Alexey N. Averkin, Dmitry I. Ignatov, Sushmita Mitra, Jonas Poelmans, Valery B. Tarasov (Eds.)

SKAD'11 – Soft Computing Applications and Knowledge Discovery

Workshop co-located with the 13th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC-2011) and the 4th International Conference on Pattern Recognition and Machine Intelligence (PReMI-2011)
June 2011, Moscow, Russia

The proceedings are published online on the CEUR-Workshop web site with ISSN 1613-0073.

Volume Editors

Alexey N. Averkin
Dorodnicyn Computing Centre of the Russian Academy of Sciences, Russia

Dmitry I. Ignatov
School of Applied Mathematics and Information Science
National Research University Higher School of Economics, Moscow, Russia

Sushmita Mitra
Machine Intelligence Unit
Indian Statistical Institute, Kolkata, India

Jonas Poelmans
Faculty of Business and Economics
Katholieke Universiteit Leuven, Belgium

Valery B. Tarasov
Bauman Moscow State Technical University, Russia

Printed in Russia by the National Research University Higher School of Economics.

Preface

Soft computing is a collection of methodologies, which aim to exploit tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness and low cost solution in real life tasks. This volume contains the papers presented at SCAKD-2011: The International Workshop on Soft Computing Applications and Knowledge Discovery held on June 24, 2011 in Moscow. This workshop was initiated with the aim of presenting high quality scientific results and promising research in the areas of soft computing and data mining, particularly by young researchers, with an objective of bringing them to the focus while promoting collaborative research activities. The main goal of this workshop was to gather researchers all areas of Soft Computing Applications and Knowledge Discovery, including but not limited to the following: Pattern Recognition, Data Mining & Knowledge Discovery, Fuzzy & Neural Networks, Evolutionary & Probabilistic Computing, Swarm Intelligence, Collective Intelligence, Machine Learning, Information Retrieval, Rough Sets, Soft Computing, Bio-informatics, Biometrics, Computational Biology, Clustering, Formal Concept Analysis, Ontology Learning, Decision Support Systems & Business Intelligence (OLAP and BI, Data Warehouse Modeling, ETL techniques and technologies, and Data Visualization, Recommender Systems, Modeling of user behavior, Applications of Soft Computing. By holding the workshop in conjunction with PReMI and RSFDGrC, we hope to provide the contributers exposure and interaction with eminent scientists, engineers, professionals, and researchers in related fields. We are proud that in total, 15 papers were accepted for oral presentation and publication in the proceedings. Finally we would like to say a word of thank to the administration of the Higher School of Economics who took care of all arrangements to make this conference pleasant and enjoyable.

June, 2011
Moscow

Alexey N. Averkin
Dmitry I. Ignatov
Sushmita Mitra
Jonas Poelmans
Valery B. Tarasov

Organization

This SCAKD'11 workshop was held in June 2011 in Moscow, Russia co-located with the 13th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC-2011) and the 4th International Conference on Pattern Recognition and Machine Intelligence (PReMI-2011) at the National Research University Higher School of Economics.

Program Chairs

Alexey N. Averkin	Dorodnicyn Computing Centre of the Russian Academy of Sciences, Russia
Dmitry I. Ignatov	State University Higher School of Economics, Russia
Sushmita Mitra	Indian Statistical Institute, India
Jonas Poelmans	Katholieke Universiteit Leuven, Belgium
Valery B. Tarasov	Bauman Moscow State Technical University, Russia

Program Committee

Mehdi Kaytoue, France
Yuri Kudryavtsev, Russia
Sergei Kuznetsov, Russia
Xenia Naidenova, Russia
Andrey Savchenko, Russia
Dominik Slezak, Poland
Laszlo Szathmary, Canada
Rustam Tagiew, Germany

Sponsoring Institutions

ABBYY, Moscow
Russian Foundation for Basic Research, Moscow
Poncelet Laboratory (UMI 2615 du CNRS), Moscow
State University Higher School of Economics, Moscow
Yandex, Moscow
Witology, Moscow
Dynasty Foundation, Moscow

Table of Contents

A New Method of DDB Logical Structure Synthesis Using Distributed Tabu Search

Eduard Babkin[1] and Margarita Karpunina[2],

National Research University “Higher School of Economics”
Dept. of Information Systems and Technologies,
Bol. Pecherskaya, 25,
6030155 Nizhny Novgorod, Russia
[1] eababkin@hse.ru
[2] karpunina-margarita@yandex.ru

Abstract. In this paper we propose a parallel tabu search algorithm based on the consecutive tabu algorithm constructed by us earlier to solve the problem of the distributed database optimal logical structure synthesis. Also we provide a reader with some information about the performance of our new parallel algorithm and the quality of the solutions obtained with help of it.

Keywords: Neural networks, tabu search, genetic algorithms, parallel programming, distributed databases.

1 Introduction

The problems of decomposition of complex data structures play an extremely important role in many critical applications varying from cloud computing to distributed databases (DDB) [7]. In later class of applications that problem is usually formulated as synthesis of optimal logical structure (OLS). In accordance with [3] it consists of two stages. The first stage is decomposition of data elements (DE) into logical record (LR) types. The second stage is irredundant replacement of LR types in the computing network. For each stage various domain-specific constraints are introduced (like irredundant allocation, semantic contiguity of data elements, available external storage) as well as optimum criteria are specified. In our work the criterion function is specified as a minimum of total time needed for consecutive processing of a set of DDB users’ queries [8].

From mathematical point of view the specified problem is a NP-complete non-linear optimization problem of integer programming. So far different task-specific approaches were proposed such as branch-and-bound method with a set of heuristics (BBM) [8], probabilistic algorithms, etc. However not many of them exploit benefits of parallel processing and grid technologies [4], [5], [6], [9].

In previous works the authors developed the exact mathematical formalization of the OLS problem and offered the sequential tabu search (TS) algorithm which used different Tabu Machines (TMs) for each stage of the solution [1], [2]. The constructed algorithm produced solutions with good quality, but it was computationally efficient for small mock-up problems.

In the present article we propose a new distributed model of TM (DTM) and a computationally efficient parallel algorithm for solutions of complex data structure decomposition problems. The article has the following structure. In Section 2 we outline critical elements of TM. In Section 3 general description of DTM algorithm is given and Section 4 specifies it in details. Section 5 briefly describes evaluation of the obtained parallel algorithm. Overview of the results in Section 6 concludes the article.

2 Short Overview of Tabu Machine Model and Dynamics

In our work we use the generic model of TM as it was specified by Minghe Sun and Hamid R. Nemati [10] with the following important constituents.

$S = \{s_1, ..., s_n\}$ is the current state of the TM, it is collectively determined by the states of its nodes.

$S_0 = \{s_1^0, ..., s_n^0\}$ is the state of the TM with the minimum energy among all states which are obtained by the current moment within the local period (or within the short term memory process (STMP)).

$S_{00} = \{s_1^{00}, ..., s_n^{00}\}$ is the state of the TM with the minimum energy among all states which are obtained by the current moment (within both the STMP and the long term memory process (LTMP)).

$T = \{t_1, ..., t_n\}$ is a vector to check the tabu condition.

$E(S)$ is the TM energy corresponding to the state S.

$E(S_0)$ is the TM energy corresponding to the state S_0.

$E(S_{00})$ is the TM energy corresponding to the state S_0.

k is the number of iterations (i.e. the number of neural network (NN) transitions from the one state to another) from the outset of the TM functioning.

h is the number of iterations from the last renewal the value of $E(S_0)$ within the STMP.

c is the number of the LTMPs carried out by the current moment.

The following variables stand as parameters of the TM-algorithm:

l – the tabu size,

β – the parameter determining the termination criterion of the STMP,

C – maximum number of the available LTMPs inside the TM-algorithm.

The state transition mechanism of the TM is governed by TS and performed until the predefined stopping rule is satisfied. Let’s name this sequence of state transitions as a work period of the TM. It is advisable to run the TM for several work periods. It is better to begin a new work period of the TM using information taken from the previous work periods, from a “history” of the TM work by applying a LTMP. In such a case a TS algorithm finds a node which has not changed its state for the longest time among all neurons of the TM. And then this node is forced to switch its state.

3 A Consecutive TM-Algorithm for OLS Problem

As [3] states, the general problem of DDB OLS synthesis consists of two stages.

1. **Composition of logical record (LR) types** from data elements (DE) using the constraints on: the number of elements in the LR type; single elements inclusion in the LR type; the required level of information safety of the system. In addition, LR types synthesis should take into account semantic contiguity of DE.
2. **Irredundant allocation of LR types** among the nodes in the computing network using the constraints on: irredundant allocation of LR types; the length of the formed LR type on each host; the total number of the synthesized LR types placed on each host; the volume of accessible external memory of the hosts for storage of local databases; the total processing time of operational queries on the hosts.

The synthesis objective is to minimize the total time needed for consecutive processing of a set of DDB users' queries. Such problem has an exact but a very large mathematical formalization. So, we provide it in the Appendix I and Appendix II of this paper due to its limited size and should refer to [1], [2], [3], [8] for further details.

In our previous work [2] we have offered a new method for formalization of the described problem in the terms of TM and have constructed TMs' energy functions as follows. TM for the first stage consists of one layer of neurons, connected by complete bidirectional links. The number of neurons in the layer is equal to I^2, where I is the number of DEs. Each neuron is supplied with two indexes corresponding to numbers of DEs and LRs. For example, $OUT_{xi} = 1$ means, that the DE x will be included to the i-th LR. All outputs OUT_{xi} of a network have a binary nature, i.e. accept values from set $\{0,1\}$. The following TM energy function for LR composition was proposed:

$$E = -\frac{1}{2} \cdot \sum_{i=1}^{I} \sum_{j=1}^{I} \sum_{x=1}^{I} \sum_{y=1}^{I} \left[-A_1 \cdot \delta_{xy} \cdot \left(1 - \delta_{ij}\right) + B_1 \cdot \delta_{ij} \cdot \left(1 - \delta_{xy}\right) \cdot \left(2 \cdot a_{xy}^{g} - 1\right) - D_1 \cdot \delta_{ij} \cdot \right.$$

$$\left. \cdot \left(incomp\_gr_{xy} + incomp\_gr_{yx}\right)\right] \cdot OUT_{xi} \cdot OUT_{yj} + \sum_{i=1}^{I} \sum_{x=1}^{I} \left[\frac{B_1}{2} \cdot \sum_{\substack{y=1 \\ y \neq x}}^{I} \left(a_{xy}^{g}\right)^2 + \frac{C_1}{2 \cdot F_i} \right] \cdot OUT_{xi} \qquad \mathbf{(1)}$$

Here $w_{xi,yj} = -A_1 \cdot \delta_{xy} \cdot \left(1 - \delta_{ij}\right) + B_1 \cdot \delta_{ij} \cdot \left(1 - \delta_{xy}\right) \cdot \left(2 \cdot a_{xy}^{g} - 1\right) - D_1 \cdot \delta_{ij} \cdot \left(incomp\_gr_{xy} + \right.$ $\left. + incomp\_gr_{yx}\right)$ are weights of neurons, $T_{xi} = \left(\frac{B_1}{2} \cdot \sum_{\substack{y=1 \\ y \neq x}}^{I} \left(a_{xy}^{g}\right)^2 + \frac{C_1}{2 \cdot F_i} \right)$ are the neurons' thresholds.

For the second stage of irredundant LR allocation we offered TM with the same structure as TM for LR composition, but the number of neurons in the layer is equal to $T \cdot R_0$, where T is the number of LRs, synthesized during LR composition, R_0 is the number of the hosts available for LR allocation.

As a result of constraints translation into the terms of TM the following TM energy function for the LR allocation was obtained:

$$E = -\frac{1}{2} \cdot \sum_{r_1=1}^{R_0} \sum_{r_2=1}^{R_0} \sum_{t_1=1}^{T} \sum_{t_2=1}^{T} \left[-A_2 \cdot \delta_{t_1 t_2} \cdot \left(1 - \delta_{r_1 r_2}\right) \right] \cdot OUT_{t_1 r_1} \cdot OUT_{t_2 r_2} + \sum_{r_1=1}^{R_0} \sum_{t_1=1}^{T} \left[\frac{B_2 \cdot \psi_0}{2 \cdot \theta_{t_1 r_1}} \cdot \right. \qquad \mathbf{(2)}$$

$$\left. \cdot \sum_{i=1}^{I} \left(x_{it_1} \cdot \rho_i \right) + \frac{C_2}{2 \cdot h_{r_1}} + \frac{D_2 \cdot \psi_0}{2 \cdot \eta_{r_1}^{EMD}} \cdot \sum_{i=1}^{I} \left(\rho_i \cdot \pi_i \cdot x_{it_1} \right) + \frac{E_2 \cdot \left(t_{r_1}^{srh} + t_{r_1} \right)}{2} \cdot \sum_{p=1}^{P_0} \left(\frac{SN_{pt_1}}{T_p} \right) \right] \cdot OUT_{t_1 r_1}$$

Here $w_{t_1 r_1, t_2 r_2} = -A_2 \cdot \delta_{t_1 t_2} \cdot \left(1 - \delta_{r_1 r_2}\right)$ are weights of neurons,

$$T_{t_1 r_1} = \frac{B_2 \cdot \psi_0}{2 \cdot \theta_{t_1 r_1}} \cdot \sum_{i=1}^{I} \left(x_{it_1} \cdot \rho_i \right) + \frac{C_2}{2 \cdot h_{r_1}} + \frac{D_2 \cdot \psi_0}{2 \cdot \eta_{r_1}^{EMD}} \cdot \sum_{i=1}^{I} \left(\rho_i \cdot \pi_i \cdot x_{it_1} \right) + \frac{E_2 \cdot \left(t_{r_1}^{srh} + t_{r_1} \right)}{2} \cdot \sum_{p=1}^{P_0} \left(\frac{SN_{pt_1}}{T_p} \right)$$

are the neurons' thresholds. Here the I is the number of DEs, $z_{pr_1}^{t_1} = OUT_{t_1 r_1} \cdot SN_{pt_1}$ and SN_{pt_1} is introduced as a normalized sum, i.e. $SN_{pt_1} = \begin{cases} 1, \text{ if } \sum\limits_{i=1}^{I} w_{pi}^{Q} x_{it_1} \geq 1 \\ 0, \text{ if } \sum\limits_{i=1}^{I} w_{pi}^{Q} x_{it_1} = 0 \end{cases}$, where w_{pi}^{Q} is the matrix of dimension $\left(P_0 \times I\right)$, that matrix shows which DEs are used during processing of different queries.

In [2] we also compared the developed TM-algorithm with other methods like [10] to estimate an opportunities and advantages of TS over our earlier approaches based on Hopfield Networks or their combination with genetic algorithms (NN-GA-algorithm) [3]. For complex mock-up problems we obtained the TM solutions with the quality higher than the quality of solutions received with help of NN-GA-algorithm on average 8,7%, the quality of solutions received with help of BBM on average 23,6% (refer to Fig. 1), and CPU time for LR composition was on average 36% less that the same spent by the Hopfield Network approach. So, our TM is able to produce good solutions. But nevertheless this algorithm is time consuming on high-dimensional tasks, and therefore it is needed to construct a parallel TM-algorithm in order to validate our approach on the high-dimensional tasks and increase the performance. Moreover, the parallel algorithm helps us to reveal the influence of the tabu parameters on the tasks' solution process and to determine the dependency between the tabu parameters and characteristics of our problem in order to obtain the better solutions faster.

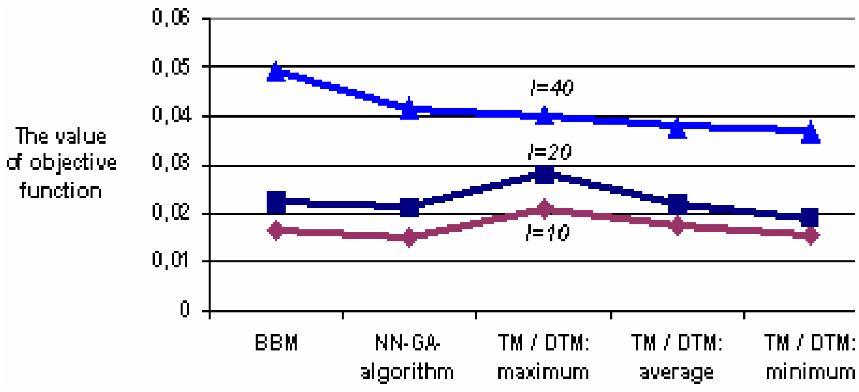


Fig. 1. The value of objective function on mock-up problems solutions.

4 A General Description of DTM Functioning

The proposed parallel algorithm of TM exploits parallelization capabilities of the following procedures: finding a neuron to change its state; changing the value of $\Delta E(S_i)$ of neurons for using it on the next iteration; the calculation of energy function value; the calculation of values of auxiliary functions used in aspiration criteria of TM; the transition from one local cycle to the other.

