two classes briefly mentioned before.

Term 2. Noise point is an outlier which differs very little from the rest of the data in clusters and is of no use to the person analyzing outliers.

Noise just represents some fluctuations and errors. No useful information can be obtained from noise in the analysis conducted. Noise points form noise class, the second of the two classes mentioned in the beginning of this paper.

The difference between interesting points and noise can be easily understood by a human (e.g. in the case of two dimensions), but not a computer. To divide outliers into the classes briefly described above we propose usage of basic instruments of fuzzy logic, clustering algorithms, and optimization methods. It is evident, that such a division cannot be performed without previous clusterization of data because outliers division into the classes of interesting points and noise points requires knowledge about the data distribution and some of the data parameters.

The problem of dividing outliers into the classes of interesting and noise points arises in different fields of study. This problem's solution can be very useful for many applications, such as: geology (distinguishing minerals from ore), fraud detection (distinguishing fraudulent usage of credit cards from a mere change in a customer behavior), search for unusual patterns in researches (distinguishing exceptions from the noise provided by sensors and detectors), search for critical errors in equipment functionality (distinguishing critical issues from minor problems), distinguishing valuable information from background noise (SETI program, cosmology, telescope images), etc.

It can be seen, that the problem stated in the beginning of the paper has high level of topicality for different issues. Of course, this problem requires efficient and logical way of solving. A possible approach to solving this problem is shown in the paper. Following section covers some basic assumptions needed for understanding of the approach developed.

3. BASICS OF THE APPROACH

In order to make the approach proposed work all the data must be clusterized and divided into clusters and outliers. DBSCAN is one of the algorithms for solving this subtask. Let us briefly describe DBSCAN.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm. This algorithm grows regions with high density into clusters and discovers clusters of arbitrary shape in spatial database with noise. DBSCAN defines a cluster as a maximal set of density-connected points.

In order to provide understandable description of the algorithm some definitions must be presented.

The neighborhood within a radius *ε* of a given object is called the ε-neighborhood of the object.

If the ε-neighborhood of an object contains at least a minimum number, *MinPts*, of objects, then the object is called a core object.

Given a set of objects, *D*, we say that an object *p* is directly density-reachable from object *q* if *p* is within the ε-neighborhood of *q*, and *q* is a core object.

An object *p* is density-reachable from object *q* with respect to *ε* and *MinPts* in a set of objects, *D*, if there is a chain of objects $p_1,...p_n$, where $p_1 = q$ and $p_n = p$ such that p_{i+1} is directly density-reachable from p_i with respect to *ε* and *MinPts*, for $1 \le i \le n$, $p_i \in D$.

An object *p* is density-connected to object *q* with respect to *ε* and *MinPts* in a set of objects, *D*, if there is an object $o \in D$ such that both *p* and *q* are density-reachable from *o* with respect to *ε* and *MinPts.*

Density reachability is the transitive closure of direct density reachability, and this relationship is asymmetric. Only core objects are mutually density reachable. Density connectivity is a symmetric relation.

A density-based cluster is a set of density-connected objects that is maximal with respect to density-reachability. Every object not contained in any cluster is considered to be noise.

The concept that underlies DBSCAN is rather simple. This algorithm searches for clusters by checking the ε- neighborhood of each point in the database. If the ε-neighborhood of a point *p* contains more than *MinPts*, a new cluster with *p* as a core object is created. DBSCAN then iteratively collects directly density-reachable objects from these core objects, which may involve the merge of a few density-reachable clusters. The process terminates when no new point can be added to any cluster.

Figure 1 illustrates DBSCAN's work.

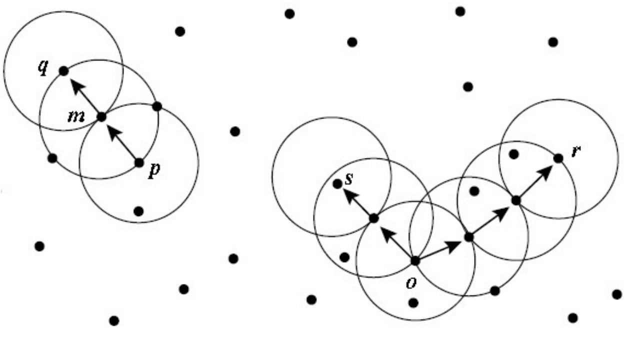


Fig. 1 – Density reachability and density connectivity in density-based clustering

In order to find values of the membership function for each of the outlier points we need to use one of the optimization approaches. Here a brief description of the interior-point method is provided.

Actually, **Interior point methods** (or barrier methods) are a certain class of algorithms to solve linear and nonlinear convex optimization problems.

The basic elements of the method consist of a self-concordant barrier function used to encode the convex set. Contrary to the simplex method, it reaches an optimal solution by traversing the interior of the feasible region.

In constrained optimization, a field of mathematics, a barrier function is a continuous function whose value on a point increases to infinity as the point approaches the boundary of the feasible region. It is used as a penalizing term for violations of constraints. The two most common types of barrier functions are inverse barrier functions and logarithmic barrier functions.

The class of primal-dual path-following interior point methods is considered the most successful. The primal-dual method's idea is easy to demonstrate for constrained nonlinear optimization. For simplicity consider all-inequality version of a nonlinear optimization problem:

$$\begin{gathered}\text{Minimize } f(x) \text{ subject to } c(x) \geq 0 \\ x \in R^n, c(x) \in R^m\end{gathered} \tag{1}$$

The logarithmic barrier function associated with (1) is

$$B(x,\mu) = f(x) - \mu \ln(c(x)) \tag{2}$$

Here μ is a small positive scalar, sometimes called the "barrier parameter". As μ converges to zero the minimum of $B(x,\mu)$ should converge to a solution of (1).

The barrier function gradient

$$g_b = g - A^T \mu / c(x) \tag{3}$$

Where g is the gradient of the original function *f(x)* and the matrix A is the constraint *c(x)* Jacobian.

In addition to the original ("primal") variable x we introduce a Lagrange multiplier inspired dual variable λ

$$c(x)\lambda = \mu \tag{4}$$

(4) is sometimes called the "perturbed complementarity" condition, for its resemblance to "complementary slackness" in KKT conditions.

We try to find those (x_μ, λ_μ) which turn gradient of barrier function to zero. Applying (4) to (3):

$$g - A^T \lambda = 0 \tag{5}$$

Applying Newton's method to (4) and (5) we get an equation for (x,λ) update (p_x, p_λ):

$$\begin{pmatrix} W & -A^T \\ \Lambda A & C \end{pmatrix} \begin{pmatrix} p_x \\ p_\lambda \end{pmatrix} = \begin{pmatrix} -g + A^T \lambda \\ \mu l - C\lambda \end{pmatrix} \tag{6}$$

W is the Hessian matrix of *f(x)* and Λ is a diagonal matrix of λ.

Because of (1), (4) the condition $\lambda \geq 0$ should be enforced at each step. This can be done by choosing appropriate α:

$$(x,\lambda) \to (x + \alpha p_x, \lambda + \alpha p_\lambda) \tag{7}$$

So, it seems that interior point method is good-enough for our approach.

4. OUTLIERS DIVIDING ALGORITHM

The main part of our approach is the algorithm that divides all the outliers into two classes described in section 1 of this paper. We need to make a few assumptions about data set in order to simplify our task.

Some basic assumptions about the data for the algorithm developed:

1. Data has numerical type, i.e. it can be presented as a point in a linear space;
2. Data has been divided into two sets by the means of the data clustering technique (e.g. DBSCAN, OPTICS, DENCLUE, etc [2]). Results of the division are as follows:
   - *C* data clusters;
   - *M* outliers.

The first assumption has been included for the simplicity of experiments with the algorithm and can be removed. The second assumption is necessary because developed algorithm needs information about clusters of data in order to clearly specify for which cluster current point is a noise or an interesting point. Simple but good fuzzy based algorithm for purpose of outlier detection can also be found in [3]. Unfortunately, this algorithm can only be used to detect outliers but not to divide them.

As it is seen, outliers can be both interesting points and noise simultaneously, so we need to use some basic methods of fuzzy sets theory in order to conclude in which class current outlier must be included. This is a task for the classification algorithm described further in this section. We have decided to use fuzzy sets and optimization methods to classify outliers into two described classes. We need a few more definitions to present our algorithm.

Let $Cl_i = \{X_1, X_2, ..., X_{num(i)}\}$ be the i^{th} ($i = \overline{1, C}$ as was defined in the assumptions above) cluster found by a clusterization algorithm (e.g. DBSCAN). *num(i)* is a number of data points in the i^{th} cluster. Let us also assume that we are working with *D*-dimensional data, i.e. each point can be presented as a point in a linear space with *D* coordinates: $X_j = \left(x_1^{(j)}, x_2^{(j)}, ... x_D^{(j)}\right)$ where $j = \overline{1, N}$ X_j is a j^{th} point, and *N* is a total number of data points (including outliers).

Definition 1. Let us assume that point X_k is an outlier, and Cl_i is a cluster closest to this point (i.e. Euclidean distance from this point to the center of this cluster is the smallest one compared to the distances from this point to other clusters). Then the point X_j in the cluster Cl_i is called a projection of the point X_k if the distance between $X_j^{(i)}$ and X_k is lesser than any distance between X_k and every other point of this cluster, i.e.

$$p^{(i)}(X_k) = \\ = \left\{X_j^{(i)} : dist\left(X_j^{(i)}, X_k\right) = \min_{j=\overline{1,num(i)}} dist\left(X_j^{(i)}, X_k\right)\right\}$$

The *dist* function can be defined as Euclidean distance.

Let us also assume that each outlier (X_k) belongs to noise class of one of the clusters (Cl_i) with some degree $\mu_i(X_k) \in [0;1]$ and to the class of interesting data points with degree $\theta_i(X_k) = 1 - \mu_i(X_k)$. This is the only element (also called 'membership function' [4]) of fuzzy sets theory that we use in the paper for dividing outliers into two classes. A criterion based on a value of membership function for dividing outliers into two classes will be described further in the paper.

Definition 2. Minimal distance between the outlier X_k and the cluster Cl_i is a distance between outlier and its projection on this cluster, i.e.: $d_{\min}\left(X_k, Cl_i\right) = dist\left(X_k, p^{(i)}\left(X_k\right)\right)$.

Let us assume that we have found values of membership functions for every outlier X_k and its closest cluster Cl_i, i.e. we know values of $\mu_i(X_k)$ and $\theta_i(X_k)$ for the point X_k, and we also know the minimal distance $d_{\min}\left(X_k, Cl_i\right)$ between outlier point X_k and its closest cluster Cl_i. Let us also define a decision boundary for the i^{th} cluster as:

$$bnd_i = \frac{\sum_{k=1}^{q} \mu_i(X_k) \cdot d_{\min}(X_k, Cl_i)}{\sum_{k=1}^{q} d_{\min}(X_k, Cl_i)}, \qquad (8)$$

where *q* is a number of outliers closest to the i^{th} cluster.

Then we can use following criterion to decide which class current outlier belongs to:

$$dec(X_k) = \begin{cases} \text{noise, if } \mu_i(X_k) > bnd_i, \\ \text{interest. point, otherwise.} \end{cases} \qquad (9)$$

This criterion is the most suitable for the current task.

Now let us describe a general idea behind the algorithm and then provide a full description of our approach. For simplicity we will use this notation: $\mu_k^{(i)} = \mu_i(X_k)$.

The idea that was partially taken from [5] is as follows. We can substitute classification task with a task of maximizing *C* linear functions of q_i variables (q_i is a number of outliers closest to the i^{th} cluster, such that $\sum_{i=1}^{C} q_i = M$), i.e. we must find global maximum for each of the following functions:

$$f_i(\mu_1^{(i)}, \mu_2^{(i)}, ..., \mu_{q_i}^{(i)}) = \sum_{k=1}^{q_i} \frac{\mu_k^{(i)}}{d_{\min}(X_k, Cl_i)}, \qquad (10)$$
$$i = \overline{1, C}.$$

We characterize structure of the i^{th} cluster with a parameter R_i that can be computed as a radius of the i^{th} cluster, i.e. it equals half of a maximum distance between two points in this cluster. In order to take into consideration structure of the cluster, we must also define constraints on production of value of membership function and minimal distance between outlier and the closest cluster:

$\mu_k^{(i)} d_{\min}(X_k, Cl_i) \le R_i, k = \overline{1, q_i}, i = \overline{1, C}$

The system is written for i^{th} cluster where $i = \overline{1, C}$:

$$\begin{cases} f_i(\mu_1^{(i)}, \mu_2^{(i)}, ..., \mu_{q_i}^{(i)}) \to \max \\ \mu_k^{(i)} \cdot d_{\min}(X_k, Cl_i) \le R_i, k = \overline{1, q_i}. \\ 0 \le \mu_k^{(i)} \le 1, k = \overline{1, q_i} \end{cases} \quad (11)$$

We have C such systems. It is evident, that we must use optimization methods that take constraints into consideration (e.g. method of interior point or Lagrange multipliers method). For numerical solution we decided to use method of interior point [6] (penalty functions method) described earlier in this paper.

Interesting situation occurs when outlier is a valuable point for all the data clusters. It is then can be considered as a so-called *global valuable point*. It can be also shown that these global valuable points require special treatment because they can belong to the class of error as well. Let us briefly discuss this problem.

If we have many global valuable points (i.e. their frequency of appearance in dataset is large) then it is highly possible that detector (device that provides us with the data) is broken. If the frequency is not so large then it occurs that there are some super-exceptional data points that are to be considered first. The formal criterion for deciding whether the device providing data is broken or not is as follows:

$$\begin{cases} num_{cl\_\text{int}erest}(X_j) = C, \\ \dfrac{w_g}{M} \ge 0.5, j = \overline{1, w} \end{cases} ., \quad (12)$$

where C is a total number of data clusters, $num_{cl\_\text{int}erest}(X_j)$ is an amount of classes for which the data point in parenthesis is valuable point, w_g is an amount of global valuable points, M is a total amount of outliers, w is a total amount of valuable data points.

We can also consider the following value:

$$P(X_j) = \frac{num_{cl\_\text{int}erest}(X_j)}{C}, \quad (13)$$

Due to discrete nature of valuable (interesting) points we can only consider discrete values like (13) for $j = \overline{1, w}$. But we can also construct some approximation based on the values $P(X_j)$ and X_j. This can be done by means of interpolation techniques. Construction of such hyperplane can help us to predict amount of classes for which the point given is a valuable point. This can be used to consider the problem discussed earlier (see eq. (12)).

5. OUTLIERS PARTITIONING APPROACH

Now we can fully and consequently describe our simple approach:

1. Divide data set into C clusters by one of the clustering techniques (e.g. DBSCAN that can handle clusters of arbitrary form).
2. For every outlier find closest cluster.
3. For every cluster and every outlier bound to this cluster (closest cluster, see step 2) find projection of outlier on this cluster.
4. For every outlier and its projection find outlier's minimal distance to the closest cluster (steps 3 and 4 can be unified based on the definition of projection given in this paper).
5. For every i^{th} cluster define starting point for optimization method (e.g. $\mu_1^{(i)} = 0, \mu_2^{(i)} = 0, ..., \mu_{q_i}^{(i)} = 0$) that satisfies initial conditions of system (11).
6. For every cluster find solution for system (4).
7. For every cluster compute decision boundary using formula (8).
8. For every cluster and outlier closest to cluster decide (based on the criterion (9)), whether outlier under consideration belongs to the noise class or to the interesting points.

This approach is rather simple and modulate. For example, we can use another data clustering technique or another decision criterion.

6. ILLUSTRATIVE EXAMPLES

In this section we apply the approach described above to sets of data points defined further in this paragraph. These data sets are distributions of points on a two-dimensional plane. First random distribution can be easily divided into two groups by clustering algorithm (in our realization we have used DBSCAN [7]). The second one can be divided in three classes. For test purposes application MATLAB® was used. MATLAB code for the first distribution described above is as follows:

```
X=[randn(30,2)*.4;randn(40,2)*.5
              +
      ones(40,1)*[4 4]];
```

We have changed coordinates of nine points in this set in order to convert them to outliers. Coordinates of these points are provided in table 1. Third column presents the class of each outlier defined by approach from section 5.

Table 1. Outliers and classes.

Point №	X coord.	Y coord.	Class
1	11.00	10.00	Interesting
2	2.00	4.00	Noise
3	0.43	4.00	Noise
4	-1.00	-1.33	Noise
5	15.00	17.00	Interesting
6	33.00	4.00	Interesting
7	-3.00	-5.00	Noise
8	-3.55	4.98	Noise
9	9.00	4.30	Noise

Figure 2 illustrates our example. Symbol '*' denotes point from 1st cluster, '+' – from 2nd. Symbol '○' is reserved for noise, and symbol 'X' denotes interesting points (number 1, 5, and 6 in table 1).

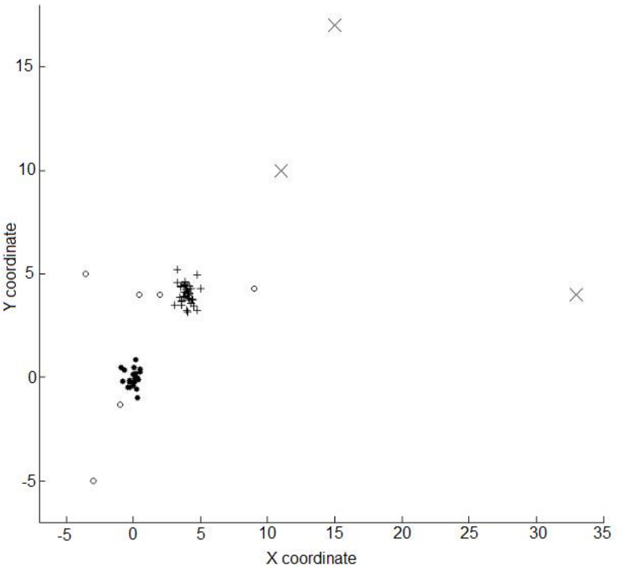


Fig. 2 – Results of applying approach to the test data set #1

The second example shows us that form of clusters has no effect on results of application of approach developed. Repetitive experiments on the data show that slight change in placement of outliers can result in changing classes for all outliers (e.g. from noise to interesting). This problem occurs only for small data sets. For larger data sets with many more outliers this problem with robustness influences results very little. In table 2 we present data for second example.

Table 2. Data for second example.

X	Y	Class	X	Y	Class
0.500	0.500	1	2.375	1.750	2
0.625	0.625	1	2.500	1.750	2
0.625	0.500	1	2.375	1.625	2
0.500	0.625	1	2.500	1.625	2
0.750	0.750	1	2.500	1.500	2
0.500	0.750	1	0.250	3.500	3
0.750	0.500	1	0.375	3.500	3
0.750	0.625	1	0.250	3.625	3
0.625	0.750	1	0.375	3.625	3
0.875	0.875	1	0.500	3.625	3
0.750	0.875	1	0.625	3.625	3
0.875	0.750	1	0.750	3.625	3
0.875	0.625	1	0.875	3.625	3
0.625	0.875	1	1.000	3.625	3
0.500	0.875	1	0.500	3.750	3
0.875	0.500	1	0.625	3.750	3
1.000	1.000	1	0.750	3.750	3
0.875	1.000	1	0.875	3.750	3
0.750	1.000	1	1.000	3.750	3
0.625	1.000	1	0.875	3.875	3
0.500	1.000	1	1.000	3.875	3
1.000	0.500	1	1.125	3.875	3
1.000	0.625	1	1.250	3.875	3
1.000	0.750	1	1.125	4.000	3
1.000	0.875	1	1.250	4.000	3
2.000	2.000	2	1.500	4.500	Noise
2.125	2.000	2	0.000	2.000	Int. p.
2.250	2.000	2	1.750	0.750	Noise
2.375	2.000	2	1.875	5.000	Noise
2.500	2.000	2	1.750	3.000	Noise
2.125	1.875	2	2.750	0.500	Noise
2.250	1.875	2	3.000	0.000	Noise
2.375	1.875	2	1.000	8.000	Int. p.
2.500	1.875	2	0.000	5.000	Int. p.
2.250	1.750	2	1.500	1.500	Noise

Figure 3 illustrates this example.

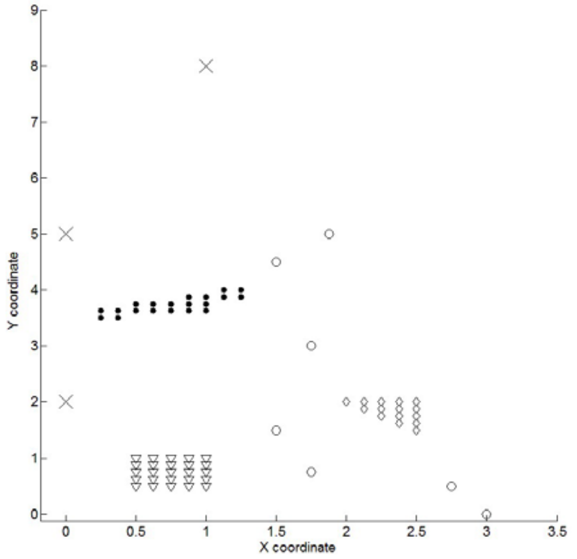


Fig. 3 – Results of applying approach to the test data set #2

For these simple examples we can imagine membership function as a mesa in three-dimensional space, where third coordinate is a value of the membership function. Points of the clusters are on the plain part of the mesa. Noise points lie higher on mesa's slopes than the boundary value. Interesting points are under the boundary value on the slopes. These illustrations give us some basic insights on this approach.

7. COMPARISON

Unfortunately, it seems that the problem addressed in this paper almost hasn't been considered in other papers. So, no other similar approaches have been found because almost all of the papers in the field of AI consider only the problem of clusterization at large.

The approach provided has time complexity $O(n^2)$. This approach is rather unique because we do not use clusterization technique to extract outliers from data and then two-class classifiers on the outliers extracted. Such method will fail because it does not consider clusters at all. But the approach provided in this paper is based on the cluster characteristics. And therefore we can easily divide outliers into noise and valuable points.

8. CONCLUSION

The problem addressed in the paper is interesting for many reasons and can be met in such fields as: handling research data, fraud detection, search for minerals, etc. The approach proposed is mathematically convenient, simple and modulate. Developed approach can be used in its initial form or it can be modified for the sake of optimization and simplicity. There are a great number of algorithms that can be used for solving such a problem, but described approach can lead to simple mathematical expressions. The approach presented is also very intuitive, simple and flexible in comparison to other approaches (neural networks classification, SVM-based classification, decision trees classifiers, some fuzzy based algorithms, etc). So, this approach also briefly introduced in [8] is important and can be used for variety of tasks. Further researches in this field may be conducted in order to obtain sustainable and useful solutions for the problem stated in the beginning of the paper.

9. REFERENCES

[1] F. Rehm, F. Klawonn, R. Kruse, A novel approach to noise clustering for outliers detection, *Soft Computing,* (11) 5 (2007), pp. 489-494.

[2] J. Han, M. Kamber, *Data Mining: Concepts and Techniques,* Morgan Kaufmann Publishers, USA, 2006, 743 p.

[3] D. Viattchenin, Direct algorithms of fuzzy clustering based on the transitive closure operation and their application to outliers detection, *Artificial Intelligence,* (3) (2007), pp. 205-216. (in Russian)

[4] A. Kaufmann, *Introduction to Fuzzy Sets Theory,* Radio and Contact, Moscow, 1982, 432 p. (in Russian)

[5] D. Viattchenin, *Fuzzy Methods of Automatic Classification,* Minsk, Technoprint, 2004, 219 p. (in Russian)

[6] M. Gautam, R. Levkovitz, *Interior Point Methods for Linear Programming Optimization: Theory and Practice,* John Wiley & Sons, USA, 1996, 300 p.

[7] M. Daszykowski, B. Walczak, D.L. Massart, Looking for Natural Patterns in Data. Part 1: Density Based Approach. Chemom. Intell. Lab. Syst. (56) (2001), pp. 83-92.

[8] V.E. Podolskiy, Simple fuzzy based optimization approach to the problem of dividing outliers into classes of valuable and noise points, *Pattern Recognition and Information Processing (PRIP'2011): proceedings of the 11th International Conference (18-20 May, Minsk, Republic of Belarus),* Minsk: BSUIR, 2011, pp. 176-179.

Vladimir Eduardovich Podolskiy*, student at Computer Systems and Networks Department of Informatics and Control Systems faculty at Bauman Moscow State Technical University. Areas of scientific interest include: fuzzy systems, fuzzy methods in sociology, data mining, clusterization, parallel computations, global politics, and human psychology.*

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

FACE RECOGNITION SYSTEM FOR MACHINE READABLE TRAVEL DOCUMENTS

Sergey Tulyakov [1], Rauf Kh. Sadykhov [2]

Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus
sergei.tulyakov@gmail.com, rsadykhov@bsuir.by

Abstract: *This paper presents an upright frontal face recognition system, aimed to recognize faces on machine readable travel documents (MRTD). The system is able to handle large image databases with high processing speed and low detection and identification errors. In order to achieve high accuracy eyes are detected in the most probable regions, which narrows search area and therefore reduces computation time. Recognition is performed with the use of eigenface approach. The paper introduces eigenface basis ranking measure, which is helpful in challenging task of creating the basis for recognition purposes. To speed up identification process we split the database into males and females using high-performance AdaBoost classifier. At the end of the paper the results of the tests in speed and accuracy are given.*

Keywords: *face detection, face recognition, eigenface basis ranking measure, gender determination.*

1. INTRODUCTION

During two last decades the technologies of face detection and recognition have attracted considerable research interest. These technologies are used in biometric systems, border control systems, video surveillance, human-computer interface, access control systems, face expression recognition, content based image retrieval.

The human ability to recognize faces is remarkable. We can easily recognize faces that have seen only several times. Such skills are quite robust to time, face expression, pose, and other distractions such as presence or absence of beard, mustache, glasses, traces of injuries and surgical operations.

However, the tasks of face detection and recognition are serious problems for automation due to many factors such as orientation, imaging conditions, presence or absence of structural components such as beards, mustache, glasses [1].

Methods to detect faces can be divided into four main groups: knowledge-based methods, feature invariant approaches, template matching methods and appearance-based methods [1]. Knowledge-based methods use expert knowledge to encode typical face. Feature invariant approaches are aimed to find facial features even when imaging conditions (pose, viewpoint or illumination) vary. Template matching methods store several standard patterns to describe a whole face or facial features separately.

Nowadays the appearance-based methods gained most popularity due to their hit rate and speed [1]. These methods don’t use a priori knowledge in the data that is present on the image. Instead, they try to extract different variation modes of the database and provide a set of subclasses which represent them best. Appearance-based methods are based on scanning the input image on different scales with fixed window size in order to find faces. Each window then is given as an input to some classifier, trained to separate two classes of objects (face/non-face).

There are many models that can be used as a classifier. To separate faces and non-faces Osuna et al. [2] and later Romdhani et al. [3] use approach based on support vector machines. Schneiderman propose to apply a statistical method to the problem of face detection [4]. Rowley et al. [5] present a face detection system based on artificial neural network. This approach has become an early standard in face detection.

Later Viola and Jones present a much faster detector than any of their contemporaries [6]. They use rectangular features instead of using pixels directly. The reason for this is that feature-based systems operate much faster. In order to compute rectangular features very fast they introduce an intermediate representation for the image called integral image. This integral image is used as an input for the cascade of weak classifiers. As a learning algorithm they use AdaBoost learning [7].

Our approach is based on cascade of weak classifiers. In order to increase detection speed and hit rate we introduce some heuristics that will be discussed in the next sections. Recognition stage is done using eigenface approach [8].

To speed up the identification process of a particular individual we split the database into groups according to gender information. There are several known algorithms capable of determining gender with high performance and accuracy. Moghaddam and Yang in [10] use support vector machines to classify gender with the use of thumbnail images. In [11] AdaBoost classifier is applied to features proposed by Viola and Jones in [6] to detect faces. We have chosen the approach proposed by Baluja and Rowley in [12] for simplicity and efficiency: gender can be classified with the comparison of a small number of pixels in the image. The authors have run various experiments to prove that their approach outgoes competitors in performance and accuracy.

2. SYSTEM OVERVIEW

Our system works with Machine Readable Travel Documents (MRTD). MRTD is an official document, conforming to the specifications contained in Doc 9303 [13].

MRTD portraits must conform the quality requirements contained in Doc 9303 (e.g. the portrait shall be color neutral showing the applicant with the eyes open and clearly visible; the portrait shall show the eyes clearly with no light reflection off the glasses and no tinted lenses).

The proposed face recognition program consists of several stages (see Fig. 1).

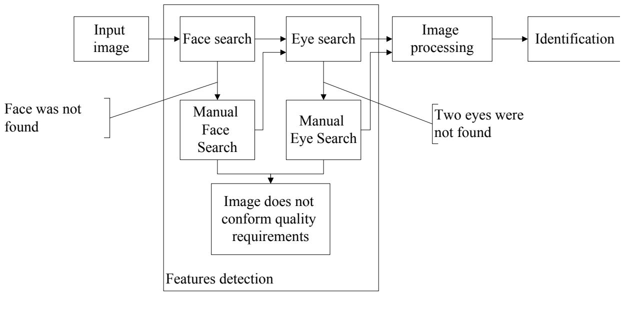


Fig. 1 – Structure of the proposed computer face recognition system

After input image is loaded, system starts feature detection stage. This stage consists of face search and eye search. If system failed to find face or eyes on image it prompts the user to mark them manually. If the user didn't manage to find features, then the image either does not conform to quality requirements or is not a portrait at all. When this stage is successfully completed, system moves to the next stage, where the image processing starts. This stage ends up with storing the image prepared for identification in the database. Identification stage uses known images to determine the most similar individual to the query one.

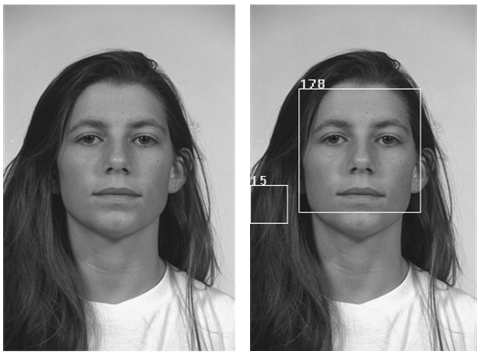
3. FEATURES DETECTION

The aim of this stage is to find candidates to be faces and to find eyes on them. To detect features the proposed system uses cascade classifier introduced by Viola et al. [6] along with extended set of Haar-like features proposed by Lienhart et. al. [14].

As mentioned before appearance-based methods scan input image at some scale levels with a fixed-size window. So, if region represents a face, it is found several times. If we know exactly that there is only one face in the image, we should choose region with maximum number of neighbors.

Image B in Fig. 2 contains two face-candidates. Left one was found 15 times, whereas right one was found 178 times (white label in the upper-left corner of the rectangle). Thus, as we know, that there is only one face on MRTD, we should consider right face-candidate to be a face. Alternatively, if we use "find the biggest face" strategy, we would also be able to reject false hit.

We have run experiments in order to determine which approach is better and "the biggest face approach" showed six times higher speed, while accuracy remains the same.



Source image
A)

Face-candidates found
B)

Fig. 2 – Face-candidates found

In order to increase speed of eye search we narrow the region of interest. We have computed regions on image that most likely will contain eyes. These regions are determined using masks that are applied to face area. This idea helps significantly increase detection speed. If eyes were found inside these regions the system proceeds to next step. Here we can also apply "the biggest eye" approach. But our experiments showed that in terms of hit rate it is better to search for eye regions that have most neighbours.

If there were no eyes in predicted regions found this can mean the following: eyes are closed or incorrectly covered with glasses, image has low quality or image is rotated. The correction of image slope is discussed in the next section.