For the case of the homogeneous computational parallel cluster with multiple identical nodes the following general scheme of new parallel functionality is proposed. The set of neurons of the whole TM is distributed among all nodes' processors according to the formula $N_p = \begin{cases} n_1+1, \text{ if } p<n_2 \\ n_1, \text{ otherwise} \end{cases}$, where $n_1 = \left\lfloor \frac{N}{P} \right\rfloor, n_2 = N \bmod P$, N – the number of neurons in the whole TM, $p=\overline{0,(P-1)}$ – the index of processor, P – the number of processors. The number of Tabu sub-machines (TsMs) is equal to the number of available processors. So, one TsM is located on each processor and TsM with index p consists of N_p neurons. During the initialization stage neural characteristics are set to each neuron. The scheme of DTM is depicted on Fig. 2. The same figure shows how the weight matrix of each TsM $W_p = \left\{w_{ij}^p; i=\overline{1,N_p}; j=\overline{1,N}\right\} = \left\{w_{ij}; i=\overline{\sum_{k=0}^{p-1} N_k+1, \sum_{k=0}^{p} N_k}; j=\overline{1,N}\right\}$ is constructed from the weight matrix $W=\left\{w_{ij}; i,j=\overline{1,N}\right\}$ of the whole TM. When the optimal state of DTM is achieved, the results from all TsMs are united. The proposition that the energy of the whole TM is additive on the energies of TsMs including in the DTM, i.e. $E=E_0+E_1+\ldots+E_{P-1}=\sum_{p=0}^{P-1} E_p$, is formulated and proofed by the authors but due to lack of the space is omitted in that article.

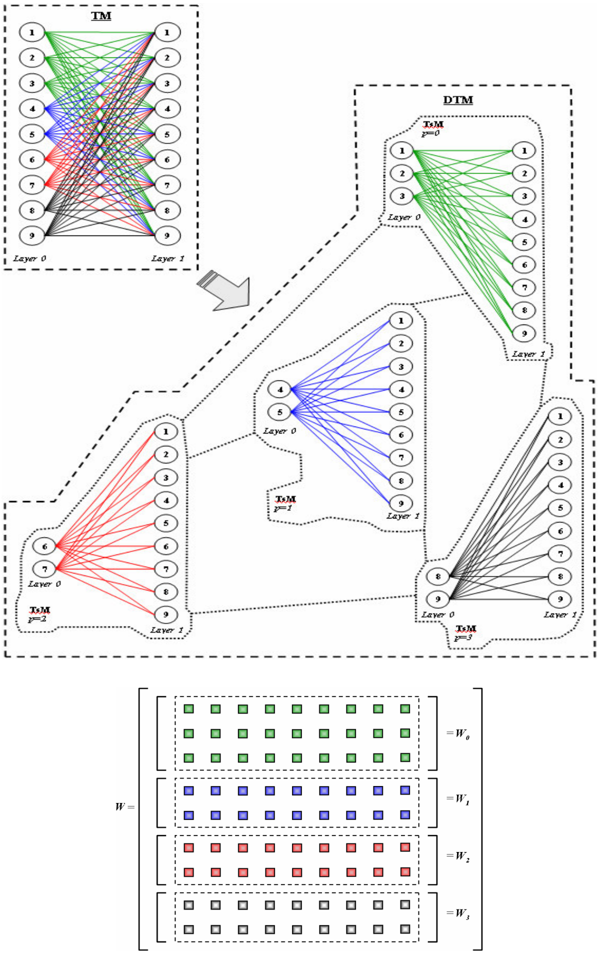


Fig. 2. DTM scheme (*up*) and method of W_p construction from the whole matrix W (*bottom*).

Let's consider a common implementation of DTM taking into account a parallel implementation of foregoing procedures.

Initialization. At this stage we assume that TsMs included into DTM are constructed and initialized. Construction and initialization are conducted following the mentioned above scheme of distribution of DTM neurons among the set of available processors. After the structure of each TsM is defined, TsMs are provided with the following characteristics: the matrix of neurons weights, vector of neurons thresholds, and vector of neurons biases. Thus, on the current stage we have the set of TsMs, and the elements of this set are

$$subTM_p = \left\{ W_p, I_p, T_p, In_p \right\}, \quad p = \overline{0,(P-1)} \ , \tag{3}$$

where $subTM_p$ – p-th TsM, W_p – the matrix of its neurons weights, I_p – the vector of neurons biases, T_p – the vector of neurons thresholds, and In_p – the vector of initial states of TsM's neurons. Matrixes W_p and vectors I_p and T_p are defined according to the following formulas:

$$W = \left\{ w_{ij}; i, j = \overline{1,N} \right\} = \begin{bmatrix} W_0 \\ W_1 \\ \vdots \\ W_{P-1} \end{bmatrix} = \begin{bmatrix} \left\{ w_{ij}^0; i = \overline{1,N_0}; j = \overline{1,N} \right\} \\ \left\{ w_{ij}^1; i = \overline{1,N_1}; j = \overline{1,N} \right\} \\ \vdots \\ \left\{ w_{ij}^{P-1}; i = \overline{1,N_{P-1}}; j = \overline{1,N} \right\} \end{bmatrix} = \begin{bmatrix} \left\{ w_{ij}; i = \overline{1,N_0}; j = \overline{1,N} \right\} \\ \left\{ w_{ij}; i = \overline{N_0+1, N_0+N_1}; j = \overline{1,N} \right\} \\ \vdots \\ \left\{ w_{ij}; i = \overline{\sum_{k=0}^{P-2} N_k + 1, \sum_{k=0}^{P-1} N_k}; j = \overline{1,N} \right\} \end{bmatrix} \tag{4}$$

$$I = \left\{ i_j; j = \overline{1,N} \right\} = \begin{bmatrix} I_0 \\ I_1 \\ \vdots \\ I_{P-1} \end{bmatrix} = \begin{bmatrix} \left\{ i_j^0; j = \overline{1,N_0} \right\} \\ \left\{ i_j^1; j = \overline{1,N_1} \right\} \\ \vdots \\ \left\{ i_j^{P-1}; j = \overline{1,N_{P-1}} \right\} \end{bmatrix} = \begin{bmatrix} \left\{ i_j; j = \overline{1,N_0} \right\} \\ \left\{ i_j; j = \overline{N_0+1, N_0+N_1} \right\} \\ \vdots \\ \left\{ i_j; j = \overline{\sum_{k=0}^{P-2} N_k + 1, \sum_{k=0}^{P-1} N_k} \right\} \end{bmatrix} \tag{5}$$

$$T = \left\{ t_j; j = \overline{1,N} \right\} = \begin{bmatrix} T_0 \\ T_1 \\ \vdots \\ T_{P-1} \end{bmatrix} = \begin{bmatrix} \left\{ t_j^0; j = \overline{1,N_0} \right\} \\ \left\{ t_j^1; j = \overline{1,N_1} \right\} \\ \vdots \\ \left\{ t_j^{P-1}; j = \overline{1,N_{P-1}} \right\} \end{bmatrix} = \begin{bmatrix} \left\{ t_j; j = \overline{1,N_0} \right\} \\ \left\{ t_j; j = \overline{N_0+1, N_0+N_1} \right\} \\ \vdots \\ \left\{ t_j; j = \overline{\sum_{k=0}^{P-2} N_k + 1, \sum_{k=0}^{P-1} N_k} \right\} \end{bmatrix} \tag{6}$$

Vector In of initial states of the whole TM neurons is random generated, and then cut on P parts, each of which (i.e. In_p) is corresponded to the concrete TsM.

The local cycle of the TM. Let's consider the local cycle of DTM.

Choose the neuron-candidate for the next move. At the first step of the TM local cycle we search for neuron on each TsM, which should change its state on current iteration. The criterion to choose such a neuron is defined as the following:

$$\Delta E_p(S_j) = \left\{ \min\left\{ \Delta E_p(S_i) \mid i = \overline{1,N_p} \right\} \ : \ k - t_j \le l \ \vee \ E_p(S) + \Delta E_p(S_j) < E_p(S_0) \right\} . \\ p = \overline{0,(P-1)} \tag{7}$$

Thus, the search of neurons satisfied to the condition (7) is performed in parallel on the hosts of CN.

The comparison of found neurons. After the neuron satisfied to the condition (7) is found on each host, the search with help of STMP reduce operations defined by authors for MPI_Allreduce function is performed within the whole DTM to find the neuron j^*, such that $\Delta E(S_{j^*}) = \min\left\{\Delta E_p(S_j) \mid p = \overline{0,(P-1)}\right\}$.

Change the energy value of neurons. After the required neuron j^* has been found, and each TsM has information about it, each neuron of $subTM_p$, $p = \overline{0,(P-1)}$ changes its $\Delta E(S_i)$ value. The calculation of DTM energy function change is done in parallel on each $subTM_p$. Further the cycle is repeated following described scheme until the condition of exit from the local cycle of the TM is satisfied.

The global cycle of the TM. We select neuron, that didn't change its state longest, on each TsMs. The number j of this neuron on each $subTM_p$ is defined according to the following criteria:

$$\left(t_j\right)_p = \min\left\{t_i \mid i = \overline{1,N_p}\right\}, \quad p = \overline{0,(P-1)}. \qquad \textbf{(8)}$$

The search of $\left(t_j\right)_p$ is done on the available processors in parallel according to the formula (8).

The comparison of found neurons. After the neuron satisfied to the condition (8) is found on each host, the search with help of LTMP reduce operations defined by authors for MPI_Allreduce function is performed within the whole DTM to find the neuron j^*, such that $t_{j^*} = \min\left\{\left(t_j\right)_p \mid p = \overline{0,(P-1)}\right\}$.

Change the energy value of neurons. After the required neuron j^* has been found, and each TsM has information about it, each neuron of $subTM_p$, $p = \overline{0,(P-1)}$ changes its $\Delta E(S_i)$ value. The calculation of DTM energy function change is done in parallel on each $subTM_p$. Further the cycle is repeated following described scheme until the number of LTMP calls will exceed $C : C \in Z^+, C \geq 0$ times. After that the search is stopped and the best found state is taken as the final DTM state.

5 The Algorithm of DTM Functioning

Let's try to represent the general description as an algorithm outlined step by step. We will use the following notations: N – the number of neurons in the DTM, i.e. $|S| = |S_0| = |S_{00}| = N$; N_p – the number of neurons including into the TsM $subTM_p$, where $p = \overline{0,(P-1)}$; P – the number of processors on which DTM operates.

Step 1. Construct TsMs $subTM_p$ and randomly initialize initial states of its neurons. Define the tabu-size l of DTM. Let $h=0$, $k=0$ – counters of iterations in the frame of the whole DTM. Let $c=0$ and $C\geq 0$ – the maximum number of LTMP calls in the frames of the whole DTM. Let $\beta>0$ is defined according to inequality $\beta\cdot N>l$ in the frames of the whole DTM too.

Step 2. Find the local minimum energy state S_0. Calculate $E(S_0)$ and

$$\Delta E(S)=\begin{bmatrix}\Delta E(S_1)\\ \Delta E(S_2)\\ \vdots\\ \Delta E(S_N)\end{bmatrix}=\begin{bmatrix}\Delta E_0(S_i),\ i=\overline{1,N_0}\\ \Delta E_1(S_i),\ i=\overline{N_0+1,N_0+N_1}\\ \vdots\\ \Delta E_{P-1}(S_i),\ i=\overline{\sum_{k=0}^{P-2}N_k+1,\sum_{k=0}^{P-1}N_k}\end{bmatrix},\quad i=\overline{1,N}\,. \qquad \textbf{(9)}$$

The values of $E_p(S_0)$ and $\Delta E_p(S_i)$ for $p=\overline{0,(P-1)}$ are calculated in parallel on P processors. Let $S_{00}=S_0$ is the best global state, and $E(S_{00})=E(S_0)$ is the global minimum of energy. Let $S=S_0$ and $E(S)=E(S_0)$. Let $t_i=-\infty,\forall i=\overline{1,N}$.

Step 3. In the frames of each $subTM_p$ choose the neuron j with $\Delta E_p(S_j)$ satisfied to $\Delta E_p(S_j)=\left\{\min\left\{\Delta E_p(S_i)\,|\,i=\overline{1,N_p}\right\}\ :\ k-t_j\leq l\ \vee\ E_p(S)+\Delta E_p(S_j)<E_p(S_0)\right\}, p=\overline{0,(P-1)}$.

Step 4. Using STMP reduce operations defined by authors, form the set $\left\{j^*,\ \Delta E(S_{j^*}),\ \ s_{j^*}\right\}$, where j^* – the index of neuron (in the frames of the whole DTM) changing its state at the current moment, $\Delta E(S_{j^*})$ – the change of DTM energy function value after the neuron j^* has changed its state, s_{j^*} – the new state of neuron j^*.

Step 5. If $subTM_p$ contains the neuron j^*, then $t_{j^*}=k$, $s_{j^*}=1-s_{j^*}$.

Step 6. Let $t_{j^*}=k$, $k=k+1$, $h=h+1$, $S=S_{j^*}$, $E(S)=E(S)+\Delta E(S_{j^*})$ in the frames of the whole DTM.

Step 7. Update $\Delta E(S)$ using (9). The values of $\Delta E_p(S_i)$ are calculated in parallel on P processors.

Step 8. Determine if the new state S is the new local and / or global minimum energy state: if $E(S)<E(S_0)$, then $S_0=S$, $E(S_0)=E(S)$ and $h=0$; if $E(S)<E(S_{00})$, then $S_{00}=S$ and $E(S_{00})=E(S)$ in the frames of the whole DTM.

Step 9. If $h<\beta\cdot N$, go to **Step 3.**, else – to **Step 10.**

Step 10. If $c\geq C$, then the algorithm stops. S_{00} is the best state. Else, in the frames of each $subTM_p$ choose in parallel the neuron j with $\left(t_j\right)_p$ satisfied to $\left(t_j\right)_p=\min\{t_i\,|\ i=\overline{1,N_p}\},\ p=\overline{0,(P-1)}$. Using LTMP reduce operations defined by authors, form the set $\left\{j^*,\ \ \Delta E(S_{j^*}),\ \ s_{j^*}\right\}$, where j^* – the index of neuron (in the frames of the whole DTM) changing its state at the current moment, $\Delta E(S_{j^*})$ – the change of DTM energy function value after the neuron j^* has changed its state, s_{j^*} – the new state of neuron j^*. Let $S_0 = S_{j^*}$ and $E(S_0) = E(S) + \Delta E(S_{j^*})$, $c = c + 1$ and $h = 0$. Go to **Step 6.**

It's worth mentioning that on the **Step 10.** the new state of local energy minimum $E(S_0)$ is set without any auxiliary checks, i.e. is can be worse than the previous S_0. Exploiting this technique we exclude stabilization in local energy minimums and expand areas of potential solutions.

6 Performance Evaluation

In order to evaluate the performance of constructed DTM the set of experiments on mock-up problems with DTM consisting of $N = 100$, $N = 400$ and $N = 1600$ neurons were done on multi-core cluster. 372 trial solutions were obtained for each mock-up problem depending on the values of < *l, C, β* > parameters of DTM.

We proposed to use an average acceleration as the metric to evaluate the efficiency of DTM. The dependencies of average acceleration on the number of processors for mock-up problems with $N = 100$, $N = 400$ and $N = 1600$ are depicted on Fig. 3. DTM gives a linear acceleration.

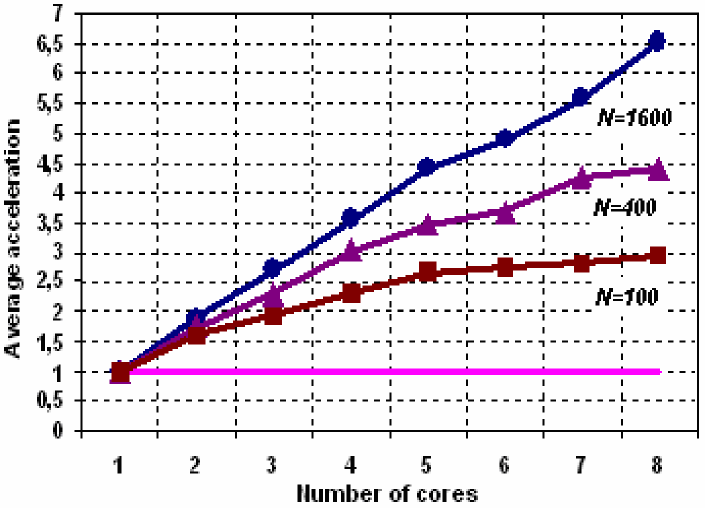


Fig. 3. Average acceleration on mock-up problems.

7 Conclusion

In this paper we proposed parallel TS algorithm for DDB OLS synthesis problem. The constructed DTM was validated and compared with the sequential TM. As expected, both approaches give the same results with the solutions quality higher than the quality of solutions received by NN-GA-algorithm [1], [3] on average 8,7% and by BBM [8] on average 23,6% on mock-up problem with higher dimension.

It is worth mentioning that during the DTM cycles intensive data communication between processors is carried out in the proposed algorithm. Therefore, we can speak about the significant increasing of DTM performance in compare with its consecutive analogue for the high-dimensional problems. This statement is not contrary to our objectives, because the problem of DDB OLS synthesis is important today in view of high dimensionality.

References

1. Babkin E., Karpunina M. Comparative study of the Tabu machine and Hopfield networks for discrete optimization problems. Information Technologies'2008. Proc. Of the 14th International Conference on Information and Software Technologies, IT 2008. Kaunas, Lithuania, April 24-25. ISSN 2029-0020. pp. 25-41. (2008)
2. Babkin E., Karpunina M. The analysis of tabu machine parameters applied to discrete optimization problems // Proceedings of 2009 ACS/IEEE International Conference on Computer Systems and Aplications, AICCSA'2009. – May 10-13, 2009. – Rabat, Morocco. – P.153-160. Sponsored by IEEE Computer Society, Arab Computer Society, and EMI, Morocco. (2009) IEEE Catalog Number: CFP09283-CDR. ISBN: 978-1-4244-3806-8. Library of Congress: 200990028. http://www.congreso.us.es/aiccsa2009.
3. Babkin E., Petrova M. Application of genetic algorithms to increase an overall performance of artificial neural networks in the domain of synthesis DDBs optimal structures. Proc. Of The 5th International Conference on Perspectives in Business Informatics Research (BIR 2006) October 6-7, 2006 Kaunas University of Technology, Lithuania. ISSN: 1392-124X Information Techonology and Control, Vol.35, No. 3A. pp.285-294. (2006)
4. Chakrapani J., Skorin-Kapov J. Massively parallel tabu search for the quadratic assignment problem, Annals of Operations Research 41. pp. 327-341. (1993)
5. Fiechter C.-N. A parallel tabu search algorithm for large traveling salesman problems. Discrete Applied Mathematics Vol. 51. ELSEVIER. pp. 243-267. (1994)
6. Garcia B.-L. et al. A parallel implementation of the tabu search heuristic for vehicle routing problems with time window constraints, Computers Ops Res, Vol.21 No. 9. pp. 1025-1033, (1994)
7. Kant K., Mohapatra P. Internet Data Centers. Computer, Published by the IEEE Computer Society. 0018-9162/04. (2004)
8. Kulba V.V., Kovalevskiy S.S., Kosyachenko S.A., Sirotyuck V.O. Theoretical backgrounds of designing optimum structures of the distributed databases. M.: SINTEG. (1999)
9. Porto Stella C. S., Kitajima Joao Paulo F. W., Ribeiro Celso C. Performance evaluation of a parallel tabu search task scheduling algorithm. Parallel Computing Vol. 26. ELSEVIER. pp. 73-90. (2000)
10. Sun M., Nemati H. R. Tabu Machine: A New Neural Network Solution Approach for Combinatorial Optimization Problems, Journal of Heuristics, 9:5-27, (2003)

Service Centers Finding by Fuzzy Antibases of Fuzzy Graph

Leonid Bershtein[1,1], Alexander Bozhenyuk[2], Igor Rozenberg[3],

[1] Taganrog Institute of Technology of Southern Federal University,
Nekrasovskiy 44, 347928, Taganrog, Russia
[2] Scientific and Technical Center "Intech" of Southern Federal University,
Oktyabrskaya Square 4, 347922, Taganrog, Russia
[3] Public Corporation "Research and Development Institute of Railway Engineers",
Nizhegorodskaya Street, 27/1, 109029, Moscow, Russia
Avb@itt.net.ru, Avb002@yandex.ru, I.kudreyko@gismps.ru

Abstract. In this paper the questions of definition optimum allocation of the service centres of some territory are observed. It is supposed that territory is described by fuzzy graph. In this case a task of definition optimum allocation of the service centres may be transformed into the task of definition of fuzzy antibases of fuzzy graph. The method of definition of fuzzy antibases is considered in this paper. The example of founding optimum allocation of the service centres as definition of fuzzy antibases is considered too.

Keywords: fuzzy directed way, accessible degree, fuzzy transitive closure, fuzzy reciprocal transitive closure, fuzzy set of antibases.

1 Introduction

There are many tasks of optimum allocation of the service centres [1]. They are an allocation of radio and TV station in some region; an allocation of military bases, which control some territory; an allocation of shops, which serve some region and so on.

However, the information about the allocation of the service centres is inaccurate or not reliable very frequently [2]. The calculation of a service degree (or quality) can be carried out by several, including to contradicting each other, criteria. For example, the definition of number and allocation of shops can be made by taking into account quality of roads, cost of ground in the given area, distance from other areas, and other criteria. Ranking of such criteria is frequently made subjectively, on the basis of the human factor.

We consider that some territory is divided into *n* areas. There are *k* service centres, which may be placed into these areas. It is supposed that each centre may be placed

[1] This work has been supported by the Russian Foundation
for Basic Research project № 10-01-00029a.

into some stationary place of each area. From this place the centre serves all area, and also some neighbor areas with the given degree of service. The service centres can fail during the exploitation (for example, for planned or extraordinary repair). It is necessary for the given number of the service centres to define the places of their best allocation. In other words, it is necessary to define the places of *k* service centres into *n* areas such that the control of all territory is carried out with the greatest possible degree of service.

2 Main concepts and definitions

In this paper we suppose that the service degree of region is defined as the minimal meaning of service degrees of each area. Taking into account, that the service degree can not always have symmetry property (for example, by specific character and relief of the region) the model of such task is a fuzzy directed graph $\widetilde{G}=(X,\widetilde{U})$ [3]. Here, set $X=\{x_i\}$, $i\in I=\{1,2,...,n\}$ is a set of vertices and $\widetilde{U}=\{<\mu_U<x_i,x_j>/<x_i,x_j>>\}$, $<x_i,x_j>\in X^2$ is a fuzzy set of directed edges with a membership function $\mu_U:X^2\to[0,1]$. The membership function $\mu_U<x_i,x_j>$ of graph $\widetilde{G}=(X,\widetilde{U})$ defines a service degree of area *j* in the case when a service center is placed into area *i*. We assume, that the service degree has property of transitivity, i.e. if the service centre is in the area x_i and serves area x_j with a degree $\mu_U<x_i,x_j>$, and if the service centre is in area x_j and serves area x_k with a degree $\mu_U<x_j,x_k>$ then a degree of service of area x_k from area x_i not less than $\mu_U<x_i,x_j>\&\mu_U<x_j,x_k>$.

For consideration of questions of optimum allocation of the service centres we shall consider concepts of a fuzzy directed way and fuzzy antibase of the fuzzy graph [4].

Definition 1. Fuzzy directed way $\widetilde{L}(x_i,x_m)$ of fuzzy directed graph $\widetilde{G}=(X,\widetilde{U})$ is called the sequence of fuzzy directed edges from vertex x_i to vertex x_m:

$$\widetilde{L}(x_i,x_m)=<\mu_U<x_i,x_j>/<x_i,x_j>>,<\mu_U<x_j,x_k>/<x_j,x_k>>,..,<\mu_U<x_l,x_m>/<x_l,x_m>>.$$

Conjunctive durability of way $\mu(\widetilde{L}(x_i,x_m))$ is defined as:

$$\mu(\widetilde{L}(x_i,x_m))=\underset{<x_\alpha,x_\beta>\in\widetilde{L}(x_i,x_m)}{\&}\mu_U<x_\alpha,x_\beta>.$$

Fuzzy directed way $\widetilde{L}(x_i,x_m)$ is called simple way between vertices x_i and x_m if its part is not a way between the same vertices.

Obviously, that this definition coincides with the same definition for nonfuzzy graphs.

Definition 2. Vertex *y* is called fuzzy accessible of vertex *x* in the graph $\widetilde{G}=(X,\widetilde{U})$ if exists a fuzzy directed way from vertex *x* to vertex *y*.

The accessible degree of vertex *y* from vertex *x*, $(x\neq y)$ is defined by expression:

$$\gamma(x,y)=\max_\alpha(\mu(\widetilde{L}_\alpha(x,y))),\ \alpha=1,2,...,p,$$

where *p* - number of various simple directed ways from vertex *x* to vertex *y*. Let's consider, that each vertex $x\in X$ in the graph $\tilde{G}=(X,\tilde{U})$ is accessible from itself with an accessible degree $\gamma(x,x)=1$.

Example 1. For the fuzzy graph 1 presented on Fig.1, vertex x_5 is fuzzy accessible vertex from x_1 with an accessible degree:
$\gamma(\mathrm{x}_1,\mathrm{x}_5)=\max\{(0{,}7\ \&\ 0{,}3);(0{,}6\ \&\ 0{,}8)\}=\max\{0{,}3;0{,}6\}\ =0{,}6.$

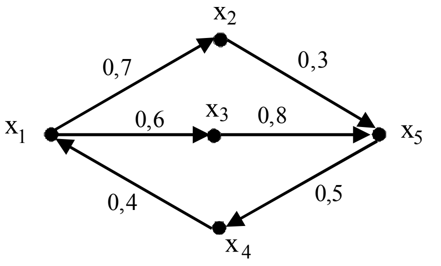


Fig. 1. Fuzzy graph 1.

Let a fuzzy graph $\tilde{G}=(X,\tilde{U})$ is given. Let's define fuzzy multiple-valued reflections $\tilde{\Gamma}^1,\ \tilde{\Gamma}^2,\ \tilde{\Gamma}^3,...,\tilde{\Gamma}^k$ as:
$\tilde{\Gamma}^1(x_i)=\{<\mu_{\Gamma^1(x_i)}(x_j)/(x_j)>\}$, here $(\forall x_j\in X)[\mu_{\Gamma^1(x_i)}(x_j)=\mu_U<x_i,x_j>]$,
$\tilde{\Gamma}^2(x_i)=\tilde{\Gamma}\{\tilde{\Gamma}(x_i)\}$, $\tilde{\Gamma}^3(x_i)=\tilde{\Gamma}\{\tilde{\Gamma}^2(x_i)\}$, ...,
$\tilde{\Gamma}^k(x_i)=\tilde{\Gamma}\{\tilde{\Gamma}^{k\text{-}1}(x_i)\}=\{<\mu_{\Gamma^k(x_i)}(x_j)/x_j>\}$, here
$(\forall x_j\in X)[\mu_{\Gamma^k(x_i)}(x_j)=\underset{\forall x_l\in X}{\vee}\mu_{\Gamma^{k-1}(x_i)}(x_l)\ \&\ \mu_U<x_l,x_j>]\cdot$

It is obvious, that $\tilde{\Gamma}^k(x_i)$ is a fuzzy subset of vertices, which it is accessible to reach from x_i, using fuzzy ways of length *k*.

Example 2. For the fuzzy graph presented on Fig.1, we have: $\tilde{\Gamma}^1(x_1)=\{<0{,}7/(x_2)>,<0{,}6/x_3>\}$, $\tilde{\Gamma}^2(x_1)=\{<0{,}6/x_5>\}$.

Definition 3. Fuzzy transitive closure $\tilde{\tilde{\Gamma}}(x_i)$ is fuzzy multiple-valued reflection:

$$\tilde{\tilde{\Gamma}}(x_i)=\tilde{\Gamma}^0(x_i)\cup\tilde{\Gamma}(x_i)\cup\tilde{\Gamma}^2(x_i)\cup...=\bigcup_{j=0}^{\infty}\tilde{\Gamma}^j(x_i)\cdot$$

Here, by definition: $\tilde{\Gamma}^0(x_i)=\{\ <1/x_i>\ \}$.

In other words, $\tilde{\tilde{\Gamma}}(x_i)$ is fuzzy subset of vertices, which it is accessible to reach from x_i by some fuzzy way with the greatest possible conjunctive durability. As we consider final graphs, it is possible to put, that:

$$\tilde{\tilde{\Gamma}}(x_i) = \bigcup_{j=0}^{n-1} \tilde{\Gamma}^{j}(x_i)\cdot$$

Example 3. For the fuzzy graph presented on Fig.1, fuzzy transitive closure of vertex x_1 is defined as $\tilde{\tilde{\Gamma}}(x_1) = \{<1/x_1>, <0{,}7/x_2>, <0{,}6/x_3>, <0{,}5/x_4>, <0{,}6/x_5>\}\cdot$

Let's define fuzzy reciprocal multiple-valued reflections $\tilde{\Gamma}^{-1}$, $\tilde{\Gamma}^{-2}$, $\tilde{\Gamma}^{-3}$,..., $\tilde{\Gamma}^{-k}$ as:

$\tilde{\Gamma}^{-1}(x_i) = \{<\mu_{\Gamma^{-1}(x_i)}(x_j)/(x_j)>\}$, here $(\forall x_j \in X)[\mu_{\Gamma^{-1}(x_i)}(x_j) = \mu_U < x_j, x_i >]$,

$\tilde{\Gamma}^{-1}(x_i) = \tilde{\Gamma}^{-1}\{\tilde{\Gamma}^{-1}(x_i)\}$, $\tilde{\Gamma}^{-3}(x_i) = \tilde{\Gamma}^{-1}\{\tilde{\Gamma}^{-2}(x_i)\}$, ...,

$\tilde{\Gamma}^{-k}(x_i) = \tilde{\Gamma}^{-1}\{\tilde{\Gamma}^{-(k-1)}(x_i)\} = \{<\mu_{\Gamma^{-k}(x_i)}(x_j)/x_j>\}$, here

$(\forall x_j \in X)[\mu_{\Gamma^{-k}(x_i)}(x_j) = \underset{\forall x_l \in X}{\vee} \mu_{\Gamma^{k-1}(x_i)}(x_l) \,\&\, \mu_U < x_j, x_l >]\cdot$

It is obvious, that $\tilde{\Gamma}^{-k}(x_i)$ is a fuzzy subset of vertices, from which it is accessible to reach vertex x_i, using fuzzy ways of length k.

Example 4. For the fuzzy graph presented on Fig.1, we have $\tilde{\Gamma}^{-1}(x_1) = \{<0{,}4/x_4>\}$, $\tilde{\Gamma}^{-2}(x_1) = \{<0{,}4/x_5>\}\cdot$

Definition 4. Fuzzy reciprocal transitive closure $\tilde{\tilde{\Gamma}}^{-}(x_i)$ is fuzzy reciprocal multiple-valued reflection:

$$\tilde{\tilde{\Gamma}}^{-}(x_i) = \tilde{\Gamma}^{0}(x_i) \cup \tilde{\Gamma}^{-1}(x_i) \cup \tilde{\Gamma}^{-2}(x_i) \cup \ldots = \bigcup_{j=0}^{n-1} \tilde{\Gamma}^{-j}(x_i)\cdot$$

In other words, $\tilde{\tilde{\Gamma}}^{-}(x_i)$ is fuzzy subset of vertices, from which it is accessible to reach vertex x_i by some fuzzy way with the greatest possible conjunctive durability.

Example 5. For the fuzzy graph presented on Fig.1, fuzzy reciprocal transitive closure of vertex x_1 is $\tilde{\tilde{\Gamma}}^{-}(x_1) = \{<1/x_1>, <0{,}3/x_2>, <0{,}4/x_3>, <0{,}4/x_4>, <0{,}4/x_5>\}$.

Definition 5. Graph $\tilde{G}=(X,\tilde{U})$ is named fuzzy strongly connected graph if the condition is satisfied:

$$(\forall \mathrm{x_i} \in \mathrm{X})(\mathrm{S}_{\tilde{\tilde{\Gamma}}(\mathrm{x_i})} = \mathrm{X}). \qquad \mathbf{(1)}$$

Here $\mathrm{S}_{\tilde{\tilde{\Gamma}}(\mathrm{x_i})}$ is the carrier of fuzzy transitive closure $\tilde{\tilde{\Gamma}}(\mathrm{x_i})$.

Differently, graph $\tilde{G}=(X,\tilde{U})$ is fuzzy strongly connected graph if between any two vertices there is a fuzzy directed way with the conjunctive durability which is distinct from 0.

It is easy to show, that expression (1) is equivalent to expression (2):

$$(\forall \mathrm{x_i} \in \mathrm{X})(\mathrm{S}_{\tilde{\tilde{\Gamma}}^{-}(\mathrm{x_i})} = \mathrm{X}). \qquad \mathbf{(2)}$$

Here $S_{\tilde{\tilde{\Gamma}}^{-}(x_i)}$ - is the carrier of fuzzy reciprocal transitive closure $\tilde{\tilde{\Gamma}}^{-}(x_i)$.