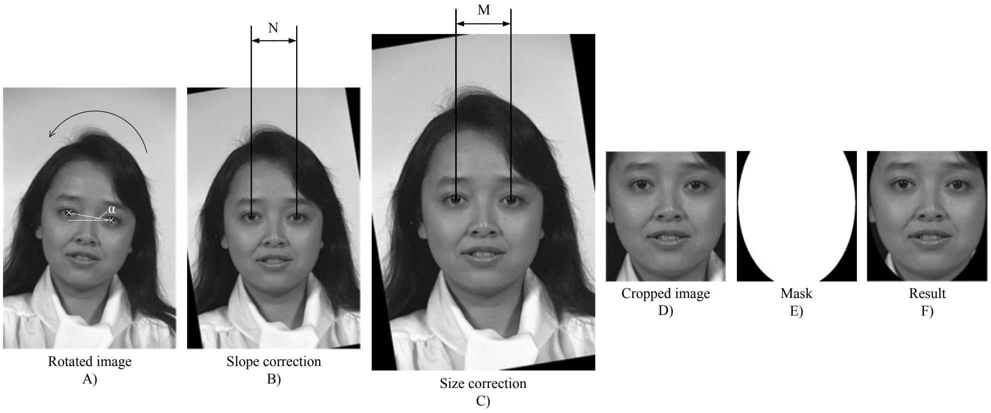


Fig. 3 – The sequence of image transformations

Due to the fact that appearance-based methods scan face at different scales, the scale parameter influences the detection speed and accuracy. By saying accuracy we mean the precise detection of face and eye point: the closer the centre of the eye rectangle is to the pupil of the eye the greater is the accuracy. Big scale parameter yields low accuracy and high speed, low scale parameter – high accuracy and low speed. However, only a small number of images require low scale parameter, most of the images are rapidly and accurately recognized even with high scale parameter.

This tells us to use variable scale parameter for rapid detection on easy images, while allowing accurate detection on complicated ones. We set separate boundaries for the value of scale parameter for face detection and eye detection. Detection starts with the highest value, which is subsequently lowered to the minimum value, only if the object was not found. This approach dramatically improved accuracy. Moreover, images being treated as not containing faces with fixed parameter were successfully recognized with scale parameter. These results are shown in the Table 1.

Table 1. Comparison of Variable and Fixed Scale Parameter

	Fixed			Variable
	1,4	1,25	1,1	
True positive rate	0,55	0,72	0,91	0,98
Speed, image/seconds	8,45	8,13	5,11	7,52

4. IMAGE PROCESSING

The goal of this step is to adjust images so that: imaginary line drawn between eyes is parallel to the top edge of the image, distance between eyes is set to a certain value and image is cropped and masked.

First, each image is rotated (see Fig. 3 A). Fig. 4 illustrates three images that were initially rotated by 23 (A), 24 (B) and 25 (C) degrees and corrected by the system. Imaginary line drawn between eyes on images B and C is not parallel to top edge of the image. So, we can expect our system to be able to correct image rotations up to approximately 23 degrees. Since MRTD are not likely to be rotated by more than 23 degrees the ability of the system to correct slope is sufficient.

Second, the distance between eyes is measured, and every image is resized to set this parameter to a certain value. This is done to ensure that all images have the same size in pixels (Fig. 3 B, C).

Let us consider that all images in our database have M pixels between the eyes. So, in order to be able to recognize new face we need to correct the size of the new image using the factor M/N, when new face has N pixels between eyes.

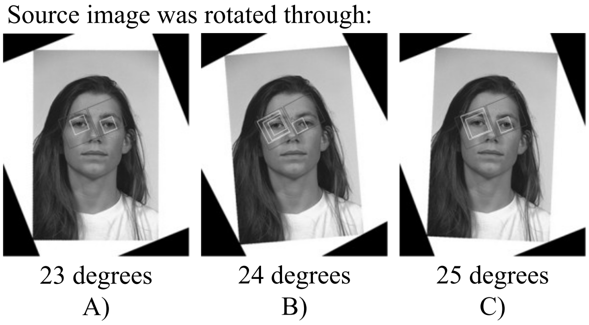


Fig. 4 – Maximum slope correction

Next, each image is cropped to reduce image size. Image C on the Fig. 3 will be stored in the database in order to help the user of the system identify person manually. Image F is used to identify person automatically.

At this step we warp image using only eye regions. If we consider resizing image using both x and y axis, we change the proportions of the face and that will probably decrease recognition abilities of the system. After this stage is complete we have a set of extracted faces.

5. IDENTIFICATION

The goal of this step is to find individuals which are the most similar to the input individual. In our work we use eigenfaces approach [8]. The reason is that this approach is able to handle large amounts of data. Face image is considered to be a two-dimensional N by M array. An image can also be treated as a vector of dimension $N \cdot M \times 1$. Thus, each image is considered to be a point in the $N \cdot M$-dimensional subspace. Face images, being similar in overall configuration, will not be randomly distributed in this huge image space. In order to decrease dimensionality principal component analysis (PCA) is used. This approach consists of the following steps: (i) compute the average face vector Ψ, (ii) obtain a set of face vectors Φ centered at expectation Ψ, (iii) compute the covariance matrix and find its K eigenvectors with the biggest eigenvalues and (iv) project each image onto new subspace to obtain coordinates of each image in this subspace.

Average face can be computed as follows:

$$\Psi = \frac{1}{n}\sum_{i=1}^{n} F_i, \qquad (1)$$

where F is training set of face vectors, n – number of faces in training set. Then, centered at expectation set of face vectors is $\Phi_i = F_i - \Psi$. Average face Ψ is computed using faces of people of different race, gender, age, color, with and without glasses, beards and mustache.

Our experiments showed that if we increase n, average face remains almost the same. The Euclidean distance between mean faces based on 100 and 200 images is 4.01; 200 and 300 images – 3.3; 300 and 400 images – 0.66; 400 and 500 images – 0.58.

Covariance matrix is determined as follows:

$$C = AA^T, \qquad (2)$$

where $A = (\Phi_1 \Phi_2 \ldots \Phi_n)$ is $N \cdot M \times n$ matrix, and C is $(N \cdot M)^2$ matrix. Matrix C is too large to compute its eigenvectors ν_i. Turk et al. [8] propose to compute ν_i as follows:

$$\nu_i = \mathrm{A}\mathrm{u}_i, \qquad (3)$$

where u_i are eigenvectors of $n \times n$ matrix $A^T A$. Now each face can be represented as a linear combination of first K eigenvectors as follows:

$$\Phi_i = \sum_{j=1}^{K} w_j \nu_j, \qquad (4)$$

where $w_j = \nu_j^T \Phi_i$. Fig. 5 shows image decomposition to mean face and eigenfaces with biggest weights.

In our work we use Euclidean distance to classify an image:

$$\varepsilon_k^2 = (\Omega - \Omega_k)^2, \qquad (5)$$

where Ω_k is a vector describing kth face class which minimizes ε_k^2. A face is classified as belonging to class k when the minimum ε_k^2 is below some threshold θ. Otherwise, image is unknown to the system and can be added to the database.

As it seen from the Fig. 5 image is treated as a vector. The coordinates can be regarded as a compact image representation.

This representation is not lossless and the exact value of loss depends on the basis. So, the more precise the basis will be the less noise the representations will contain and different images will more likely to have different coordinates. Our hypothesis is that the basis having smaller average loss over images will show greater recognition performance. The derivation of this loss function is given below.

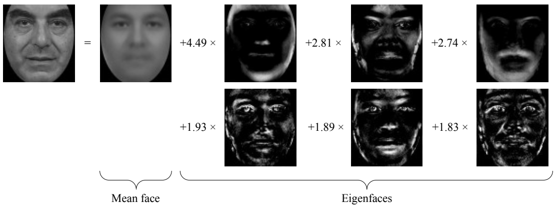


Fig. 5 – Image decomposition to mean face and eigenfaces with biggest values

Let A be a set of faces which are used to construct basis. Let B be a set of faces which are used to test basis. $|A|, |B|$ are the numbers of images in defined sets respectively. Let $L(\nu)_i$ be a loss vector:

$$L(\nu)_i = (B_i - B'(\nu)_i)^2, \qquad (6)$$

where $B'(\nu)_i$ – face images restored from basis, ν – basis:

$$B'(\nu)_i = (\nu^T W_i) + \Psi, \qquad (7)$$

where W_i– representation of ith image in the basis ν, Ψ – mean face.

The target function can be written down as follows:

$$e(\nu) = \frac{1}{|B|}\sum_{k=1}^{|B|} L(\nu)_k. \qquad (8)$$

The task to minimize $e(\nu)$ is practically impossible. However, this target function can be used to compare existing eigenface bases. To illustrate this idea let us build two different bases: ν_{A1} consists of people of all races, gender, with or without mustaches (image set $A1$); ν_{A2} – white male faces without mustache (image set $A2$). Each set of images consists of 100 images.

As expected, mean face computed from the first set of images (see Fig. 6, A) contains features of all races that were included into initial set of images. Mean face computed from the second image set is a face of a white male (see Fig. 6, B). As a test set we used a set of 45 images that consists of males and females of different races. Images from this set are not present in set $A1$ and $A2$.

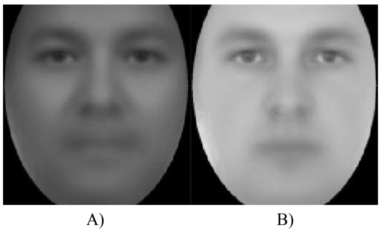


Fig. 6 – Two average faces

The following results were obtained: $e(\nu_{A1}) = 232587$ and $e(\nu_{A2}) = 421466$. Since $e(\nu_{A1}) < e(\nu_{A2})$ we can expect basis ν_{A1} to restore faces more precisely than basis ν_{A2}. Other experimental results are given in section 0.

6. GENDER DETERMINATION

One of the goals of our system is to handle indeed large MRTD databases. Although data storage optimizations, eager evaluation and repeated calculations caching reduce search time, full scan of the database takes much time. In order to obtain the preliminary result earlier, we divide the database into males and females using AdaBoost classifier [7]. It determines the gender of a particular person using only relatively small number of pixel comparisons [9]. Detailed description of this approach can be found in [12]. We will outline the main steps of the algorithm.

We extract faces so that the distance between the eyes is 10 pixels, pupils of the eyes are situated in points $(5,5)$ and $(10,5)$. The whole face must occupy the area with dimensions of 20×20 pixels. We use five types of comparisons along with their inverses. All together they represent weak classifiers:

1. $p_i > p_j$
2. $|p_i - p_j| \leq 5$
3. $|p_i - p_j| \leq 10$
4. $|p_i - p_j| \leq 25$
5. $|p_i - p_j| \leq 50$

where p_i, p_j are two pixels in the image, $i,j = \overline{1{,}400}, i \neq j$.

Although the total quantity of weak classifiers in the image with dimensions 20×20 is 1596000, AdaBoost is capable of selecting and combining a small number of weak classifiers into the strong classifier, having significant accuracy. The first features selected by AdaBoost are shown in Fig. 7.

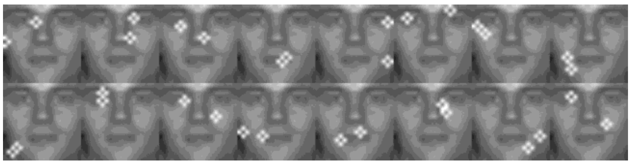


Fig. 7 – First pixel pairs (weak classifiers) selected by AdaBoost

We have configured AdaBoost to select 1000 features and combine them into one strong classifier. The dependence of classification accuracy on the number of weak classifiers is given in the Fig. 8**.**

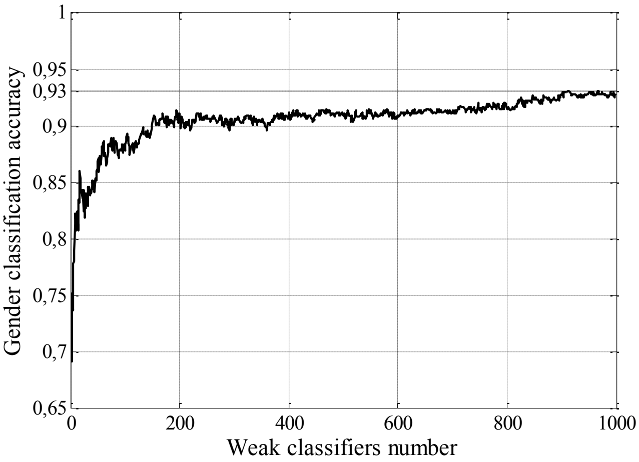


Fig. 8 – Accuracy of gender classifiers consisting from 1 to 1000 weak classifier. The highest accuracy is achieved with 911 weak nodes.

The best classifier consists of 911 nodes and gives 0,93 accuracy. We use this classifier to split the database into two groups and to determine gender of the unknown individual. ***Even though these percentiles are rather high, we cannot guarantee 100% accuracy. Because of that we use grouping of the individuals in the database to receive the preliminary searching result as fast as possible. This preliminary result in 93% cases will be identical to the final result. For example, if an individual is male, the system will start search in males group outputting results in real time. Due to small error equal to 0.07 it will proceed to search within females group, but the interim results will obtain with spending up to 40% less time.***

7. EXPERIMENTAL RESULTS

We used frontal faces in the COLOR FERET[15][16] database to build bases. We used the rest of the FERET database (those images that were

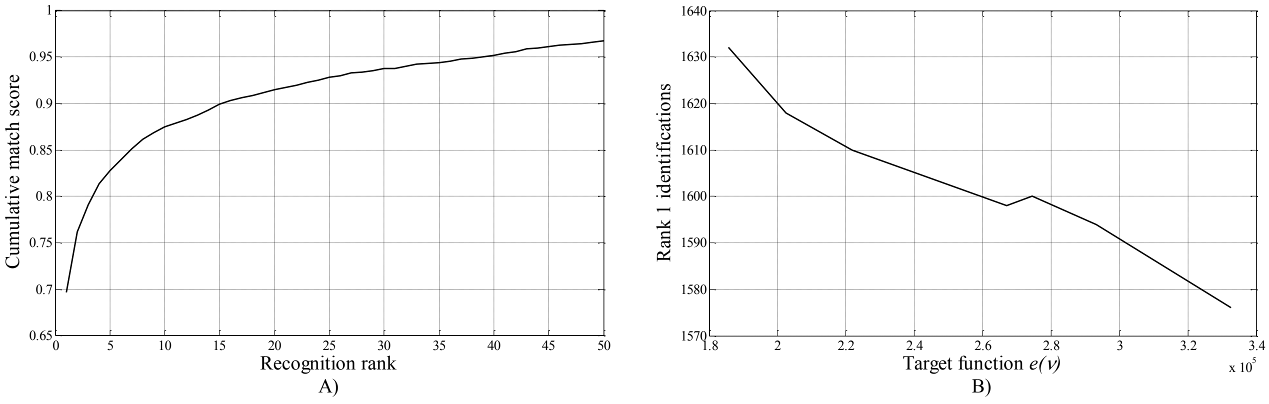


Fig. 9 – Recognition system performance (A), computed with the use of 11886 images. The dependence of the number of rank 1 true positives on the value of target function (B), computed with the use of 1983 images from the FERET database

not used to create bases) with additional 9903 frontal facial images to test the system. The total number of images made up 11886. We tried to locate individuals with facial expressions different to query image. In our previous work [17] we achieved 0.3 seconds per query over 11886 images database, which is quite promising. After splitting the database into two groups the preliminary results are ready after 0.18 seconds.

The cumulative match score of the system is shown on the Fig. 9 (A). The recognition rank at the position *N* denotes the ratio of correctly identified individuals that were placed on the *N-th* position by similarity to the query image.

Fig. 9 (B) shows the dependence of the number of rank 1 true positives upon the $e(\nu)$, derived in the previous section. Since relative values do not show significant difference, the graph shows absolute values. However, when the recognition system is to be applied in the critical environment, e.g. criminals recognition, it is worth tuning the basis to produce as much rank 1 identifications as possible. Computation of the exact number of rank 1 identifications is time consuming, while computing $e(\nu)$is very fast.

8. CONCLUSIONS AND FUTURE RESEARCHES

The described system was implemented on the C++ programming language with the use of Open Source Computer Vision Library.

Variable image scale parameter helps increase speed and detection accuracy, allowing the system handling large images databases. Image processing sequence brings each face to the same state regardless of initial rotation and size. The proposed “the biggest face” approach along with regions that most likely contain eyes significantly increase the speed of feature detection.

The proposed method of basis ranking allows choosing the best one which is useful for more successful recognition process.

Gender-based identification allows getting preliminary results taking up to 40% less time.

One of the future research directions is to use additional criteria to create more groups in the database based on race, age, or face geometry.

9. ACKNOWLEDGMENT

Portions of the research in this paper use the FERET database of facial images collected under the FERET program, sponsored by the DOD Counterdrug Technology Development Program Office.

10. REFERENCES

[1] M. Yang, Recent advances in face detection, *Tutorial, IEEE International Conference on Pattern Recognition*, Cambridge, 2004.

[2] E. Osuna, R. Freund and F. Girosi, *Support Vector Machines: Training and Applications*, Massachusetts Institute of Technology, Department of Brain and Cognitive Sciences, 1997.

[3] S. Romdhani, P. Torr, B. Scholkopf and A. Blake, Computationally efficient face detection, *In Proceeding of the 8th International Conference on Computer Vision*, vol. 2, Vancouver, 2001, pp. 695-700.

[4] H. Schneiderman, *A Statistical Approach to 3D Object Detection Applied to Faces and Cars*, Dissertation Thesis at Carnegie Mellon University, Pittsburgh, 2000.

[5] H. Rowley, S. Baluja and T. Kanade, Neural network-based face detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (20) 1 (1998), pp. 22-38.

[6] P. Viola and M. Jones, Robust real-time face

detection, *International Journal of Computer Vision*, (2004), pp. 137-154.

[7] Y. Freund and R. Shapire, A decision-theoretic generalization of on-line learning and an application to boosting, *Journal of Computer and System Sciences*, (1996) pp. 119-139.

[8] M. Turk, and A. Pentland, Eigenfaces for recognition, *Journal of Cognitive Neuroscience*, (3) 2 (1991), pp. 71-86.

[9] S. Baluja, M. Sahami, and H. Rowley, Efficient face orientation discrimination, *International Conference on Image Processing*, 2004.

[10] B. Moghaddam and M. Yang, Learning gender with support faces, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (PAMI), (24) 5 (May 2002).

[11] G. Shakhnarovich, P. Viola and B. Moghaddam, A unified learning framework for real time face detection and classification, *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, 2002

[12] S. Baluja and H. Rowley, Boosting sex identification performance, *International Journal of Computer Vision*, (71) 1 (2007).

[13] *International Civil Aviation Organization. Machine Readable Travel Documents*, Doc 9303, 2006.

[14] R. Lienhart and J. Maydt, An extended set of Haar-like features for rapid object detection, *In Proceedings of The IEEE International Conference on Image Processing*, pp. 900-903.

[15] P.J. Phillips, H. Moon, S.A. Rizvi and P.J. Rauss, The FERET evaluation methodology for face recognition algorithms, *IEEE Trans. Pattern Analysis and Machine Intelligence*, (22) (2000), pp. 1090-1104.

[16] P.J. Phillips, H. Wechsler, J. Huang and P. Rauss, The FERET database and evaluation procedure for face recognition algorithms, *Image and Vision Computing J.*, (16) 5 (1998), pp. 295-306.

[17] S. Tulyakov and R. Sadykhov, Face recognition on machine readable travel documents: algorithms and results, *In Proceedings of International Conference on Pattern Recognition and Information Processing*, 2011.

***Sergey Tulyakov,** received his BS degree with honors in computer science from Belarusian State University of Informatics and Radioelectronics in 2009. In 2010 received master degree from the aforementioned university. He is currently a PhD student at Belarusian State University of Informatics and Radioelectronics. His research areas include image processing, computer vision and machine learning.*

***Rauf Kh. Sadykhov,** habilitated doctor in computer science since 1991. Vice-chairman of the Belarusian office of the International Society of neural networks, IEEE member, member of the Belarusian Association of Image Processing. His research directions include: digital signal processing, pattern recognition and image processing, artificial intelligence, machine learning, parallel architectures for digital signal and image processing. He is an author and co-author of more than 360 scientific papers, including 3 monographs. Currently he is a head of Computer Systems Department at Belarusian State University of Informatics and Radioelectronics.*

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

INFLUENCE LEARNING FOR MULTI-AGENT SYSTEM BASED ON REINFORCEMENT LEARNING

Anton Kabysh [1], Vladimir Golovko [1], Arunas Lipnickas [2]

[1] Brest State Technical University, 224017, Moskovskaya str. 267, Brest, Republic of Belarus
anton.kabysh@gmail.com, gva@bstu.by
[2] Kaunas University of Technology, Student g. 48-111, LT-51367, Kaunas, Lithuania
arunas.lipnickas@ktu.lt

Abstract: *This paper describes a multi-agent influence learning approach and reinforcement learning adaptation to it. This learning technique is used for distributed, adaptive and self-organizing control in multi-agent system. This technique is quite simple and uses agent's influences to estimate learning error between them. The best influences are rewarded via reinforcement learning which is a well-proven learning technique. It is shown that this learning rule supports positive-reward interactions between agents and does not require any additional information than standard reinforcement learning algorithm. This technique produces optimal behavior of multi-agent system with fast convergence patterns.*

Keywords: *reinforcement learning, influence learning, multi-agent learning, multi-joined robot.*

1. INTRODUCTION

Multi-Agent approach – it is an entire paradigm in the development of complex systems consisting of interacting autonomous agents, which are operating with local knowledge and limited capacity, however, can be expected, in general, the behavior of the system. No agent lives in a vacuum, but typically must interact with other agents to achieve its goals. Distributed artificial intelligence is the subfield of artificial intelligence that focuses on complex systems that are inhabited by multiple agents. The main goal of research in this area is to provide principles for the construction and application of multi-agent systems as well as means for coordinating the behavior of multiple independent agents [1].

For example, many application domains are envisioned in which teams of software agents or robots learn to cooperate amongst each other and with human beings to achieve global objectives. Interacting agents may also be essential in many non-cooperative domains such as economics and finance. Teams of agents have the potential for accomplishing tasks that are beyond the capabilities of a single agent. An excellent and demanding example of multi-agent cooperation is in robot soccer. At the same time, Multi-Agent learning (MAL) poses significant theoretical challenges, particularly in understanding how agents can learn and adapt in the presence of other agents that are simultaneously learning and adapting [2,3].

In general multi-agent systems each agent possesses its own, arbitrary reward function which may be entirely unrelated to the rewards other agents receive. For this realm of problems a number of corresponding multi-agent reinforcement learning algorithms have been suggested [4, 5, 7, 8]. The *Nash-Q* learning algorithm shown multi-agent system as general-sum games where agents tend to find Nash equilibrium in common environment. This algorithm assumed an environment where each agent has full knowledge over the actions taken by other agents. Aiming at the reduction of necessary preconditions to be met for this algorithm to be applicable, Littman proposed the *Friend-or-Foe Q* learning algorithm, which, however, requires every other agent to be classified as cooperative or competitive a priori, and Greenwald and Hall developed *Correlated-Q-learning* which generalizes Nash-Q and FoF-Q.

Multi-agent systems with cooperative agents which collectively aim at achieving a common goal is to be found very frequently in practical applications and, thus, of special importance. Some practical usages include [4]:

- Mobile telephony (e.g. for channel allocation)

- Network technology (e.g. for data packet routing)
- Computing power management (e.g. for load balancing across servers)
- Autonomous robots (e.g. for robotic soccer) and distributed robotics.
- Job-shop scheduling [4].

This numeration of mostly real-life applications misses another important area the field of production planning and factory optimization, where machines may act cooperatively with the goal of achieving maximal joint productivity.

The universal multi-joined robot learning problem via multi-agent learning described in this paper. The section 2 describes reinforcement learning approach and section 3 influence learning framework to multi-agent learning. Section 4 describes and eplains particular methodology of adaptation reinforcement learning to multi-agent influence learning. Next sections, 5 and 6 describes model of universal multi-joined and learning experiments. Conclusion contains perspectives, prospects and limitation of presented learning framework.

2. REINFORCEMENT LEARNING

Reinforcement learning is an approach to artificial intelligence that emphasizes learning by the individual from its interaction with its environment that produces optimal behavior [4]. It is often used as one of the control techniques, especially for learning autonomous agents in unknown environment. It emerged at the intersection of dynamic programming, machine learning, biology, studies the reflexes and reactions of living organisms: reflex theory, animal cognition [5]. RL is highly popular for learning autonomous agents, for example autonomous robotics, negotiating agents and so on. The math foundation of RL is Markov Decision Process (MDP), so it widely used for learning in game theory, e.g. TD-Gammon [6].

The core of all most Reinforcement Learning methods is a Temporal Difference (TD) learning [4]. Temporal Difference technique measures the inconsistency between difference of quality for two *actions* done in some *state* and received reward, shows expectation of agent.

Agent execute action *a* in particular state *s*, goes to next state *s'* and receives reward *r* as a feedback of recent action. During learning agent try to select the best action in some state (best action usually more rewarded in future). Visually, iteration of RL-agent on MDP is shown at Fig. 1.

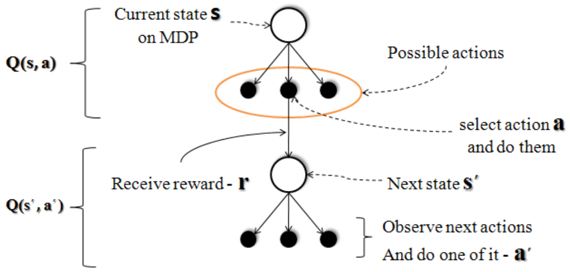


Fig. 1 – One iteration of Reinforcement learning

where α – learning rate, γ - discount factor.

Learning goal is to approximate *Q*-function, e.g. finding true *Q*-values of *Q*-function for each *action* in every *state*. Formulas 1,2 shows *SARSA* learning rule. Estimated value δ is *Temporal Difference error*.

$$\delta = r + Q(s',a') - Q(s,a) \qquad (1)$$

$$\Delta Q(s,a) = \alpha\delta \qquad (2)$$

The natural extension of standard RL algorithm is usage *eligibility traces* to remember previously visited states. Eligibility trace is a temporary record of the occurrence of an event, such as the visiting of a state or the taking of an action [4]. At every time step, when a TD error occurs, only the eligible states or actions are updated.

$$\Delta Q(s,a) = \alpha[r + Q(s',a') - Q(s,a)]e(s) \qquad (3)$$

$$e(s) = \begin{cases} \lambda\gamma e(s) & if\ s \neq s_t \\ \lambda\gamma e(s) + 1 & oterwise \end{cases} \qquad (4)$$

Formula (2) called for every previously visited state *s* if, where $e(s)$ – is a eligibility value, λ – is a eligibility discount factor, decay in time past eligibilities.

Reinforcement Learning works well in case of one agent and where external environment can be represented compactly via *Q*-function. In a multi-agent environment with multiple cooperative agents in general it is difficult to define appropriate state-action spaces for all interacting agents – the *course of dimensionality* problem. For example, in multi-agent system with *N* agents RL algorithm should search in *N*-dimensional state-space. Most often the tiling of the state space has to be rather fine to over all possibly relevant situations and there can also be a wide variety of actions to choose from. As a consequence there exists a combinatorial explosion problem when trying to explore all possible actions from all possible states. Solutions to this problem apply scale-spacing methods and/or function approximation methods to reduce and/or interpolate the searchable value-space.

To solve these problems in this work we use *influence learning framework* to define formal model of agent's interaction.

3. INFLUENCE LEARNING

Multi-Agent learning (MAL) poses significant theoretical challenges, particularly in understanding how agents can learn and adapt in the presence of other agents that are simultaneously learning and adapting [10]. In a distributed system, a number of individually acting agents coexist. If they strive to accomplish a common goal, i.e. if the multi-agent system is a cooperative one, then the establishment of coordinated cooperation between the agents is of utmost importance [2].

Influence – is active offer from one agent to another which can be accepted or not. If influence is accepted, than it affects the agent's behavior and (or) translates it into a new state. If influence is not accepted, than it can be returned to the sender with negative mark. Moreover if influence is accepted by receiver than it also can send a feedback showing how influence was good.

The main idea of influence learning is that good influences should be rewarded and negative should be excluded. Agents should cooperate to achieve the common goal finding in adaptive process best influences to each other. Only way to work together determines whether or not achieved the goal and only final goal determine what influences is valid from the others. Therefore, the adaptive learning process should be used to find best influences and their sequence that are correct for achieving the goal. In this paper we used reinforcement learning for these purposes, but this is not strict.

So, the influence learning – is simple and agile learning framework for cooperating agents in combination with an adaptive algorithm to find the best influences. In other words, influence learning is a *learning to coordinate behaviors*. The advantage of this approach is its universality; any multi-agent systems can be easily described in terms of influences. The limitation of this approach is that the agent should have a connection with each other. In any case, if the connection does not exist, then influence learning is simply reduced to the standard techniques of single agent learning.

4. REINFORCEMENT LEARNING ADAPTATION TO INFLUENCE LEARNING

Let's see how the reinforcement learning can be adapted to influence learning. Typically, in reinforcement learning, agent estimate error temporal difference error between two states and learn. In multi-agent system actions from one agent may be directed to another agents and change their states. The actions in this case will be the influences. If agent state is changed via influence the received reward in new state is transferred to influencing agent. It can lean via reinforcement learning and in these way positive influences is selected among others.

Let's see to interconnected agents *A* and *B* in states s_a and s_b respectively. Agent *A* in state s_a execute action *a* over agent *B*, and put them it into new state s_b. Agent in new state *B* select action *b* and execute it somewhere (on another agent, or on environment). Actions *a* and *b* has their Q-*values* $Q(s_a, a)$ and $Q(s_b, b)$ respectively. This situation is shown at Fig. 2.

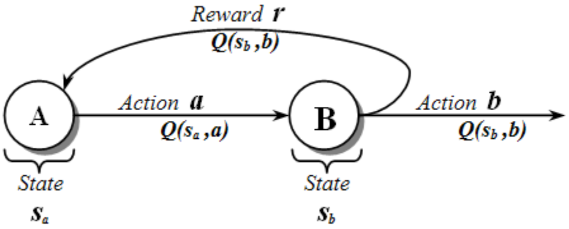


Fig. 2 – Influences between two agents. Reinforcement Learning view

Executing action and receiving reward agent *B* can send this reward as a learning feedback to *A*. This feedback include *Q-value* $Q(s_b, b)$ and reward value *r*. Receiving this feedback agent *A* can calculate *Temporal Difference* error for selected influence (action *a*) and learn. Learning process done using formulas (5, 6).

$$\delta_{AB} = r + \gamma Q(s_b, b) - Q(s_a, a) \qquad (5)$$

$$\Delta Q(s_a, a) = \alpha \delta_{AB} \qquad (6)$$

Formula (5) defines an *influence error* as a *temporal difference error* between agent *A* and *B*. Expression $r + \gamma Q(s_b, b)$ – is a feedback from agent *B*.

The most one important change in I-RL is that we suppose a $Q(s_b, b)$ - is a *"future"* Q-value of agent *A*.

In the end of learning, agent *A* can build the optimal influence selection policy for agent *B*. Feedback between agents included into update rule produces coherence of their behaviors. The advantage of this rule is simplicity and all single-agent RL algorithms can be used without serious changes. Easy to see that update rule is identical to standard reinforcement learning approach.

To support indirect interactions lets include eligibility traces into agent update rule. Introduce one more agent *C***.** This situation is shown at Fig. 3.