Definition 6. Let fuzzy transitive closure for vertex x_i looks like $\tilde{\tilde{\Gamma}}(x_i) = \{<\mu_{i1}(x_1)/x_1>, <\mu_{i2}(x_2)/x_2>, \ldots, <\mu_{in}(x_n)/x_n>\}$, then the volume $\rho(\tilde{G}) = \underset{j=\overline{1,n}}{\&}\ \underset{i=\overline{1,n}}{\&} \mu_{ij}(x_j)$ we name a degree of strong connectivity of fuzzy graph $\tilde{G}$.

Let $\tilde{G}=(X,\tilde{U})$ is fuzzy graph with degree of strong connectivity $\rho(\tilde{G})$, and $\tilde{G}'=(X',\tilde{U}')$ is fuzzy subgraph with degree of strong connectivity $\rho(\tilde{G}')$.

Definition 7. Fuzzy subgraph $\tilde{G}'=(X',\tilde{U}')$ we name maximum strong connectivity fuzzy subgraph or fuzzy strong component connectivity if there is no other subgraph $\tilde{G}''=(X'',\tilde{U}'')$ for which $\tilde{G}' \subset \tilde{G}''$, and $\rho(\tilde{G}') \le \rho(\tilde{G}'')$.

Definition 8. A subset vertices $\overline{B}_\alpha \subset X$ is called fuzzy antibase with the degree $\alpha \in [0,1]$, if some vertex $y \in \overline{B}_\alpha$ may be accessible from any vertex $x \in X / \overline{B}_\alpha$ with a degree not less α and which is minimal in the sense that there is no subset $\overline{B}' \in \overline{B}_\alpha$, having the same accessible property.

Let's designate through $\tilde{R}(\overline{B})$ a fuzzy set of vertices, from which the subset $\overline{B}$ is accessible, i.e.:

$$\tilde{R}(\overline{B}) = \bigcup_{x_i \in \overline{B}} \tilde{\tilde{\Gamma}}^{-}(x_i)\ ,$$

Here, $\tilde{\tilde{\Gamma}}^{-}(x_i)$ is a fuzzy reciprocal transitive closure of the vertex x_i. Then the set $\overline{B}_\alpha$ is fuzzy antibase with a degree α in only case, when the conditions are carried out:

$$\tilde{R}(\overline{B}_\alpha) = \{ <\mu_j/x_j> | x_j \in X \& (\forall j=\overline{1,n})(\mu_j \ge \alpha)\}\ , \tag{3}$$

$$(\forall B' \subset \overline{B}_\alpha)[\tilde{R}(B') = \{ <\mu'_j/x_j> | x_j \in X \& (\exists j=\overline{1,n})(\mu'_j < \alpha)\}]\ . \tag{4}$$

The condition (3) designates, that any vertex either is included into set $\overline{B}_\alpha$, or is accessible from some vertex of the same set with a degree not less α. The condition (4) designates that any subset $\overline{B}' \in \overline{B}_\alpha$ does dot have the property (3).

The following property follows from definition of fuzzy antibase:

Property 1. In fuzzy antibase $\overline{B}_\alpha$ there are no two vertices which are entered into same strong connectivity fuzzy subgraph with degree above or equal α.

Let $\{\mu_1, \mu_2, ..., \mu_L\}$ is a set of all values of membership function which are attributed to edges of graph $\widetilde{G}$. Then the following properties are true:

Property 2. In any fuzzy circuit-free graph exists exactly *L* fuzzy antibases with degrees $\{\mu_1, \mu_2, ..., \mu_L\}$.

Property 3. In any fuzzy circuit-free graph there is just one fuzzy antibase with degree α.

Property 4. If in a fuzzy circuit-free graph an inequality $\alpha_1 < \alpha_2$ is executed, then inclusion $\overline{B}_{\alpha_1} \supset \overline{B}_{\alpha_2}$ is carried out.

Let's note interrelation between fuzzy antibases and strong connectivity fuzzy subgraphs. Following properties are true.

Property 5. If subset $\overline{B}_\alpha$ is fuzzy antibase with degree α, then there is such subset $X' \subseteq X$, that $\overline{B}_\alpha \subset X'$, and fuzzy subgraph $\widetilde{G}' = (X', \widetilde{U}')$ has degree of strong connectivity not less α.

Property 6. If subset $\overline{B}_\alpha$ is fuzzy antibase with degree α, then there is not such subset $X' \subseteq X$, that $X' \subseteq \overline{B}_\alpha$ and fuzzy subgraph $\widetilde{G}' = (X', \widetilde{U}')$ has degree of strong connectivity α.

The following consequence follows from property 6:

Consequence 1. That in fuzzy graph $\widetilde{G}$ there was fuzzy antibase with degree α, consisting of only one vertex, it is necessary that fuzzy graph $\widetilde{G}$ was strong connectivity with degree α.

Property 7. Let $\gamma(x_i,x_j)$ is an accessible degree of vertex x_j from vertex x_i. Then the following statement is true:

$$(\forall x_i,x_j \in \overline{B}_\alpha)[\gamma(x_i,x_j)<\alpha]. \qquad \textbf{(5)}$$

Differently, the accessible degree of any vertex $x_j \in \overline{B}_\alpha$ from any other vertex $x_i \in \overline{B}_\alpha$ is less than meaning α.

Let a set $\tau_k=\{X_{k1}, X_{k2},...,X_{kl}\}$ be given, where X_{ki} is a fuzzy *k*-vertex antibase with the degree of α_{ki}. We define as α_k = max $\{\alpha_{k_1}, \alpha_{k_2}, ..., \alpha_{k_l}\}$. In the case $\tau_k=\varnothing$ we define $\alpha_k=\alpha_{k-1}$. Volume α_k means that fuzzy graph $\widetilde{G}$ includes k-vertex subgraph with the accessible degree α_k and doesn't include k-vertex subgraph with an accessible degree more than α_k.

Definition 5. A fuzzy set

$$\widetilde{B}^- = \{<\alpha_1/1>, <\alpha_2/2>, ..., <\alpha_n/n>\}$$

is called a fuzzy set of antibases of fuzzy graph $\widetilde{G}$.

Property 8. The following proposition is true:

$$0 \le \alpha_1 \le \alpha_2 \le ... \le \alpha_n = 1.$$

The fuzzy set of antibases defines the greatest degree of service of all territory by the k centres of service ($k\in\{1,2,\dots,n\}$).
Thus, it is necessary to determine a fuzzy set of antibases for a finding of the greatest degree.

3 Method for finding of fuzzy set of antibases

We will consider the method of finding a family of all fuzzy antibases with the highest degree. The given method is an analogue method for definition of all minimal fuzzy dominating vertex sets [5] and it is a generalization of the Maghout's method for crisp graphs [6].

Assume that a set $\overline{B}_\alpha$ is a fuzzy base of the fuzzy graph $\widetilde{G}$ with the degree α. Then for an arbitrary vertex $x_i\in X$, one of the following conditions must be true.
a) $x_i\in \overline{B}_\alpha$;
b) if $x_i\notin \overline{B}_\alpha$, then there is a vertex x_j such that it belongs to the set $\overline{B}_\alpha$ with the degree $\gamma(x_i,x_j)\geq\alpha$.
In other words, the following statement is true:

$$(\forall x_i \in X)[x_i \in \overline{B}_\alpha \vee (x_i \notin \overline{B}_\alpha \to (\exists x_j \in \overline{B}_\alpha | \gamma(x_i,x_j) \geq \alpha))]. \qquad \textbf{(6)}$$

To each vertex $x_i\in X$ we assign Boolean variable p_i that takes the value 1, if $x_i\in \overline{B}_\alpha$ and 0 otherwise. We assign a fuzzy variable $\xi_{iji}=\alpha$ for the proposition $\gamma(x_i,x_j)\geq\alpha$. Passing from the quantifier form of proposition (4) to the form in terms of logical operations, we obtain a true logical proposition:

$$\Phi = \underset{i}{\&}(p_i \vee (\overline{p}_i \to (\underset{j}{\vee}(p_j \& \gamma_{ij})))).$$

Taking into account interrelation between implication operation and disjunction operation ($\alpha\to\beta= \overline{\alpha}\vee\beta$), we receive:

$$\Phi = \underset{i}{\&}(p_i \vee p_i \vee \underset{j}{\vee}(p_j \& \gamma_{ij})).$$

Supposing $\xi_{ii} = 1$ and considering that the equality $p_i \vee \underset{j}{\vee} p_i \& \xi_{ij} = \underset{j}{\vee} p_j \xi_{ij}$ is true for any vertex x_i, we finally obtain:

$$\Phi = \underset{i}{\&}(\underset{j}{\vee}(p_j \& \gamma_{ij})). \qquad \textbf{(7)}$$

We open the parentheses in the expression (7) and reduce the similar terms the following rules:

$$a\vee a\&b=a;\ a\&b\vee a\&\ \overline{b}=a;\ \xi'\&a\vee\xi''\&a\&b. \qquad \textbf{(8)}$$

Here, $a,b\in\{0,1\}$, $\xi'\geq\xi','$ $\xi',\xi''\in[0,1]$.
Then the expression (7) may be rewrite as:

$$\Phi = \underset{i=\overline{1,l}}{\vee}(p_{1_i} \ \& \ p_{2_i} \ \& \ldots \& \ p_{k_i} \ \& \ \alpha_i) \cdot \qquad \textbf{(9)}$$

Property 9. If in expression (9) further simplification on the basis of rules (8) is impossible, then everyone disjunctive member i defines antibase with the highest degree α_i.

Proof. Let's consider, that further simplification is impossible in expression (9). Let, for definiteness, disjunctive member

$$(p_1 \ \& \ p_2 \ \& \ldots \& \ p_k \ \& \ \alpha), k < n, \alpha \in (0,1] \qquad \textbf{(10)}$$

is included in the expression (9).

Let's assume, that the subset $X' = \{x_1, x_2, \ldots, x_k\}$ is not antibase with degree α. Then there is some vertex, for example $x_{k+1} \in X / X'$, for which the statement $(\forall i = \overline{1,k})(\gamma(x_{k+1}, x_i) < \alpha)$ is true. In other words, the accessible degree of any vertex of subset X' from vertex x_{k+1} is less than value α.

We present the expression (7) in a kind:

$$\begin{aligned}\Phi = {} & (1p_1 \vee \xi_{12}p_2 \vee \ldots \vee \xi_{1n}p_n) \ \& \ (\xi_{21}p_1 \vee 1p_2 \vee \ldots \vee \xi_{2n}p_n) \ \& \ldots \& \\ & (\xi_{k+1,1}p_1 \vee \xi_{k+1,2}p_2 \vee \ldots \xi_{k+1,k}p_k \vee 1p_{k+1} \vee \ldots \vee \xi_{k+1,n}p_n) \ \& \ldots \\ & \& \ (\xi_{n1}p_1 \vee \xi_{n2}p_2 \vee \ldots \vee 1p_n).\end{aligned} \qquad \textbf{(11)}$$

In expression (11) all coefficients $\xi_{k+1,i} < \alpha, \forall i = \overline{1,k}$. Therefore, in expression (9) all disjunctive members which do not contain variables $p_{k+1}, p_{k+2}, \ldots, p_n$, necessarily contain coefficients smaller value α. From here follows, that the disjunctive member (10) is not included in the expression (9). The received contradiction proves, that subset $X' = \{x_1, x_2, \ldots, x_k\}$ is antibase with degree α.

Let's prove now, that the disjunctive member (10) is minimum member. We will assume the return. Then following conditions should be carried out:

a) subset $X' = \{x_1, x_2, \ldots, x_k\}$ is antibase with degree $\beta > \alpha$;

or

b) there is a subset $X'' \subset X'$ which is antibase with degree α.

Let the condition a) is satisfied. Then the next statement is true:

$$(\forall x_j, j = \overline{k+1,n})(\exists x_i, i \in \overline{1,k} \mid \gamma(x_j, x_i) \geq \beta) \cdot$$

Let's present expression Φ in the form of (11). If to make logic multiplication of each bracket against each other without rules of absorption (8) we will receive n^2 the disjunctive members containing exactly n of elements, and, on one element from each bracket of decomposition (11). We will choose one of n^2 disjunctive members as follows:

- From the first bracket we will choose element $1p_1$;
- From the second bracket - element $1p_2$; …;

- From *k*th bracket - element $1p_k$;

- From (*κ+1*)th bracket we will choose element $\xi_{k+1,i_1} p_{i_1}$ such, that index $i_1 \in [1,k]$, and volume $\xi_{k+1,i_1} > \beta$;

- From (*κ+2*)th bracket - element $\xi_{k+2,i_2} p_{i_2}$, for which index $i_2 \in [1,k]$, and volume $\xi_{k+2,i_2} > \beta$, etc.;

- From *n*th bracket - element $\xi_{n,i_{n-k1}} p_{i_{n-k}}$, for which index $i_{n-k} \in [1,k]$, and volume $\xi_{n,i_{n-k1}} > \beta$.

Using rules of absorption (8), the received disjunctive member can be led to kind $(p_1 \ \& \ p_2 \ \& \ldots \& \ p_k \ \& \ \beta')$, in which the volume $\beta' = \min\{ \xi_{k+1,i_2}, \xi_{k+2,i_2}, \ldots, \xi_{n,i_{n-k}} \} \geq \beta > \alpha$ and which will be necessarily absorbed disjunctive member (10). (Decomposition (9) the disjunctive member cannot include the received contradiction (10)) proves impossibility of case a).

Let's assume now, that the condition is satisfied. Let, for definiteness, $X'' = \{x_1, x_2, \ldots, x_{k-1}\}$. Considering expression Φ in the form of decomposition (8), we will choose a disjunctive member as follows:

- From the first bracket we will choose element $1p_1$,

- From the second bracket - element $1p_2$, …,

- From *(κ-1)*th bracket - element $1p_{k-1}$,

- From *k*th bracket we will choose element $\xi_{k,i_1} p_{i_1}$ such, that index $i_1 \in [1,k-1]$, and volume $\xi_{k,i_1} \geq \alpha$,

- From *(κ+1)*th bracket - element $\xi_{k+1,i_2} p_{i_2}$, for which index $i_2 \in [1,k-1]$, and volume $\xi_{k+1,i_2} \geq \alpha$, etc.,

- From *n*th bracket - element $\xi_{n,i_{n-k+11}} p_{i_{n-k+1}}$, for which index $i_{n-k+1} \in [1,k-1]$, and volume $\xi_{n,i_{n-k+1}} \geq \alpha$.

Using rules of absorption (8) the received disjunctive member can be led to kind $(p_1 \ \& \ p_2 \ \& \ldots \& \ p_{k-1} \ \& \ \alpha)$, in which size $\beta = \min\{ \xi_{k,i_2}, \xi_{k+1,i_2}, \ldots, \xi_{n,i_{n-k+1}} \} \geq \alpha$ and which will be necessarily absorbed by a disjunctive member (10). (Decomposition (9) the disjunctive member cannot include the received contradiction (10)) proves impossibility of case b).

Property 9 is proved.

The following method of foundation of fuzzy antibases may be proposed on the base of property 9:

- We write proposition (7) for given fuzzy graph $\widetilde{G}$;
- We simplify proposition (7) by proposition (8) and present it as proposition (9);
- We define all fuzzy antibases, which correspond to the disjunctive members of proposition (9).

Example 6. Find all fuzzy antibases for the fuzzy graph 2 presented in Fig.2:

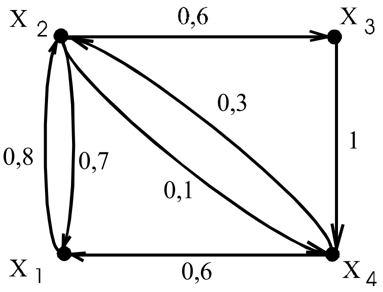


Fig. 2. Fuzzy graph 2.

The vertex matrix for this graph has the following form:

$$R = \begin{pmatrix} 0 & 0{,}8 & 0 & 0 \\ 0{,}7 & 0 & 0{,}6 & 0{,}1 \\ 0 & 0 & 0 & 1 \\ 0.6 & 0{,}3 & 0 & 0 \end{pmatrix}.$$

On the basis of the made above definition of fuzzy accessible vertex we can construct an accessible matrix *N*, containing accessible degrees for all pairs of vertices:

$$N=\left\|\gamma_{ij}\right\|_n = \bigcup_{k=0}^{n-1} R^k \cdot$$

Here, $\gamma_{ij}=\gamma(x_i,x_j)$, x_i, $x_j \in X$, R^k – the vertex matrix power *k* for graph. Matrix R^0 is an identity matrix:

$$R^0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

We raise the contiguity matrix to 2, 3 powers. Uniting them, we find an accessible matrix:

$$N = R^0 \cup R \cup R^2 \cup R^3 = \begin{pmatrix} 1 & 0{,}8 & 0{,}6 & 0{,}6 \\ 0{,}7 & 1 & 0{,}6 & 0{,}6 \\ 0{,}6 & 0{,}6 & 1 & 1 \\ 0{,}6 & 0{,}6 & 0{,}6 & 1 \end{pmatrix}.$$

The corresponding expression (7) for this graph has the following form:

$$\begin{aligned}\Phi &= (p_1 \vee p_2 0{,}8 \vee p_3 0{,}6 \vee p_4 0{,}6) \,\&\, (p_1 0{,}7 \vee p_2 \vee p_3 0{,}6 \vee p_4 0{,}6) \,\& \\ &\& (p_1 0{,}6 \vee p_2 0{,}6 \vee p_3 \vee p_4) \,\&\, (p_1 0{,}6 \vee p_2 0{,}6 \vee p_3 0{,}6 \vee p_4 0{,}6).\end{aligned}$$

Multiplying parenthesis 1 and 2, parenthesis 2 and 4 and using rules (8) we finally obtain:

$$\Phi = p_1 0{,}6 \vee p_1 p_4 0{,}7 \vee p_1 p_2 p_4 \vee p_2 0{,}6 \vee p_2 p_4 0{,}8 \vee p_3 0{,}6 \vee p_4 0{,}6.$$

It follows from the last equality that the graph $\widetilde{G}$ has 7 fuzzy antibases, and fuzzy set of antibases is defined as:

$$\widetilde{B}^{-} = \{< 0{,}6/1 >, < 0{,}8/2 >, < 1/3 >, < 1/4 >\}.$$

The fuzzy set of antibases defines the next optimum allocation of the service centres: If we have 3 or more service centres then we must place these centres into vertices 1, 2, and 4. The degree of service equals 1 in this case. If we have 2 service centres then we must place these centres into vertices 2, and 4. The degree of service equals 0,8 in this case. If we have only one service centre then we can place it in any vertex. The degree of service equals 0,6 in last case.

3 Conclusion

The task of definition of optimal allocation of the service centres as the task of definition of fuzzy antibases of fuzzy graph was considered. The definition method of fuzzy antibases is the generalization of Maghout's method for nonfuzzy graphs. It is necessary to mark that the suggested method is the method of ordered full selection, because these tasks are reduced to the task of covering, i.e. these tasks are NP-compete tasks. However, this method is effective for the graphs which have not homogeneous structure and not large dimensionality.

References

1. Christofides, N.: Graph theory. An algorithmic approach. Academic press, London (1976)
2. Malczewski, J.: GIS and multicriteria decision analysis. John Willey and Sons, New York (1999)
3. Kaufmann, A.: Introduction a la theorie des sous-ensemles flous. Masson, Paris (1977)
4. Bershtein, L.S., Bozhenuk, A.V.: Fuzzy graphs and fuzzy hypergraphs. In: Dopico, J., de la Calle, J., Sierra, A. (eds.) Encyclopedia of Artificial Intelligence, Information SCI, pp. 704--709. Hershey, New York (2008)
5. Bershtein, L.S., Bozhenuk, A.V.: Maghout method for determination of fuzzy independent, dominating vertex sets and fuzzy graph kernels. J. General Systems. 30: 45--52 (2001)
6. Kaufmann, A.: Introduction a la combinatorique en vue des applications. Dunod, Paris (1968)

Forecasting the U.S. Stock Market via Levenberg-Marquardt and Haken Artificial Neural Networks Using ICA&PCA Pre-Processing Techniques

Golovachev Sergey

National Research University, Higher School of Economics, Moscow
Department of World Economics and International Affairs

Abstract. Artificial neural networks (ANN) is an approach to solving different tasks. In this paper we forecast U.S. stock market movements using two types of artificial neural networks: a network based on the Levenberg-Marquardt learning mechanism and a synergetic network which was described by German scientist Herman Haken. The Levenberg-Marquardt ANN is widely used for forecasting financial markets, while the Haken ANN is mostly known for the tasks of image recognition. In this paper we apply the Haken ANN for the prediction of the stock market movements. Furthermore, we introduce a novation concerning preprocessing of the input data in order to enhance the predicting power of the abovementioned networks. For this purpose we use Independent Component Analysis (ICA) and Principal Component Analysis (PCA). We also suggest using ANNs to reveal the "mean reversion" phenomenon in the stock returns. The results of the forecasting are compared with the forecasts of the simple auto-regression model and market index dynamics.

Keywords: artificial neural network, back-propagation, independent component analysis, principal component analysis, forecast.

1 The Levenberg-Marquardt Network

Artificial neural networks are a modern approach to various problem-solving tasks. For example, they are used for image recognition and in different biophysics researches. One of the possible applications of ANNs is forecasting and simulation of financial markets. The idea is the following: a researcher tries to construct such an ANN, so that it can successfully imitate the decision-making process of the "average" stock market participant. This hypothesis results from the fact that ANNs, in turn, try to imitate the design of the biological neural networks, in particular the ones which exist in human brain.

A market participant is an investor whose individual actions have no influence on the price fluctuations, for example a trader operating with insignificant sums of money. Moreover, we argue that the market participant makes his decisions solely on the analysis of the previous dynamics of the stock – thus we assume endogenous price-making mechanism. Furthermore, we set the homogeneity of the investors so that they all have the same decision-making algorithms (that is why we call them “average”).

While designing the Levenberg-Marquardt ANN it is essential to set some of the key parameters of the network. Firstly, we must set the architecture of the network (number of layers, number of neurons in each, including number of input and output neurons). In our research we use simple three-layer ANN with 2 input neurons, 2 neurons in the hidden layer and 1 output neuron. The results show that such architecture is quite effective while it does not lead to lengthy computational procedure. Secondly, we determine the activation function in the hidden layer which performs a non-linear transformation of the input data. We use a standard logistic function with the range of values [0;1].

The key feature of the Levenberg-Marquardt ANN is using of back-propagation of the errors of the previous iterations as a learning mechanism. The idea of back-propagation rests on the attempt of communicating the error of the network (of the output neuron, in particular) to all other neurons of the network. As a result, after a number of iterations the network optimizes the weights with which neurons in different layers are connected, and the minimum of error is reached. Propagation of the error through the network also requires usage of the Jacobian matrix which contains first derivatives of the elements of the hidden and input layers.

The computational mechanism is as follows (1):

$$w_{new} = w_{old} - (Z^T Z + \lambda I)^{-1} * Z^T * \varepsilon(w_{old}), \tag{1}$$

where
w_{old}– weight vector of the previous iteration;
w_{new} – weight vector of the current iterations;
Z – Jacobian matrix with the dimensionality m×n; m – is the number of learning examples for each iteration, n – total number of weights in the network;
λ – learning ratio;
I – identical matrix with the dimensionality n×n;
ϵ- vector of n elements, which contains forecast errors for each learning example.

To enhance the predicting power of our model we introduce here pre-processing techniques of the Independent Component Analysis (ICA). This is a method of identifying the key and important signals in the large, noisy data. ICA is often compared with another useful processing tool – Principal Component Analysis (PCA). However, the general difference of ICA from PCA is that we obtain purely independent vectors on which a process can be decomposed, whereas PCA requires only non-correlatedness of such vectors. Moreover, ICA allows non-Gaussian distributions, which is quite useful and realistic assumption, especially for financial data.

The ICA stems from the so-called “cocktail party” problem in acoustics. The problem is the following: assume that we have i number of people(s) talking in the room and j number of microphones(x) which record their voices. For two people and two microphones signals from the microphones are as follows in (2):

$$
\begin{aligned}
x_1 &= a_{11} * s_1 + a_{12} * s_2 \\
x_2 &= a_{21} * s_1 + a_{22} * s_2
\end{aligned}
\tag{2}
$$

Consequently, we should set mixing matrix A which transforms voices into the recordings, (1):

$$
A = \left\{ \begin{matrix} a_{11} \; a_{12} \cdots \; a_{li} \\ a_{21} \; a_{22} \cdots \; a_{2i} \\ a_{j1} \; a_{j2} \cdots \; a_{ji} \end{matrix} \right\} \tag{3}
$$

The task for the researcher consists then in finding a demixing matrix A^{-1} which enables to get the vector of voices s knowing only the vector of the recordings x, (4):

$$
s = A^{-1} * x \tag{4}
$$

When we apply ICA for the stock market we assume that the empirical stock returns are the "recordings", the noisy signals of the original "voices" which determine the real process of price movements. Consequently, when we obtain a de-mixing matrix A^{-1}then we get a powerful tool for extracting the most important information about the price movements. Furthermore, ICA allows us to reduce the dimensionality of the empirical data without losing significant information. It is very important while using ANNs, because, on the one hand we should present the network as much relevant information as possible, but, on the other hand, too much input information leads to lengthy computational procedures and problems with convergence to a nontrivial solution.

As it was mentioned above, we use two types of inputs in the Levenberg-Marquardt ANN. First input is the logarithmic return of the stock for the day which precedes the day of the forecast. Second input is derived from the processing of ten previous logarithmic returns with ICA algorithm: we get the de-mixing matrix A^{-1} and the subsequent vector of independent components s. Then we transform this vector to the scalar value considering the most influential independent component.

In the section "Results" we show that such pre-processing turns out to be very useful for stock market forecasting. Moreover, it is worth mentioning that ICA can be used as a self-sufficient forecasting tool for various financial markets.

2 The Haken network

The second ANN which is used for forecasting U.S. stock market is quite different from the Levenberg-Marquardt network. It is the network of Herman Haken, German scientist and the founder of synergetics.

This ANN is self-learning and uses a "library" of pre-set values which by default represent all possible states of the process. Therefore, during a number of iterations the network converges to one of these values. The Haken ANN is widely used for image recognition. For example, when we set a task of recognizing letters of the alphabet we use the whole alphabet as a pre-set "library". It is obvious, because any letter which is presented to the network is essentially a part of the alphabet.

However, we aim to apply the Haken ANN for the stock market forecasting, and the situation here is much more complicated. We must choose the "library" which contains all possible states of the market. To solve this task we resort to two important assumptions. Firstly, we argue that all necessary information which is needed for the forecast is contained in the returns of the stock during ten trading days before the day for which the forecast is calculated. Secondly, we assume that the using of processing techniques of the ICA and PCA, which eventually reduces the information dimensionality, allows us to extract most important and valuable information signals.

Thus, to obtain the "library" of pre-set values we use the eigenvectors of the covariance matrix of the subsequent empirical vectors of stock returns (PCA) or the de-mixing matrix of the empirical vectors obtained from ICA.

The network functions as follows, (5):

$$q^* = q + \sum_{k=1}^{M} \lambda_k v_k^T q v_k + B \sum_{k=1}^{M} (v_k^T q)^2 (v_k^T q) v_k + C(q^T q) q, \tag{5}$$

where

q – vector of M elements which the network tries to optimize. Initially this vector is deliberately made noisy to ignite the process of learning, thus we assume that the real data on the stock market is also noisy in the similar way;

q^*- vector which the network finally reconstructs;

V – matrix which plays a role of the "library" and contains pre-set values which are obtained from PCA or ICA;

λ – learning ratio;

B – computational parameters which calibrating has an influence on the convergence of the network and the speed of learning.

The final forecasting signal is obtained by subtracting the empirical vector from the reconstructed one.

3 Trading rules

Now we present trading rules which were used while working with the Levenberg-Marquardt and the Haken ANNs.

Firstly, we should specify the data which we forecast. We predict price movements of the 30 liquid stocks of the U.S. index S&P 500[11] in the period from November, 7th, 2008 to May,2nd, 2010.

For each trading day t we make forecast via our ANNs for each stock. When the forecasts are made we range them according to their absolute value. The final selection of the stocks in the virtual portfolio is based on two opposing trading rules. According to the Rule A we select from 1 to 5 stocks with the highest forecast value[22] (note that at this step we do not know the real return of day t which makes our forecast truly out-of-sample). According to the Rule B we select from 1 to 5 stocks with the lowest forecast values.

The reason for using Rule B is widely recognized phenomenon of “mean reversion” in the financial data. Thus, if Rule B is successful, then our ANN is capable of detecting this property of the market.

The dynamics of our trade portfolio will be compared to the dynamics of the S&P 500 index and the dynamics of the portfolio if the decision-making was based on the simple auto-regression model (while the trading rules A and B are retained), (6):

$$r_t^* = \alpha_t + \beta_t * r_{t-1}, \tag{6}$$

where
r_t^* - forecast value of the logarithmic return of the stock for the trading day t,
α_t, β_t-auto-regression coefficients,
r_{t-1}- logarithmic return of the stock for the trading day t-1.

4 Results

Now we present some of the results of the forecasting using the Levenberg-Marquardt and the Haken ANNs. Due to the limited space of this paper we demonstrate here only most successful examples.

Figure 1 demonstrates relative dynamics of our virtual portfolio (red line) using the Levenberg-Marquardt ANN and trading Rule B (five stocks with “worst” forecasts). Blue line is a portfolio when the decision-making is based on the auto-regression model. Blue line is the S&P 500 index. The horizontal axis is time and t indicates trading days. The vertical axis displays the value of the portfolio with the initial value of 1.

Figure 2 demonstrates relative dynamics of our virtual portfolio (red line) using the Haken ANN with the PCA pre-processing and trading Rule B (one stock with the “worst” forecast). Blue line is a portfolio when the decision-making is based on the auto-regression model. Green line is the S&P 500 index.

[1] We use closing prices of the following stocks: ExxonMobil, Apple, Microsoft, General Electric, Procter&Gamble, Johnson&Johnson, Bank of America, JPMorgan Chase, Wells Fargo, IBM, Chevron, Sisco Systems, AT&T, Pfizer, Google, Coca Cola, Intel, Hewlett Packard, Wal Mart, Merck, PepsiCo, Oracle, Philip Morris International, ConocoPhillips, Verizon Communications, Schlumberger, Abbott Labs, Goldman Sachs, Mcdonalds, QUALCOMM.

[2] $2Notethatinthismodelweuseonlylongpositions, shortsellingisnotallowed.$

The horizontal axis is time and t indicates trading days. The vertical axis displays the value of the portfolio with the initial value of 1.

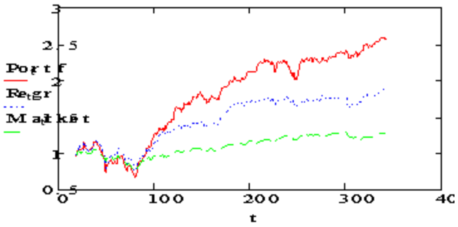


Fig. 1.

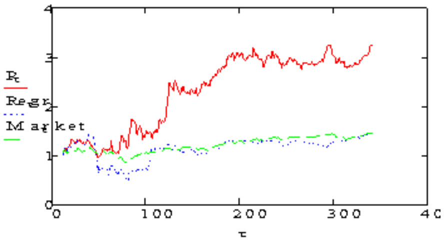


Fig. 2.

Figure 3 demonstrates relative dynamics of our virtual portfolio (red line) using the Herman Haken ANN with the ICA pre-processing and trading Rule A (one stock with the "best" forecast). Blue line is a portfolio when the decision-making is based on the auto-regression model. Green line is the S&P 500 index. The horizontal axis is time and t indicates trading days. The vertical axis displays the value of the portfolio with the initial value of 1.

5 Conclusions

Using of pre-processing techniques of the ICA and PCA with the ANNs proved to be a reliable decision-support mechanism for trading on the liquid stock market. Dynamics of the subsequent portfolios outperform portfolios which follow simple auto-regression forecast or linked to the stock index. Furthermore, the Levenberg-Marquardt and Haken ANNs displayed the ability to reveal the "mean reversion" phenomenon in the complex market data and use it for future forecasts.

However, despite the success of the Levenberg-Marquardt and the Haken ANNs and proper pre-processing techniques we still face difficulties in making up a strategy which will guarantee robust and stable growth of the portfolio over continuous period of time. Moreover, more theoretical research is needed to

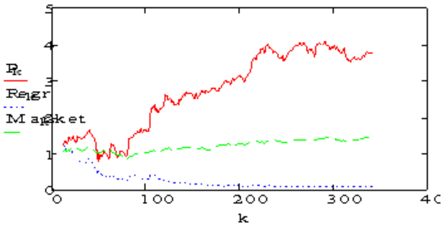


Fig. 3.

justify the argument that it is the neural network decision-making mechanism which is used by traders in real life. It is also obvious that more in-depth study is needed to explain the phenomenon of “mean reversion”. Some of these issues will be the topics of the future research.