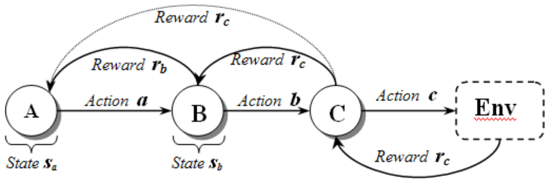


Fig. 3 – Influences between three agents. Reinforcemen learning view

As we can see at fig. 3, there are direct interaction between *A*-*B*, and *B*-*C*, and *indirect* interaction between *A* and *C*.

To support indirect interactions lets introduce *influence value* $i(d)$, where *d* is a *distance* between agents; This distance shows how *structurally* far away produced influence to this agent. For direct interactions *d* is equal to 0; for indirect interaction *d > 0*, depending on how many intermediate influences done between indirect agents. For direct interactions *A*-*B* and *B*-*C* the influence distance *d* equal to 1. For indirect interaction *A*-*C* influence distance is equal to 2, and so on.

The update rule changed in following way:

$$\delta_{AC} = r_c + \gamma Q(s_c, c) - Q(s_a, a) \qquad (7)$$

$$\Delta Q(s_a, a) = \alpha \delta_{AC} i(d) \qquad (8)$$

$$i(d) = \lambda^d \qquad (9)$$

where δ_{AC} - is a influence error between agent A and C, λ – is a coefficient of influence discount factor (0<λ<1).

Influence value $i(d)$ is depends from discount factor λ and reduced with increasing influence distance between agents. If $\lambda = 1$, then all influences will be updated with full power. If $\lambda = 0$ only direct interactions will be updated. If $0 < \lambda < 1$ then indirect interactions will be updated with decay.

5. MODEL OF MULTI-JOINED ROBOT

Multi-Joined Robot (MJR) learning task is a simple decentralized model, where simulate robot arm with N-degrees of freedom, where N – is a number agents in MAS. Every segment – is an intellectual agent learned via Reinforcement Learning with own *Q*-function. The goal of experiment is to learn MJR reach some target point. The goal can be achieved only if all actions were approval. Moreover, this model includes distributed and indirect interactions, which are also worth considering. This problem requires synchronization of local agent's.

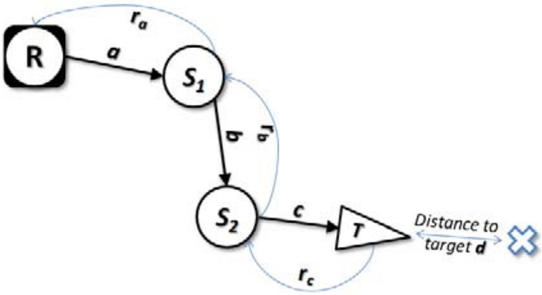


Fig. 4 – Multi-Joined Robot with 4 segments *R, S1, S2, T.* a,b,c **- Agent actions.** r_a, r_b, r_c **– Feedback reward corresponds to actions**

MJR contains one root segment ***R***, several intermediate segments ***S₁***, ***S₂***, ..., ***Sₘ***, and one terminal segment ***T*** connected into chain from ***R*** to ***T*** (Fig. 6). Every segment, excluding terminal, can rotate at full circle (360°) all next segments. At one time step each segment, excluding terminal, can rotate all next segments at 5° to left or right, or do nothing.

First acts root segment ***R***, then first intermediate ***S₁***, then second ***S₂*** and so on, until ***Sₘ***,. Root segment can't move, can't be moved and don't change their position. Terminal segment verify reaching the target and receive actions from previous segments that change their own position.

Every segment – is an intellectual agent learned via reinforcement learning. Goal of multi-agent system is reaching a target grid. After learning MJR must reach by oneself any acceptable target cell of grid world.

Used next learning procedure (one training start):

1. MJR moved to initial position.
2. Every segment selects and executes action in order to structure of MJR. States of all next agents are changed.
3. Terminal segment calculate distance to target point.
4. If target is reached then MJR count grand-prix reward and learned. Go to 1.
5. Else, terminal segment produce feedback reward for previous agent to learn it. Feedbacks are propagated into MJR, so agents learn via RTD until root segment will be reached.
6. If simulation time is ended (1000 simulation steps) go to 1. If average RTD-Error (7) lover than limit value, then learning is over.
7. Next time step. Go to 2

6. EXPERIMENTAL RESULTS

RL parameters include: α (learning rate) = 0.05~0.1; γ (discount factor) = 0.7; λ (eligibility discount factor) = 0.7~0.99, d (influence discount factor) = 0.5~0.7.

In experiments we used MJR with 5 active segments. It is good compromise between model complexity and decentralization of agent actions.

The normal convergence process using *Q-Learning* update rule illustrated at fig 5.

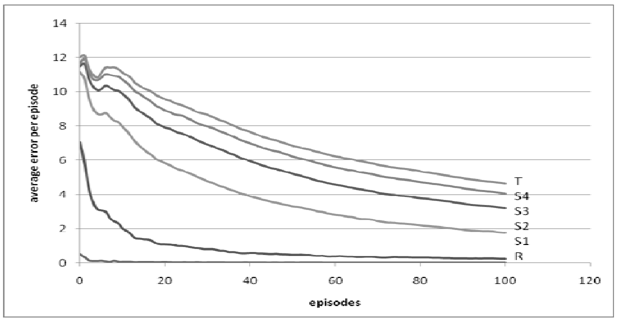


Fig. 5 – Influences between three agents. Reinforcemen learning view

As been shown, there are direct dependencies between agent's actions and errors. Every next segment can be placed at more states (has biggest state-action space), than others, and its error value become higher and convergence is longer.

Compare convergence speed with the overall multi-agent learning technique, *Joint-Action Learners* (JAL) which tries to train all agents at once. In I-RL approach in a single time step the system of learning expression $\{\, \Delta Q_R(s_r, a_R), \Delta Q_{S1}(s_{S1}, a_{S1}), ..., \Delta Q_{R4}(s_{S4}, a_{S4}) \,\}$ is evaluated. In contrast, Joint-Action learning framework try to build the optimal policy overall multi-agent system approximating one joint Q-function $\Delta Q^*(s^* = \{s_R, s_{S1}, ..., s_{S4}\}, a^* = \{a_R, ..., a_{S4}\})$ composing state and action from all agents.

The advantage of I-RL is that it tries to approximate the number of small policies, one per agent instead of one complex. The fig. 6 shows the convergence result of I-RL in comparison with JAL.

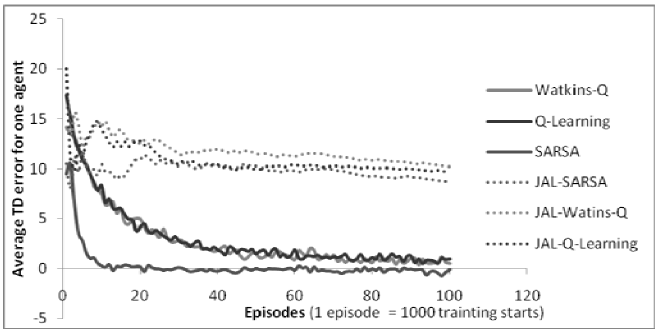


Fig. 6 – Average I-RL error for one agent per episode

Quality of convergence depends from number of segments. If MJR has more than 6 segments then probability of convergence is much lower. The main problem in this case is decentralization of agent actions. For example, if one agent in the beginning of MJR learns unsuccessfully (broken agent), then behavior of all next segments can becomes unstable, if they are can't compensate wrong action from broken agent. Every next agent should be sure that executed on it action is correct, in other words every selected action should guarantee convergence of learning and reaching the target. The described I-RL learning rule should be updated with these properties.

Behavior policy variously changed in way of use different algorithms. RL algorithms with influence tract (SARSA(λ), Watkins-*Q*(λ)) shown more smooth behavior and better synchronization than algorithms without it (*Q*-Learning).

Another unobvious result was seen in robot behavior. For algorithms with eligibility traces robot prefer rotation about a fixed root point with segment reconfiguration on new round to reach the target. Nevertheless, for *Q*-Learning (without eligibility traces) robot prefer reach the target in a straight way.

7. CONCLUSION

It is easy to see that I-RL converges much faster than JAL. This is an illustration of how *the course of dimensionality* affects the convergence of the algorithm. I-RL takes recommendations how to make decomposition of state-action space to much smaller subspaces, where only best influences are rewarded.

This approach helps to solve almost all the difficulties facing the multi-agent learning supportive without losing its simplicity. Also an advantage of this technique is that it can be applied to almost any problem of multi-agent reinforcement learning. The disadvantage of this work is that no comparison was made with more sophisticated multi-agent reinforcement learning algorithms, such us Correlated-Q.

I-RL has two known limitations. One is decentralization problem for long indirect influences, and second one is problem of agent influences insurance. The agent should be sure, that received influences as much optimal, as possible. It is a guarantee of I-RL algorithm convergence.

8. Acknowledgments

This work was supported by a Belorussian-Lithuanian grant F11LIT-003 from Belarusian Republican Foundation for Fundamental Research and Educational Grant GB11/121 from Ministry of Education of Republic of Belarus.

9. REFERENCES

[1] Vidal H.M., *Fundamentals of Multiagent Systems with Net Logo Examples*, E-book: www.multiagent.com, (2008).

[2] Panait L. and Luke S., Cooperative multi-agent learning: the state of the art, *Autonomous Agents and Multi-Agent Systems*, (11) 3 (2005), pp. 387-434.

[3] Alonso E., D'Inverno M., Kudenko D., Luck M., Noble J., *Learning in Multi-Agent Systems. Science Report, Discussion*. UK's Special Interest Group on Multi-Agent Systems, (2001).

[4] Gabel T., *Multi-Agent Reinforcement Learning Approaches for Distributed Job-Shop Scheduling Problems*, PhD Thesis, University of Osnabrueck, 2009.

[5] Bab A., Brafman R.I., Multi-agent reinforcement learning in common interest and fixed sum stochastic games: an experimental study, *Journal of Machine Learning Research*, (9) (2008), pp. 2635-2675.

[6] Richard S. Sutton, Andrew G. Barto, *Reinforcement Learning: An Introduction*, MIT Press., (1998).

[7] Worgotter F., Porr B., Temporal sequence learning, prediction and control – A review of different models and their relation to biological mechanisms, *Neural Computation*, (17) (2005), pp. 245-319.

[8] Tan Ming, Multiagent reinforcement learning. Independent vs cooperative agents, *Autonomous Agents and Multiagent Systems*, (10) 3 (2005), pp. 273-328.

[9] Shoham Y., Powers R., Grenager T., If multi-agent learning is the answer, what is the question, *Journal of Artificial Intelligence*, (2006).

[10] Stone P., Multiagent learning is not the answer. It is a question, *Artificial Intelligence*, (171) (2007), pp. 402-405.

[11] Kabysh A., Golovko V., Mikhniayeu A., Lipnickas A., Behaviour patterns of adaptive multi-joined robot learned by multi-agent influence reinforcement learning, *Proceedings of Pattern Recognition and Image Processing* (PRIP–2011), 2011, 19–21 May, BSU, Minsk, pp. 392-297.

Anton Kabysh *Starting Ph.D. "Learning algorithms of Intelligent Multi-Agent Systems". in Brest State Technical University at 2009. Member of teaching staff, Department of Intelligent Information Technologies, Brest State Technical University. Member of: Laboratory of Artificial Neural Networks; Laboratory of Robotics "BrSTU Robotics" www.robobics.bstu.by.*

Areas of research interests: Multi-Agent Systems, Multi-Agent Learning, Reinforcement Learning, Intellectual Robotics.

Prof. Vladimir Golovko, *received M.E. degree in Computer Engineering in 1984 from the Moscow Bauman State Technical University. In 1990 he received PhD degree from the Belarus State University and in 2003 he received doctor science degree in Computer Science from the United Institute of Informatics problems national Academy of Sciences (Belarus). At present he work as head of Intelligence Information Technologies Department and Laboratory of Artificial Neural Networks of the Brest State Technical University.*

His research interests include Artificial Intelligence, neural networks, autonomous learning robot, signal processing, chaotic processes, intrusion and epilepsy detection.

Dr. Arunas Lipnickas, *graduated in 1996 the faculty of Electrical Engineering and Control Systems, department of Applied Electronics, Kaunas University of Technology. In 1998 graduated with honours gaining the degree of Master of Science in Electrical Engineering at Kaunas University of Technology. In 2002 he obtained PhD degree in Informatics Engineering. At present is a Senior Assoc. Researcher, Mechatronics Centre for Studies and Research, Head of Laboratory of Robotics, Kaunas University of Technology.*

Areas of research interests: artificial intelligence, neural networks, signal processing, decision support systems, fault diagnosis, robotics, mechatronics.

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

MULTI-AGENT PARALLEL IMPLEMENTATION OF PHOTOMASK SIMULATION IN PHOTOLITHOGRAPHY

Syarhei M. Avakaw [1], Alexander A. Doudkin [2], Alexander V. Inyutin [2],
Aleksey V. Otwagin [2,3], Vladislav A. Rusetsky [1]

[1] Research and Production Republican Unitary Enterprise “KBTEM-OMO” of Planar Corporation, 2 Partizansky Ave, Minsk, Belarus, mve@kbtem.avilink.net
[2] United Institute of Informatics Problems of NAS of Belarus, Minsk, doudkin@newman.bas-net.by
[3] Belarusian State University of Informatics and Radioelectronics, 6 P. Brovka st, Minsk, Belarus, otwagin@bsuir.by

Abstract: *A framework for paralleling aerial image simulation in photolithography is proposed. Initial data for the simulation representing photomask are considered as a data stream that is processed by a multi-agent computing system. A parallel image processing is based on a graph model of a parallel algorithm. The algorithm is constructed from individual computing operations in a special visual editor. Then the visual representation is converted into XML, which is interpreted by the multi-agent system based on MPI. The system performs run-time dynamic optimization of calculations using an algorithm of virtual associative network. The proposed framework gives a possibility to design and analyze parallel algorithms and to adapt them to architecture of the computing cluster.*

Keywords: *Aerial Image Simulation, Multi-agent, Integrated Circuit, Photolithography, Parallel Algorithm.*

1. INTRODUCTION

A photolithography process is a major step in conversion of integrated circuit (IC) layout pattern on a surface of a semiconductor wafer. A simulation of the lithography process is very complicated. It can be roughly simulated in two steps:

1) mask shapes are projected into a photoresist as an aerial image;

2) a distribution of an absorbed intensity of a emission in the photoresist are calculated and patterned based on the aerial image intensity.

The image projection is simulated by the Hopkins equation [1], which is a four-dimensional integral. This calculation is too slow to simulate across the full chip and parallel algorithms can be used to speed up the simulation [2,3]. Currently, the known number of photolithography simulation software systems is implemented both on clusters of multiprocessor personal computers and workstations [4-6, 7] and supercomputers [9]. Hardware-accelerated computational lithography tools are also built [8]. The need of solving additional tasks for planning and optimizing the structure of a parallel application in the design process prevents their widespread use.

In many cases, a development of the parallel applications is carried out based on existing sequential algorithms and their composition. Computational operations as parts of the parallel algorithm often have a universal character and can be applied to various problems of information processing. Implementations of the operations in the portable forms allow going to component design, when the program is constructed from large blocks.

In addition, a process of a parallel program execution requires modern methods of planning and optimization of load characteristics of computational nodes. In many cases, the nodes of parallel systems have heterogeneous characteristics, both spatial and temporal. For an effective implementation of parallel programs such systems should provide tools for organizing and monitoring parallel computing and its dynamic reconfiguration.

There are a number of machine vision systems that use parallel and distributed processing [10-13]. Different technologies such as CORBA [13] or a multi-agent approach [10] are used as a architectural core of these systems.

We propose to use for designing and organizing parallel computations an integrated set of tools (framework) that includes a visual editor, a compiler, an optimization system and a parallel computing support system based on MPI [14]. Having framework for design, analysis and planning of parallel applications and built-in specific mechanisms for the implementation of parallelism, IC designer can significantly accelerate the processing flow of photomask and layout images in the operational analysis of IC. MPI makes our system is widely applicable to various parallel computers. The main features of the proposed framework are the following:

1) visual representation of a parallel application based on graph model of a computation. A concept of computational grains is used. The grains are independent modules developed in different programming languages (C + +, Java, MPI) and have a specific interface for integration into a parallel algorithm;

2) a support of a portability, implemented as libraries. The computing grains are added to the system algorithms at a run time;

3) static and dynamic optimization of the parallel applications using an algorithm of virtual associative network (VAN). This algorithm is a kind of hybrid genetic algorithm (GA) and provides a quick search for a solution close to optimal. It has two modifications: the first one is used in the analysis phase of the design (the static optimization), the second one – on the stage of program execution (the dynamic optimization);

4) assignment of operations of the parallel application on the computational nodes (processors) of a computer system, taking into account information obtained during the optimization.

The paper describes the implementation of parallel simulating of image formation in photoresist wafer during photolithography. Init data for processing are the images of the original topology of VLSI photomasks. Results of simulating give a possibility to do a subsequent automatic mask inspection and determine a significance of photolithography defects. The defects of the topology are significant if they lead to the formation of defects on the wafer that should be corrected at the stage of the production.

The task of simulation of an aerial image is calculation the light intensity distribution of the wafer surface and to obtain a latent image with regard to characteristics of the optical system and lighting conditions. Section 2 represents an algorithm that simulates the aerial image on the photoresist. In Section 3 we consider the problem of parallel processing and describe a graph model representation of the parallel algorithm based on concept of computational grains. Section 4 describes the basic elements of the computational platform and their interaction in the development of parallel applications. Section 5 represents an example of an implementation of the simulation algorithm and some experimental results of the optimization process.

2. ALGORITHM FOR SIMULATING AERIAL IMAGE ON THE PHOTORESIST

Algorithm for simulating the image on the photoresist surface is composed of the following steps [15, 16]:

- calculation of a pupillary function;
- calculation of a vector amplitude of the object;
- calculation of a transfer matrix of the projection lens
- calculation of two-dimensional distribution of intensity in a given position of the plane
- calculation of two-dimensional distribution of intensity in different positions of the plane;
- calculation of image intensity in semi-coherent light;
- calculation of the volume distribution of intensity.

The influence of the vector properties of light is taken into account by the so-called vector factors (multipliers) applied to the pupil function. Using the factors allowed describing the influence of the vector nature of electromagnetic waves on the image of thin periodic structures, whose dimensions are within the resolution of optical systems, significantly reducing the computation time of the aerial image.

To describe the effect of an anterior aperture of the optical system it is necessary to use the factor that accounts the diffraction of a plane linearly polarized wave at the input of the optical system. In this case, we consider the effect of the optical system to redistribute the energy in the spectrum regardless of the direction of polarization and the direction of wave propagation. Another factor takes into account the influence of the entrance aperture at the entrance of the optical system. Based on these data, we can simulate the effect of an influence of the numerical entrance aperture to the distribution of any Fourier component exactly as a vector field (a vector nature of electromagnetic waves is considered at the inlet and outlet of the optical system). As a result, it is possible to calculate the vector field of the image both in coherent and partially coherent light.

Using the factors allows describing the influence of a linearly polarized wave to the image quality. For the case non-polarized or partially polarized light it is necessary to use both electric and magnetic vectors which are implemented in the proposed algorithm.

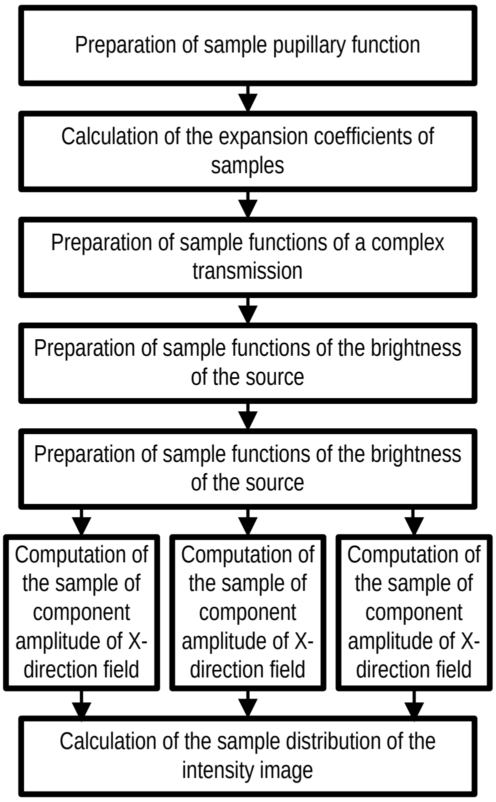

Preparation of sample pupillary function
Calculation of the expansion coefficients of samples
Preparation of sample functions of a complex transmission
Preparation of sample functions of the brightness of the source
Preparation of sample functions of the brightness of the source
Computation of the sample of component amplitude of X-direction field
Computation of the sample of component amplitude of X-direction field
Computation of the sample of component amplitude of X-direction field
Calculation of the sample distribution of the intensity image

Fig. 1 –An algorithm for constructing the aerial image

The main features of the algorithm are the following.

1) Integrating the vector nature of the light field based on the wording of the electric and the magnetic vector of the amplitudes as functions of the three space Cartesian coordinates, as well as two coordinates in the pupil of the optical system. This formulation provides a correct account of the aberrations of the optical system and an influence of high numerical aperture to an image formation without a significant complication of the mathematical apparatus. In contrast to the currently used models, the proposed model is based on a strict conformity physical nature of the processes and much simpler, it is favorable for constructing fast algorithms.

2) Using the partial coherence theory the image intensity is calculated the most economical way based on a system of Eigen functions. Eigen functions are simulated by Zernike polynomials and are used to describe the mutual intensity of different pixels of the image. Such technique significantly reduces the number of integrals over the light source, a calculation of which remains to be the most time-consuming step in the simulation after applying the features mentioned above.

These principles supplement each other in developing the most efficient algorithm for calculating the intensity distribution of the aerial image and do not individually represent a value what they find together. The implementation of the algorithm in the form of single-process applications has shown the following: if the data level is large, then the processing time is unacceptably large (over 1 min / frame). Since all of the layout images are processed by a common program and may be called by a defect detection system at the same time, it is expedient to use a parallel computer system for the simultaneous processing of input data stream.

3. A PARALLEL APPLICATION MODEL AND A COMPUTATIONAL GRAIN CONCEPT

The basic principles of creation a graph-oriented parallel program representation are defined in [17]. The scenarios for data processing are represented in the form of Directed Acyclic Graph (DAG). DAG is represented as a tuple $G = (V, E, W, C)$, where:

V is a set of graph nodes, that represents decomposition of a parallel dataflow processing program on the separated operations $v_i \in V, 1 \le i \le N$;

E is a set of graph edges, that represents a precedence relation between operations in the scenario and determines a data transfer between these nodes, $\{e_{i,j} = (v_i, v_j)\} \in E, i = \overline{1,N}, j = \overline{1,N}, i \ne j$;

W is an operation cost matrix;

C is an edge cost set, where $c_{i,j} \in C$ determines the communication volume between two data processing operations, which is transferred by edge $e_{i,j} \in E$. We consider those operations, which are related and connected by the edge, use an identical data format for a predecessor output and a successor input. For all the scenarios, particular edges have an equal cost.

The development of a dataflow processing application includes the following three stages:

1) creation of a part of DAG scenario that describes logical structure of application;

2) assignment and editing of operations parameters of DAG scenario for each data type;

3) mapping of DAG scenario to cluster architecture.

Each computational operation in DAG scenario is realized as a separate unit called a grain. The grain uses specific interface for integration into framework and data exchange. A design of the grain makes possible a rapid adaptation of existing processing algorithms into a parallel application. These algorithms are transformed to objects that are capable to form their own calling context on the base of received parameters. Each operation interprets its parameter string by convenient way and converts the parameters to a variable name or to a constant value. The order and rules of a parameter transform are determined by an operation specification.

All operations work with a specific data storage mechanism that is incorporated into framework architecture. The storage realizes a shared memory abstraction for source data and results of processing. A variant of shared memory is realized on a shared file system that is common for many cluster architectures. A storage interface provides operations for writing and reading of data. There exists also an intermediate storage mechanism in local memory of each processor, where the results of this processor operation are stored. This one allows reducing time expenses for variable reading in case of repeated access.

The parameters of operation are read from storage. Each parameter is identified by its object name, represented as a string. Each parameter value is placed into corresponding internal grain variable, thus all parameters form a calling context. Further, the operation is executed and results of processing are placed in the storage. At this moment these values are accessible for other grains in parallel application.

The grains are collected in specific libraries that are dynamically linked into the parallel application. The grain is loaded from the library in due time and identified by the operation name. The realization of specific grain libraries from different classes of processing algorithms allows expanding the application area of the proposed framework.

An example of the parallel program graph is presented in Fig.2, where each operation is denoted as *Ci* with a cost vector. A cost of an information transfer between contiguous operations is equal for all data types. Some operations are strictly oriented on a specific processor while others can be placed on each processor in cluster. If the operation cost for some type of data is equal to zero, then this operation must be skipped for the selected type of data.

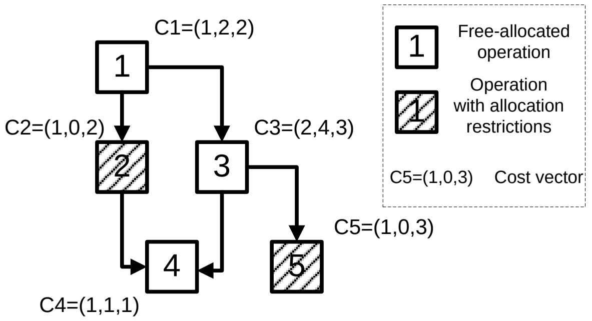


Fig. 2 – An example of program graph and denotation semantic

A matrix of restrictions *Z*(*O*,*P*) is formed according to the following rules:

Z(*O*,*P*) = 1, if processor *p* allows execution of operation *O*;

Z(*O*,*P*) = 0, otherwise.

The matrix of restrictions is used in optimization procedures and prevents an erroneous allocation of the specified operations on some processors. The restrictions arise because of a heterogeneous cluster structure and different operations requirements.

A main task of a multi-agent system is a planning and an optimization of a parallel program execution with a simultaneous provision of reliable computations and guaranteed processing. There exist many algorithms of DAG scheduling that use various optimization techniques and heuristics. The techniques include priority based list scheduling, for example, the algorithms HLF (Highest Level First), LP (Longest Path) and CP (Critical Path) [18-20]. Another technique is a clusterization, and such algorithm, as DSC (Dominating Sequence Clustering) [21, 22] belongs to this technique. However all static scheduling algorithms are constructed mostly for special graph topologies, or use special constraints, such as a zero communication time between nodes or an unbounded number of processors. Because of a stochastic nature of input information the static scheduling approach can not realize an effective optimization for many cases of parallel processing.

Another perspective search techniques use an evolutionary optimization. These techniques are based on such algorithms, as tabu search [23], simulated annealing and genetic algorithms. GA combined with VAN algorithm is the most powerful technique among them. VAN algorithm is based on a concept of associations between the particular operations and dedicated processors [24]. Each operation *O* and processor *P* are associated by means of virtual link of strength $\omega_{O,P}$. Some structure of an associative memory, which consists of the associations, is constructed for optimization. This memory is learned by an experience, accumulated in a solution search process. VAN algorithm is based on GA representation of solutions in a form of population of chromosomes. Each chromosome represents a variant of program graph decomposition.

4. THE FRAMEWORK ARCHITECTURE AND AGENT BEHAVIOR

The architecture of agent framework for parallel processing is presented in Fig. 3.

The framework architecture is based on MPI and allows a fast communication between agents by means of an internal MPI virtual machine. A basic multi-agent structure is presented in Fig. 4.

An input for the multi-agent system is a parallel program graph and a set of data objects that are different by their types. The parallel graph is represented as XML file, which allows specifying all the characteristics of separate operations for all types of data objects.

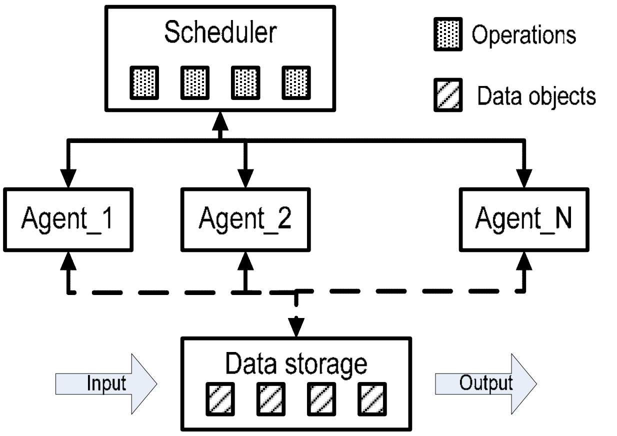


Fig. 3 – The architecture of parallel processing framework

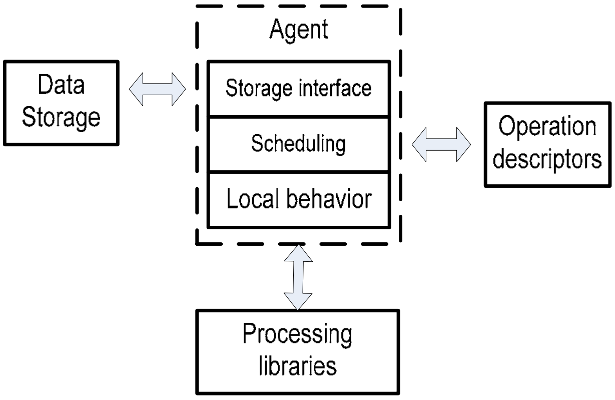


Fig. 4 – The internal structure of agent

Each agent performs parsing of the whole graph and builds internal structures that are used for an execution of specified operations with correct parameters for each object type. These operations are represented by descriptors, which are used by a scheduler to control precedence relations and an overall process.

The data object is represented by a descriptor too. The descriptor contains an identifier, type attributes and some additional information, for example, a name of data file, which contains information for this object. When an operation requires additional data for processing, this descriptor must be extended for specified applications in appropriate way.

As the descriptors are transferred between processors of the parallel application, therefore the application code must contain serialization mechanisms. These mechanisms are realized for an interaction with MPI facilities for messaging. The code is included in a message transfer interface that is extensible and allows the use of alternative message transport systems.

Data objects are stored in a shared data storage that is realized as descriptor storage. Each descriptor is linked with a universal container for storing of different data objects. The storage interface allows interaction with global storage for each agent in the system. This interface has some facilities for an object search, on-demand loading of remote objects and deletion of unused objects from the storage. Each agent has a local copy of the storage and uses it as a write-through cache.

The purpose of the scheduler interface is a processing and a scheduling control. It contains a special component, which is called a scheduler and makes decisions about the next processing operation that must be placed in a descriptor queue. All descriptors of the operations for processed objects are stored in the descriptor storage that contains three sets of descriptors: ready, working and finished pools. The scheduler chooses the next processed operation from the ready pool and sends its descriptor to an appropriate processor agent. After processing this descriptor is placed to the finished operations pool and the information about next stage of processing is changed. The process is repeated while the ready pool is not empty.

For reliability of computations there exists an intermediate working pool of descriptors. This pool is used to mark the descriptors that are now executed by agents. When some agent is broken, then the corresponding descriptor remains in this pool a long period of time. The scheduler periodically checks a descriptor state and moves these waiting descriptors back to the ready pool. The descriptors then have a possibility to allocate on a different working agent.

The agents are dynamically linked up the library of image processing operations. Each processor executes the operations that are specified by the descriptors. The processor receives the descriptor from the coordinator, determines the next operation and executes it using the descriptor data. After a completion of data processing, the descriptor is returned to the scheduler. The processor works while a stop instruction is not received.

Besides the process coordination, the runtime agents check a system state and characteristics. These characteristics are collected and used for a runtime optimization. The optimization is based on a measuring of a data processing speed. When the input data change a system pattern significantly, the system must adapt to this situation. The adaptation consists in a reconfiguration of the operation subsets for all processor agents. The system tries to adapt to changed conditions and to achieve a high processing speed.

The agents use two different policies to choose of next operation from the descriptor pool. The first one consists in choosing operations on the base of agent's preferences. These preferences are formed in a working process by the means of VAN algorithm. Each agent has a vector of weights, and a probability of a selection of operation *O* for this agent A is:

$$P = \frac{\omega_{O,A}}{\sum\limits_{i=1,O} \omega_{i,A}} \cdot Z(O, A). \tag{1}$$

When the agent performs some operation and its performance characteristics are increased, then the corresponding operation weight is corrected according to:

$$\omega_{O,A}(t+1) = \omega_{O,A}(t) + \alpha\,, \tag{2}$$

where α is a learning coefficient.

The weights of the remaining operations are corrected according to:

$$\omega_{O,A}(t+1) = \omega_{O,A}(t) - \alpha/(N-1)\,, \tag{3}$$

where *N* means overall amount of the agents. The agent can choose from a subset of operations, taking first ready operation.

The second choosing policy is a greedy one that consists in choosing of first ready operation from the ready pool. This policy is introduced to eliminate a situation, when some descriptors are not chosen by long time. The greedy agents execute these operations and later they can choose this operations type as preferable. Each agent can switch between two scheduling policies randomly.

5. EXAMPLE APPLICATIONS AND EXPERIMENTAL RESULTS

For simulating we use 20×20 µm topological structure (Fig. 5), where black color indicates the exposed part on positive photoresist UV 210 Shipley with a puncture defect.

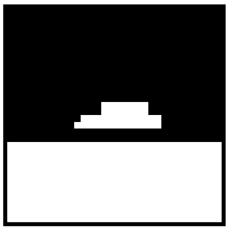


Fig. 5 – Example of topological structure for testing

The parameters for the simulation are: wavelength – 248 nm; generating normalization – 1.0; linear and circular polarization – 0, lens magnification – 0, numerical aperture – 0.6, number of points on the diameter – 19; size of the object – 8.0×8.0 nm, resolution – 0.16, number of points on the object – 50, amplitude transmission – 1.0; index of refraction – 1.0, defocusing within the confines of 1.0-2.0; step defocusing – 0.5.

The results of the simulating are shown in Fig. 6. X-axis is the distance from the center point of the object, the axis Y is the value of the lighting intensity.

Some experiments were conducted to measure a simulation performance depending against the size of the source data and the number of used processors. Checking the result of the simulation was carried out by comparison with the results obtained by leading industry SIGMA C microlithography simulator (Photronics, Inc.) on VLSI layouts with defects according to the standard SEMI-P22-0699 that have been detected EM-6329 [25].

The first group of experiments showed the dependence of the calculation time on the number of processors allocated to run the application (Table 1).

Table 1. The calculation time for different number of processors

Number of objects	Number of processors			
	1	2	3	4
50	77,8	44,67	41,63	24,48
100	154,92	89,23	83,19	49,51
200	310,11	178,47	166,29	101,42

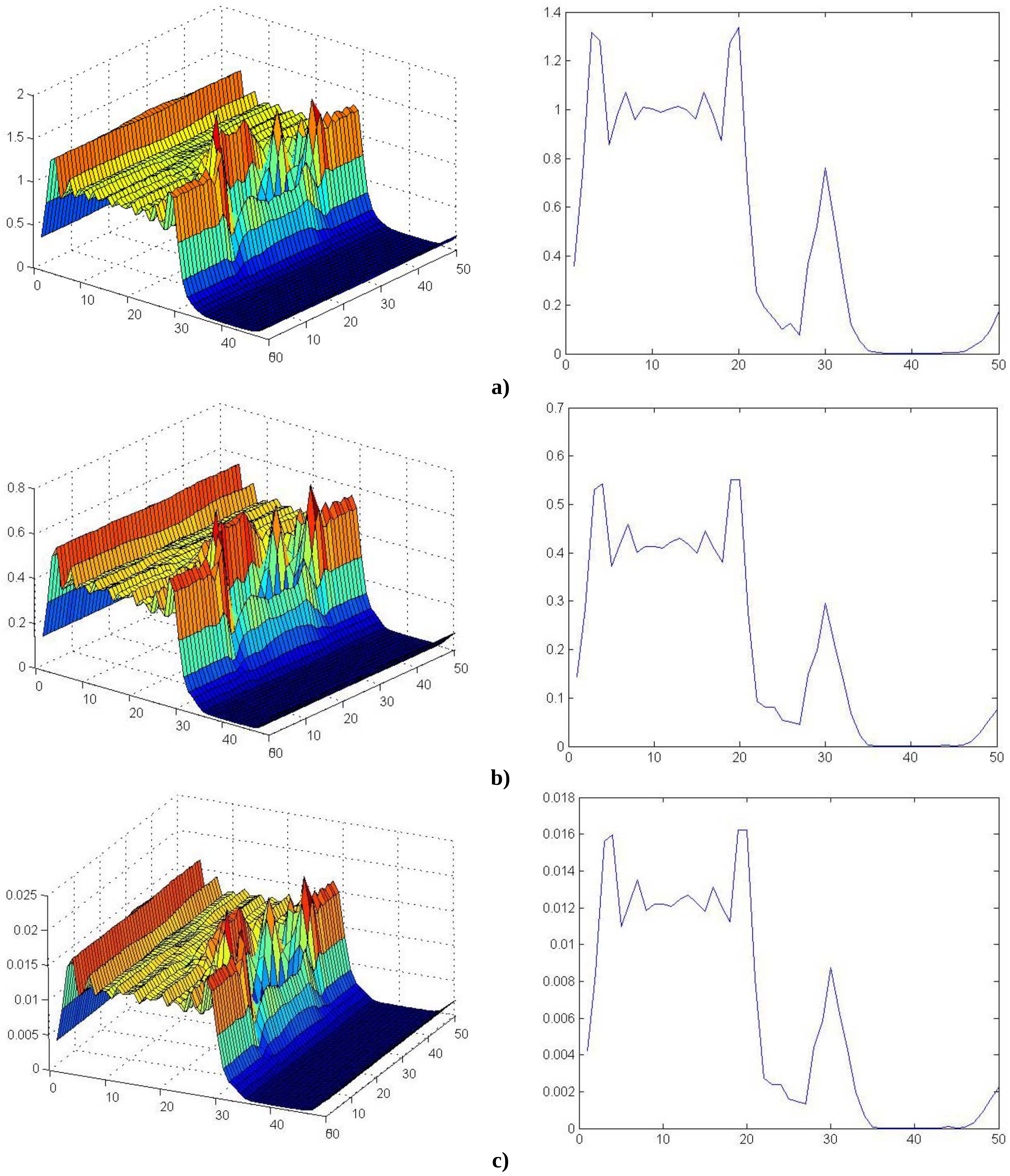
The overall speedup is about 3 times for the case of 4 processors. This result is due to the fact that the parallel application has a nonlinear structure with a different duration of the operations, so its parallelism is also non-linear. Nevertheless, the resulting acceleration can substantially speed up the processing of the photomasks.

The second group of experiments shows the dependence of the performance of parallel applications on the size of the input data (Table 2, the number of objects is equals 100, the number of processors – 4).

Table 2. The calculation time for different data

Number of objects	Number of points of the pupil		
	50	100	200
100	49,51	156,36	534.56

Judging by the results, the algorithm shows almost linear scalability. When the number of points of the pupil increases by 2 times in X and Y coordinates, the amount of data processing is increased by 4 times. In this case, the processing time increases by 3.16 times for 100 pixels, and by 3.42 times for 200 points (about the time for 100 pixels). This demonstrates the scalability of the algorithm, since the processing time is almost linearly dependent on the incoming data volume.



a)

b)

c)

Fig. 6 – Aerial image – intensity distribution on the surface of photoresist (a), past intensity – intensity distribution in the layer of photoresist at the beginning of the exposure (b), absorbed intensity – intensity distribution in the layer of photoresist at the end of exposure (c)

The results of the experiments with the scenarios for calculating the characteristics of aerial images are shown in Table 3 (the processing flow of objects in seconds). It is clear that the optimum time is achieved for 2 processors. Obviously this is due to the large volume of communications between the operations. In this case, a high degree of a locality is required for a quick access to the data.

Table 3. Calculation of the characteristics of aerial pictures

Number of objects	**Number of processors**			
	1	**2**	**3**	**4**
20	223,21	168,80	178,34	187,47
50	566,99	427,36	442,9	461,27
100	1150,89	865,34	880,74	902,45

The scenario was tested on a sequence of 20 images. The following results of the execution time are presented in Table 4.

Table 4. Calculation of the intensity distribution in the photoresist layer

Number of images	Number of processors			
	1	2	3	4
20	621,35	320,231	263,04	204,36

The results indicate a high degree parallelism of the scenario allowing achieving substantial speed up the processing even for relatively short flows simulated images.

Experimental data flows had an irregular structure and were generated randomly. The results of the static optimization for deterministic flows are presented in the form of comparing the times of stream processing according to the schedules obtained by the classical GA and VAN algorithm. The results are given for different numbers of CPU involved in the processing (Fig. 7).

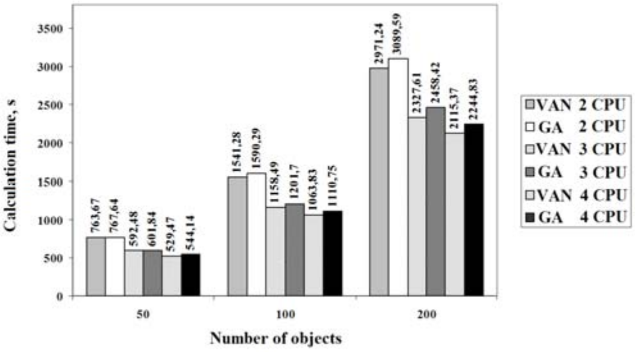


Fig.7 – Improving of a performance for the static optimization

The optimal static schedules, obtained in the first series of the experiments, were used in the second series of the experiments with stochastic flows. These schedules were compared with the schedules which were implemented by dynamically reconfigurable applications (Fig. 8). The results show a change in processing time for the static (S) and dynamic (D) VAN algorithm.

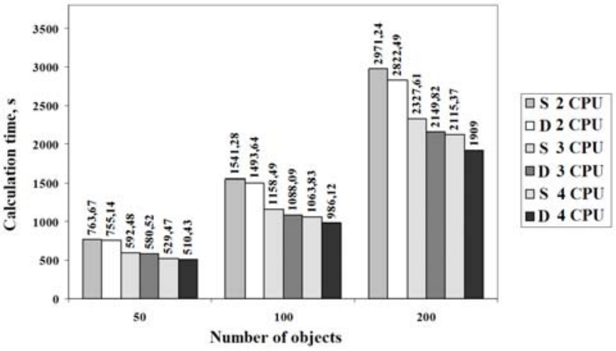


Fig.8 – Improving of a performance for the dynamic optimization

The results of the first series of experiments indicate that VAN algorithm finds the best VAN schedules, and the performance of the algorithm increases with extension of the search space. The VAN algorithm also has a lower computational complexity and finds solutions faster than the classical GA.

The results of the second series of experiments show that the virtual network algorithm significantly improves performance in case of stochastic processing flow of images by operating of dynamic optimization system.

6. CONCLUSION

The implementation of parallel applications in specialized component architectures allows users to significantly accelerate the process of creating, analyzing and optimizing programs. Architecture of data stream processing based on the use of multi-agent systems can be easily adapted for many applications involving parallel processing. Using the component approach defines a flexible framework of a parallel application that allows adapting the computational process for the operation. Design tools can be easily extended with new operations, algorithms, and data types for implementing application processing flow of information from different subject areas.

7. REFERENCES

[1] M. Born, E. Wolf, *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*, seventh ed., Cambridge University Press, 1999.

[2] Y.C. Pati, T. Kailath, Phase-shifting masks for microlithography: automated design and mask requirements, *Journal of the Optical Society of America* A 11 (1994), pp. 2438-2452.

[3] D. Yu, Z. Pan and C.A. Mack, Fast lithography simulation under focus variations for OPC and layout optimizations, *Proc. SPIE 6156*, Apr. 2006, pp. 397-406.

[4] C. Spence, Full-chip lithography simulation and design analysis – how OPC is changing IC design, *Proc. SPIE,* (21) 10 (2005), pp. 1-14.

[5] Eric R. Poortinga, Comparing software and hardware simulation tools on an embedded-attenuated PSM / Eric R. Poortinga [et al.] [Electronic resource]. – 2007. – Mode of access: www.micromagazine.com/archive/00/06/ poortinga.html. – Date of access: 12.07.2008.

[6] Optolith – 2D Optical Lithography Simulator [Electronic resource]. – 2005. – Mode of access: http://www.silvaco.com/products/vwf/athena/

optolith/optolith_datasheet.html. – Date of access: 11.07.2008.

[7] N. Cobb and Y. Granik, New concepts in OPC. *Proc. SPIE 5377*, 2004, pp. 680-690.

[8] J. Ye, Y.-W. Lu, Y. Cao, L. Chen, and X. Chen, System and method for lithography simulation, *Patent US 7,117,478* B2, Jan. 18, 2005.

[9] G.A. Gomba, Collaborative innovation: IBM's immersion lithography strategy for 65 nm and 45 nm halfpitch nodes & beyond, *Proc. SPIE 6521*, 2007.

[10] D. Argiro, S. Kubica, M. Young, and S. Jorgensen, Khoros: an integrated development environment for scientific computing and visualization, *Whitepaper, Khoral Research, Inc.*, 1999.

[11] M. Zikos, E. Kaldoudi, S. Orphanoudakis, DIPE: a distributed environment for medical image processing, *Proceedings of MIE'97*, Porto Carras, Sithonia, Greece, May 25-29, 1997. – pp. 465-469.

[12] M. Guld, B. Wein, D. Keysers, C. Thies et al., A distributed architecture for content-based image retrieval in medical applications, *Proceedings of the 2nd International Workshop on Pattern Recognition in Information Systems.* 2002, pp. 299-314.

[13] J. Wickel, P. Alvarado, P. Dörfler et al., Axiom – a modular visual object retrieval system, M. Jarke, J. Koehler, and G. Lakemeyer, editors. *Advances in Artificial Intelligence LNAI 2479*. Springer, 2002. p. 253–267.

[14] W. Gropp, E. Lusk, and A. Skjellum, *Using MPI: Portable Parallel Programming with the Message Passing Interface.* – MIT Press. – 1995.

[15] C.J.R. Sheppard, P. Torok, Approximate forms for diffraction integrals in high numerical aperture focusing, *Optik,* (105) 2 (1997), pp. 77-82.

[16] N.B. Voznesensky, A.V. Belozubov, Polarization effects on image quality of optical systems with high numerical apertures, *Proc. SPIE, 1999, Vol.3754*, p.366-373.

[17] K. Hwang, Z. Xu, *Scalable Parallel Computing – Technology, Architecture, Programming,* McGraw-Hill, USA, 1998.

[18] O. Beaumont, A. Legrand, Y. Robert, Static scheduling strategies for heterogeneous systems, *Computing and Informatics,* (21) (2002), pp. 413-430.

[19] Y.-K. Kwok, I. Ahmad, Static scheduling algorithms for allocating directed task graphs to multiprocessors, *ACM Computing Surveys,* (31) 4 (1999), pp. 406-471.

[20] T. Hagras, J. Janecek, A fast compile-time task scheduling heuristic for homogeneous computing environments, *International Journal of Computers and Their Applications,* (12) 2 (2005), pp. 76-82.

[21] A. Gerasoulis, T. Yang, A comparison of clustering heuristics for scheduling directed acyclic graphs onto multiprocessors, *Journal of Parallel and Distributed Computing,* (4) 16 (1992), pp. 276-291.

[22] M.A. Palis, J.-C. Liou, and D.S.L. Wei, Task clustering and scheduling for distributed memory parallel architectures, *Trans. Parallel and Dist. Systems*. (7) 1 (1996), pp. 46-55.

[23] S. Porto, A.C. Ribeiro, A tabu search approach to task scheduling on heterogeneous processors under precedence constraints, *International Journal of High-Speed Computing,* (2) 7 (1995), pp. 45-71.

[24] Y.M. Yufik, T.B. Sheridan, Virtual networks: new framework for operator modeling and interface optimization in complex supervisory control systems, *Annual Reviews in Control,* (20) (1996), pp. 179-195.

[25] S. Avakaw, A. Korneliuk, A. Tsitko, A prospective modular platform of the mask pattern automatic inspection using die-to-database mask method, *Proc. of SPIE, Photomask and Next-Generation Lithography Mask Technology XII*, Yokohama, Japan, 13-15 April, 2005. – Vol. 5853. – Bellingham, Washington: SPIE, 2005, pp. 965-976.

__Syarhei M. Avakaw__ Candidate of Sciences degree (an equivalent of Ph.D.) in 2002, Doctor of Engineering in 2008. Director of the KBTEM-OMO (Planar Corporation). Since 1984 has been working on development of the equipment for automatic inspection of planar structures. The main research interests of Dr. S. Avakaw is automatic inspection in precise equipment used for manufacture of high-precision master patterns of the electronics products.

__Alexander A. Doudkin__ Candidate of Sciences degree (an equivalent of Ph.D.) in CAD in 1987. Doctor of Engineering in 2010. Principal researcher at the laboratory of systems identification, United Institute of Informatics Problems, NAS of Belarus. Member of Belarusian branch of IAPR and an international society of neural *networks. Research interests: digital signal and image processing; development of methods, algorithms and technologies for integrated circuit layouts digital processing in respect to creation of computer vision system for VLSI design and manufacture inspection; neural network application, methods of parallel data processing.*

***Alexander Inyutin** graduated in 1998 from Belarusian State University of Transport BSUT, Gomel, received the MS degree in 1999. He works in laboratory of systems identification, United Institute of Informatics Problems of National Academy of Sciences of Belarus. His research interests include image processing, mathematical morphology, defect detection, methods of parallel data processing.*

***Aleksey V. Otwagin** is graduated in Computer Science in BSUIR at 1998. Candidate of Sciences degree (an equivalent of Ph.D.) in 2007. He is associate professor of Belarusian State University of Informatics and Radioelectronics. His research interests include multi-agent systems, parallel processing, optimization of parallel program, genetic algorithms.*

***Vladislav A. Rusetsky** is graduated in Computer Science in BSUIR in 2008, received the MS degree in 2009. Now hi is postgraduate at the Department of Electronic Engineering and Technology, BSUIR. His research interests include development of the equipment for photomask production.*

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

ALGEBRAIC APPROACH TO INFORMATION FUSION IN ONTOLOGY-BASED MODELING SYSTEMS

Irina Artemieva [1], Alexander Zuenko [2], Alexander Fridman [2]

[1] Far Eastern State University, 8 Sushanova str., 690950 Vladivostok, Russia; iartemeva@mail.ru
[2] Institute for Informatics and Mathematical Modelling of Technological Processes of the Russian Academy of Sciences, 24A Fersman str., 184209 Apatity, Russia; {fridman, zuenko}@iimm.kolasc.net.ru

Abstract: *In this paper we discuss the possibilities to use algebraic methods (in particular, n-tuple algebra developed by the authors) to improve the functioning of convenient ontology-based modeling systems. An illustrative example shows the ways to unify representation and processing of two major parts of subject domain ontologies.*

Keywords: *ontology-based system, modeling system, subject domain ontology, n-tuple algebra.*

1. INTRODUCTION

An intelligence system for a domain with complicated structure belongs to knowledge-based systems, which allow for accumulating knowledge relating to different chapters of the domain as well as sub-domains ontologies and data archives in order to support solving various applied tasks by domain specialists.

To represent ontology and knowledge, developers often use graph-oriented structures [1]. Then a graph traversal can help to solve a task since access to information stored in a graph leaf requires finding a path from the root of graph to this leaf. This is time consuming.

Databases (DBs) give an alternative way to represent ontology and knowledge. In this case, we formulate a task solving method by means of a query language. To accelerate access to information, all descriptors of an ontology element including its properties, functions and relations, which form its knowledge base are stored in one database table. To become practically useful for specialists of a complicated subject domain, information components of a program system have to contain data archives, ontology and other domain knowledge. A special data ontology provides interpretation of information stored in archives [2]. Data archives assist in solving different classes of applied tasks including various data analysis for knowledge acquisition.

Using of database tools to represent information components of an intelligence system for subject domains with complicated structure gives a possibility to formulate database queries representing different integrity restrictions on knowledge or/and data as well as rules to coordinate knowledge with data and data with knowledge.

In [3], we considered a method to represent ontology and knowledge by database tables and to use them for developing an intelligence system.

Thus, we can state that developers of modern intelligence systems face certain challenges resulting from fundamentally different approaches used in constructing DBs and knowledge bases (KBs). KB design is based on a mathematical system that is named by a number of terms: formal approach, axiomatic method, symbolic logic, theory of formal systems (TFS). In TFS, inference rules are defined in the way that allows to interpret new symbol constructions as corollaries to or new theorems from the symbol constructions or statements that are axioms or theorems in the given formal system. Algebraic techniques, e.g. those of relational algebra are most commonly used in constructing data processing systems.

Ontology-based systems also have problems with integration their subsystems into a common software environment. In our opinion, algebraic approach seems to be a rational supplement to traditional formal methods in logic in order to unify information representation and processing as well as to improve logical analysis techniques. Below we will mostly dwell on the first part of these problems, namely the unification. As a mathematical and software basis, we use *n-tuple algebra* (NTA) [4, 5] briefly introduced below in Section 5. Then we describe NTA capabilities in dealing with graphs and semantic networks (see Section 6). To exemplify theoretical statements, we refer to chemical subject domain [6] and some training tasks for intelligence systems in this domain.

2. TWO PARTS OF THE DOMAIN ONTOLOGY

Domain ontologies for development of intelligence systems able to not only store, search and edit subject domain ontologies and knowledge but to solve other types of applied tasks as well, have to contain a definition of a notion system used to determine input data for the applied tasks and represent results of solving these tasks. For example, such a notion system for chemistry must allow for describing different properties of physical and chemical processes that take place at a certain period of time and under certain external conditions (see Fig. 1).

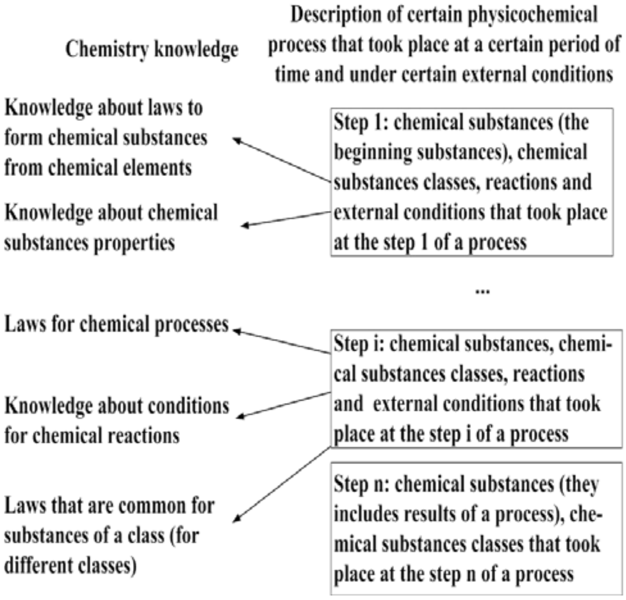

Chemistry knowledge
Description of certain physicochemical process that took place at a certain period of time and under certain external conditions
Knowledge about laws to form chemical substances from chemical elements
Knowledge about chemical substances properties
Step 1: chemical substances (the beginning substances), chemical substances classes, reactions and external conditions that took place at the step 1 of a process
...
Laws for chemical processes
Knowledge about conditions for chemical reactions
Step i: chemical substances, chemical substances classes, reactions and external conditions that took place at the step i of a process
Laws that are common for substances of a class (for different classes)
Step n: chemical substances (they includes results of a process), chemical substances classes that took place at the step n of a process

Fig. 1 – Two parts of a chemistry ontology

Thus, the domain ontology can comprise two parts. The first one consists of terms representing the subject domain knowledge. These terms allow us to describe different properties of subject domain objects and to define names of their sets or subsets. For example, terms *Chemical substances, Chemical reactions* are names of nonempty sets; terms *Metal oxides, Metals* are names of subsets. Terms *Atomic weight*, *Current number* specify properties of chemical elements. As another example, let us define the term *Reagents of reaction* as an own property of a chemical reaction. So the definitional domain of the function defining this property is the set of chemical reactions, and the value area is the set of all subsets of all chemical substances. If the term is defined as a property of a reagent of a reaction, then its first argument designates the name of reaction, the second one is the name of a reagent of this reaction.

The second part of the subject domain ontology includes terms, which are used describe different properties of physical and chemical processes that take place at a certain period of time and under certain external conditions. Process descriptions represent results of experiments realized by chemistry researchers. A set of descriptions of such experiments forms data archives, which are information components of intelligence chemistry systems. Data archives allow for testing the regularity of knowledge stored inside an intelligence system during monitoring of information components.

Let us now describe some examples of terms belonging to the second part of the domain ontology. The term *Process reactions* is a function. Its argument is a number of the process step, and the result of the function is a set of reactions that take place at this step. It is obvious that the result of the function is a subset of the set with the name *Chemical reactions.*

Archives can also be used to analyze and generalize experiment results in order to discover new domain knowledge. In this case, data archives either provide input data for a system that automatically realizes such generalizations or take part in regularity tests of the generalizations carried out manually and added to information components of an intelligence system [7].

3. DATABASE STRUCTURE

Information representation structure in a knowledge base is defined by means of the first part of ontology. The subject domain ontology allows to specify the database schema as a set of terms and their interconnections. If a term is defined in the ontology model as a set, it will be represented in a database as a table containing two fields: unique ID (key field) and a value. If the term is defined as a function, it will be expressed as a table where the number of fields is by one greater then the sum of arguments numbers plus the number of elements in the representation of the result (if the result is a Cartesian product rather than a single value, then each element of this product corresponds to one table field). If the result is a predicate, it is regarded as a function with Boolean output.

The type of each field is specified by means of value restrictions from the ontology model [7].

4. TASK TYPES

Here we analyze the constraints, which graph-oriented formalisms of knowledge representation impose on solutions of applied tasks in different subject domains.

In such a case, the constraints, by which the domain ontology may constrain the knowledge contents, have to be expressed as restrictions of types “part-whole”, “set-subset” and so on. Task solving methods are represented by graph traversal algorithms. Therefore World Wide Web Consortium (W3C) has developed a special formalism to represent other task solving methods as rules [8] for ontology-based program systems.

Let us now consider possibilities we have if we use algebraic methods to represent knowledge.

There are agreements belonging to a set of domain ontological agreements, which can be expressed by equalities. They define relations among values of several terms. Such equalities can be used to automatically manage information components of the intelligence system. They allow to compute values of some terms using given values of all other terms of an equality. This kind of computations can be easily represented by a database query language.

Some typical examples of tasks for finding ways to synthesize substances look as follows. Input data of the tasks can specify:

- a substance to synthesize;
- a set of initial substances (starting points of the synthesis);
- a set of intermediate substances, which can be used during the synthesis.

If a goal substance is specified, the task result can be obtained by a query to the database, which stores information about reactants and results of chemical reactions.

If a set of initial substances is fixed, the task result is a set of reactions whose set of reactants has initial substances as a subset and whose set of reaction results contains the goal substance. If no suitable reactions exist in the database, the task result is a set of reactions’ sequences. The reactants of the first reaction of each sequence contain initial substances as a subset. The results of the last reaction of each sequence contain the goal substance. Each sequence is the result of a query to the database.

A set of intermediate substances forms additional conditions for queries to the database.

Having the structures of the domain ontology and database described, we are going to demonstrate that all necessary structures can be expressed in similar algebraic objects, and the latter ones can be processed by unified algebraic procedures to solve standard tasks of an ontology-based modeling system. However, previously we have to introduce the mathematical basis of these representations and processing procedures.

In the two following sections, we will briefly describe possibilities to use an algebraic system, namely NTA, for solving the problems under discussion.

5. BASICS OF N-TUPLE ALGEBRA

NTA was developed for modelling and analysis of multiplace relations. Unlike relational algebra used for formalization of databases, NTA can use all mathematical logic’s means for logic modelling and analysis of systems, namely logical inference, corollary trueness’ check, analysis of hypotheses, abductive inference, etc. NTA is based on the known properties of Cartesian products of sets, which correspond to the fundamental laws of mathematical logic. In NTA, transitional results can be obtained without representation the NTA structures as sets of elementary *n*-tuples since every NTA operation uses sets of components of attributes or *n*-tuples of components [9, 10].

Definition 1. N-tuple algebra is an algebraic system whose support is an arbitrary set of multiplace relations expressed by specific structures, namely elementary *n*-tuple, *C*-*n*-tuple, *C*-system, *D*-*n*-tuple, and *D*-system, called *n-tuple algebra objects*.

So, apart from the elementary *n*-tuple, NTA contains additional structures providing a compact expression for sets of elementary *n*-tuples.

Names of NTA objects consist of a name proper, sometimes appended with a string of names of attributes in square brackets; these attributes determine the relation diagram in which the *n*-tuple is defined. For instance, if an elementary *n*-tuple $T[XYZ] = (a, b, c)$ is given, then T is the name of the elementary *n*-tuple (a, b, c), X, Y, Z are names of *attributes,* and $[XYZ]$ is the *relation diagram* (i.e. space of attributes), $a \in X$, $b \in Y$ и $c \in Z$. A *domain* is a set of all values of an attribute. Hereafter attributes are denoted by capital Latin letters which may sometimes have indices, and the values of these attributes are denoted by the lower-case Latin letters. A set of attributes representing the same domain is called a *sort*. Structures defined on the same relation diagram are called *homotypic* ones. Any totality of homotypic NTA objects is an algebra of sets.

N-tuple algebra is based on the concept of a flexible universe. A *flexible universe* consists of a certain totality of *partial universes* that are Cartesian products of domains for a given sequence of attributes. A relation diagram determines a certain partial universe.

In a space of properties $\boldsymbol{S}$ with attributes X_i (i.e. $\boldsymbol{S} = X_1 \times X_2 \times \ldots \times X_n$), the flexible universe will be comprised of different projections, i.e. subspaces that use a part of attributes from $\boldsymbol{S}$. Every such subspace corresponds to a partial universe.

Definition 2. An *elementary n-tuple* is a sequence of elements each belonging to the domain of the corresponding attribute in the relation diagram. An example of an elementary *n*-tuple $T[XYZ]$ is given above.

Definition 3. A *C-n-tuple* is an *n*-tuple of sets (*components*) defined in a certain relation diagram; each of these sets is a subset of the domain of the corresponding attribute.

A *C-n*-tuple is a set of elementary *n*-tuples; this set can be enumerated by calculating the Cartesian product of the *C-n*-tuple's components. *C-n*-tuples are denoted with square brackets. For example, $R[XYZ] = [A, B, C]$ means that $A \subseteq X$, $B \subseteq Y$, $C \subseteq Z$ and $R[XYZ] = A{\times}B{\times}C$.

Definition 4. A *C-system* is a set of homotypic *C-n*-tuples that are denoted as a matrix in square brackets. The *C-n*-tuples that such a matrix contains are rows of this matrix.

A *C*-system is a set of elementary *n*-tuples. This set equals to the union of sets of elementary *n*-tuples that the corresponding *C-n*-tuples contain.

In order to combine relations defined on different projections within a single algebraic system isomorphic to algebra of sets, NTA introduces *dummy attributes* formed by using *dummy components*. There are two types of these components. One of them called a *complete component* is used in *C-n*-tuples and is denoted by "*". A dummy component "*" added in the *i*-th place in a *C-n*-tuple or in a *C*-system equals to the set corresponding to the whole range of values of the attribute X_i. In other words, the domain of this attribute is the value of the dummy component. Another dummy component (∅) called an *empty set* is used in *D-n*-tuples.