References

1. Back A.D., Weigend A.S. A First Application of Independent Component Analysis to Extracting Structure from Stock Returns// International Journal of Neural Systems, Vol. 8, No.5 (October, 1997).
2. Bishop C.M. Neural Networks for Pattern Recognition. Oxford University Press, 1995 – 483 p.
3. Bell J.I., Sejnowsi T. J. An information-maximisation approach to blind separation and blind deconvolution//Neural Computation, 7, 6, 1004-1034 (1995).
4. Grriz J.M., Puntonet C.G., Moiss Salmern, E.W. Lang Time Series Prediction using ICA Algorithms//IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications 8-10 September 2003, Lviv, Ukraine.
5. Hyvrinen A., Oja E. Independent Component Analysis: Algorithms and Applications//Neural Networks, 13(4-5):411-430, 2000.
6. Krse B., van der Smagt P. An Introduction To Neural Networks, Eight Edition, November 1996.
7. Lu C.-J., Le T.-S., Chiu C.-C. Financial time series forecasting using independent component analysis and support vector regression//Decision Support Systems 47 (2009) 115–125.

Estimating Probability of Failure of a Complex System Based on Partial Information about Subsystems and Components, with Potential Applications to Aircraft Maintenance

Christelle Jacob[1], Didier Dubois[2], Janette Cardoso[1], Martine Ceberio[3], and Vladik Kreinovich[3]

[1] Institut Supérieur de l'Aéronautique et de l'Espace (ISAE), DMIA department,
Campus Supaéro, 10 avenue Édouard Belin, Toulouse, France
jacob@irit.fr, cardoso@isae.fr

[2] Institut de Recherche en Informatique de Toulouse (IRIT), ADRIA department,
118 Route de Narbonne 31062 Toulouse Cedex 9, France
dubois@irit.fr

[3] University of Texas at El Paso, Computer Science Dept., El Paso, TX 79968, USA
mceberio@utep.edu, vladik@utep.edu

Abstract. In many real-life applications (e.g., in aircraft maintenance), we need to estimate the probability of failure of a complex system (such as an aircraft as a whole or one of its subsystems). Complex systems are usually built with redundancy allowing them to withstand the failure of a small number of components. In this paper, we assume that we know the structure of the system, and, as a result, for each possible set of failed components, we can tell whether this set will lead to a system failure. In some cases, for each component A, we know the probability $P(A)$ of its failure; in other cases, however, we only know the lower and upper bounds $\underline{P}(A)$ and $\overline{P}(A)$ for this probability. Sometimes, we only have expert estimates for these probabilities, estimates that can be described as fuzzy numbers.

Usually, it is assumed that failures of different components are independent events, but sometimes, we know that they are dependent – in which case we usually do not have any specific information about their correlation. Our objective is to use all this information to estimate the probability of failure of the entire the complex system. In this paper, we describe methods for such estimation.

Keywords: complex system, probability of failure, interval uncertainty, fuzzy uncertainty, stochastic dependence

1 Formulation of the Problem

It is necessary to estimate the probability of failure for complex systems. In many practical situations applications, we need to estimate the probability of failure of a complex system. The need for such estimates come from the fact that in practice, while it is possible (and desirable) to minimize the risk, it is not possible to completely eliminate the risk: no matter how many precautions we take, there are always some very low probability events that can potentially lead to a system's failure. All we can do is to make sure that the resulting probability of failure does not exceed the desired small value p_0. For example, the probability of a catastrophic event is usually required to be at or below $p_0 = 10^{-9}$.

For example, in aircraft design and maintenance, we need to estimate the probability of a failure of an aircraft as a whole and of its subsystems. On the design stage, the purpose of this estimate is to make sure that this probability of failure does not exceed the allowed probability p_0. On the maintenance stage, we perform this estimate to decide whether a maintenance is needed: if the probability of failure exceeds p_0, this means that we need to perform some maintenance to decrease this probability to the desired level p_0 (or below).

Information available for estimating system's probability of failure: general description. Complex systems consist of subsystems, which, in turn, consists of components (or maybe of sub-subsystems which consist of components). So, to estimate the probability of failure of a complex system, we need to take into account:

- when the failure of components and subsystems lead to the failure of the complex system as a whole,
- how reliable are these components and subsystems, and
- are the component failures independent events or they are caused by a common cause, and

When the failure of components and subsystems lead to the failure of the complex system as a whole? Complex systems are usually built with redundancy allowing them to withstand the failure of a small number of components. Usually, we know the structure of the system, and, as a result, for each possible set of failed components, we can tell whether this set will lead to a system failure. So, in this paper, we will assume that this information is available.

How reliable are components and subsystems? What do we know about the reliability of individual components? For each component A, there is a probability $P(A)$ of its failure. When we have a sufficient statistics of failures of this type of components, we can estimate this probability as the relative frequency of cases when the component failed. In some case, we have a large number of such cases, and as a result, the frequency provides a good approximation to the desired probability – so that, in practice, we can safely assume that we know the actual values of these probabilities $P(A)$.

In some case, we only have a few failure cases, not enough to get an accurate estimate for $P(A)$. In this case, the only information that we can extract from the observation is the interval $\mathbf{P}(A) = [\underline{P}(A), \overline{P}(A)]$ that contains the actual (unknown) value of this probability.

This situation is rather typical for aircraft design and maintenance, because aircrafts are usually built of highly reliable components – at least the important parts of the aircraft are built of such components – and there are thus very few observed cases of failure of these components.

In some cases, especially on the design stage, we do not yet have failure statistics, so we have to rely on expert estimates, estimates based on the expert's experience with similar components. These estimates are usually formulated by using words from natural language, like "about 1%". A natural way to describe such expert estimates is to use fuzzy techniques (see, e.g., [9, 14]), i.e., to describe each such estimate as a *fuzzy number* $\mathcal{P}(A)$. A fuzzy number means, crudely speaking, that for different degrees of uncertainty α, we have an interval $\mathcal{P}(A, \alpha)$ that contains the actual (unknown) probability $P(A)$ with this degree of uncertainty: e.g., the interval $\mathcal{P}(A, 0)$ contains $P(A)$ with guarantee (uncertainty 0), while the interval $\mathcal{P}(A, 0.5)$ contains $P(A)$ with uncertainty 0.5.

Are the component failures independent events or they are caused by a common cause? In many practical situations, failures of different components are caused by different factors. For example, for an aircraft, possible failures of mechanical subsystems can be caused by the material fatigue, while possible failures of electronic systems can be caused by the interference of atmospheric electricity (e.g., when flying close to a thunderstorm). As a result, usually, it is assumed that failures of different components are independent events

However, sometimes, we know that the failures of different components are caused by a common cause. In this case, the failures of different components are no longer independent events. In such situations, often, do not have any specific information about the correlation between these failures. This is a typical situation in aircraft design and maintenance – we have very few failures of each component, not enough to determine the exact probability of each such failure, and we have even fewer situations when two components failed, so an empirical determination of such correlations is out of question.

What we do in this paper. Our objective is to use all this information to estimate the probability of failure of the entire the complex system. In this paper, we describe methods for such estimation.

2 Simplest Case: Component Failures are Independent and Failure Probabilities $P(A)$ Are Exactly Known

Let us start our analysis with the simplest case when the component failures are independent and the failure probabilities $P(A)$ for different components A are known exactly.

As we mentioned, we assume that there exists an efficient algorithm that, given a list of failed components, determines whether the whole system fails or not.

In this case, it is always possible to efficiently estimate the probability P of the system's failure by using Monte-Carlo simulations. Specifically, we select the number of simulations N. Then, for each component A, we simulate a Boolean variable *failing(A)* which is true with probability $P(A)$ and false with the remaining probability $1 - P(A)$. This can be done, e.g., if we take the result r of a standard random number generator that generates values uniformly distributed on the interval $[0, 1]$ and select *failing(A)* to be true if $r \le P(A)$ and false otherwise: then the probability of this variable to be true is exactly $P(A)$.

Then, we apply the above-mentioned algorithm to the simulated values of the variables *failing(A)* and conclude whether for this simulation, the system fails or not. As an estimate for the probability of the system's failure, we then take the ratio $p \stackrel{\text{def}}{=} f/N$, where f is the number of simulations on which the system failed. From statistics, it is known that the mean value of this ratio is indeed the desired probability, that the standard deviation can be estimated as $\sigma = \sqrt{p \cdot (1-p)/N} \le 0.5/\sqrt{N}$, and that for sufficiently large N (due to the Central Limit Theorem), the distribution of the difference $P - p$ is close to normal. Thus, with probability 99.9%, the actual value P is within the three-sigma interval $[p - 3\sigma, p + 3\sigma]$.

This enables us to determine how many iterations we need to estimate the probability P with accuracy 10% (and certainty 99.9%): due to $\sigma \le 0.5/\sqrt{N}$, to guarantee that $3\sigma \le 0.1$, it is sufficient to select N for which $3 \cdot 0.5/\sqrt{N} \le 0.1$, i.e., $\sqrt{N} \ge (3 \cdot 0.5)/0.1 = 15$ and $N \ge 225$. It is important to emphasize that this number of iterations is the same no matter how many components we have – and for complex systems, we usually have many thousands of components.

Similarly, to estimate this probability with accuracy 1%, we need $N = 22,500$ iterations, etc. These numbers of iterations work for all possible values P. In practical applications, the desired probability P is small, so $1 - P \approx 1$, $\sigma \approx \sqrt{P/N}$ and the number of iterations, as determined by the condition $3\sigma \le 0.1$ or $3\sigma \le 0.01$, is much smaller: $N \ge 900 \cdot P$ for accuracy 10% and $N \ge 90,000 \cdot P$ for accuracy 1%.

Comment. In many cases, there are also efficient analytical algorithms for computing the desired probability of the system's failure; see, e.g., [5].

3 Important Subcase of the Simplest Case: When Components are Very Reliable

In many practical applications (e.g., in important subsystems related to aircrafts), components are highly reliable, and their probabilities of failure $P(A)$ are very small. In this case, the above Monte-Carlo technique for computing the probability P of the system's failure requires a large number of simulations, because otherwise, with high probability, in all simulations, all the components will be simulated as working properly.

For example, if the probability of a component's failure is $P(A) = 10^{-3}$, then we need at least a thousand iteration to catch a case when this component fails; if $P(A) = 10^{-6}$, we need at least a million iterations, etc.

In such situations, Monte-Carlo simulations may take a lot of computation time. In some applications, e.g., on the stage of an aircraft design, it may be OK, but in other case, e.g., on the stage of routine aircraft maintenance, the airlines want fast turnaround, and any speed up is highly welcome.

To speed up such simulations, we can use a re-scaling idea; see, e.g., [7]. Specifically, instead of using the original values $P(A)$, we use re-scaled (larger) values $\lambda \cdot P(A)$ for some $\lambda \gg 1$. The value λ is chosen in such a way that the resulting probabilities are larger and thus, require fewer simulations to come up with cases when some components fail. As a result of applying the above Monte-Carlo simulations to these new probabilities $\lambda \cdot P(A)$, we get a probability of failure $P(\lambda)$.

In this case, one can show that while the resulting probabilities $\lambda \cdot P(A)$ are still small, the probability $P(\lambda)$ depends on λ as $P(\lambda) \approx \lambda^k \cdot P$ for some positive integer k.

Thus, to find the desired value P, we repeat this procedure for two different values $\lambda_1 \neq \lambda_2$, get the two values $P(\lambda_1)$ and $P(\lambda_2)$, and then find both unknown k and P from the resulting system of two equations with two unknowns: $P(\lambda_1) \approx \lambda_1^k \cdot P$ and $P(\lambda_2) \approx \lambda_2^k \cdot P$.

To solve this system, we first divide the first equation by the second one, getting an equation $P(\lambda_1)/P(\lambda_2) \approx (\lambda_1/\lambda_2)^k$ with one unknown k, and find $k \approx \ln(P(\lambda_1)/P(\lambda_2))/(\lambda_1/\lambda_2)$. Then, once we know k, we can find P as $P \approx P(\lambda_1)/\lambda_1^k$.

4 Cases When We Know the Probabilities $P(A)$ with Uncertainty: Fuzzy Uncertainty Can Be Reduced to Interval One

In many cases, we do not know the exact probabilities $P(A)$ of the component's failure, we only know the intervals $\mathbf{P}(A)$ that contain these probabilities, or, even more generally, a fuzzy number $\mathcal{P}(A)$ that describes this probability.

In the case of intervals, different combinations of values $P(A) \in \mathbf{P}(A)$ lead, in general, to different values P. Let us denote the dependence of P on the values $P(A)$ by f: $P = f(P(A), P(B), \ldots)$. In this case, we want to know the range of possible values of the desired probability P:

$$\mathbf{P} = [\underline{P}, \overline{P}] = f(\mathbf{P}(A), \mathbf{P}(B), \ldots) \stackrel{\text{def}}{=}$$

$$\{f(P(A), P(B), \ldots) : P(A) \in \mathbf{P}(A), P(B) \in \mathbf{P}(B), \ldots\}.$$

The problem of computing such an interval is a particular case of the general problem of interval computations, i.e., a problem of computing the range of a given function $f(x_1, \ldots, x_n)$ when each of the variables x_i takes value from a given interval $\mathbf{x}_i$; see, e.g., [6, 8, 12] and references therein.

When each probability is described by a fuzzy number $\mathcal{P}(A)$, i.e., by a membership function $\mu_A(P)$ that assigns, to every real number P, a degree to which this number is possible as a value of $P(A)$, we want to find the fuzzy number

$\mathcal{P}$ that describes $f(P(A), P(B), \ldots)$. A natural way to define the corresponding membership function $\mu(P)$ leads to Zadeh's extension principle:

$$\mu(P) = \sup\{\min(\mu_A(P(A)), \mu_B(P(B)), \ldots) : f(P(A), P(B), \ldots) = P\}.$$

It is known that from the computational viewpoint, the application of this formula can be reduced to interval computations.

Specifically, for each fuzzy set with a membership function $\mu(x)$ and for each $\alpha \in (0, 1]$, we can define this set's α-*cut* as $\mathcal{X}(\alpha) \stackrel{\text{def}}{=} \{x : \mu(x) \geq \alpha\}$. Vice versa, if we know the α-cuts for all α, we, for each x, can reconstruct the value $\mu(x)$ as the largest value α for which $x \in \mathcal{X}(\alpha)$. Thus, to describe a fuzzy number, it is sufficient to find all its α-cuts.

It is known that when the inputs $\mu_i(x_i)$ are fuzzy numbers, and the function $y = f(x_1, \ldots, x_n)$ is continuous, then for each α, the α-cut $\mathcal{Y}(\alpha)$ of y is equal to the range of possible values of $f(x_1, \ldots, x_n)$ when $x_i \in \S_i(\alpha)$ for all i:

$$\mathcal{Y}(\alpha) = f(\mathcal{X}_1(\alpha), \ldots, \mathcal{X}_n(\alpha));$$

see, e.g., [4, 9, 13, 14]. This is how processing fuzzy data is often done – by reducing to interval computations.

$$\mathbf{P} = [\underline{P}, \overline{P}] = f(\mathbf{P}(A), \mathbf{P}(B), \ldots).$$

In particular, for our problem, once know the α-cuts $\mathcal{P}(A, \alpha)$ corresponding to different components A, we can find the α-cuts $\mathcal{P}(\alpha)$ corresponding to the desired probability P as the corresponding interval range:

$$\mathcal{P}(\alpha) = f(\mathcal{P}(A, \alpha), \mathcal{P}(B, \alpha), \ldots).$$

So, if we know how to solve our problem under interval uncertainty, we can also solve it under fuzzy uncertainty – e.g., by repeating the above interval computations for $\alpha = 0, 0.1, \ldots, 0.9, 1.0$.

In view of this reduction, in the following text, we will only consider the case of interval uncertainty.

5 Component Failures are Independent, Failure Probabilities $P(A)$ Are Known with Interval Uncertainty: Monotonicity Case

In view of the above analysis, let us now consider the case when the probabilities $P(A)$ are only known with interval uncertainty, i.e., for each component A, we only know the interval $[\underline{P}(A), \overline{P}(A)]$ that contains the actual (unknown) value $P(A)$. We still assume that failures of different components are independent events.

Let us start with the simplest subcase when the dependence of the system's failure on the failure of components is monotonic. To be precise, we assume that if for a certain list of failed components, the system fails, it will still fail if we add one more components to the list of failed ones. In this case, the smaller the probability of failure $P(A)$ for each component A, the smaller the probability P that the system as a whole will fail. Similarly, the larger the probability of failure $P(A)$ for each component A, the larger the probability P that the system as a whole will fail.

Thus, to compute the smallest possible value $\underline{P}$ of the failure probability, it is sufficient to consider the values $\underline{P}(A)$. Similarly, to compute the largest possible value $\overline{P}$ of the failure probability, it is sufficient to consider the values $\overline{P}(A)$. Thus, in the monotonic case, to compute the range $[\underline{P}, \overline{P}]$ of possible values of overall failure probability under interval uncertainty, it is sufficient to solve two problems in each of which we know probabilities with certainty:

- to compute $\underline{P}$, we assume that for each component A, the failure probability is equal to $\underline{P}(A)$;
- to compute $\overline{P}$, we assume that for each component A, the failure probability is equal to $\overline{P}(A)$.