A C-n-tuple that has at least one empty component is empty.

Below, we will show that usage of dummy components and attributes in NTA allows to transform relations with different relation diagrams into ones of the same type, and then to apply operations of theory of sets to these transformed relations. The proposed technique of defining dummy attributes differs from the known techniques essentially because new data are inputted into multiplace relations as sets rather than elementwise which significantly reduces both computational laboriousness and memory capacity for representation of the structures.

Operations (intersection, union, complement) and checks of relations of inclusion or equality for these NTA objects correspond to the known properties of Cartesian products.

Theorem 1. $P \cap Q = [P_1 \underset{\cdot}{\cap} Q_1\ \ P_2 \underset{\cdot}{\cap} Q_2\ \ \ldots\ \ P_n \underset{\cdot}{\cap} Q_n]$.

Theorem 2. $P \underset{\cdot}{\subseteq} Q$, if and only if $P_i \subseteq Q_i$ for all $i = 1, 2, \ldots, n$.

Theorem 3. $P \cup Q \subseteq [P_1 \underset{\cdot}{\cup} Q_1\ \ P_2 \underset{\cdot}{\cup} Q_2\ \ \ldots\ \ P_n \underset{\cdot}{\cup} Q_n]$, equality is possible in two cases only:

(i) $P \subseteq Q$ or $Q \subseteq P$;

(ii) $P_i = Q_i$ for all corresponding pairs of components except one pair.

Note that in NTA, according to Definition 4, equality $P \cup Q = \begin{bmatrix} P_1 & P_2 & \cdots & P_n \\ Q_1 & Q_2 & \cdots & Q_n \end{bmatrix}$ is true for all cases.

Theorem 4. Intersection of two homotypic *C*-systems equals to a *C*-system that contains all non-empty intersections of each *C-n*-tuple of the first *C*-system with each *C-n*-tuple of the second *C*-system.

Theorem 5. Union of two homotypic *C*-systems equals to a *C*-system that contains all *C-n*-tuples of the operands.

To introduce some algorithms for calculating complements of the NTA objects, we need the following

Definition 5. A *complement* ($\overline{P_j}$) of any component P_j of an NTA object is defined as a complement to the domain of the attribute corresponding to this component.

Theorem 6. For an arbitrary *C-n*-tuple $P = [P_1\, P_2 \ldots P_n]$

$$\overline{P} = \begin{bmatrix} \overline{P_1} & * & \cdots & * \\ * & \overline{P_2} & \cdots & * \\ \cdots & \cdots & \cdots & \cdots \\ * & * & \cdots & \overline{P_n} \end{bmatrix}. \qquad (1)$$

In the above *C*-system (1) whose dimension is $n \times n$, all the components except the diagonal ones are dummy components. We shall call such *C*-systems *diagonal C-systems.* To denote diagonal *C*-systems as one *n*-tuple of sets, we use *reversed* square brackets. Such a "reduced" expression for a diagonal *C*-system makes up a new NTA structure called a *D-n*-tuple.

Definition 6. A *D-n-tuple* is an *n*-tuple of components enclosed in reversed square brackets which equals a diagonal *C*-system whose diagonal components equal the corresponding components of the *D-n*-tuple.

According to Definition 6, the complement of a *C-n*-tuple can be directly recorded as a *D-n*-tuple.

Definition 7. A *D-system* is a structure that consists of a set of homotypic *D-n*-tuples and equals the intersection of sets of elementary *n*-tuples that these *D-n*-tuples contain.

Theorem 7. The *complement of a C-system* is a *D*-system of the same dimension, in which each component is equal to the complement of the corresponding component in the initial *C*-system.

It is easy to see that relations between *C*-objects (*C-n*-tuples and *C*-systems) and *D*-objects (*D-n*-tuples and *D*-systems) are in accordance with de Morgan's laws of duality. Due to this fact, they are called ***alternative classes.*** Let us now introduce some theorems regulating this transformation.

Theorem 8. Every *C*-*n*-tuple (*D*-*n*-tuple) *P* can be transformed into an equivalent *D*-system (*C*-system) in which every non-dummy component p_i corresponding to an attribute X_i of the initial *n*-tuple is expressed by a *D*-*n*-tuple (*C*-*n*-tuple) that has p_i in the attribute X_i and dummy components in all the rest attributes.

Theorem 9. A *D*-system *P* containing *m* *D*-*n*-tuples is equivalent to a *C*-system equal to the intersection of *m* *C*-systems obtained by transformation every *D*-*n*-tuple belonging to *P* into a *C*-system.

Theorem 10. A *C*-system *P* containing *m* *C*-*n*-tuples is equivalent to a *D*-system equal to the union of *m* *D*-systems obtained by transforming every *C*-*n*-tuple belonging to *P* into a *D*-system.

Transformations of NTA objects into ones of alternative classes allow to realize all operations of theory of sets on NTA objects without having to represent the objects as sets of elementary *n*-tuples.

Let us call relations and operations of algebra of sets with preliminary addition of missing attributes to NTA objects *generalized operations and relations* and denote them in this way: $\cap_G$, $\cup_G$, $\setminus_G$, $\subseteq_G$, $=_G$, etc. The first two operations completely correspond to logical operations $\wedge$ and $\vee$. NTA relation $\subseteq_G$ corresponds to deducibility relation in predicate calculus. Relation $=_G$ means that two structures are equal if they have been transformed to the same relation diagram by adding certain attributes. This technique offers a fundamentally new approach to constructing logical inference and deducibility checks [5, 8, 9, 10].

6. NTA: DATA AND KNOWLEDGE REPRESENTATION

6.1. GRAPHS AND SEMANTIC NETWORKS

In artificial intelligence systems, logical inference in graphs and semantic networks is implemented through algorithms of search for accessible vertices or through construction of the transitive closure of a graph. However, such algorithms are not efficient enough and hard to parallel. Let us now consider the way graphs are expressed in NTA. We will use the graph presented in Fig. 2 as an example.

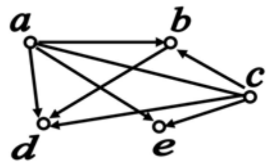


Fig. 2 – Example of a graph

This graph can be expressed as a *C*-system $G[XY] = \begin{bmatrix} \{a\} & \{b,c,d,e\} \\ \{b\} & \{d\} \\ \{c\} & \{a,b,d,e\} \end{bmatrix}$ isomorphic to the adjacency matrix of this graph.

Composition of graphs $G \circ G$, e.g. composition of a graph with itself, is used quite often. This operation is shortly denoted as G^2. Greater "degrees" of composition can also be used, e.g. $G^3 = G \circ G \circ G$ and so on.

It is often necessary to determine the set of all the accessible vertices for each vertex of a graph *G*. This information is contained in the *transitive closure* of the graph (suppose that it contains *n* vertices), which is the graph G^+ each of whose vertices is connected with all its accessible vertices with an arc. Transitive closure can be constructed with the following sequence of operations:

$$G^+ = G \cup G^2 \cup G^3 \cup \ldots \cup G^k,$$

where $k \le n$. Practically in all cases, the operation of transformation of a finite graph *G* into graph G^+ ends before the last "degree" G^k is found. The reason for ending this operation early is the fact that at some step the next "degree" of the graph does not have any arcs that have not been in the graph before.

Let us consider the way inference in semantic networks is implemented in NTA [4]. Any semantic network can be represented as a totality of binary relations. In semantic networks, inference rules are expressed as productions whose left part contains joins or compositions of some of these relations, and the right part is a relation that is substituted for the left part in the semantic network or is added to the semantic network as a new relation. Suppose that in an initial semantic network, existing relations R_1 and R_2 (see Fig. 3) infers an additional link R_3 between the domain of the relation R_1 (vertex *K*) and the co-domain of the relation R_2 (vertex *N*) as it is shown in Fig. 4 where *A*, *B*, *C* are variables whose values can be the vertices of the described semantic networks.

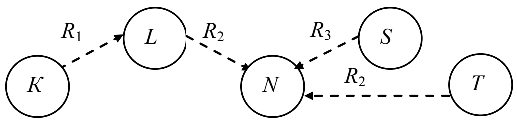


Fig. 3 – Initial semantic network

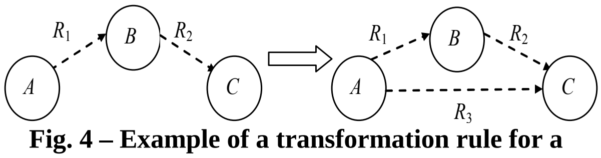


Fig. 4 – Example of a transformation rule for a network

In NTA language, this network can be recorded as a totality of *C*-systems, namely $R_1[XY] = [\{K\},\{L\}]$, $R_2[YW] = [\{L,T\},\{N\}]$, $R_3[X,W] = [\{S\},\{N\}]$.

6.2. CORRESPONDENCE BETWEEN N-TUPLE ALGEBRA AND PREDICATE CALCULUS

In trivial case (when individual attributes do not correspond to multiplace relations), an *n*-tuple corresponds to *conjunction* of one-place predicates with different variables. For example, a *C*-n-tuple $P[XYZ] = [P_1\ P_2\ P_3]$ where $P_1 \subseteq X$; $P_2 \subseteq Y$; $P_3 \subseteq Z$ corresponds to a logical formula $H = P_1(x) \wedge P_2(y) \wedge P_3(z)$. A *D*-*n*-tuple $\overline{P} =]\overline{P_1}\ \overline{P_2}\ \overline{P_3}[$ corresponds to the negation of the formula *H* (*disjunction* of one-place predicates) $\neg H = \neg P_1(x) \vee \neg P_2(y) \vee \neg P_3(z)$. An elementary n-tuple that is a part of a non-empty NTA object corresponds to a *satisfying substitution* in a logical formula. An empty NTA object corresponds to an *identically false formula*. An NTA object that equals any particular universe corresponds to a valid formula, or a *tautology*. A non-empty NTA object corresponds to a *satisfiable formula*.

In NTA, attribute domains can be any arbitrary sets that are not necessarily equal to each other. This means that NTA structures correspond to formulas of *many-sorted predicate calculus*. One can find rules of quantification in NTA in [8].

7. VERY SIMPLE CASE STUDY

To clarify the general idea of using NTA in chemistry, we imagined a very simple set of reactions, namely:

$$\begin{aligned}&\text{R1) } MgO + H_2 = Mg + H_2O;\\ &\text{R2) } CuO + H_2 = Cu + H_2O; \qquad (2)\\ &\text{R3) } Zn + H_2SO_4 = ZnSO_4 + H_2.\end{aligned}$$

For the given example, these reactions form the set *Chemical reactions* described above in Section 2.

Formalization of this set uses the following attributes and domains corresponding to sorts in predicate calculus and to the term *Chemical substances* from Section 2:

$$\begin{aligned}&\text{S1) Metal oxides } X = \{MgO, CuO\};\\ &\text{S2) Water } Y = \{H_2O\};\\ &\text{S3) Metals } Z = \{Mg, Cu, Zn\}; \qquad (3)\\ &\text{S4) Salts } W = \{ZnSO_4\};\\ &\text{S5) Gases } V = \{H_2\};\\ &\text{S6) Acids } U = \{H_2SO_4\}.\end{aligned}$$

Let us also define *elementary types* (*elements*) producing substances (3):

$$\begin{aligned}&\text{E1) Oxidizers } P = \{O\};\\ &\text{E2) Acid balances } Q = \{SO_4\}. \qquad (4)\end{aligned}$$

The reactions (2) describe two types of relations among *Reagents* and resulting substances with the following diagrams corresponding to the concept of *Chemical reactions* defined as a function in Section 2:

$$\begin{aligned}&\text{F1) Metal reduction } X \times V \rightarrow Z \times Y;\\ &\text{F2) Metal oxidation } Z \times U \rightarrow W \times V. \qquad (5)\end{aligned}$$

Considering (4), the diagrams (5) look like

$$\begin{aligned}&\text{F1') Metal reduction } Z \times P \times V \rightarrow Z \times Y;\\ &\text{F2') Metal oxidation } Z \times V \times Q \rightarrow Z \times Q \times V. \qquad (6)\end{aligned}$$

These relations constitute the flexible universe [5] of the problem under investigation and form the knowledge base for this example.

Actually, representations (5) and (6) reflect the way to minimize the number of relations in the knowledge base by using variables instead of constants the way that is typical for expert systems. Since our case study is very simple, it requires no variables. So, we will use equations (2), (3) directly.

In NTA terms, reactions (2) look as follows.

$$\begin{aligned}&\text{R1: the left part } R1^{in}[XV] =]\{MgO\}\ \{H_2\}[,\\ &\text{the right part } R1^{out}[ZY] =]\{Mg\}\ \{H_2O\}[; \quad (7)\\ &\text{R2: the left part } R2^{in}[XV] =]\{CuO\}\ \{H_2\}[,\\ &\text{the right part } R2^{out}[ZY] =]\{Cu\}\ \{H_2O\}[; \quad (8)\\ &\text{R3: the left part } R3^{in}[ZU] =]\{Zn\}\ \{H_2SO_4\}[,\\ &\text{the right part } R3^{out}[WV] =]\{ZnSO_4\}\ \{H_2\}[.(9)\end{aligned}$$

Here we analyze two types of task settings from the set described in Section 4, which are similar to forward and backward inference in expert systems. They are:

T1) Database contains reagents {Zn, H_2SO_4, MgO, CuO}; we need to find all possible resulting reactions and substances;

T2) It is necessary to find out whether {Mg} can be produced from the given reagents {MgO, Zn, H_2SO_4}.

T1 implementation

Initial data stored in the DB can be recorded as a *D*-*n*-tuple because they can be used independently:

$$DBase[XZU] =]\{MgO, CuO\}\{Zn\}\{H_2SO_4)[. \quad (10)$$

To apply each of the rules (2), we need to pass through two stages. First, the solver compares the left part of a rule Ri^{in} with the current DB contents

Dbase. If they match, the DB is modified according to the right part of the rule Ri^{out}. Let us assume that we just add substances from the right part into the DB.

To check truthfulness of the left side of a rule in NTA, we have to find out whether the relation

$$\mathrm{Ri}^{in} \subseteq_G Dbase \qquad (11)$$

is true. If it is true, the solver corrects the DB this way:

$$Dbase := Dbase \cup_G \mathrm{R}i^{out}. \qquad (12)$$

If the solver checks the rules (2) similarly to Markov's algorithm (starting from the first rule and applying the first applicable one), it will decide that the first and the second rule are not applicable as the check of the relation (11) for them gives the negative result. For instance, for the first rule this check will look like

$$]\{\mathrm{MgO}\}\ \{\mathrm{H_2}\}\ \varnothing\ \varnothing[\ \subseteq\ ]\{\mathrm{MgO, CuO}\}\ \varnothing\ \{\mathrm{Zn}\}\ (\mathrm{H_2SO_4}\}[\quad (13)$$

where relations are reduced to the same diagram [*XVZU*].

Checking the third rule within the diagram [*XZU*]:

$$]\varnothing\ \mathrm{Zn}\ \mathrm{H_2SO_4}[\ \subseteq\ ]\{\mathrm{MgO, CuO}\}\ \mathrm{Zn}\ \mathrm{H_2SO_4}[\quad (14)$$

will give the positive answer, and the DB will be modified according to (12):

$$\begin{aligned} &DBase[XZUWV] = [DBase[XZU] \cup_G \mathrm{R3}^{out}[WV] = \\ &=]\{\mathrm{MgO, CuO}\}\ \{\mathrm{Zn}\}\ \{\mathrm{H_2SO_4}\}\ \{\mathrm{ZnSO_4}\}\ \{\mathrm{H_2}\}[. \quad (15) \end{aligned}$$

T2 implementation

In NTA, this task is realized by recursive passing through the following steps.

1. Setting the initial data.
2. Setting the goal data.
3. Inclusion check of the goal data in right parts of rules. If the result is negative, backtrack to the closest previous fork of the derivation tree. If there is no possible backtracking, the inference is over with the negative answer.
4. If the inclusion check finds a suitable rule, its left part has to replace its right part in the set of goals. When the set of goals contains initial (leaf) data only, its inclusion into the DB is checked. The positive (negative) result of the inclusion check witnesses the positive (negative) reply to the query.

For our example, the initial data can be expressed as a *D*-*n*-tuple

$$S =]\{\mathrm{MgO}\}\ \{\mathrm{Zn}\}\ \{\mathrm{H_2SO_4}\}[, \qquad (16)$$

and the initial set of goals (at the step number 0) equals

$$G^0 =]\{\mathrm{Mg}\}[. \qquad (17)$$

Comparing (17) with the right parts of the rules (2), the solver finds the only suitable rule (7), for which

$$G^0 \subseteq_G \mathrm{R1}^{out}[ZY] \qquad (18)$$

is true and forms the next goal set as

$$G^1 =]\{\mathrm{MgO}\}\ \{\mathrm{H_2}\}[. \qquad (19)$$

Then the solver finds $\{H_2\}$ in R3^{out}[*WV*] (9) and forms

$$G^2 =]\{\mathrm{MgO}\}\ \{\mathrm{Zn}\}\ \{\mathrm{H_2SO_4}\}\ [. \qquad (20)$$

The set (20) contains leaf data only (no its components belong to right parts of the rules (2)), and $G^2 \subseteq_G S$ is true. So, the inference answers "yes" to the query.

For tasks T1 and T2, consequences of admissible reactions can be formed by saving numbers of applied rules.

8. CONCLUSION

NTA provides storing and processing of both data and knowledge structures by similar techniques. The novelty of NTA lies in creating some new mathematical structures allowing to represent both data and knowledge. This feature simplifies combining data and knowledge bases within a single software system. In NTA, the subject domain ontology and various factual information can be recorded as a number of multiplace relations, and different queries are expressed by some operations analogous to those of theory of sets. The main idea of this processing is as follows. An initial relation defined on a Cartesian product *D* can be often split into blocks corresponding to relations on some projections of *D*, which greatly reduces laboriousness of operations on this relation by using its matrix properties. This allows to process every block separately using known features of Cartesian products, for instance, by paralleling the necessary operations. This idea provides new opportunities to express complex methods of reasoning by comparatively simple algorithms, which can be easily modelled in the computer.

If all information is stored in NTA objects of the same class (like *D*-*n*-tuples used for the case study in

Section 6), these operations are polynomially complex [4]. Generally speaking, traversals of a derivation tree in NTA results in NP-complete algorithms just as in conventional approaches. Besides usage of matrix properties of NTA objects, new structural and statistical classes of conjunctive normal forms with polynomially identifiable satisfiability properties were discovered in NTA. Consequently, we can implement many algorithms whose complexity evaluation is theoretically high, e.g. exponential, in polynomial time, on the average. As for making databases more intelligent, NTA can be considered an extension of relational algebra to knowledge processing. In the authors' opinion, NTA can become a methodological basis for creating knowledge processing languages.

NTA structures provide expressing multiplace predicates but have ordinary sets as components, these sets corresponding to unary predicates. Algebra of conditional *n*-tuples [9] was developed to expand abilities of NTA by dealing with binary predicates.

In this paper, we have illustrated solving of comparatively simple tasks in chemical synthesis when only initial and goal substances are fixed. Solving some harder tasks where we must use certain intermediate products during the synthesis, will probably require for abductive-like reasoning [5], but this matter needs additional studies we plan to conduct in near future.

9. ACKNOWLEDGEMENTS

This work was supported in part by the Russian Foundation for Basic Researches (grant 11-08-00641), OITVS of the Russian Academy of Sciences (RAS) (project # 2.3) and the Chair of RAS (project 4.3 of the Programme # 15).

10. REFERENCES

[1] M. Denny, Ontology Building: a Survey of Editing Tools: http: //www.xml.com/pub/a/ 2004/07/14/onto.html.

[2] A. Kleshchev, I. Artemjeva, Mathematical models of domain ontologies, *Int. Journal on Inf. Theories and Appl.* (14) 1 (2007), pp. 35-43.

[3] I. Artemieva, N. Reshtanenko, V. Tsvenikov, Upgradable tree levels editor of metaontologies, ontologies and knowledge for chemistry, *Int. Book Series "Information Science and Computing"* (5) 2 (2008), pp. 119-124.

[4] B. Kulik, A. Fridman, A. Zuenko, Logical Analysis of Intelligence Systems by Algebraic Method, *Proceedings of Twentieth European Meeting on Cybernetics and Systems Research (EMCSR 2010)*, Vienna, Austria 6-9 April 2010, pp. 198-203.

[5] B. Kulik, A. Zuenko, A. Fridman, Modified reasoning by means of N-tuple algebra, *Proceedings of the 11th International Conference "Pattern Recognition and Information Processing (PRIP'2011)",* Minsk, Republic of Belarus 18- 20 May 2011, pp. 271-274.

[6] I. Artemieva, Multilevel modular chemistry ontology: structure and management, *Proceedings of the First Russia and Pacific Conf. on Computer Technology and Applications (RPC 2010)*, Vladivostok, Russia 6-9 September 2010, pp. 12-17.

[7] I. Artemieva, A. Zuenko, A. Fridman, Integration of ontologies, knowledge and data archives into ontology-based modeling systems, *Proceedings of the 11th International Conference "Pattern Recognition and Information Processing (PRIP'2011)",* Minsk, Republic of Belarus 18- 20 May 2011, pp. 303-306.

[8] *RIF Production Rule Dialect.* W3C Working Draft 11 February 2010. URL: http://www.w3.org/TR/2010/WD-rif-prd-20100211/.

[9] B. Kulik, A generalized approach to modelling and analysis of intelligent systems on the cortege algebra basis, *Proceedings Sixth International Conference on System Identification and Control Problems (SICPRO'07)*, Moscow, Russia, 2007, pp. 679-715. (in Russian).

[10] A. Zuenko, A. Fridman, Development of N-tuple algebra for logical analysis of databases with the use of two-place predicates, *Journal of Computer and Systems Sciences International,* (48) 2 2009, pp. 254-261.

__Irina L. Artemieva__ graduated from Far Eastern State University and worked for the Institute of Automation and Control Processes of the Far Eastern Branch of the Russian Academy of Sciences from 1978 to 2010. Since 2010, she works for the Institute of Applied Mathematics of the FEBRAS. Her scientific interests are within artificial intelligence. She got her PhD in 1992, her Doctor of Science (Computer Science) degree in 2009 and a Professor degree in 2011. At present she is a Professor of Far Eastern Federal University (FEFU). She has published 180 scientific papers.

Alexander Zuenko, *a researcher of the Institute for Informatics and Mathematical Modelling of Technological Processes (IIMM) of the Russian Academy of Sciences (RAS), graduated from the Petrozavodsk State University in 2005 and got his PhD in 2009. His scientific activities relate to developing software for modelling open subject domains, as well as to knowledge representation and processing. He has 35 scientific publications including 2 monographs.*

Alexander Fridman *graduated from the Leningrad Electrotechnical Institute in 1975 and worked in Baku (Azerbaijan) for Russian Ship-building Ministry until 1989, when he moved to Apatity (Murmansk region, Russia) and began working for RAS. He got his PhD in 1976, a Doctor of Science (Technical Sciences) degree in 2001 and a Professor degree in 2008. At present he is the head of Laboratory on Information Technologies for Control of Industry-Natural Complexes in the Institute for Informatics and Mathematical Modelling of Technological Processes of RAS and professor of Applied Mathematics Chair in Kola Branch of the Petrozavodsk State University. His scientific interests include modelling and intelligence systems. He has 215 scientific publications including 3 monographs, 21 tutorials and 16 certificates for inventions.*

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

THE ART AND SCIENCE OF GPU AND MULTI-CORE PROGRAMMING

Robert E. Hiromoto

University of Idaho, 1776 Science Center Drive, Idaho 83402, USA,
hiromoto@cs.uidaho.edu

Abstract: *This paper examines the computational programming issues that arise from the introduction of GPUs and multi-core computer systems. The discussions and analyses examine the implication of two principles (spatial and temporal locality) that provide useful metrics to guide programmers in designing and implementing efficient sequential and parallel application programs. Spatial and temporal locality represents a science of information flow and is relevant in the development of highly efficient computational programs. The art of high performance programming is to take combinations of these principles and unravel the bottlenecks and latencies associate with the architecture for each manufacturer computer system, and develop appropriate coding and/or task scheduling schemes to mitigate or eliminate these latencies.*

Keywords: *Performance, Speedup, Parallelism, Temporal Spatial Locality.*

1. INTRODUCTION

Commodity multithreaded processors in dual- and multi-core processors, and the graphic processing unit (GPU) comprise today's computer market place. The progress in the nanometer process technology has advanced the design and manufacturing of these ambitious computer architectures. This new era of computing capabilities place new demands on the programmers and requires potentially new tricks or programming techniques for veteran programmers experienced in the art of parallel programming. With each new generation of more sophisticated processor design, the processor structure, the organization of memory hierarchy, the implicit and explicit scheduling of executable threads (tasks), and the programming software environment reappear as a set of new criteria that defines the parameter space of programming paradigms. Encouraged by the promise that these new systems can deliver higher performance, a new surge of interest and activity has emerged in areas of multi-core and GPU computing. As a consequence, the GPU is now manufactured with higher precision arithmetic and programming software to allow general-purpose computing on GPUs (GPGPU), where traditionally the applications were handled by the CPU. The GPGPU programming paradigm relies on streaming data through lightweight threads of execution, where banked memory provides non-blocking access of operands (data) between competing threads. Consequently, hardware support for GPUs and multi-core systems require more complex memory hierarchies, and hardware technologies to maintain memory consistency and cache coherence under a dynamic parallel execution model. Cache coherence in multicore systems is an example of the hardware organizational complexity required to address memory latencies.

New technologies, like the GPU and multi-core processor, introduce new challenges to the applications programmer. The novelty of these systems lies in their hardware organization, where an optimal mapping between hardware and software can easily be compromised when multithreaded parallel execution is introduced. The challenges for a parallel code (program) developer is to bend, reshape, and retrofit existing codes onto new and ever changing architectures. For the novice developer, this requires navigating a desperate terrain strewn with hidden and undetected detours.

Fortunately, there are physical principles that can provide code developers with a framework to reason about multithreaded programming complexities exhibited by new and intricate hardware configurations. These principles are *time* and *space* metrics upon which performance is measured. The faster the clock cycle, the higher the potential CPU performance; or the longer in response (latency) time, the lower in performance that can be expected. The rate of *information* flow, ultimately dictates the achievable performance whether in the activities of the financial markets or the execution of applications on computer systems. Minimizing the time for operand access, repeated reuse of data or instructions, minimizing latencies between input/output (I/O) requests are examples where *temporal* and the *spatial* locality between references play important roles in optimizing performance. *Temporal* and *spatial* localities are two simple principles that all programmers, interested in designing and implementing optimally performing application codes, should be guided by.

In the next sections, we introduce the concept of *spatial* and *temporal* locality and the implied consequences to the organization of memory hierarchies and coding of the matrix multiplication program; the notion of parallelism and Amdahl's law; and architectures of the multi-core and the GPU.

2. SPATIAL AND TEMPORAL LOCALITY

The principle of locality [1, 2] refers to two basic types of reference locality. *Spatial* locality refers to the use of data elements stored within a relatively close proximity of one another. *Temporal* locality refers to the reuse of data (instructions) or other resources within relatively short time periods. These principles, in various forms, are applied in performance optimization for cache utilization and memory prefetching technology, code motion affecting memory access patterns, and process scoreboarding to enhance processor performance. Although there are other specific terminologies of locality, those more dynamic predictive assertions (branch or most probable access predictor) are typically beyond the control of the programmer.

In this paper, we take the liberty to expand the definitions of both *spatial* and *temporal* locality as measures of both near and distant proximity of data references. Under these expanded definitions, *spatial* and *temporal* localities are principles that encapsulate the basic notions of length and time. They are measurable and when taken separately or in combinations, they can be reasoned about and formulated in predictive analysis.

From a programmer's perspective, such metrics lend themselves in code planning for I/O tasks and data layouts, code restructuring for both sequential and parallel execution, and reasoning within the context of a memory hierarchy with associated referencing costs. The programmer must understand the hardware organization and resources available to the application code. As with every organizational structure, computer system has their particular idiosyncrasies; however, these peculiarities can be measured in both space and time.

2.1. MEMORY HIERARCHY

Hierarchical memory is a hardware organization that is arranged to benefit from *spatial* and *temporal* references. Levels of memory characterize a typical memory hierarchy; where caches, populated near the CPU, have low latency and low memory capacity devices. As the distance between the CPU and the levels (from low to high) of the memory hierarchy increase, so to do their access latencies and memory capacities increase. As the distance between the CPU and the levels (from low to high) of the memory hierarchy increase, so to do their access latencies and memory capacities increase. The design of hierarchical memory is organized to read blocks of slower (higher) level memory into the next successive (lower) level of the memory hierarchy. Predictive algorithms (hardware and OS) attempt to predict which data might be accessed next or least and then react by moving the appropriate data through the memory hierarchy as needed. Temporal latencies and memory device capacities are illustrated in Fig. 1, where the cost of accessing data from the memory hierarchy that are more distant from the CPU increases from a few clock cycles to orders of magnitude.

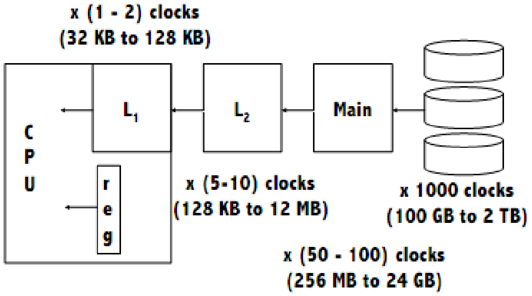
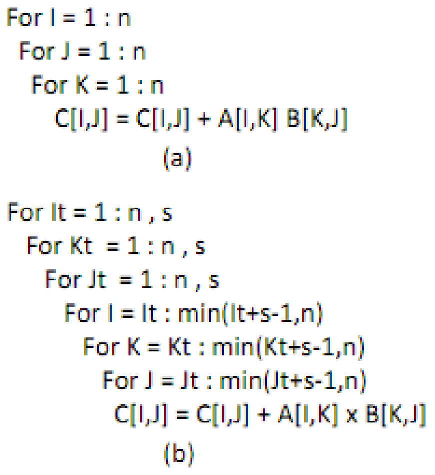


Fig. 1 – Memory Hierarchy

The organization of a memory hierarchy illustrates a principle of economy of scale. A programmer should be aware of this principle when dealing with I/O considerations. A cache is designed to be faster by restricting operand address lookup to within a small memory capacity, and by requiring shorter wires (links) to connect to the CPU. In exchange for limited cache capacity, cache technology employs temporal and *spatial* locality analyses to maintain recently referenced data and/or near recently referenced data. Programmers have considerable control over the locality of data referencing by structuring their codes in ways to increase the *spatial* and *temporal* locality of both operand and instruction references, and the locality offered by the various hierarchical storage devices. The result of this analysis can lead to a higher cache utilization per unit cycle time; translating into higher performance.