6 In Practice, the Dependence is Sometimes Non-Monotonic

Let us show that in some practically reasonable situations, the dependence of the system's failure on the failure of components is non-monotonic. This may sound counter-intuitive at first glance: adding one more failing component to the list of failed ones suddenly makes the previously failing system recover, but here is an example when exactly this seemingly counter-intuitive behavior makes perfect sense.

To increase reliability, systems include duplication: for many important functions, there is a duplicate subsystem ready to take charge if the main subsystem fails. How do we detect that the main system failed? Usually, a subsystem contains several sensors; sensors sometimes fail, as a result of which their signal no longer reflect the actual value of the quantity they are supposed to measure. For example, a temperature sensor which is supposed to generate a signal proportional to the temperature, if failed, produces no signal at all, which the system will naturally interpret as a 0 temperature. To detect the sensor failure, subsystems often use statistical criteria. For example, for each sensor i, we usually know the mean m_i and the standard deviation σ_i of the corresponding quantity. When these quantities are independent and approximately normally distributed, then, for the measurement values x_i, the sum $X^2 \stackrel{\text{def}}{=} \sum\limits_{i=1}^{n} \dfrac{(x_i - m_i)^2}{\sigma_i^2}$ is the sum of n standard normal distributions and thus, follows the chi-square distributed with n degrees of freedom. So, if the actual value of this sum exceeds the threshold corresponding to confidence level $p = 0.05$, this means that we can confidently conclude that some of the sensors are malfunctioning.

If the number n of sensors is large, then one malfunctioning sensor may not increase the sum X^2 too high, and so, its malfunctioning will not be detected, and the system will fail. On the other hand, if all n sensors fail, e.g., show 0 instead of the correct temperature, each term in the sum will be large, the sum will exceed the threshold – and the system will detect the malfunctioning. In this case, the second redundant subsystem will be activated, and the system as a whole will thus continue to function normally.

This is exactly the case of non-monotonicity: when only one sensor fails, the system as a whole fails; however, if, in addition to the originally failed sensor, many other sensors fail, the system as a whole becomes functioning well.

7 What If We Have Few Components, With Respect to Which the Dependence is Non-Monotonic

Let us start with the simplest case when there are only few components with respect to which the dependence is non-monotonic, and with respect to all other components, the dependence *is* still monotonic. In this case, for all monotonic components B, as before, we can take $P(B) = \overline{P}(B)$ when we are computing $\overline{P}$, and take $P(B) = \underline{P}(B)$ when we are computing $\underline{P}$.

For non-monotonic components A, for computing each of the values $\underline{P}$ and $\overline{P}$, we need to to take into account all possible values $P(A) \in [\underline{P}(A), \overline{P}(A)]$. For each such component, by using the formula of full probability, we can represent the probability P of the system's failure as follows:

$$P = P(A) \cdot P(F|A) + (1 - P(A)) \cdot P(F|\neg A),$$

where $P(F|A)$ is the conditional probability that the system fails under the condition that the component A fails, and $P(F|\neg A)$ is the conditional probability that the system fails under the condition that the component A does not fail. The conditional probabilities $P(F|A)$ and $P(F|\neg A)$ do not depend on $P(A)$, so the resulting dependence of P on $P(A)$ is linear. A linear function attains it minimum and maximum at the endpoints. Thus, to find $\underline{P}$ and $\overline{P}$, it is not necessary to consider all possible values $P(A) \in [\underline{P}(A), \overline{P}(A)]$, it is sufficient to only consider two values: $P(A) = \underline{P}(A)$ and $P(A) = \overline{P}(A)$.

For each of these two values, for another non-monotonic component A', we have two possible options $P(A') = \underline{P}(A')$ and $P(A') = \overline{P}(A')$; thus, in this case, we need to consider $2 \times 2 = 4$ possible combinations of values $P(A)$ and $P(A')$.

In general, when we have k non-monotonic components $A_1, \ldots, A_k$, it is sufficient to consider 2^k possible combinations of values $\underline{P}(A_i)$ and $\overline{P}(A_i)$ corresponding to each of these components. When k is small, this is doable – and, as our preliminary experiments show, works very well.

This procedure requires times which grows as 2^k. As we mentioned earlier, when k is large, the needed computation time becomes unrealistically large.

8 General Case: the Problem is NP-Hard, and Even Checking Whether a Component Is Monotonic Is NP-hard

Natural question. The fact that the above algorithm requires unrealistic exponential time raises a natural question: is it because our algorithm is inefficient or is it because the problem itself is difficult?

The problem is NP-hard. In the general case, when no assumption is made about monotonicity, the problem is as follows:

- we have a propositional formula F with n variables A_i – each variable A_i is true if the corresponding component fails, and the formula F is true if the system as whole fails;
- for each component i, we know the interval $[\underline{P}(A_i), \overline{P}(A_i)]$ that contains the actual (unknown) $P(A_i)$ that this component fails;
- we assume that the failures of different components are independent events.

Different values $P(A_i) \in [\underline{P}(A_i), \overline{P}(A_i)]$ lead, in general, to different values of the probability P that F is true (i.e., that the system failed). Our objective is to compute the range $[\underline{P}, \overline{P}]$ of possible values of this probability.

One can easily show that, in general, this problem is NP-hard (for precise definitions, see, e.g., [15]). Indeed, it is well known that the following *propositional satisfiability* problem SAT is NP-hard: given a propositional formula F, check whether this formula is satisfiable, i.e., whether there exist values A_i that make it true. We will prove that our problem is NP-hard by reducing SAT to it: for every particular case F of SAT, there is a particular case of our problem whose solution leads to a solution to the F. Indeed, for every formula F, let us take $[\underline{P}(A_i), \overline{P}(A_i)] = [0, 1]$ for all variables i.

If the formula F is not satisfiable, then F is always false, so the probability of its being true is 0. In this case, the range of possible values of P consists of a single value 0: $[\underline{P}, \overline{P}] = [0, 0]$.

On the other hand, if F is satisfiable, e.g., if F is true for A_1 true, A_2 false, etc., then we can take $P(A_1) = 1$, $P(A_2) = 0$, etc., and conclude that under these probabilities, F is always true, i.e., $P = 1$. Thus, in this case, $\overline{P} = 1$.

So, by computing $\overline{P}$, we will get either $\overline{P} = 0$ or $\overline{P} = 1$. In the first case, F is satisfiable, in the second case, it is not. The reduction proves that our problem is NP-hard.

Even checking monotonicity is NP-hard. Let us show that even checking monotonicity is NP-hard. Intuitively, monotonicity with respect to a component A means that if the system was in a failing state, and we change the state of A to failing, the system remains failing. In precise terms, monotonicity means that if the formula F was true, and we change the value of the variable A from false to true, then F remains true.

Let us prove that the problem of checking monotonicity is NP-hard by reducing SAT to this problem. Indeed, for every propositional formula $F(A_1, \ldots, A_n)$, we can form a new formula $F' \stackrel{\text{def}}{=} F(A_1, \ldots, A_n) \,\&\neg A_0$.

When the original formula F is not satisfiable, then F is always false and thus, the new formula F' is also always false. In this case, F' is, by definition, monotonic with respect to A_0.

When F is satisfiable, this means that F is equal to "true" for some values of A_1, ..., A_n. In this case, for $A_0 =$"false", the new formula F' is true, but if we make $A_0 =$"true", F' becomes false. Thus, in this case, F' is not monotonic with respect to A_0.

Summarizing: the new formula F' is monotonic with respect to A_0 if and only if the original formula F was satisfiable. This reduction proves that checking monotonicity is indeed NP-hard.

9 Case of Narrow Intervals: Cauchy Deviate Method

In many practical situations, intervals are narrow. In this case, we can use an efficient Cauchy deviates method to find the range of the resulting probability P; see, e.g., [10].

10 What If We Do Not Assume Independence

Exact methods. If we do not assume independence, then, in principle, we can have different probabilities P of failure even when the failure probabilities $P(A_i)$ of all components are known exactly. For example, if the system consists of two components and it fails if both components fail, i.e., if $F = A_1 \,\&\, A_2$, then possible values of P take an interval

$$[\underline{P}, \overline{P}] = [\max(P(A_1) + P(A_2) - 1, 0), \min(P(A_1), P(A_2)].$$

In general, to describe probabilities of all possible combinations of statements A_i, it is sufficient to describe 2^n probabilities of *atomic* statements $A_1^{\varepsilon_1} \,\&\, \ldots \,\&\, A_n^{\varepsilon_n}$, where $\varepsilon_i \in \{-, +\}$, A^+ means A, and A^- means $\neg A$. These probabilities satisfy the condition that their sum is 1; each given probability $P(A_i)$ and the desired probability P can be described as a sum of some such probabilities, so the problem of finding the range of P becomes the particular case of *linear programming* problems, when we need to find the minimum and maximum of a linear function under linear constraints; see, e.g., [16].

For example, in the above case $n = 2$, we have four non-negative unknowns $P_{++} = P(A_1 \,\&\, A_2)$, $P_{+-} = P(A_1 \,\&\, \neg A_2)$, $P_{-+} = P(\neg A_1 \,\&\, A_2)$, and $P_{--} = P(A_1 \,\&\, A_2)$ that satisfy the constraints $P_{++} + P_{+-} + P_{-+} + P_{--} = 1$, $P_{++} + P_{+-} = P(A_1)$, and $P_{++} + P_{-+} = P(A_2)$. Here, $P = P_{++}$, so, e.g., to find $\underline{P}$, we need to minimize P_{++} under these constraints.

Limitations of the exact methods. This works well if n is small, but for large n, this method requires an unrealistically long time.

Heuristic approximate methods. In principle, we can use technique similar to straightforward interval computations. Indeed, for simple formulas F like $\neg A_1$, $A_1 \vee A_2$, or $A_1 \,\&\, A_2$, we have explicit formulas for the range of the probability $P(F)$. So, to estimate the range of the probability P for an arbitrary formula F, we can do the following:

- we *parse* the expression F, i.e., represent is as a sequence of simple boolean operations – the same sequence that a computer computing F would follow, and
- replace each computation step with corresponding operations with probability ranges.

In this algorithm, at each moment of time, we keep the bounds on the probabilities $P(A_i)$ and on the probabilities $P(F_j)$ of the corresponding intermediate formulas.

The problem with this approach is that at each step, we ignore the dependence between the intermediate results F_j; hence intervals grow too wide. For example, the estimate for $P(A \vee \neg A)$ computed this way is not 1, but an interval containing the correct value 1.

A more accurate algorithm was proposed in [1–3]. In this algorithm, at each stage, besides the bounds on the values $P(F_j)$ (including the original bounds on $P(A_i)$, or – if available – exact values of $P(A_i)$), we also compute the bounds for the probabilities $P(F_j \,\&\, F_k)$, $P(F_j \,\&\, \neg F_k)$, $P(\neg F_j \,\&\, F_k)$, and $P(\neg F_j \,\&\, \neg F_k)$.

On each computation step, when we add a new intermediate result F_a (e.g., $F_a = F_b \,\&\, F_c$), we add bounds for the probabilities of the new statement F_a and of all possible combinations that include F_a, i.e., on the probabilities $P(F_a \,\&\, F_k)$, $P(F_a \,\&\, \neg F_k)$, $P(\neg F_a \,\&\, F_k)$, and $P(\neg F_a \,\&\, \neg F_k)$. To compute each of these probabilities, we use the known bounds on the probabilities of combinations of F_b and F_k, F_c and F_k, F_b and F_c, and get the desired bounds on the combinations of F_a and F_k by solving the corresponding linear programming problem. In this linear programming problem, we consider, as variables, $2^3 = 8$ probabilities of atomic statements $F_b^{\varepsilon_b} \,\&\, F_c^{\varepsilon_c} \,\&\, F_k^{\varepsilon_k}$.

At the end of the process, we get an interval $\widetilde{P}$. Similarly to interval computations, we can prove, by induction, that every possible value P of the system's failure is contained in this interval, i.e., that the interval $\widetilde{P}$ is an enclosure for the desired range $[\underline{P}, \overline{P}]$: $[\underline{P}, \overline{P}] \subseteq \widetilde{P}$.

This algorithm requires more computation time that the above straightforward algorithm – since for s intermediate steps, we now need s^2 estimations instead of s – but as a result, we get more accurate accurate, with smaller "excess width" for the resulting enclosure. For example, the probability $P(A \vee \neg A)$ is estimated as 1.

To get an even better accuracy, instead of probabilities of all possible combinations of two intermediate results, we can compute, on each step, probabilities of all possible combinations of three such results: F_i, F_j, and F_k. In this case, computation time grows as s^3 but the resulting enclosure is even more accurate. Similarly, we can have combinations of fours, fives, etc.

Acknowledgments. C. Jacob was supported by a grant from @MOST Prototype, a join project of Airbus, Institut de Recherche en Informatique de Toulouse (IRIT), Laboratoire dAnalyse et dArchitecture des Systémes (LAAS), ONERA, and Institut Supérieur de l'Aéronautique et de l'Espace (ISAE). V. Kreinovich was supported by the Nat'l Science Foundation grants HRD-0734825 and DUE-0926721 and by Grant 1 T36 GM078000-01 from the Nat'l Institutes of Health.

The authors are thankful to the anonymous referees for valuable suggestions.

References

1. Ceberio, M., Kreinovich, V., Chopra, S., Longpré, L., Nguyen, H. T., Ludaescher, B., and Baral, C.: Interval-type and affine arithmetic-type techniques for handling uncertainty in expert systems, Journal of Computational and Applied Mathematics, 199(2), 403–410 (2007)
2. Chopra, S.: Affine arithmetic-type techniques for handling uncertainty in expert systems, Master's thesis, Department of Computer Science, University of Texas at El Paso, 2005.
3. Chopra, S.: Affine arithmetic-type techniques for handling uncertainty in expert systems, International Journal of Intelligent Technologies and Applied Statistics, 1(1), 59–110 (2008)
4. Dubois, D., Prade, H.: Operations on fuzzy numbers, International Journal of Systems Science, 9, 613–626 (1978)
5. Dutuit, Y., Rauzy, A.: Approximate estimation of system reliability via fault trees, Reliability Engineering and System Safety, 87(2), 163–172 (2005)
6. Interval computations website http://www.cs.utep.edu/interval-comp
7. Jaksurat, P., Freudenthal, E., Ceberio, M., Kreinovich, V.: Probabilistic Approach to Trust: Ideas, Algorithms, and Simulations, Proceedings of the Fifth International Conference on Intelligent Technologies InTech'04, Houston, Texas, December 2–4, 2004 (2004)
8. Jaulin, L., Kieffer, M., Didrit, O., Walter, E.: Applied Interval Analysis, Springer, London (2001)
9. Klir, G., Yuan, B.: Fuzzy Sets and Fuzzy Logic, Prentice Hall, Upper Saddle River, NJ (1995)
10. Kreinovich, V., Ferson, S.: A new Cauchy-based black-box technique for uncertainty in risk analysis, Reliability Engineering and Systems Safety, 85(1–3), 267–279 (2004)
11. Kreinovich, V., et al.: Computational Complexity and Feasibility of Data Processing and Interval Computations, Kluwer, Dordrecht (1997)
12. Moore, R. E., Kearfott, R. B., Cloud, M. J.: Introduction to Interval Analysis, SIAM Press, Philadelphia, Pennsylvania (2009)
13. Nguyen, H. T., Kreinovich, V.: Nested intervals and sets: concepts, relations to fuzzy sets, and applications, In: Kearfott, R. B., Kreinovich, V., eds., Applications of Interval Computations, Kluwer, Dordrecht, 245–290 (1996)
14. Nguyen, H. T., Walker, E. A.: A First Course in Fuzzy Logic, Chapman & Hall/CRC, Boca Raton, Florida (2006)
15. Papadimitriou, C.: Computational Complexity, Addison Welsey, Reading, Massachusetts (1994)
16. Walley, P.: Statistical reasoning with imprecise probabilities. Chapman & Hall: New York (1991)

Stepwise Feature Selection Using Multiple Kernel Learning

Vilen Jumutc

Riga Technical University, Meza 1/4, LV-1658 Riga, Latvia
Jumutc@gmail.com

Abstract. In this paper we propose a novel more flexible approach for the simultaneous feature selection and classification using Support Vector Machine and recent major advances of it, namely Multiple Kernel Learning. Using a quite simple kernel assembly scheme in the following paper we will indicate that feature selection and classification could be done in one step without applying computationally intensive and maybe inadequate filtering or wrapper approach. Later imply that to achieve dimensionality reduction, tractable and more compact as well as comprehensively accurate model it is necessary to accomplish all of above goals by "training" SVM only once. Actually we apply some additional prerequirement that resulted in a ranking criteria that could be provided by any domain expert or created by our algorithm using Linear SVM by itself. Provided experimental results verify that our approach is comparable or even more accurate and robust than other feature extraction/selection schemes tested on public UCI datasets.