2.2. MATRIX MULTIPLICATION: ANALYSIS

An interesting application of spatial and temporal locality is the coding of the matrix multiplication operation [3]. In most introductory lectures, the multiplication of two matrices is commonly defined through the loop structure as illustrated in Fig. 2 (a). The inner-loop index K references elements of matrix A in a row-wise manner (in cache), whereas, the elements of matrix B are referenced column-wise. In a programming language that assigns the elements of a matrix in a row-wise fashion, the cache strategy for predicting the most probable next reference will load the elements of matrix B in a row-wise fashion as well. The affect of program (a) and the caching predictive strategy results in an inefficient cache utilization for matrix B, where fewer elements of its column-wise references are available in the cache resulting in more cache misses; as compared to cache misses experienced by row-wise references of matrix A. In this example, a simple remedy requires only an interchange between the J and K loop indexes. In this reorganized loop structure, matrix B more optimally complies with maximizing *spatial* locality, whereas, the matrix element A[I,K] now exhibits *temporal* locality. Fig. 2(b) illustrates a blocking technique that further improves *temporal* and *spatial* locality. In this more complicated set of loops, matrix multiplication is computed using a series of sub-block calculations where sub-blocks of A and B can be used several times (increasing *temporal* and *spatial* locality) before the next sub-blocks are accessed by the cache in a block consecutive fashion.



```
For I = 1 : n
 For J = 1 : n
  For K = 1 : n
    C[I,J] = C[I,J] + A[I,K] B[K,J]
                 (a)

For It = 1 : n , s
  For Kt  = 1 : n , s
    For Jt  = 1 : n , s
      For I = It : min(It+s-1,n)
        For K = Kt : min(Kt+s-1,n)
          For J = Jt : min(Jt+s-1,n)
            C[I,J] = C[I,J] + A[I,K] x B[K,J]
                 (b)
```

Fig. 2 – Titled Matrix Multiplication (See [3])

A simple space-time analysis of program (a) resulted in a more optimal sequential implementation of the matrix multiplication operation. Yet a more surprising result of program (b) is that it also represents a parallel formulation of matrix multiplication. This is a very rich result as no parallel programming considerations are explicitly employed. Guided only by the analyses of spatial and temporal locality, a rather deep and unexpected result has emerged. As a consequence, it is challenging to speculate on the applicability of spatial and temporal locality as an analysis tool in developing more optimal sequential and parallel (not necessary more parallel but more optimal in performance) algorithms. The notion here is to provide distance and time metrics to determine the rate or radius of information that influences the final solution. Once such an algorithmic analysis is completed the details of computer implementation then follow. Program (b) represents a blocking strategy that parallel algorithms employ in certain situations of their implementation. The GPU in particular is heavily reliant on this technique termed *tiled* memory to facilitate memory accesses and kernel executions.

3. CATEGORIES OF PARALLELISM

Parallel programming is an optimization technique to increase computational performance. The idea is as old as human and pre-human activities in scavenging for food, where independent journeys are made to search and returned to their home base with their share of findings. Computational parallelism incorporates a similar strategy. The solution of a single problem is divided up into as many independent tasks (i.e., no information flows between these tasks) that can then be executed concurrently and then accumulated to obtain the final solution. Parallelism may be categorized into two broad groups: (1) unbounded and (2) bounded parallelism. Unbounded (or near unbounded) parallelism requires little or no communication (transfer of information) between any of the independently executing tasks. The term “embarrassingly parallel” is coined to describe such parallelism. Unbounded parallelism ushered in the notions of scalability and scaled speedup [4] as desired characteristics for trends in the future development of parallel algorithms. Unbounded parallelism may be attained in searches, algorithms that use data (information) inefficiently, transforming or marshaling data, rendering in computer graphics, ray tracing, or brute-force searches in cryptography to name a few examples. Bounded parallelism represents a more typical class of constrained parallelism, where the solution specifications requires frequent or nominal exchange of solution critical data that are computed incrementally among the various parallel tasks. All real-world solution-oriented processes represent this behavior. Fig. 3 illustrates this point in an analogy of the well-known space-time diagram to the space-time division between unbounded and bounded parallelism. Within the space-time cone of influence, solution algorithms are governed by the exchange of solution-oriented information. All communication is bounded by the speed of communication (*C*). Outside the cone of influence, a region of unbounded parallelism reigns; however, for this parallelism to make a solution contribution their region must collapse into the cone of influence.

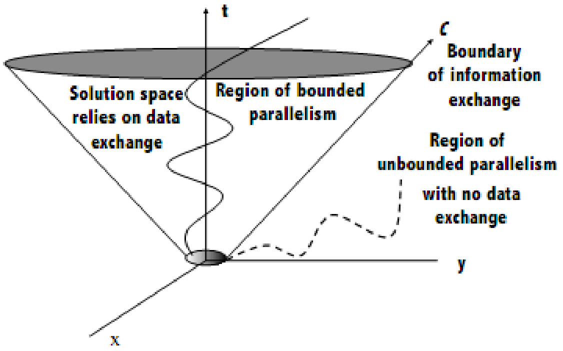


Fig. 3 – Regions of Bounded and Unbounded Parallelism

3.1. AMDAHL'S LAW

Amdahl's law [5] is an algebraic argument that parallelism, for a fixed size problem, is limited by the percentage of code that cannot be parallelized. The argument provides a simple but profound insight into the limits of performance that is attainable through the application of parallel techniques and hardware. Amdahl assumes a simple model by neglecting potential bottlenecks such as memory bandwidth and I/O bandwidth. Still the results provide an ideal upper bound for performance expectations. Amdahl's analysis derived the following overall speedup improvement when p processors are applied to the portion of a sequential algorithm that supports f_p percent of parallel improvement.

$$S_p = \frac{1}{f_s + \frac{f_p}{p}}$$

where f_s is the percent of scalar (sequential) code and the condition that $f_p + f_p = 1$.

Speedup is a useful metric to assess performance improvement; however, it is important to realize that it may be misleading as a performance metric since performance is a metric measured in time. Equation (1) arises from the ratio between the sequential execution time of an algorithm over its parallelized execution time. As a result, speedup is a dimensionless improvement metric, independent of time.

Fig. 4 depicts the comparison of speedups between two different algorithms constructed to exhibit high and low percentages of parallelism. Algorithm 1 is constructed to exhibit 90% parallelism, by selecting values for f_s and f_p to be 0.1 and 0.9, respectively. Algorithm 2 is constructed to exhibit low parallelism at 30%. The corresponding values for f_s and f_p are chosen at 0.7 and 0.3, respectively. However, the single processor performance of algorithm 2 is adjusted to be five (5) times the performance of algorithm 1. As might be expected, algorithm 2 never surpasses the speedup of algorithm 1. On the other hand, Fig. 5 provides a more sobering assessment of performance and emphases the pitfalls of the speedup metric. In this comparative plot, algorithm 1 fails to attain a parallel performance faster than algorithm 2, no matter the number of parallel processors employed. The constructed algorithms 1 and 2 could very well represent two different algorithms for the solution of the same computational problem; e.g., variants of the Jacobi and Gauss-Seidel iteration schemes. In this context, more parallelism does not necessarily mean better performance.

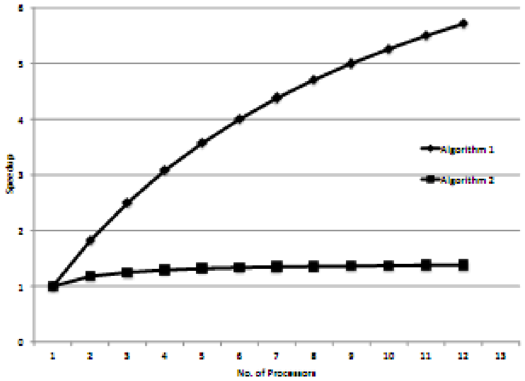


Fig. 4 – Speedup Improvement

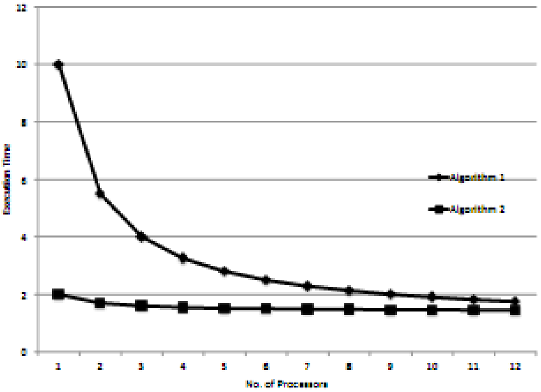


Fig. 5 – Parallel Performance

The problem with the use of speedup as a metric for performance is the introduction of time-normalization based on a non-optimal sequential algorithm (algorithm 1). Fortunately, speedup can be salvaged if the time-normalization is made with respect to the best available sequential algorithm [6], in this case algorithm 2. In other words, the time base applied to the metric of speedup must to explicitly stated; otherwise, the use of the metric is meaningless. Time, as mentioned earlier, is the critical metric that describes performance.

4. GPU AND MULTI-CORE ARCHITECTURES

In this section, an over review of generic architectures for both the multi-core and GPU processors are provided. Saline issues of the architecture are presented and the challenges that they pose for programmers to overcome.

4.1. MULTI-CORES

Traditional processors are designed with only a single CPU that performs the reading and executing of instructions. A multi-core processor combines two or more cores (CPUs) on a single integrated circuit (IC). It is also known as a chip multiprocessor (CMP), and offers the promise of enhanced performance for the processing of independent tasks, a reduced power consumption (by lowering the clock speed on each core), and low-level hardware support for the synchronization of parallel tasks. Ideally, a dual-core processor should have twice the performance as a single core processor. In several benchmarks, performance gains are found to be in the range of fifty percent to one-and-a-half times as powerful as a single core processor [7].

Many-core refers to many cores in a computer. They may or may not all populate the same chip. They may be organized as many single-core chips, a collection of single-core and multi-core chip, or many multi-core chips. A typically “many-core” might refer to 32 or more cores, while “multi-core” are fewer in number. These terminologies are technology driven and certain to change over time.

Fig. 6. depicts a generic dual-core processor with CPU-level 1 (L1) caches and a shared level 2 (L2) cache. A dual- (multi-) core configuration is distinguished from traditional multiprocessor systems by the tight shared-coupling of the L2 cache between two cores. Although this provides a “faster” connection between the coupled cores, it introduces a departure from the hierarchical memory strategies previously discussed. The capability of high-speed accessing of the L2 cache by shared cores presents challenges absent in traditional single-core multiprocessor systems. The multi-core organization places more demands on the predictive data access strategies applied to the shared L2 cache. This demand has the potential to increase reference latencies and increase contention between cores in supplying instructions and data; significantly degrading performance [8, 9]. As a consequence, multi-core designers have provided multiple hardware contexts [10, 11] or multithreading. Multithreading has the advantage that it can handle arbitrarily complex access patterns even in cases where it is impossible to predict the accesses before hand. Multithreading simply reacts to cache misses that occur, rather than attempting to predict them. Multithreading tolerates latency by attempting to overlap the latency of one context with the computation of other concurrent contexts. As with all advantages, there are disadvantages: (i) multithreading relies on available concurrency within an application, which may not exist; (ii) an overhead is incurred in switching between contexts; and (iii) significant hardware support is required to minimize context-switching overhead delays.

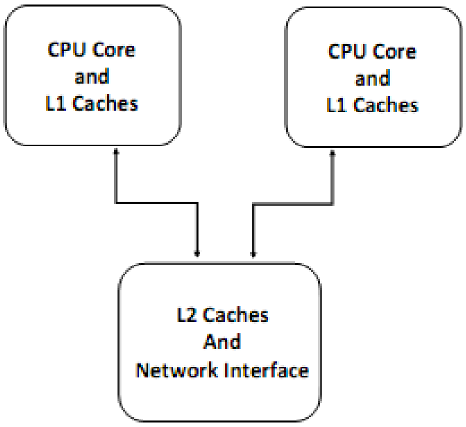


Fig. 6 – Generic Dual-core Processor

In support of multithreading, on-chip synchronization primitives are provided in hardware and enables low synchronization algorithm capabilities [12,13,14]. Low synchronization algorithms offer relaxed ordering constraints on memory operations and reduce or eliminate the synchronization overheads from locking (lock-free) [15]. Ultimately, low synchronization algorithms can improve efficiency and scalability in multithreading. The need for on-chip inter-core communication to handle issues like synchronization and data sharing are being addressed by the developments in Network-on-Chip (NOC) [16,17]. The NOC network links (wires) are shared by many signals. This capability allows NOC to operate on different data packets at one time, achieving a high level of data parallelism, and providing separation between computation and communication on a multi-core processor system. As a consequence, NOC provides high throughput performance and scalability when compared to point-to-point or shared bus communication architectures. The benefits of NOC to multi-core processors could be considerable.

As multi-core systems mature, efforts like NOC may address the problem of cache coherence for which multi-core performance can suffer. The cache coherence problem in a parallel multiprocessor environment arises when copies of a shared resource in memory are maintained in local caches and modified without maintaining the consistency between the cache and memory [18]. This problem is amplified in multi-core and many-core systems where the sharing of the cache between cores introduces an intervening level of *spatial* complexity between local caches and the multi-core processor memory. *Temporal* locality of cache consistency also suffers.

Sorting out the performance issues of a generic multi-core processor system is left in the hands of the application's developer. The details of achieving optimal performance on a dual- or multi-core system is, in some sense, an extension of code development for traditional single-core multiprocessors. One difference arises from the hierarchical cache organization of multi-core systems. This one modification can skew the *spatial* and *temporal* locality of references depending on the multithreaded demands of the application. The skewing of references may result in positive or negative performance behaviors. However, it is a new programming or thread scheduling issue that must be confronted by the code developer. The art of multi-core programming is to devise multithreaded scheduling or throttling schemes to deal with the interactions between shared cores and the shared L2 cache resource. As the multi-core processor industry matures, manufacturers and their designers may provide better hardware support to facilitate the programmer's tasks. Until that time, it is the struggles of the programmer in extracting the last ounce of performance, and in so doing provide examples that guides the industry in developing new and better architectures.

4.2. GPUS

In this section, the architecture of the Nvidia GeForce GTX 280 is described as an example of a generic GPU architecture. The GTX 280 supports shared memory, atomic operations, double precision floating point instructions, 240 cores that run at 1.3GHz, and manufacturing that uses the 65 nm fabrication process. The Nvidia GPU is a streaming SIMD processor, whose kernel represent single program multiple data (SPMD) programming strategies. This interesting hybrid model is flexible but introduces constraints that must be understood to avoid performance degradations.

Fig. 7. is a high-level view of the Thread Processing Cluster (TPC). The TPC consists of 3 stream multiprocessors (SM) each with eight (8) streaming processors (SPs). Within a SM, each block of eight (8) SPs and raster operation processors (ROP) for graphics share an Instruction Unit (IU), and executes threads in a Single Instruction Multiple Data (SIMD) mode. Any diverging (branching, data stalling, etc.) threads may not execute in parallel. Local shared memory in each of the three SMs allows associated SPs to share data with other intra-SPs. The shared memory organization is essential to the performance of SPs. Shared memory is used to minimize the need to read or write to/or from the global memory subsystem (shown in Fig.8), which are typically very slow. Inside each TPC, eight texture-filtering units (TF) are provided for graphic rendering tasks. A typical memory hierarchy may have 32-bit registers per SP (see Fig.9). SMs also have access to constant and texture caches. Constant and texture caches are read-only and important to SMS performance since they can be accessed quickly.

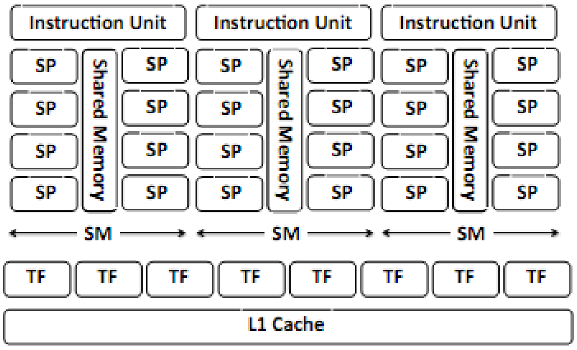


Fig. 7 – Thread Processing Cluster (TPC)

Fig. 8, depicts the GeForce GTX 280 architecture. The GeForce GTX 280 provides 10 TPCs, with three (3) SMs per TPC. The GPU employs synchronous multithreading. It has a hardware-based thread scheduler manages that schedules threads across the TPC. If a thread is stalled on a memory access, the scheduler can context switch to another thread with little overhead. A typical GPU has upwards of 30,720 threads on a chip. The architecture includes texture caches and memory interface units. The texture caches are used to combine memory accesses more efficiently and provide higher bandwidth memory read/write operations. The texture cache is a level 2 (L2) cache and is shared by all TPCs. The "atomic" operations perform atomic read-modify-write operations to global memory and facilitates parallel reductions and parallel data structure management. Atomic operations are integer operations in global memory that provide associative operations on signed/unsigned ints; add, sub, min, max, and, or, xor; increment, decrement; exchange; and compare and swap operations. Table 1 provides a summary of some important parameter values that define the Nvidia GTX 200 series [19,20].

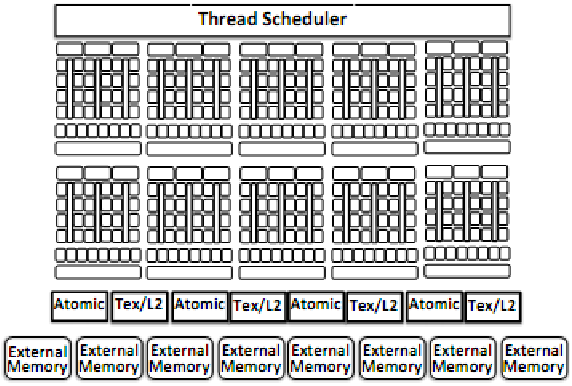


Fig. 8 – GeForce GTX 280

Table 1.– GPU Specifications

SM Resources	
SP	8 per SM
Registers	16,384 per SM
Shared Memory	16KB per SM
Caches	
Constant Cache	8KB per SM
Texture Cache	6 – 8KB per SM
Shared Memory	16KB per SM
Programming Model	
Wraps	32 threads
Blocks	512 threads max
Registers	128 per thread max
Memory	8 ×128MB, 64-bit
Constant Memory	64 KB total

CUDA is the synonym for the Compute Unified Device Architecture. It is a software specification-programming platform to assist the GPU program developer to interface to the logical GPU processor. Using CUDA, the programmer can specify the thread layout that are organized in blocks and in turn organized into grids. Threads executing within a block can synchronize among themselves. Thread communication between different blocks must be performed through slower global memory. Threads in a block have IDs and can be indexed into 1, 2, or 3 dimensions. A grid consists of multiple thread blocks of the same dimension and size. All threads in a grid execute the same CUDA kernel (SPMD). The CUDA architecture allows a SM to execute blocks one at a time. A "warp" of 32 threads or a half warp of 16 threads defines the number of threads that can execute in parallel. Each block is executed in 32 thread Warps. Warps are the scheduling units for the SM; and thus, thread scheduling is measured in warps. A block size can be defined to be 1 to 512 concurrent threads. At any time, only one warp is executed by a SM. Warps whose next instruction has its operands ready for execution are placed into the ready queue. Ready warps are selected for execution on a prioritized scheduling policy. All threads in a warp execute the same instruction (SIMD) when selected. The hardware is free to assign blocks to any SM at any time. Each block can execute in any order relative to other blocks. Threads in the same block share data and synchronize while doing their portion of the work. Threads in different blocks cannot cooperate.

Fig. 9, illustrates a CUDA memory architecture of the GPU. Global memory (off-chip) is the main means of communicating R/W data between host and memory. Content and Texture memories (off-chip) are available to all threads but when cached, their access latency is very short. Shared memory (on-chip) is shared between threads in the same block. Shared memory is divided into "memory banks" of equal size and is as fast as register when no bank conflicts occur. The amount of shared memory per SM is 16 KB organized into 16 banks.

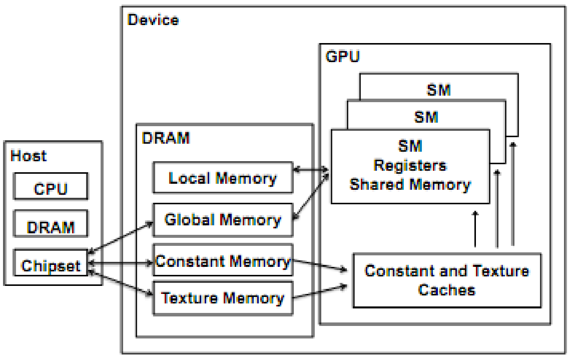


Fig. 9 – CUDA Memory Model

Since the wrap size is 32 threads, a half wrap size of 16 is used for conflict-free bank memory access. Each thread has a private local memory (off-chip). Local memory is to replace the functionality of registers if the kernel requires more registers than are available to a thread. Load/store instructions swap register memory loads and stores in and out of the local memory, which slows down the program performance. CUDA threads may access data from multiple memory spaces during their execution. Finally, all threads have access to the same global memory. Registers are very fast and play a significant role in performance tuning.

The CUDA SPs support lightweight threads designed to execute in SIMD mode. CUDA cores lack their own register files or L1 caches. They do not have multiple function units for each data type (floating point and integer) or load/store unit for accessing memory. Each CUDA core has a pipelined floating-point unit (FPU), a pipelined integer unit, some logic for dispatching instructions and operands to these units, and a queue for holding results.

CUDA SPs are optimized for multithreaded performance but not optimized for single-threaded performance [21].

GPUs provide fine-grain multithreading to offset the extreme latencies incurred from slow main memory accesses. The characteristics of fine-grain multithreading require very high thread utilization [22,23]. Multithreading for the GPU requires explicit optimization of the number of registers used by the kernel and the number of threads per block. The SIMD nature of each kernel execution, dictates that simpler and smaller kernels (that are written in SPMD fashion) can provide better performance. Smaller computational kernels require less registers and are less likely to have data stalls, branching or frequent communications. GPU kernels must be organized and scheduled to maintain a balance between *temporal* and *spatial* locality. Application programs that exhibit regularity in their computational structure are, therefore, prime candidates to profit from the use of GPUs as general-purpose applications on GPUs (GPGPU). The program developers have the primary responsibility to perform microsurgery on applications for implementation on the GPU. The programming task is arduous but can be rewarding. The GPU program developers must remind themselves that fine-grained SIMD processing can return high performance gains but only if there are enough resources and allows only limited communications. Of course not all program applications can support the number of threads required to sustain high GPU performance. As a consequence, the GPU has emerged as a co-processor to the CPU [24].

In summary, the GPU architecture and the multithreading interactions can be a very complicated and dynamic environment for which to design optimal programs. Multithreading, if poorly coordinated, can result in large latencies in operand accesses and contenting thread delays. The design of efficient GPU programs requires assessing the latencies within the components of the memory hierarchy; and uses this knowledge to develop a mapping of multithreading strategies that accrue benefits from the aggregation of data utilization according to regular access patterns. Ultimately, it is the *spatial* and *temporal* localities that include read/write accesses, speeds of the different memory modules, and the threaded structure of the kernel's execution blocks that dictates the performance that can be achieved.

5. CONCLUSION

Assessing the programming issues of modern processors appears to be a confusing and challenging task. This presentation argues that by employing two simple principles of an expanded notion of *spatial* and *temporal* locality some of the most intricate pitfalls and traps can quickly be exposed and reasoned about. This is not to imply that the programming techniques will be easy to derive but rather that performance issues can be isolated and dealt with by using a more rational algorithmic approach. In multi-core systems, the sharing of the cache between several cores has the unintended downside of incurring higher data access congestion between cores. In addition, cache coherence also becomes a performance consideration over which the programmer has some degree of explicit control. In future evolutions of multi-core systems, new hardware and software technologies may mitigate or solve these problems.

The massively parallel programming paradigm of the GPU exemplifies the code developer's dilemma in fully understanding the organization of the system's architecture. Before attempting to write a single line of program code, the GPU's *spatial* and *temporal* organization should be studied in considerable detail. The optimal performance of a multithreaded GPU architecture demands that all threads be active at any one time and that they are all making uniform progress. This is a challenge that requires practice and knowledge based upon trial and error. GPU programming practices can be gained by learning techniques (tricks) from experienced GPU program developers; however, it is more important to learn how to design one's own programming strategies, as modern computer systems will definitely evolve away from current technologies and hardware organizations.

The art of high performance programming is a combination of first principles taken as a foundation to unravel the bottlenecks and latencies associated with the architectures of different systems. Equipped with this information, the programmer must articulate appropriate coding techniques, task-scheduling schemes, and resource allocations to mitigate or eliminate perceived latencies and bottlenecks within the framework of a given architecture.

6. REFERENCES

[1] P.J. Denning, S.C. Schwartz, Properties of the working-set model, *Communications of the ACM*, (15) 3 (1972), pp. 191-198.

[2] P.J. Denning, The locality principle, *Communications of the ACM*, (48) 7 (2005), pp. 19-24.

[3] M. Wolfe, *High-Performance Compilers for Parallel Computing*, Addison Wesley, June 16, 1995.

[4] J.L. Gustafson, Reevaluating Amdahl's law, *Communications of the ACM*, (31) 5 (1988), pp. 532-533.

[5] G. Amdahl, Validity of the single processor approach to achieving large-scale computing capabilities, *AFIPS Conference Proceedings*, (30) (1967), pp. 483-485.

[6] D.H. Bailey, Twelve Ways to Fool the Masses When Giving Performance Results on Parallel Computers, *RNR Technical Report* RNR-91-020, June 11, 1991.

[7] S. Saini, D. Talcott, D. Jespersen, J. Djomehri, H. Jin, and R. Biswas, scientific application-based performance comparison of SGI Altix 4700, IBM POWER5+, and SGI ICE 8200 Supercomputers, *Proceedings of the ACM/IEEE conference on Supercomputing, SC'08*, 2008.

[8] S. Hily and A. Seznec, Contention on 2nd level cache may limit the effectiveness of simultaneous multithreading, *Technical Report* PI-1086, IRISA, 1997.

[9] J. Huh, C. Kim, H. Shafi, L. Zhang, D. Burger, and S.W. Keckler, A NUCA substrate for flexible CMP cache sharing, *Proceedings of the 19th annual international conference on Supercomputing ICS'05*, 2005, pp. 31-40.

[10] W.-D. Weber and A. Gupta, Exploring the benefits of multiple hardware contexts in a multiprocessor architecture: Preliminary results, *Proceedings of the 16th Annual International Symposium on Computer Architecture*, June 1989, pp. 273-280.

[11] B.J. Smith, Architecture and applications of the HEP multiprocessor computer system, *SPIE*, (298) (1981), pp. 241-248.

[12] G.R. Andrews, Paradigms for process interaction in distributed programs, *ACM Computing Surveys*, 1991.

[13] P.E. McKenney and J. Slingwine, Efficient kernel memory allocation on shared-memory multiprocessors, *In USENIX Conference Proceedings*, Berkeley CA, February 1993.

[14] M.I. Reiman and P.E. Wright, Performance analysis of concurrent-read exclusive-write, *ACM*, (February 1991), pp. 168-177.

[15] J. Sartori and R. Kumar, Low-overhead, high-speed multi-core barrier synchronization, high performance embedded architectures and compilers, *Lecture Notes in Computer Science*, (5952) (2010), pp. 18-34.

[16] M. Amde, T. Felicijan, A. Efthymiou, D. Edwards, and L. Lavagno, Asynchronous on-chip networks, *IEE Proceedings Computers and Digital Techniques*, (152) 02 (2005).

[17] A.O. Balkan, M.N. Horak, G. Qu, U. Vishkin, Layout-accurate design and implementation of a high-throughput interconnection network for single-chip parallel processing, *Proc. IEEE Symp. on High Performance Interconnection Networks* (Hot Interconnects), August 2007.

[18] P. Stenstrom, A survey of cache coherence schemes for multiprocessors, Computer, (23) 6 (1990), pp. 12-24.

[19] Nvidia, "Compute Unified Device Architecture Programming Guide Version 2.0," http://developer.download.nvidia.com/compute / cuda/2 0/docs/NVIDIA CUDA Programming Guide 2.0.pdf.

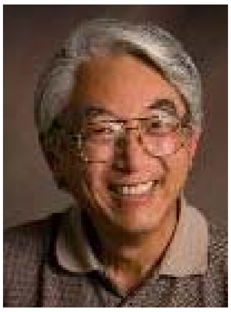
[20] Nvidia, "NVIDIA GeForce GTX 200 GPU Architectural Overview," http://www.nvidia. com/docs/IO/55506/GeForce GTX 200 GPU Technical Brief.pdf, May 2008.

[21] T. Halfhill, Looking Beyond Graphics NVIDIA's Next-Generation CUDA Compute and Graphics Architecture, http://www.nvidia. com/content/PDF/fermi_white_papers/T.Halfhill_Looking_Beyond_Graphics.pdf

[22] A. Kumar, Tips for speeding up your algorithm in the CUDA programming, http://www. mindfiresolutions.com/Tips-for-speed-up-your-algorithm-in-the-CUDA-programming-399.php.

[23] N. Satish, M. Harris and M. Garland, Designing efficient sorting algorithms for manycore GPUs, *Proc. 23rd IEEE International Parallel and Distributed Processing Symposium*, May 2009.

[24] A. Lippert, NVIDIA GPU Architecture for General Purpose Computing, http://www.cs.wm. edu/~kemper/cs654/slides/nvidia.pdf



Dr. Robert E. Hiromoto*, received his Ph.D. degree in Physics from University of Texas at Dallas. He is professor of computer science at the University of Idaho. His areas of research include information-based design of sequential and parallel algorithms, decryption techniques using set theoretic estimation, parallel graphics rendering algorithms for cluster-based systems, and secure wireless communication protocols.*

CiSj

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

USE OF ARTIFICIAL INTELLIGENCE TECHNIQUES FOR PROGNOSTICS: NEW APPLICATION OF ROUGH SETS

Janusz Zalewski [1], Zbigniew Wojcik [2]

[1] Deptartment of Software Engineering, Florida Gulf Coast University
Fort Myers, FL 33965, USA
zalewski@fgcu.edu, http://www.fgcu.edu/zalewski/

[2] Smart Machines
13703 Morningbluff Drive, San Antonio, TX 78216, USA
wojcik.zbigniew@sbcglobal.net, http://smart-ma.com/

Abstract: *The objective of this paper is to set the context for the potential application of rough sets in prognostics. Prognostics is a field of engineering, which deals with predicting faults and failures in technical systems. Engineering solutions to respective problems embrace the use of multiple Artificial Intelligence (AI) techniques. The authors, first, review selected AI techniques used in prognostics and then propose the application of rough sets to build the system health prognostication model.*

Keywords: *Prognostics, Model-based Algorithms, Data Driven Algorithms, Rough Sets.*

1. INTRODUCTION

This paper builds on a previous limited survey of AI techniques in prognostics [1]. Since the first comprehensive survey of AI methods used for prognostics, completed in 2007 [2], a number of new algorithms based on various AI approaches have been either developed or applied to prognostics problems. Moreover, the entire discipline of Prognostics and Health Management (PHM) has been significantly developed during this time, and even though no specific new survey paper has been published, perhaps with one exception [3], several articles appeared, which summarize the accomplishments thus far, or set up the scene for the future [4-8].

For the purpose of this paper, we adopt the terminology and classification of AI methods for prognostics as outlined in [2]. Fault *diagnostics* is defined as fault isolation and fault identification, that is, making attempts to "determine the location of fault" (fault isolation) and actually "determining what is wrong" (fault identification). Fault *prognostics* is defined as "determining when a failure will occur based conditionally on anticipated future usage", which essentially means predicting how much time is left before a failure occurs. The fundamental term used in prognostics is the Remaining Useful Life (RUL) of a device or other equipment defined as "an estimation of a remaining life of a component prior to occurrence of a failure" [8].

This paper concerns only prognostics of technical systems, including electromechanical and electronic systems, leaving out other disciplines where prognostics is essential for proper operation, such as production systems and medical applications, for example. Respective survey papers exist, which review issues and techniques related to these other areas, for example [9] in medical prognostics.

Limited to the discussion of prognostics in technical systems, this paper is structured as follows. Section 2 discusses the taxonomy of prognostics algorithms, which is followed by a brief review of two basic categories of these algorithms in Section 3 (model based algorithms) and Section 4 (data-driven algorithms), including a discussion of the application of rough sets in prognostics. This is followed by a conclusion in Section 5.

2. PROGNOSTICS ALGORITHMS TAXONOMY

The essential categorization of prognostics algorithms, shown in Fig. 1, is based on the distinction between using analytical models (mathematical equations) for prediction, which leads to *model-based* approaches, and deriving conclusions about system behavior from the analysis of measurement data, which leads to *data-driven* approaches [2]. This categorization is not disjoint, however, because analytical models can be enhanced and refined with the use of experimental data, and behavioral data can be analyzed using mathematical methods, which leads to *hybrid* approaches. Nevertheless, for the purpose of this paper such categorization seems logical, since it captures the essential difference between two views of approaching prognostics.

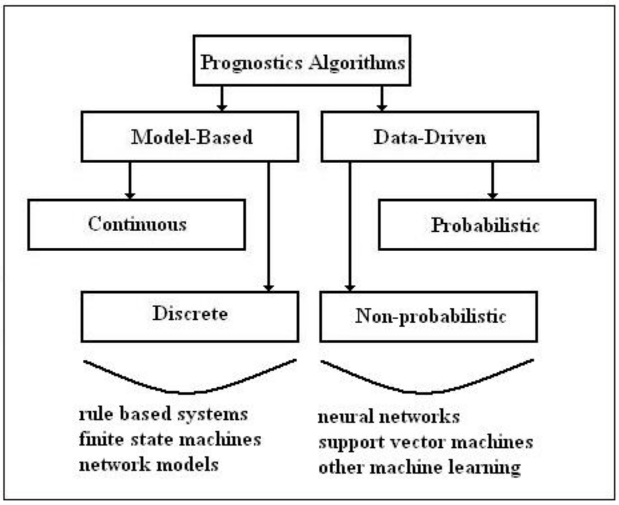


Fig. 1 – Taxonomy of prognostics algorithms

Model-based approaches can vary significantly, since there are a variety of mathematical theories one can use to describe the operation of technical systems, and they may include: differential equations, finite state machines, rule based systems, queuing theory, and other discrete models, such as various sorts of networks: Petri nets, Bayesian belief networks, and others. Many of these models are not purely analytical and use inference techniques typical to AI.

Data-driven models, on the other hand, rarely include analytical knowledge, therefore they all involve some sort of inference. When probability densities governing the distribution of data are available, then one can use well defined statistical methods, for example, linear regression. However, exact knowledge of probability distributions is rarely the case, so methods, which handle incomplete information are applied, including neural networks, support vector machines, and other machine learning approaches, in general.

Even though model-based systems are ideally based on mathematical equations and may not involve any AI reasoning techniques, this is rarely the case. So before venturing into the true AI methods, we first describe the principle of model-based approaches.

3. MODEL BASED ALGORITHMS

Due to a relatively large amount of various theories on which prognostics models can be based, model-based prognostics methods range, accordingly, from rather simple, deterministic, physics based models, to more complicated models, where the determination of critical model parameters may be very involved. Essentially, model-based prognostics can be derived from model-based techniques for fault detection, isolation and diagnosis, which have been used for a relatively long time [10-11]. The process illustrated in Fig. 2, developed for fault diagnosis, can be extended for prognostics by adding prediction estimates for respective model parameters. Most of the applications of model-based techniques do not necessarily use AI methods, unless they are hybrid, if analytical models are accurate enough not to require additional techniques to reason about data.

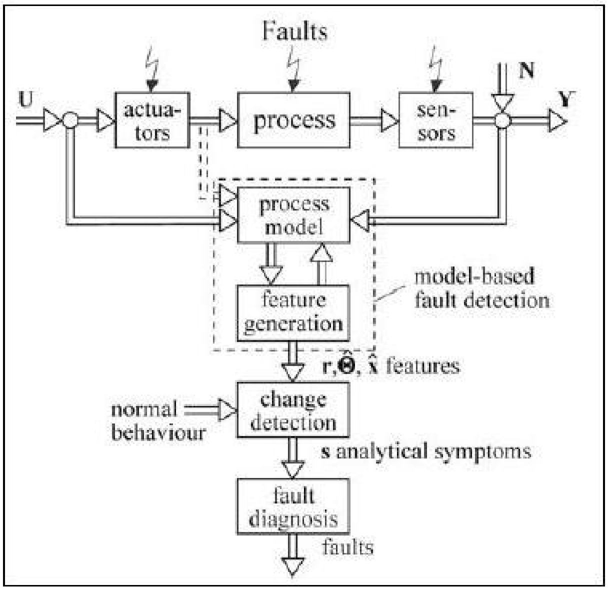


Fig. 2 – General scheme of model-based detection and diagnosis [10]

3.1. REVIEW OF SELECTED MODEL BASED TECHNIQUES

Luo et al. [12] describe such a model-based prognostic process, which starts with building a system model by using singular perturbation techniques of control theory. Based on the system model, Monte Carlo simulations are performed under different load conditions. Then, a prognostic model for component degradation, using Interacting Multiple Models (IMM) and parameter estimation method, is constructed based on simulated data. The degradation measure is tracked using and IMM estimator and the remaining life of the system is estimated under different future usage scenarios. The process is illustrated in a suspension system of an automobile, with a failure mode involving a crack in the suspension spring caused by fatigue.

Thumati and Jagannathan [13] propose a more sophisticated scheme addressing both state and output faults by considering separate time profiles. The faults are modeled using input and output signals of the system. The fault detection comprises online approximator in discrete time with an adaptive term. An output residual, generated by comparing the fault estimator with that of measured system output, is used as a fault indicator when it exceeds a predefined threshold. A parameter update law is developed to estimate the time to failure, considered as a first step to prognostics. The effectiveness of this approach is demonstrated using a fourth-order multiple-input multiple-output satellite system.

There have been other multiple papers published on model-based prognostics. Hines and Garvey [14] developed a nonparametric prognostic model named PACE (path classification and estimation), which is applied to predict RUL of the steering system of a drill used for deep oil exploration. Sankavaram et al. [15] describe a hybrid approach that uses both a model-based prognosis applied to automotive suspension system and data-driven prognosis to Li-Ion batteries and electronic systems. Application of model-based prognostics methods extends beyond typical machinery and electronics, and is also used in production systems, for example [16].

3.2. PARTICLE FILTERS

Particle filters are the most recently used technique in prognostics and deserve special attention. They are best explained in the context of Kalman filters, used to estimate and predict internal state (over time) of a system described by a set of differential equations of a certain form, given a series of measurements corrupted by noise. The Kalman filter works best in the special case where the system of equations is linear and the system and process noise is Gaussian. Then the Kalman filter provides an optimal analytic solution.

There are several generalizations of the Kalman filter, such as the extended Kalman filter, that make simplifying assumptions about the system's non-linearity, so that the techniques of the linear Kalman filter may be applied to it. However this may cause problems in situations where the system is highly nonlinear or the distribution of the noise is multimodal or highly skewed. It turns out that Particle Filters introduced less than two decades ago [17] are suitable to deal with these sorts of problems.

Particle filters were introduced to solve the problem of state estimation in circumstances where other methods, such as Kalman filters, are not particularly effective [18]. Particle filters use simulation to provide a numerical solution and are able to handle many more general situations than Kalman filter extensions. The only restriction is that the system must meet the Markov condition, which means that when the system makes an update, the new state depends only on the current state and is independent of any of the past states.

Particle filters are numerical methods for estimating the internal states of Bayesian models that use simulation. The system model describes the way the system updates over time and is given by

$$x_k = f_k(x_{k-1}, z_k)$$

where x_k has the distribution $p(x_k|x_{k-1})$ for $k > 0$ and x_0 has the distribution $p(x_0)$.

The vector x is called the state vector, z is a random vector called the process noise that has a known distribution, and f is called the state transition function. The value of the state at time k depends only on the value of the state at $k-1$ and is independent of the values at all previous times.

The measurement model is given by the following equation:

$$y_k = h_k(x_k, v_k)$$

where the vector y is called the measurement vector, v is a random vector called the measurement noise with a known distribution, and h is the observation function. The value of the measurement at time k depends only on the state at time k and is independent of previous measurements.

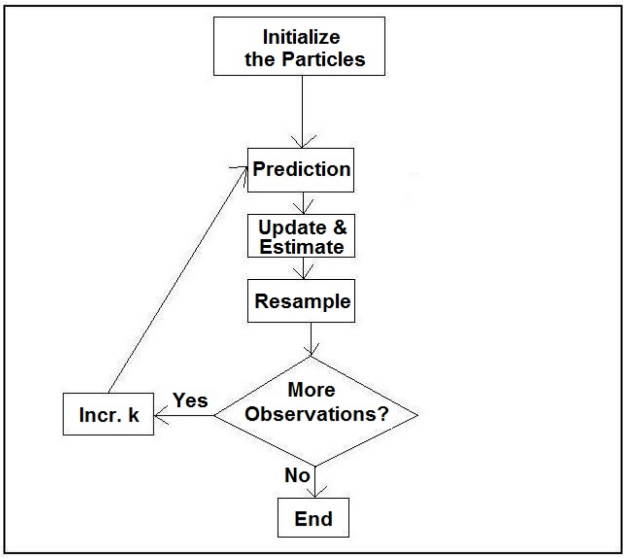


Fig. 3 – Basic steps in the particle filter algorithm

The computational algorithm uses simulation to approximate $p(x_k|y_{1:k-1})$, which is unknown. However it may not be possible to sample from this distribution, so another distribution $q(x_k|y_{1:k-1})$, called importance distribution (proposal) function, is sampled from instead. Although any distribution with a support (set of values for which the distribution is not zero) that is a subset of the support of the prior may be used for q, the algorithm will perform better for a distribution with a shape close to that of $p(x_k|y_{1:k-1})$.

The algorithm is initialized by taking N samples, called *particles*, from the distribution of the initial state, $q(x_0)$. Let's denote the *i*-th particle as x_0^i for each *i* belonging to [1, N]. Each particle is assigned a weight of $w_0^i = p(x_0^i)/q(x_0^i)$. The weights are then normalized by dividing each weight by the sum of all the weights so that the normalized weights $\sim w_0^i$ sum to one. The algorithm then proceeds in three steps for each iteration, as described below and illustrated in Figure 3.

The first step, called *prediction* is to approximate the distribution $p(x_k|y_{1:k-1})$. Finding this exact distribution requires integrating $p(x_k|x_{k-1})p(x_k|y_{1:k-1})$ with respect to x_{k-1} and this integral is not tractable in general, so simulation is used to approximate it. The particles from the previous iteration are updated by taking a sample z_k^i from the process noise for each particle x_{k-1}^i and then evaluating the state transition function: $x_k^i = f(x_{k-1}^i, z_k^i)$.

The second step called *updating* is to estimate the posterior, $p(x_k|y_{1:k})$. Each particle is assigned an initial weight, $w_k^i = p(x_k^i|x_{k-1})/q(x_k^i|x_{k-1})$. The weights are then normalized by dividing each weight by the sum of all the weights so that the normalized weights $\sim w_k^i$ sum to one. Then $p(x_k|y_{1:k})$ may be approximated by the discrete distribution that is equal to $\sim w_k^i$ at each particle x_k^i. An estimate of the state at time *k*, x_k*, may now be found by taking the weighted sum of all the particles.

The third step is called *resampling*. Here a sample of size *N* is taken from this discrete distribution to replace the current set of particles and each new particle is assigned a weight of *1/N*. It is necessary because the set of particles may consist of many particles from highly improbable regions that will result in all but one particle having negligible weights after several iterations, a situation called "degeneracy". When resampling is done, particles with low weights are unlikely to be selected and particles with high weights will likely be selected several times. This ensures that more particles are at regions where the true value of the state is likely to be and helps to avoid degeneracy.

The prediction step is now taken for time *k+1* and the algorithm continues until there are no more observations y_k to consider.

On-going research has shown great promise for application of particle filters to prognostics. They can be effectively used to track progression of system state in order to make estimations of remaining useful life (RUL), which is at the core of system prognostics and health management [19-25].

4. DATA DRIVEN ALGORITHMS

Data-driven algorithms are the ones which are truly based on AI techniques, since they require reasoning about data patters. In this paper, we discuss briefly one such technique, Support Vector Machines, and then pursue toward description of a new technique based on rough sets, which to our knowledge has not been used in prognostics before.

4.1. SUPPORT VECTOR MACHINES

One specific method used in prognostics, which has been mentioned in some surveys [2] but not extensively covered, is Support Vector Machines (SVM), also known as maximum margin classifier. SVM is a method considered as a nonlinear regressive model [26], in which the dependence of a scalar *d* on a vector *X* is described by the formula

$$d = f(X) + v$$

where both the function *f()* and the statistic properties of the additive noise *v* are unknown.

All the available information is a set of training data

$$\{(x_i, d_i)\}$$

where $i = 1 \ldots N$; x_i is the sample value of the input vector *X*, and d_i is a corresponding value of the output *d*. The problem is to give an estimate of the dependence of *d* on *X*. A nonlinear regression is performed by mapping the input vector *X* into a high-dimensional feature space, in which then a linear regression is performed.

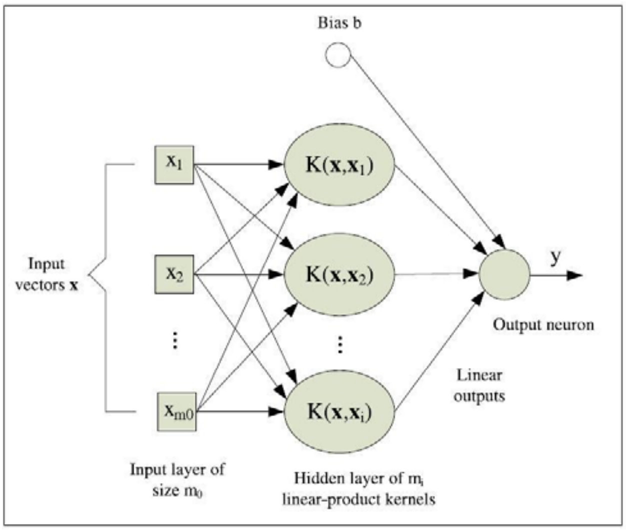


Fig. 4 – Architecture of an SVM [26].

In the architecture of the SVM shown in Fig. 4 [26], the kernel function $K(x, x_i) = \Phi^T(x)\Phi(x_i)$, and $i = 1, 2, \ldots, m_i$ is the dimension of the feature space. It is also assumed that $\Phi(X)$ is defined *a priori* for all *j*. Given such a set of nonlinear transformations, one may define a formula for *y*, an estimate of *d*, as follows

$$y = \Sigma\, w_j\Phi_j(X) + b, \text{ for } j=1...m_j$$

where w_j for $j=1...m_j$ denotes a linear weight connecting the feature space to the output, and *b* is a bias.

Caesarendra et al. [26] use SVM as a part of a more sophisticated approach, in a combination with a probability based approach, to predict a final time of a failure in a bearing. SVM is only a part of the entire procedure trained by kurtosis and the target vector to build the prediction model, which is then utilized to predict the final failure time of individual bearing.

While Caesarendra et al. [26] applied this method to the simulated bearing failure data as well as experimental bearing run-to-failure data, Kim et al. in their paper [27] verified the proposed model on bearing failure data of a high pressure LNG pump. Although the training error for classification of six classes using an SVM classifier was relatively high (18.75%), the authors claim that the estimated RUL follows closely the real remaining life of the machine, thus validates the proposed concept as potentially useful for application in industry.

Most recently, Widodo and Yang [28] used SVM in conjunction with survival analysis in machine health prognostics. In their work, survival analysis utilizes censored and uncensored data collected from condition monitoring (CM) and then estimates the survival probability of failure time of machine components. SVM is trained on data input from CM histories that correspond to target vectors of estimated survival probability. After validation, the SVM is employed to predict failure time of individual units of machine components. The method has been verified for simulation and experimental bearing data.

Multiple other applications of SVM's in prognostics exist, including specific technical systems, such as automotive [29], as well as non-technical systems, such as human patients in medicine [30].

4.2. USE OF ROUGH SETS FOR PROGNOSTICS

Rough sets, invented by Zdzisław Pawlak in 1981, are a mathematical method of dealing with incomplete information and uncertainty [31-32]. As such, they seem ideal to making predictions and prognostics. Nevertheless, there have not been any applications of rough sets to prognostics, except one we've found [33]. In this section, we present a method of using rough sets to build a system health prognostics model, and discuss how it fits into the overall taxonomy of AI methods in prognostics.

The rationale behind using a rough sets approach for health degradation estimation and prediction is three-fold:

a) It sets the focus on splitting the data into two parts: completely healthy and partly degraded (with different levels of degradation),

b) Offers tools to measure the level of health ambiguity and health degradation, and

c) Facilitates intelligent data mining, by considering relationships between unambiguous and ambiguous data, known from the psychology of human perception as intelligent.

A. Mathematical Background. The rough sets approach incorporates the Universe *U* of degradation signals (called test points) and a set *R* of equivalence relations. Examples of a test points are: (a) trf, the set of measurements of turbine fan vibration level over time, (b) tit, the set of Turbine Inlet Temperatures over time.

Each test point *u* □ *U* has its own corresponding equivalence relation *r* □ *R*. Each first level equivalence relation *r* for each test point *u* describes the equivalence class of the measurements *"in healthy condition"* not looked for, because the focus is on health degradation. Health degradation of a part described by the test point *u* is represented by the complement of *u*/*r* in *u.*

Diagnostics of a health problem is frequently based on a specific non-empty intersection of the equivalence relations *"in degraded condition"*. For instance, a disintegration of an engine is likely to happen when

$$TRF/degraded \cap TIT/degraded \neq \emptyset$$

where *degraded* ≡ *"'in degraded condition"*.

Engine *disintegration* is thus representable by non-empty indiscernibility relation [34] over two or more specific equivalence relations, e.g.

$$disintegration \leftarrow IND\{trf\_degraded;\ tit\_degraded\}$$
$$= trf/degraded \cap tit/degraded$$

executed on paired elements that occur at the same time.

Unhealthy situations are detected by rough sets variance. Rough sets variance prevents obscuration of degradation information by measuring distance to healthy interval. Statistical population variance of variable *X*:

$$Var(X) = 1/N\, \Sigma(x\text{-}x^*)(x\text{-}x^*)$$

where $(x\text{-}x^*)$ is a measure of deviation of the values *x* from their mean x^*

The statistical variance obscures degradation information, because it collects into the sum also healthy measurements.

There may be no significant difference between: (a) a signal of a few unhealthy measurements accompanied with a large number of small deviations from the mean and (b) a signal of no unhealthy measurements accompanied with a large number of slightly higher (but still normal) deviations from the mean. Therefore, using rough sets variance makes sense to measure degradations, because rough sets variance of variable X for N measurements is equal [34]

$$rsVar(X) = Median|x - L(X)|$$

where $L(X)$ is the value of the healthy range of X, nearest to x. The value of x must be outside of the healthy range, otherwise: $x - L(X)=0$ and is rejected from considering to the median. Only non-zero degradations from the healthy range $L(X)$ count.

Rough sets variance is non-linear. For a large number of elements, the rough sets variance may also be expressed by [34]

$$rsVar(X) = 1/N \Sigma(x - L(X))(x - L(X))$$

or by

$$rsVm(X) = 1/N \Sigma|x - L(X)|$$

Statistical covariance between variables X and Y is equal:

$$Cov(X, Y) = 1/N \Sigma(x - x^*)(y - y^*))$$

Rough sets covariance between variables X and Y for N measurements is as follows [34]:

$$rsCov(X,Y) = Median(x - L(X))(y - L(Y))$$

Also, rough sets covariance provides very high signal-to-noise ratio, as opposed to the statistical covariance, which incorporates the mean. For a large number of elements the rough sets covariance may also be expressed by

$$rsCov(X) = 1/N \Sigma(x - L(X))(\mathrm{y} - L(Y))$$

Bringing this to the world of health management, the equivalence class of measurements x taken at test point X of a part represented by the equivalence relation r = *in degraded condition* equals

$$x/r = \cap\{x \varepsilon X: |x| > rsVar(X)\}$$

where $|x| > rsVar(X) \equiv r \equiv$ *"degraded"*. The sign ∩ means that r ≡ "degraded" is the indiscernibility relation over all historical data of the test point X.

With the approximate knowledge denoted as *k=(X,"degraded")*, the degraded health at test point X can be approximated by two subsets [32]

$$LOxr = U \{ x \varepsilon x|r: x \subset X\}$$

$$UPxr = U \{ x \varepsilon x|r: x\ crossSection\ X.neq.\ \emptyset\}$$

called, respectively, the lower and upper approximation of health degradation represented by the test point X.

B. Applicability of the Theory. The proposed rough sets mechanism analyzes data recorded from interrelated system components and makes predictions using the degradation levels of these components and health status measurements from historical data.

Statistical failure time does not predict failure of a particular item based on its specific level of degradation, but on statistics of a large number of items. Repairs can be made on a specific part that is less degraded than the statistical one, and an enormous risk is with any part that is degraded more than that statistical. Intelligence is of value here: if one knows, which part is less degraded than expected, one can let it work for some time longer. But if it is used up too much than expected, then it is too dangerous not to make a repair or replacement somewhat earlier than expected.

Conditional maintenance not only costs less, but prevents unexpected crushes. Influence of one part on degradation of another part is also needed to make better predictions, because one cannot have a sensor of degradation at any part of the system. Influences can be learned and collected into a knowledge database. Without the deterministic knowledge of the degradation level and influences on degradations of other parts, no other health status can be determined except the probability of survival for the item being measured. This assumption is true, e.g., when material fatigue leads to a failure.

Application of the proposed model needs knowledge of events co-existing in time and properties of changes of observations over time. All this knowledge can be acquired in real time from historical databases. The proposed approach makes sense of knowledge.

The proposed data fusion system described in [34] consists of terms, which are the products of ratios of degradation and the ratios of influences on the degradations. The parts degradation data fusion system is deterministic for data sequenced by time. It exploits available degradation signals and gets optimized by removal of insignificant terms, e.g., when one part does not influence another. The influences are computed by rough sets correlations. The model is hierarchical and does not limit the object's complexity. Reasoning is deterministic and rule-like chaining is applicable.

Determinism of time sequences allows using simple inequalities with weights learned at the time of major events such as failure times, near-failures or major repairs. Patterns of degradation are created from historical data at times of known failures or major repairs. Predictions are made by: (i) matching current object conditions with the patterns of degradation learned through historical data, and (ii) computing and applying the increments of degradations and of the influences on the degradations learned from the patterns matched, minute-by-minute, to see when possible failures could happen.

A method is offered for computing elementary increments to extrapolate the actual and predicted patterns of degradation to the points of predicted failures and repairs. The system avoids obscuration of rare degradation measurements by healthy measurements by using rough sets: any healthy status is immediately considered irrelevant. Logistics operations can be optimized by accurately preparing to predicted repairs. Influences of environmental degradations that start to affect system at the earliest stages can be traced back.

The existing structural modeling and regression models are equalities, not inequalities. The new proposed reasoning is based on testing the inequality in each modeling record, making a rule that fires an alarm if the measurement data implies a significant level of degradation. Records of lower-level components are listed and executed first, and whenever inequality is true, using its right side as an operand of one of the inequalities that is listed later as a higher-level component. Coefficients are not computed as in structural modeling by minimizing an error, but by a new machine learning from time-sequenced events.

By incorporating rough sets degradation covariances, that is, influences of degradations of parts on each other, the proposed method directly rejects obscuration by all measurements indiscernible by healthy status, automatically sets to 0 all influences of no impact on the system degradation, and retains only critical influences degrading the system. By this the proposed system correctly assesses the critical factors for prediction. Relating health status measurement to the equivalence class of the healthy ranges and not to the mean or to median is the key difference of the proposed approach with respect to statistics. Ratios directly representing health problem level are incorporated, and not the values of the variables as commonly used in statistics.

A numerical example of the application of this method to health prognostics of a jet engine can be found in [34].

5. CONCLUSION AND FUTURE WORK

The objective of this paper was to place rough sets in the context of AI techniques used for equipment health prognostics and outline the applicability of rough set theory for the evaluation of health degradation. In this view, it seems that rough sets allow for accurate data mining and prediction, with such new tools as rough sets variance, rough sets covariance, time-sequenced data fusion model, and deterministic machine learning. The rough sets compound covariance proves successful to mine and detect engine disintegration, and new time-sequenced data fusion model to predict it in real time. Determinism of these new tools assures higher accuracy of predictions and increased precision when mining of historical data.

The proposed rough sets prognostic model considers only data sequenced by time [34]. The time makes data mining deterministic in the domain of health degradations. Predictions are more accurate when data are deterministic, or much less amount of data is needed for the same quality of predictions. Most data analysis tools are concerned with implicit static-like data (i.e., not necessarily sequenced by time) and rather probabilistic type of events, where time plays no significant role. Instead, randomness is important to be statistical. Prediction is inherently time-dependent and it is not easy to apply a statistical inference to it because statistics is basically static. Needles to say, also the basic rough sets theory is concerned with implicit static-like data. The if-then rules generated from this type of data are probabilistic in nature. In contrast to that, the new rough sets model involves only data sequenced by time.

The state-of-the-art machine learning methods do not adapt to the time of an event but to the event itself: such prediction of time of fault or time of repair is not direct and therefore incurs inaccuracies. A near-deterministic approach proposed allows making predictions based on degradation patterns gathered automatically for different applications and at the time moments of failures. Prediction of failures and the health status takes place by deterministic application of these increments to the actual patterns of degradation matched. Statistical (and non-deterministic) predictions would require more cases, which are not always available.

Machine learning is commonly understood as a slow adjustment of weights, and not just by setting the weights using knowledge. Rough sets variance and covariance set values and weights of the new deterministic data fusion model, which constitute knowledge model for time-sequenced prognostic, faster and more accurate compared to statistical or stochastic adjustment of weights.

The advantages of the proposed machine learning that computes each weight as unknown from the equations are: (i) high precision of classification; (ii) high speed of accurate machine learning even from one example to classify one class of degradation cases; and (iii) simplicity, by adjusting weights directly on the level of dichotomization concepts and by this easiness to interpret the machine learning process.

This new approach uncovers large areas of applications for rough sets, including predictions of health problems, diagnostics, mission readiness evaluation, equipment maintenance, optimization of logistics operations, avoidance of unnecessary repairs and high risks of failures with crushes, and data fusion for decision making.

6. ACKNOWLEDGEMENT

J. Zalewski would like to thank members of the NASA Prognostics Center of Excellence, especially Dr. Kai Goebel and Dr. Abhinav Saxena, for introducing him to the latest developments in

prognostics.

Part of this work has been funded by a grant SBAHQ-10-I-0250 from the U.S. Small Business Administration (SBA). SBA's funding should not be construed as an endorsement of any products, opinions, or services. All SBA-funded projects are extended to the public on a nondiscriminatory basis.

7. REFERENCES

[1] J. Zalewski, Latest development in artificial intelligence techniques for prognostics – a brief survey, *Proc. PRIP 2011, 11th International Conference on Pattern Recognition and Information Processing*, Minsk, Belarus, May 18-20, 2011, pp. 21-24.

[2] M. Schwabacher, K. Goebel, A survey of artificial intelligence for prognostics, *Proc. 2007 AAAI Fall Symposium.* Arlington, Virginia, November 8-11, 2007, AAAI Press, Menlo Park, Calif., pp. 108-115.

[3] J.W. Hines, A. Usynin, Current computational trends in equipment prognostics, *Intern. Journal of Computational Intelligence Systems*, (1) 1 (2008), pp. 94-102.

[4] Y. Gao et al., Research status and perspectives of fault prediction technologies in prognostics and health management system, *Proc. ISSCAA 2008, 2nd Int'l Symposium on Systems and Control in Aerospace and Astronautics,* Shenzhen, China, December 10-12, 2008.

[5] Y.G. Bagul, I. Zeid, S.V. Kamarthi, A framework for prognostics and health management, *Proc. 2008 IEEE Aerospace Conference*, Big Sky, Montana, March 1-8 2008.

[6] M. Pecht, R. Jaai, A prognostics and health management roadmap for information and electronics-rich systems, *Microelectronics Reliability*, (50) (2010), pp. 317-323.

[7] E. Zio, The challenges of system health management and failure prognostics, *IEEE Trans. on Reliability*, (58) 2 (2009).

[8] S. Uckun, K. Goebel, P.J.F. Lucas, Standardizing research methods for prognostics, *Proc. PHM 2008, Int'l Conference on Prognostics and Health Management.* Denver, Colo., October 6-9, 2008.

[9] P.J.F. Lucas, A. Abu-Hanna, Prognostics methods in medicine, *Artificial Intelligence in Medicine*, Vol. 15, 1999, pp. 105-119.

[10] J. Gertler, Survey of model-based failure detection and isolation in complex plants, *IEEE Control Systems*, (8) 3 (1988), pp. 3-11.

[11] R. Isermann, Model-based fault detection and diagnosis – status and applications, *Annual Review in Control*, (29) 1 (2005), pp. 71-85.

[12] J. Luo et al., Model-based prognostic techniques applied to a suspension system, *IEEE Trans. on Systems, Man and Cybernetics – Part A: Systems and Humans*, (38) 5 (2008), pp. 1156-1168.

[13] B.T. Thumati, S. Jagannathan, A model-based fault-detection and prediction scheme for nonlinear multivariable discrete-time systems with asymptotic stability guarantees, *IEEE Trans. on Neural Networks*, (21) 3 (2010), pp. 404-423.

[14] J.W. Hines, D.R. Garvey, Non-parametric model-based prognostics, *Proc. 2008 Annual IEEE 2008 Annual Reliability and Maintainability Symposium*, Las Vegas, NV, January 28-31, 2008, pp. 469-474.

[15] C. Sankavaram et al., Model-based and Data-driven Prognosis of Automotive and Electronic Systems, *Proc. 5th Annual IEEE Conference on Automation Science and Engineering*, Bangalore, India, August 22-25, 2009, pp. 96-101.

[16] P. Maier et al., Integrating model-based diagnosis and prognosis in autonomous production, *Proc. PHM'09, First International Conference on Prognostics and Health Management*. San Diego, Calif., September 27 – October 1, 2009.

[17] N.J. Gordon, D.J. Salmond, A.F.M. Smith, Novel approach to Nonlinear/Non-Gaussian Bayesian state estimation, *IEE Proceedings-F*, Vol. 140, 1993, pp. 107-113.

[18] B. Ristic, S. Arulampalam, N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications,* Artech House, 2004.

[19] M. Orchard et al., Advances in uncertainty representation and management for particle filtering applied to prognostics, *Proc. PHM 2008, 1st Intern. Conference on Prognostics and Health Management*, October 6-9, 2008.

[20] M. Daigle, K. Goebel, Model-based prognostics with fixed-lag particle filters, *Proc. PHM 2009, 2nd Annual Conference on Prognostics and Health Management*, San Diego, Calif., September 27 – October 1, 2009.

[21] M. Orchard, G. Vachtsevanos, A particle filtering approach for on-line fault diagnosis and failure prognosis, *Trans. of the Institute of Measurement and Control*, (31) 3-4 (2009), pp. 221-246.

[22] S. Butler, J. Ringwood, Particle filters for remaining useful life estimation of abatement equipment used in semiconductor manufacturing, *Proc. SysTol 2010, Conference on Control and Fault-Tolerant Systems,* Nice, France, October 6-8, 2010, pp. 436-441.

[23] M. Orchard et al., Risk-sensitive particle-filtering-based prognosis framework for estimation of remaining useful life in energy storage devices, *Studies in Informatics and Control*, (19) 3 (2010), pp. 209-218.

[24] E. Bechhoefer, S. Clark, D. He, A state space model for vibration based prognostics, *Proc. PHM 2010, 3rd Annual Conference on Prognostics and Health Management*, Portland, Ore., October 10-14, 2010.

[25] M. Daigle, K. Goebel, Multiple damage progression paths in model-based prognostics, *Proc. 2011 IEEE Aerospace Conference*, Big Sky, Montana, March 5-12, 2011.

[26] W. Caesarendra, A. Widodo, B.-S. Yang, Combination of probability approach and support vector machine towards machine health prognostics, *Probabilistic Engineering Mechanics*, Vol. 26, 2011, pp. 165-173.

[27] H.-E. Kim et al., Machine prognostics based on health state estimation using SVM, *Proc. WCEAM-IMS 2008, Third World Congress on Engineering Asset Management and Intelligent Maintenance Systems Conference*, Beijing, China, October 27-30, 2008, pp. 834-845.

[28] A. Widodo, B.-S. Yang, Machine Health Prognostics Using Survival Probability and Support Vector Machine. *Expert Systems and Applications*, Vol. 38, 2011, pp. 8430-8437.

[29] D.T. Vollmer, M. Manic, CI-PASM – Computational intelligence based prognostic automotive system model, *Proc. ICIEA 2009, 4th IEEE Conference on Industrial Electronics and Applications*, Xi'an, China, May 25-27, 2009, pp. 3714-3719.

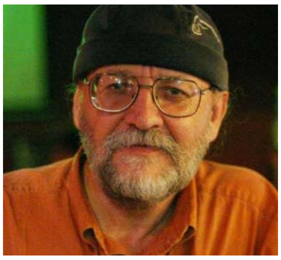
[30] V. Van Belle et al., Survival VSM: a practical scalable algorithm, *Proc. ESANN 2008, European Symposium on Artificial Neural Networks – Advances in Computational Intelligence and Learning*, Bruges, Belgium, April 23-25, 2008.

[31] Z. Pawlak, Rough Sets, *International Journal of Information and Computer Sciences*, Vol. 11, No. 5, pp. 341-356, 1982. Available at: http://chc60.fgcu.edu/Images/articles/PawlakOriginal.pdf

[32] Z. Pawlak, *Rough Sets – Theoretical Aspects of Reasoning about Data,* Kluwer Academic Publishers, 1991.

[33] S. Lee, *An Architecture for a Diagnostic/Prognostic System with Rough Set Feature Selection and Diagnostic Decision Fusion Capabilities*. PhD Dissertation, Georgia Tech, December 2002.

[34] Z. Wojcik, System Health Prognostic Model Using Rough Sets. *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. Lecture Notes in Computer Science*, Vol. 3641, pp. 522-531, 2005.



Janusz Zalewski *is a professor of Computer Science and Software Engineering at Florida Gulf Coast University, in Ft. Myers, Florida, USA. He previously worked at nuclear research labs in Europe and the U.S. and consulted for the government and industry. He also had fellowships at NASA and Air Force Research Labs. His research interests include real-time embedded systems, prognostics of complex systems, and software engineering education.*

Zbigniew Wojcik *obtained Ph.D. from the Polish Academy of Sciences in the area of pattern recognition. He is the autor of several models, algorithms and methods for prognostics and forecasting, speech signal recognition, computer vision and image processing, non-blurring image filtering and enhancement, methods of detection of complete image features independent of positions and rotations of the objects, and parallel shape recognition algorithms. He is a Principal at Smart Machines.*

Vasyl Koval, Oleh Adamiv, Anatoly Sachenko, Viktor Kapura, Hubert Roth
METHOD OF ENVIRONMENT RECONSTRUCTION USING EPIPOLAR IMAGES

In this paper the method for 3D Environment reconstruction using epipolar images is presented. Method allows to fuse stereoimages within certain time depending on acceptable computational resources.

Василь Коваль, Олег Адамів, Анатолій Саченко, Віктор Капура, Hubert Roth
МЕТОД РЕКОНСТРУКЦІЇ СЕРЕДОВИЩА З ВИКОРИСТАННЯМ ЕПІПОЛЯРНОГО ЗОБРАЖЕННЯ

В даній статті представлено метод для 3D реконструкції середовища, що використовує епіполярні зображення. Метод забезпечує злиття стереозображень в межах певного часу залежно від прийнятних обчислювальних ресурсів.

Василий Коваль, Олег Адамив, Анатолий Саченко, Виктор Капура, Hubert Roth
МЕТОД РЕКОНСТРУКЦИИ СРЕДЫ С ИСПОЛЬЗОВАНИЕМ ЭПИПОЛЯРНОГО ИЗОБРАЖЕНИЯ

В данной статье представлен метод для 3D реконструкции среды, использующий эпиполярные изображения. Метод обеспечивает слияние стереоизображения в пределах определенного времени в зависимости от приемлемых вычислительных ресурсов.

Aleksandra Maksimova
THE MODEL OF DATA PRESENTATION WITH FUZZY PORTRAITS FOR PATTERN RECOGNITION

This paper deals with a data presentation model based on fuzzy portraits. The fuzzy portraits are formed by integral characteristics of pattern classes. It is the basis for fuzzy classifier construction. It is determined that further division of some classes of images into clusters increases the quality of pattern recognition algorithm. The main idea of fuzzy clustering for fuzzy portraits creating and problem of adequate fuzzy partition choice is considered. The paper provides the stages of fuzzy production knowledge base construction on the basis of fuzzy portraits. The local validity measure for fuzzy portrait is defined. The problem of identification in chemical and food industries is considered as an application of this approach.

Олександра Максимова
МОДЕЛЬ ПРЕДСТАВЛЕННЯ ДАНИХ З НЕВИЗНАЧЕНИМИ ПОРТРЕТАМИ ДЛЯ РОЗПІЗНАВАННЯ ОБРАЗІВ

В статті розглянуто модель представлення даних нечіткими портретами. Нечіткі портрети формуються на основі інтегральних характеристик класів образів та є базисом для формування правил виводу нечіткого класифікатора. Визначено, що подальший поділ деяких класів образів на підгрупи дозволяє підвищити якість алгоритму класифікації. В роботі розглянуто основну ідею застосування нечіткої кластеризації для формування нечітких портретів класів образів і вибору адекватного розбиття класу образів на кластери. В статті представлені основні кроки щодо побудови бази знань нечіткого класифікатора. Введено локальну міру адекватності нечіткого портрета. Як приклад застосування даного підходу розглядається задача ідентифікації в хімічній та харчовій промисловості.

Александра Максимова
МОДЕЛЬ ПРЕДСТАВЛЕНИЯ ДАННЫХ С НЕОПРЕДЕЛЕННЫМИ ПОРТРЕТАМИ ДЛЯ РАСПОЗНАВАНИЯ ОБРАЗОВ

В статье рассматривается модель представления данных нечеткими портретами. Нечеткие портреты формируются на основе интегральных характеристик классов образов и являются базисом для формирования правил вывода нечеткого классификатора. Определено, что дальнейшее разделение некоторых классов образов на подгруппы позволяет повысить качество алгоритма классификации. В работе рассматривается основная идея применения нечеткой кластеризации для формирования нечетких портретов классов образов и проблема выбора адекватного разбиения класса образов на подклассы. В статье представлены основные шаги построения базы знаний нечеткого классификатора. Введена локальная мера адекватности нечеткого портрета. В качестве приложения данного подхода рассматривается задача идентификации в химической и пищевой промышленности.

Vladimir E. Podolskiy

DIVIDING OUTLIERS INTO VALUABLE AND NOISE POINTS

A great number of different clustering algorithms exists in computer science. These algorithms solve the task of dividing data set into clusters. Data points which were not included into one of these clusters are called 'outliers'. But such data points can be used for the discovery of unusual behavior of the analyzed systems. In this article we present a novel fuzzy based optimization approach for division these outliers into two classes: interesting (usable for solving the problem) outliers and noise.

Володимир Е. Подольський

ДІЛЕННЯ ВИКИДІВ НА КОРИСНІ ДАНІ І ШУМ

У інформатиці існує величезне число різних алгоритмів кластеризації. Ці алгоритми вирішують задачу розбиття набору даних на кластери. Дані, що не потрапляють ні в один з цих кластерів, називаються "викидами". Але ці дані (викиди) можуть бути використані для відкриття незвичайної поведінки аналізованих систем. У даній статті представлений новий, заснований на нечітких множинах, оптимізаційний підхід до розділення викидів на два класи: цікаві (корисні для вирішення завдання) викиди і шум.

Владимир Э. Подольский

ДЕЛЕНИЕ ВЫБРОСОВ НА ПОЛЕЗНЫЕ ДАННЫЕ И ШУМ

В информатике существует огромное число различных алгоритмов кластеризации. Эти алгоритмы решают задачу разбиения набора данных на кластеры. Данные, которые не попадают ни в один из этих кластеров, называются "выбросами". Но эти данные (выбросы) могут быть использованы для открытия необычного поведения анализируемых систем. В настоящей статье представлен новый, основанный на нечётких множествах, оптимизационный подход к разделению выбросов на два класса: интересные (полезные для решения задачи) выбросы и шум.

Sergey Tulyakov, Rauf Kh. Sadykhov

FACE RECOGNITION SYSTEM FOR MACHINE READABLE TRAVEL DOCUMENTS

This paper presents an upright frontal face recognition system, aimed to recognize faces on machine readable travel documents (MRTD). The system is able to handle large image databases with high processing speed and low detection and identification errors. In order to achieve high accuracy eyes are detected in the most probable regions, which narrows search area and therefore reduces computation time. Recognition is performed with the use of eigenface approach. The paper introduces eigenface basis ranking measure, which is helpful in challenging task of creating the basis for recognition purposes. To speed up identification process we split the database into males and females using high-performance AdaBoost classifier. At the end of the paper the results of the tests in speed and accuracy are given.

Сергій М. Туляков, Рауф Х. Садихов

ПРОГРАМНА СИСТЕМА РОЗПІЗНАВАННЯ ОБЛИЧ НА МАШИНОЗЧИТУВАНИХ ПРОЇЗНИХ ДОКУМЕНТАХ

Дана стаття описує програмну систему розпізнавання облич на машинозчитуваних проїзних документах. Дана система володіє високою швидкістю пошуку і низьким рівнем помилок виявлення та ідентифікації на великих об'ємах зображень. Для підвищення точності виявлення, а також зниження обчислювального часу, пошук очей здійснюється в найбільш вірогідних областях. Ідентифікація виконується за допомогою підходу власних облич. У статті представлена міра оцінки базису власних облич, яка може бути корисною на етапі вибору базису. Для прискорення пошуку, система ділить базу відомих облич на дві частини на основі статі людини. Класифікація статі виконується за допомогою класифікатора Adaboost. В кінці статті представлені результати випробувань системи за швидкістю і точністю.

Сергей М. Туляков, Рауф Х. Садыхов

ПРОГРАММНАЯ СИСТЕМА РАСПОЗНАВАНИЯ ЛИЦ НА МАШИНОСЧИТЫВАЕМЫХ ПРОЕЗДНЫХ ДОКУМЕНТАХ

Данная статья описывает программную систему распознавания лиц на машиносчитываемых проездных документах. Рассматриваемая система обладает высокой скоростью поиска и низким уровнем ошибок обнаружения и идентификации на больших объемах изображений. Для повышения

точности обнаружения, а также снижения вычислительного времени, поиск глаз осуществляется в наиболее вероятных областях. Идентификация выполняется с помощью подхода собственных лиц. В статье представлена мера оценки базиса собственных лиц, которая может быть полезной на этапе выбора базиса. Для ускорения поиска, система делит базу известных лиц на две части на основе пола человека. Классификация пола выполняется с помощью AdaBoost классификатора. В конце статьи представлены результаты испытаний системы по скорости и точности.

Anton Kabysh, Vladimir Golovko, Arunas Lipnickas

INFLUENCE LEARNING FOR MULTI-AGENT SYSTEM BASED ON REINFORCEMENT LEARNING

This paper describes a multi-agent influence learning approach and reinforcement learning adaptation to it. This learning technique is used for distributed, adaptive and self-organizing control in multi-agent system. This technique is quite simple and uses agent's influences to estimate learning error between them. The best influences are rewarded via reinforcement learning which is a well-proven learning technique. It is shown that this learning rule supports positive-reward interactions between agents and does not require any additional information than standard reinforcement learning algorithm. This technique produces optimal behavior of multi-agent system with fast convergence patterns.

Антон Кабиш, Володимир Головко, Arunas Lipnickas

МОДЕЛЬ НАВЧАННЯ ЧЕРЕЗ ВПЛИВ НА ОСНОВІ ПІДКРІПЛЮЮЧОГО НАВЧАННЯ ДЛЯ НАВЧАННЯ МУЛЬТИАГЕНТНИХ СИСТЕМ

У даній роботі розглядається підхід до навчання мультиагентної системи на основі навчання через вплив на базі підкріплюючого навчання. Даний метод навчання в мультиагентній системі може використовуватися для розподіленого та адаптивного управління, а також управління, що самоорганізовується. Суть цього простого способу в тому, що він використовує вплив одного агента на іншого для обчислення оцінки помилки взаємодії між ними і коректує їх поведінку на основі цього. Кращий впливи одних агентів на інших посилюються з використанням підкріплюючого навчання. У роботі показано, що такий спосіб навчання підтримує позитивні взаємодії між агентами, що забезпечують досягнення потрібної мети, і не вимагає ніякої додаткової інформації для своєї роботи. Такий підхід забезпечує оптимальну поведінку мультиагентної системи та володіє швидким сходженням.

Антон Кабыш, Владимир Головко, Arunas Lipnickas

МОДЕЛЬ ОБУЧЕНИЯ ЧЕРЕЗ ВЛИЯНИЕ НА ОСНОВЕ ПОДКРЕПЛЯЮЩЕГО ОБУЧЕНИЯ ДЛЯ ОБУЧЕНИЯ МНОГОАГЕНТНЫХ СИСТЕМ

В данной работе рассматривается подход к обучению многоагентной системы на основе обучения через влияние на базе подкрепляющего обучения. Данный метод обучения может использоваться для распределенного, адаптивного и самоорганизующегося управления в многоагентной системе. Суть этого простого способа в том, что он использует воздействия агентов друг на друга для вычисления оценки ошибки взаимодействия между ними и корректирует их поведение на основе этого. Лучшие воздействия агентов друг на друга усиливаются с использованием подкрепляющего обучения. В работе показано, что такой способ обучения поддерживает положительные взаимодействия между агентами, обеспечивающие достижение нужной цели и не требует никакой дополнительной информации для работы. Такой подход обеспечивает оптимальное поведение многоагентной системы и обладает быстрой сходимостью.

Syarhei M. Avakaw, Alexander A. Doudkin, Alexander V. Inyutin, Aleksey V. Otwagin, Vladislav A. Rusetsky

MULTI-AGENT PARALLEL IMPLEMENTATION OF PHOTOMASK SIMULATION IN PHOTOLITHOGRAPHY

A framework for paralleling aerial image simulation in photolithography is proposed. Initial data for the simulation representing photomask are considered as a data stream that is processed by a multi-agent computing system. A parallel image processing is based on a graph model of a parallel algorithm. The algorithm is constructed from individual computing operations in a special visual editor. Then the visual representation is converted into XML, which is interpreted by the multi-agent system based on MPI. The system performs run-time dynamic optimization of calculations using an algorithm of virtual associative network. The proposed framework gives a possibility to design and analyze parallel algorithms and to adapt them to architecture of the computing cluster.

Сергій М. Аваков, Олександр А. Дудкін, Олександр В. Інютін, Олексій В. Отвагін, Владислав А. Русецький

ПАРАЛЕЛЬНА МУЛЬТИАГЕНТНА РЕАЛІЗАЦІЯ ПРОЦЕСУ МОДЕЛЮВАННЯ ФОТОШАБЛОНІВ ДЛЯ ФОТОЛІТОГРАФІЇ

Запропоновані засоби паралельного моделювання повітряного зображення фотошаблонів на поверхні напівпровідникової пластини в процесі фотолітографії. Початкові дані для моделювання розглядаються як потік даних, що обробляється мультиагентною обчислювальною системою. Паралельна обробка зображення заснована на застосуванні графової моделі паралельного алгоритму, який створюється з окремих обчислювальних операцій в спеціальному візуальному редакторі та потім перетворюється у формат XML. Алгоритм інтерпретується мультиагентною системою на базі MPI з динамічною оптимізацією обчислень на основі алгоритму віртуальної асоціативної мережі. Запропоновані засоби дозволяють виконувати швидке проектування паралельних алгоритмів, їх аналіз і адаптацію до архітектури обчислювального кластера.

Сергей М. Аваков, Александр А. Дудкин, Александр В. Инютин, Алексей В. Отвагин, Владислав А. Русецкий

ПАРАЛЛЕЛЬНАЯ МУЛЬТИАГЕНТНАЯ РЕАЛИЗАЦИЯ ПРОЦЕССА МОДЕЛИРОВАНИЯ ФОТОШАБЛОНОВ ДЛЯ ФОТОЛИТОГРАФИИ

Предложены средства параллельного моделирования воздушного изображения фотошаблонов на поверхности полупроводниковой пластины в процессе фотолитографии. Исходные данные для моделирования рассматриваются как поток данных, обрабатываемый мультиагентной вычислительной системой. Параллельная обработка изображения основана на применении графовой модели параллельного алгоритма, который создается из отдельных вычислительных операций в специальном визуальном редакторе и затем преобразуется в формат XML. Алгоритм интерпретируется мультиагентной системой на базе MPI с динамической оптимизацией вычислений на основе алгоритма виртуальной ассоциативной сети. Предложенные средства позволяют выполнять быстрое проектирование параллельных алгоритмов, их анализ и адаптацию к архитектуре вычислительного кластера.

Irina Artemieva, Alexander Zuenko, Alexander Fridman

ALGEBRAIC APPROACH TO INFORMATION FUSION IN ONTOLOGY-BASED MODELING SYSTEMS

In this paper we discuss the possibilities to use algebraic methods (in particular, n-tuple algebra developed by the authors) to improve the functioning of convenient ontology-based modeling systems. An illustrative example shows the ways to unify representation and processing of two major parts of subject domain ontologies.

Ірина Артемьєва, Олександр Зуєнко, Олександр Фрідман

АЛГЕБРАЇЧНИЙ ПІДХІД ДО СПІЛЬНОЇ ОБРОБКИ ІНФОРМАЦІЇ В СИСТЕМАХ МОДЕЛЮВАННЯ НА ОСНОВІ ОНТОЛОГІЙ

У даній статті ми обговорюємо можливості використання методів алгебри, зокрема, розробленої авторами алгебри кортежів, для поліпшення функціонування сучасних систем моделювання на основі онтологій. Ілюстративний приклад показує способи уніфікації представлення і обробки двох базових частин онтологій предметної області.

Ирина Артемьева, Александр Зуенко, Александр Фридман

АЛГЕБРАИЧЕСКИЙ ПОДХОД К СОВМЕСТНОЙ ОБРАБОТКЕ ИНФОРМАЦИИ В СИСТЕМАХ МОДЕЛИРОВАНИЯ, ОСНОВАННЫХ НА ОНТОЛОГИЯХ

В настоящей статье мы обсуждаем возможности использования алгебраических методов, в частности, разработанной авторами алгебры кортежей, для улучшения функционирования современных систем моделирования, основанных на онтологиях. Иллюстративный пример показывает способы унификации представления и обработки двух базовых частей онтологий предметной области.

Robert E. Hiromoto

THE ART AND SCIENCE OF GPU AND MULTI-CORE PROGRAMMING

This paper examines the computational programming issues that arise from the introduction of GPUs and multi-core computer systems. The discussions and analyses examine the implication of two principles (spatial and temporal locality) that provide useful metrics to guide programmers in designing and implementing efficient sequential and parallel application programs. Spatial and temporal locality represents a science of information flow and is relevant in the development of highly efficient computational programs. The art of high performance programming is to take combinations of these principles and unravel the bottlenecks and latencies associate with the architecture for each manufacturer computer system, and develop appropriate coding and/or task scheduling schemes to mitigate or eliminate these latencies.

Robert E. Hiromoto

МИСТЕЦТВО І НАУКА GPU ТА МУЛЬТИЯДЕРНОГО ПРОГРАМУВАННЯ

Дана стаття розглядає обчислювальні задачі програмування, які є результатом запровадження GPU та мультиядерних обчислювальних систем. Представлені матеріали розглядають значення двох принципів (розміщення у просторі і часі), які забезпечують корисну метрику для введення програмістів в проектування та створення ефективних послідовних і паралельних прикладних програм. Розміщення у просторі та часі є наукою інформаційних потоків і доречне для проектування надзвичайно ефективних обчислювальних програм. Мистецтво паралельного (високопродуктивного) програмування полягає в тому, щоб узяти комбінації цих принципів та розплутати вузькі місця і затримки, зв'язані з архітектурою для кожного з виробників обчислювальних систем, і розробити відповідне кодування і/або схеми планування задач, щоб зменшити або виключити ці затримки.

Robert E. Hiromoto

ИСКУССТВО И НАУКА GPU И МУЛЬТИЯДЕРНОГО ПРОГРАММИРОВАНИЯ

Данная статья рассматривает вычислительные проблемы программирования, являющиеся результатом внедрения GPU и мультиядерних вычислительных систем. представленные материалы рассматривают значение двух принципов (размещение в пространстве и времени), обеспечивающие полезную метрику для введения программистов в проектирование и осуществление эффективных последовательных и параллельных прикладных программ. Размещение в пространстве и времени является наукой об информационных потоках и уместно для проектирования чрезвычайно эффективных вычислительных программ. Искусство параллельного (высокопроизводительного) программирования состоит в том, чтобы взять комбинации этих принципов и распутать узкие места и задержки связанные с архитектурой для каждого производителя вычислительных систем, и разработать соответствующую кодировку и/или схемы планирования задач, чтобы смягчить или исключить эти задержки.

Janusz Zalewski, Zbigniew Wojcik

USE OF ARTIFICIAL INTELLIGENCE TECHNIQUES FOR PROGNOSTICS:
NEW APPLICATION OF ROUGH SETS

The objective of this paper is to set the context for the potential application of rough sets in prognostics. Prognostics is a field of engineering, which deals with predicting faults and failures in technical systems. Engineering solutions to respective problems embrace the use of multiple Artificial Intelligence (AI) techniques. The authors, first, review selected AI techniques used in prognostics and then propose the application of rough sets to build the system health prognostication model.

Janusz Zalewski, Zbigniew Wojcik

ВИКОРИСТАННЯ МЕТОДІВ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ПРОГНОЗУВАННЯ: НОВЕ
ЗАСТОСУВАННЯ ПРИБЛИЗНИХ НАБОРІВ

Завдання цієї статті – встановити контекст для потенційного застосування приблизних наборів у прогнозуванні. Прогнозування є галуззю інженерії, яка має справу з прогнозом дефектів і невдач в технічних системах. Технічні рішення відповідних проблем охоплюють використання багаторазових методів штучного інтелекту (ШІ). Автори, спочатку, розглядають відібрані методи ШІ, що використовуються в прогнозуванні, а потім пропонують застосування приблизних наборів до побудови системи моделі прогнозування здоров'я.

Janusz Zalewski, Zbigniew Wojcik

ИСПОЛЬЗОВАНИЕ МЕТОДОВ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ ПРОГНОЗИРОВАНИЯ: НОВОЕ ПРИМЕНЕНИЕ ПРИБЛИЗИТЕЛЬНЫХ НАБОРОВ

Задание этой статьи – установить контекст для потенциального применения приблизительных наборов в прогнозировании. Прогнозирование является отраслью инженерии, которая имеет дело с прогнозом дефектов и неудач в технических системах. Технические решения соответствующих проблем охватывают использование многоразовых методов искусственного интеллекта (ИИ). Авторы, сначала, рассматривают отобранные методы ИИ, которые используются в прогнозировании, а затем предлагают применения приблизительных наборов к построению системы модели прогнозирования здоровья.

Prepare your paper according to the following requirements:

Unconditional requirement - paper should not be published earlier.

(i) Suggested composition (frame) of paper:

- Issue formulation stressing its urgent solving; evaluation of recent publications in the explored issue
- short formulation of paper's purpose
- description of proposed method (algorithm)
- implementation and testing (verification)
- conclusion.

(ii) Use A4 (210 x 297 mm) paper. Size of paper has to be extended up to 6-8 pages.

(iii) Please use main text two column formatting;

(iv) A paper must have an abstract and some keywords;

(v) Place a full list of references at the end of the paper. Please place the references according to their order of appearance in the text.

(vi) An affiliation of each author is wanted.

(vii) The text should be single-spaced. Use Times New Roman (11 points, regular) typeface throughout the paper.

(viii) Equations should be placed in separate lines and numbered. The numbers should be within brackets and right aligned.

(ix) The figures and tables must be numbered, have a self-contained caption. Figure captions should be below the figures; table captions should be above the tables. Also, avoid placing figures and tables before their first mention in the text.

(x) As soon as you have the complete materials, the final versions should come electronically in MS Word'97 of MS Word 2000 format to the address computing@computingonline.net.

(xi) A hardcopy of your article is needed to be sent by regular mail for our publishing house.

(xii) Please send short CVs (up to 20 lines) and photos of every author.

(xiii) There is no other formatting required. The publishing department makes all rest formatting according to the publisher's rules.

Journal Topics:

- Algorithms and Data Structure
- Bio-Informatics
- Cluster and Parallel Computing, Software Tools and Environments
- Computational Intelligence
- Computer Modeling and Simulation
- Cyber and Homeland Security
- Data Communications and Networking
- Data Mining, Knowledge Bases and Ontology
- Digital Signal Processing
- Distributed Systems and Remote Control
- Education in Computing
- High Performance Computing and GRIDS
- Image Processing and Pattern Recognition
- Intelligent Robotics Systems
- Internet of Things
- Standardization of Computer Systems
- Wireless Systems

Основні вимоги до подання і оформлення публікацій наукового журналу "Комп'ютинг":

Безумовною вимогою є те, щоб стаття не була опублікована раніше!

(i) Наукові статті повинні мати такі необхідні елементи:

- постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями;
- аналіз останніх досліджень і публікацій, в яких започатковано розв'язання даної проблеми і на які спирається автор, виділення невирішених раніше частин загальної проблеми, котрим присвячується означена стаття;
- формулювання цілей статті (постановка завдання);
- виклад основного матеріалу дослідження з повним обгрунтуванням отриманих наукових результатів;
- висновки з даного дослідження і перспективи подальших розвідок у даному напрямку.

(ii) Використовуйте A4 (210 x 297 mm) формат сторінки. Загальний розмір статті має містити 6-8 сторінок.

(iii) Використовуйте двоколонкове форматування основного тексту;

(iv) Стаття повинна обов'язково містити основний текст українською мовою, анотацію (написану на Англійській і Українській мовах) і список ключових слів;

(v) В кінці статті розмістіть список літератури. Розміщуйте список літератури в порядку її цитування.

(vi) Необхідною є інформація про наукові звання, титули та посади авторів.

(vii) Текст повинен бути набраним одинарним інтервалом із використанням шрифту Times New Roman (11 points, regular).

(viii) Формули повинні відділятись від основного тексту пустими стрічками а також пронумеровані у круглих дужках та відцентровані по правому краю.

(ix) Таблиці і рисунки повинні бути пронумерованими. Заголовки рисунків розміщують під рисунком по центру. Заголовки таблиць розміщують по центру зверху таблиці.

(x) Завершені версії статей повинні бути надісланими в електронному MS Word'97 або MS Word 2000 форматі за адресою computing@computingonline.net.

(xi) Просимо надсилати поштою роздруковані копії статей.

(xii) В кінці кожної статті потрібно подати її назву, резюме (абстракт) і ключові слова англійською мовою.

(xiii) Просимо надсилати нам короткі біографічні дані (до 20 рядків) і скановані фотографії кожного із авторів.

(xiv) Видавництво здійснює остаточне форматування тексту згідно із вимогами друку.

(xv) У закордонних читачів можуть виникнути проблеми при ознайомленні з працями на російській та українській мовах. В зв'язку з цим редакційна колегія просить авторів додатково прислати розширений реферат (резюме), щоб б містило дві сторінки тексту англійською мовою, і супроводжувалось заголовком, прізвищами та адресами авторів. Авторам рекомендується використовувати у рисунках статті позначення переважно англійською мовою, або давати переклад у дужках. Тоді у розширеному резюме можна буде посилатися на рисунки у основному тексті.

Тематика журналу:

- Алгоритми та структури даних
- Біо-інформатика
- Кластерні та паралельні обчислення, програмні засоби та середовище
- Обчислювальний інтелект
- Комп'ютерне та імітаційне моделювання
- Кібернетична безпека та захист від тероризму
- Обмін даними та організація мереж
- Видобування даних, бази знань та онтології
- Цифрова обробка сигналів
- Розподілені системи та дистанційне управління
- Освіта в комп'ютингу
- Високопродуктивні обчислення та ГРІД
- Обробка зображень та розпізнавання шаблонів
- Інтелектуальні робототехнічні системи
- Інтернет речей
- Стандартизація комп'ютерних систем
- Безпровідні системи

Основные требования к подаче и оформлению публикаций научного журнала “Компьютинг”:

Безусловное требование – чтобы статья не была опубликована ранее!

(i) Научные статьи должны иметь такие необходимые элементы:

- постановка проблемы в общем виде и ее связь с важными научными или практическими задачами;
- анализ последних исследований и публикаций, в которых начаты решения данной проблемы и на которые опирается автор, выделение нерешенных прежде частей общей проблемы, которым посвящается обозначенная статья;
- формулирование целей статьи (постановка задача);
- изложение основного материала исследования с полным обоснованием полученных научных результатов;
- выводы из данного исследования и перспективы дальнейших изысканий в данном направлении.

(ii) Используйте A4 (210 x 297 mm) формат страницы. Общий размер статьи 6-8 страниц.

(iii) Используйте двухколоночное форматирование основного текста;

(iv) Статья должна обязательно содержать основной текст на Русском языке, аннотацию (написанную на Английском и Русском языках) и список ключевых слов;

(v) В конце статьи разместите список литературы. Размещайте список литературы в порядке ее цитирования.

(vi) Необходима информация о научных званиях, титулах и должностях авторов.

(vii) Текст должен быть набранным одинарным интервалом с использованием шрифта Times New Roman (11 points, regular).

(viii) Формулы должны отделяться от основного текста пустыми строками, а также пронумерованные в круглых скобках и отцентрованные по правому краю.

(ix) Таблицы и рисунки должны быть пронумерованными. Заголовки рисунков размещают под рисунком по центру. Заголовки таблиц размещают по центру сверху таблицы.

(x) Завершенные версии статей должны быть присланы в электронном MS Word'97 или MS Word 2000 формате по адресу computing@computingonline.net.

(xi) Просим присылать распечатанные копии статей по почте.

(xii) В конце каждой статьи необходимо предоставить ее название, резюме (абстракт) и ключевые слова на английском языке.

(xiii) Просим присылать нам короткие биографические данные (до 20 строчек) и сканированные фотографии каждого из авторов.

(xiv) Издательство осуществляет окончательное форматирование текста в соответствии с требованиями печати.

(xv) У зарубежных читателей могут возникнуть проблемы при ознакомлении с трудами на русском и украинском языках. В связи с этим редакционная коллегия просит авторов дополнительно прислать расширенный реферат (резюме), который содержал бы две страницы текста на английском языке, и сопровождался заголовком, фамилиями и адресами авторов. Авторам рекомендуется использовать в рисунках статьи обозначения преимущественно на английском языке, или давать перевод в скобках. Тогда в расширенном резюме можно будет посылаться на рисунки в основном тексте.

Тематика журнала:

- Алгоритмы и структуры данных
- Био-информатика
- Кластерные и параллельные вычисления, программные средства и среды
- Вычислительный интеллект
- Компьютерное и имитационное моделирование
- Кибернетическая безопасность и защита от терроризма
- Обмен данными и организация сетей
- Добыча данных, базы знаний и онтологии
- Цифровая обработка сигналов
- Распределенные системы и дистанционное управление
- Образование в компьютинге
- Высокопроизводительные вычисления и ГРИД
- Обработка изображений и распознавание шаблонов
- Интеллектуальные робототехнические системы
- Интернет вещей
- Стандартизация компьютерных систем
- Беспроводные системы