1 Introduction

Recent advances in computer science and computational intelligence uncover vital necessity for the feature selection and dimensionality reduction methods applied to the variety of highly-dimensional data sources like biomedical CT images, cardiogram, microarray and other data with high variance and insufficient sample size. By this research we intend to resolve simultaneously several problems of previous feature selection/extraction methods like SVM-RFE [1] that solely depends on Linear SVM and like every wrapper approach evaluates classifier each iteration of feature extraction algorithm. We state that our feature selection scheme is both computationally inexpensive and outperforms resembling approaches that basically implement either forward-selection procedure to ensure crisp feature selection or backward-elimination that potentially could be very time-consuming and suffers from overall non-convexity of stated optimization problem. Embedded MKL extension provides us with strong convexity of feature selection problem and simultaneously helps to build ad-hoc classifier that incorporates only most predictive and discriminative attributes.

The upcoming sections of our paper are structured as follows: Section 2 briefly presents common SVM basics and MKL extension. Section 3 describes in details our feature selection method and presents generalized algorithm. Section

4 summarizes experimental setup and numerical results. And finally in Section 5 we analyze and compare our method with other feature extraction/selection approaches as well as conclude about further possible research area.

2 Background

In this section we present some commonly recognized SVM basics [2] and MKL extension of it [4, 5] for learning from an affine combination of regular (linear, RBF, polynomial etc.) or data-driven kernels.

2.1 Support Vector Machine

Support Vector Machine is based on the concept of separating hyperplanes that define decision boundaries using Statistical Learning Theory [2]. Using a kernel function, SVM is an alternative training method for polynomial, RBF and multilayer perceptron classifiers in which the optimal solution or decision surface is found by solving the quadratic programming problem with linear constraints, rather than by solving a non- convex, unconstrained minimization problem as stated in typical back-propagation neural network.

Further we present only dual representation of SVM primal objective that is expressed in terms of its Lagrangian multipliers λ_i and can be effectively optimized using any off-shell linear optimizer that supports constraint adaptation:

$$\max_{\lambda}\{\sum_{i=1}^{l}\lambda_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\lambda_i\lambda_i y_i y_j K(x_i, x_j)\}, \quad \lambda_i \geq 0, \ \sum_{i=1}^{l}\lambda_i y_i = 0, \tag{1}$$

where λ_i represents a Lagrangian multiplier, y_i is $\{\pm 1\}$ - valued label of data sample x_i, $K(x_i, x_j)$ is a kernel function and l is a number of training samples.

Finally corresponding classification of a new sample x' is derived by: $d = sign(\sum_i \lambda_i K(x_i, x') + b)$, where $K(x_i, x')$ and b correspond to a kernel function evaluated for a new sample and a linear offset of the optimal decision hyperplane.

2.2 Multiple Kernel Learning

Multiple Kernel Learning aims at simultaneously learning the kernel and the associated predictor in general SVM context. Recent applications of MKL have clearly proven that using multiple kernels instead of a single one can enhance the interpretability of the decision function and improve performances [4, 5]. In such cases, a convenient approach is to consider that the kernel $K(x, y)$ is actually a convex combination of basis kernels:

$$K(x, y) = \sum_{i=1}^{m} w_i K_i(x, y), \quad w_i \geq 0, \ \sum_{i=1}^{m} w_i = 1, \tag{2}$$

where m is the total number of kernels. Within this framework, the problem of data representation through the different kernels is then transferred to the choice of optimal weights w_i that minimizes the MKL objective function [5].

3 Proposed method

In this section we describe in details aforementioned feature selection method and a general kernel assembly scheme for Multiple Kernel Learning. The overall approach is given in the form of abstract algorithm that depicts a clear view of all steps needed to implement proposed method.

3.1 Ranking criteria

Before handling actual feature selection procedure we apply some additional ranking criteria that performs an ordering of all features according to their relevant importance to an evaluated classifier. Similar approach was provided by [1] in SVM-RFE method and consists of the following very simple steps:

1. Evaluate Linear Support Vector Machine and compute corresponding weight vector of dimension length: $w = \sum_i \lambda_i y_i x_i$, where λ_i is a dual variable of SVM optimization problem, y_i is a label of *i-th* training sample x_i
2. Compute the ranking $c_j = (w_j)^2$ for every *j-th* attribute
3. Sort the ranked attributes in the descending order and create corresponding ordered list of features S

3.2 Generalized algorithm

After evaluating the ranking criteria and obtaining ordered list of features we perform following kernel assembly scheme that could be effectively summarized by the generalized algorithm that incorporates several subroutines and inner algorithms such that *SimpleMKL* [3], *InitKernelMatrices* etc.:

Algorithm 1: Stepwise feature selection via kernel assembly scheme

```
input : ordered list of features $S$ of size $m$, training data $X$ of size
        $n \times m$, class labels $Y$ of size $n$
output: nonlinear SVM model: $\lambda$ defines a dual SVM solution and $b$
        corresponds to a linear offset, selected feature subsets which
        correspond to not-null elements of weight vector $w$
begin
    $K \leftarrow$ InitKernelMatrices ();
    $I_{RBF} \leftarrow$ InitRBFInterval ();
    for $i \leftarrow 1$ to $m$ do
        $S' \leftarrow S(\overline{1,i})$;
        $X' \leftarrow X(:, S')$;
        for $j \leftarrow 1$ to $|I_{RBF}|$ do
            $ind \leftarrow (i-1) \times |I_{RBF}| + j$;
            $K[ind] \leftarrow$ ComputeRBFKernel ($X'$, $I_{RBF}[j]$);
        end
    end
    $[w, \lambda, b] \leftarrow$ SimpleMKL ($Y, K$);
end
```

Finally classification using defined in Algorithm 1 SVM model could be handled using following equation:

$$d = sign(\sum_i \sum_j \lambda_i w_j K_j(x_i, x') + b), \tag{3}$$

where K_j is the RBF kernel function and x' is a test sample.

It is obvious that represented by Algorithm 1 kernel assembly scheme could be summarized as a stepwise feature subset selection from the ordered list of all attributes. Further algorithmic steps only broaden number of kernel matrices by additional parametrization of RBF kernel.

To implemented our approach we have selected to train and test our method within SimpleMKL framework [3] in order to avoid time-expensive cross-validation and provide more accurate estimation of ”tuning” parameters of RBF kernel. The later parameters are defined by *InitRBFInterval* method of our generalized algorithm and correspond to unknown optimal bandwidth γ of any RBF kernel.

To fasten computation of incredibly many kernel matrices (in Algorithm 1 number of kernel matrices is bounded by $m \times |I_{RBF}|$) we have decided to estimate optimal iteration pace of the outer ”for” loop in our generalized algorithm. In order to lower a computational effort and memory load without significant performance degradation we conducted 10-fold cross-validation on the training set and averaged total error across all folds. The pace with the lowest averaged error was selected for performing Algorithm 1.

4 Experiments

4.1 Experimental Setup

In our experiments we have tested proposed model under predefined $C = 10$ (error trade-off) value of the soft-margin SVM that showed most comprehensible performance for imbalanced data sets and varying γ value of RBF kernel that trade-offs kernel smoothness and could be effectively estimated via SimpleMKL framework [3].

To verify and test our proposed approach we have selected several highly dimensional public UCI datasets and evaluated them under following experimental setup: for datasets that weren’t originally separated into validation and training sets we performed 10-fold cross-validation and collected averaged total error and balanced error rate (BER). For others we tested our approach on presented in UCI repository validation set and collected single total error and BER. Additionally we experiment with highly dimensional gene microarray dataset, namely CNS-ET, that was very comprehensively inspected in [6]. For this dataset we apply Leave-One-Out cross-validation scheme to provide comparable results with [6] where Pomeroy et al. followed the same experimental setup.

4.2 Numerical Results

In the following subsection we have summarized numerical results for all datasets under fixed C parameter and enclosed subspace for γ parameter of RBF kernel with some initial guess of its corresponding scaling factor[1]. In the Table 1 we present performance measures obtained by our approach under SimpleMKL framework, linear/nonlinear SVM benchmark results as well as some additional performance measures for SVM with differently applied filtering or wrapper feature selection approach. In braces we give averaged number of selected features for all presented in Table 1 approaches except linear/nonlinear SVM that was "trained" using all features.

Table 1. Averaged Total Error/BER

Dataset	SVM$_{\text{linear}}$	SVM$_{\text{rbf}}$	Our method	F+SVM[a]	CSA[b]
Arrythmia	0.26/0.26	0.25/25	0.21/0.22(34)	-/-	0.26/-(28)
Arcene	0.17/0.18	0.2/0.22	0.13/0.14(1101)	-/0.21(661)	0.19/-(600)
Dexter	0.07/0.07	0.11/0.11	0.08/0.08(118)	-/0.08(209)	0.07/-(717)
CNS-ET	0.33/0.4	0.35/0.5	0.2/0.24(132)	-/-	-/-

[a] SVM with the F-score feature selection scheme [7]

[b] Contribution-Selection Algorithm with the best performing inducted classifier [8]

5 Results Analysis and Conclusion

In this paper we propose novel stepwise feature selection method that basically extends Multiple Kernel Learning approach and helps to provide classifier with the most comprehensible and meaningful subset of features and perform actual classification all in one step. As we can see from the above given experimental results our feature selection method is comparable or even more accurate and robust than other feature selection/extraction approaches. Remarkably that our approach almost anywhere attains comparable or even smaller subset of features. For UCI datasets it is clear that our stepwise feature selection algorithm brings necessary discrimination capabilities and additional accuracy to the nonlinear SVM classifier eliminating noisy and redundant features. Separately we should examine CNS-ET dataset because we do not provide performance measures for F-SVM and CSA methods. In original work of Pomeroy et al. [6] authors preselected 150 most discriminative genes and conducted SVM classifier. They

[1] We have defined range of $b_\gamma \cdot 10^{[-20\ldots20]}$ with the step 0.5 resulting in a total of 81 γ-parameters where b_γ is a corresponding scaling factor of γ stated as follows: $b_\gamma = 1/2 \cdot \sqrt{median(X)}$ where X is a vector of all dataset values.

achieved total error of 25% and balanced error rate (BER) of 29.1%. In comparison we achieve drastically more robust and accurate result without even knowing the domain field. In [6] the best possible result was achieved using the combination of three classifiers (SVM, k-NN and TrkC) and was comparable to our results: total error of 20% and BER of 24.2%. In conclusion we should highlight that our approach is a general purpose algorithm and could be used for any classification problem that suffers from ”dimensionality curse” and needs a quick and elegant feature extraction approach for the kernel-based classifiers. Our further research is closely related to the feature ranking algorithms that could definitely provide more reliable and domain-specific information about feature relevance to the problem than ordinary Linear Support Vector Machine.

References

1. Guyon, I. et al.: Gene Selection for Cancer Classification using Support Vector Machines. Machine Learning, Vol.46, Issue 1-3, 389 – 422 (2002)
2. Vapnik, V.: The Nature of Statistical Learning Theory. Springer-Verlag (1995)
3. Rakotomamonjy, A. et al.: SimpleMKL. Journal of Machine Learning Research, Vol.9, 2491–2521 (2008)
4. Lanckriet, G. et al.: Learning the Kernel Matrix with Semidefnite Programming. Journal of Machine Learning Research, Vol.5, 27–72 (2004)
5. Bach, F. et al.: Multiple kernel learning, conic duality, and the SMO algorithm. In Proceedings of the 21st International Conference on Machine Learning. Montreal, Canada, 41–48 (2004)
6. Pomeroy, S. et al.: Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature, Vol.415, 436–442 (2002)
7. Chen, Y-W. and Lin C-J.: Combining SVMs with Various Feature Selection Strategies. Studies in Fuzziness and Soft Computing, Vol.207, 315–324 (2006)
8. Cohen, S. et al.: Feature Selection Based on the Shapley Value. In Proceedings of the 19th International Joint Conference on Artificial Intelligence. Edinburgh, Scotland, UK, 665–670(2005)

Empirical reconstruction of fuzzy model of experiment in the Euclidean metric

Tatiana Kopit and Alexey Chulichkov

Department of Computer Methods of Physics, Faculty of Physics, Moscow State University, Moscow, 119991 Russia
kopit_tanya@mail.ru, achulichkov@gmail.com

Abstract. In this paper we introduce a method for the fuzzy model reconstruction and a method for measurements reduction on the basis of test signals by maximization a posteriori possibility. It ensures the maximum accuracy of the measurements reduction. It is used the model of measurement errors with fuzzy constraints on its Euclidean norms.

Keywords: mathematical modeling, fuzzy sets, decision making, analysis and interpretation of data, measurement and computing systems

Introduction. In this paper we consider a fuzzy experiment conducted by the scheme

$$\xi = \mathit{\Lambda}\varphi + \nu. \tag{1}$$

The measurement result ξ of the Euclidean space $\mathcal{R}_n$ is accompanied by a additive noise ν. By a result of ξ of measure (1) it is required to estimate the value of the parameter vector $\eta = U\varphi$, where $U \in (\mathcal{R}_N \to \mathcal{R}_M)$ is defined linear operator [4].

In the paper [1] it is shown that if (1) vector $\varphi \in \mathcal{R}_N$ and a linear operator $\mathit{\Lambda}$ priori arbitrary, and $\nu \in \mathcal{R}_n$ is the fuzzy vector in $\mathcal{R}_n$ with distribution possibilities $\pi^\nu(\cdot)$ [2,3]. And for the determination of the model of the measuring device $\mathit{\Lambda}$ there are involved the measurement results $\xi_j = \mathit{\Lambda} f_j + \nu_j, \quad j = 1, \ldots, m$, of known test signals, $f_1, \ldots, f_m$, where the error $\nu_1, \ldots, \nu_m \in R_n$ of test measurements are fuzzy elements of $\mathcal{R}_n$ with the given distribution of possibilities. The estimates of the maximum possibility $\widehat{A}$ and $\widehat{f}$ are values of fuzzy elements $\mathit{\Lambda}$ and φ respectively, as a solution of the maximin problem

$$(\widehat{A}, \widehat{f}) = \arg \max_{A,f} \min(\pi^\nu(x - Af), \pi^N(X - AF)), \tag{2}$$

and the estimate $\widehat{u}$ is value of the fuzzy element of η given by $\widehat{u} = U\widehat{f}$. Here, x, A, f, X are implementation of fuzzy elements of $\xi, \mathit{\Lambda}, \varphi, \mathit{\Xi}$, respectively. The same scheme of test measurements in matrix form is given by $\mathit{\Xi} = \mathit{\Lambda} F + N$.

Consider the solution of the problem (2), where the operator $\mathit{\Lambda}$ and element φ are a priori arbitrary, so that $\pi^{\mathit{\Lambda}}(A) = 1$ for every $A \in (\mathcal{R}_N \to \mathcal{R}_n)$ and $\pi^\varphi(f) = 1$ for any $f \in \mathcal{R}_N$, and distribution possibilities of measurement error is given by

$$\pi^\nu(z) = \mu_0(\|z\|^2), \quad z \in \mathcal{R}_n; \quad \pi^N(Z) = \mu_0(\|Z\|_2^2), \quad Z \in (\mathcal{R}_m \to \mathcal{R}_n),$$

where $\mu_0(\cdot) : [0, \infty) \to [0, 1]$ is strictly decreasing function, $\mu_0(0) = 1$, $\lim\limits_{z\to\infty} \mu_0(z) = 0$.

Then the problem (2) leads to the following minimax problem:

$$\min_{A,f} \max(\|x - Af\|^2, \|X - AF\|_2^2). \tag{3}$$

Let us denote $J_1(A, f) = \|x - Af\|^2$, $J_2(A) = \|X - AF\|_2^2$, then $J(A, f) = \max(J_1(A, f), J_2(A))$, and the problem (3) can be rewritten as

$$J(A, f) = \max(J_1(A, f), J_2(A)) \sim \min_{A,f}. \tag{4}$$

Depending on which of the minimum values $J_1(\widehat{A}_0, \widehat{f}(\widehat{A}_0))$ or $J_2(\widehat{A}_0)$, is less it is selected different methods for solving the problem (3), they are considered in [1].

Example 1. Let the unknown operator A is defined by the matrix of size 2×2, $A = \begin{pmatrix} a_{11} & a_{22} \\ a_{21} & a_{22} \end{pmatrix}$, given m test signals, $f_1 = \begin{pmatrix} f_{11} \\ f_{21} \end{pmatrix}, \ldots, f_m = \begin{pmatrix} f_{1m} \\ f_{2m} \end{pmatrix}$, forming columns of the matrix $F \in (\mathcal{R}_m \to \mathcal{R}_2)$, and test results are given as vectors $x_1 = \begin{pmatrix} x_{11} \\ x_{21} \end{pmatrix}, \ldots, x_m = \begin{pmatrix} x_{1m} \\ x_{2m} \end{pmatrix}$, forming columns of the matrix $X \in (\mathcal{R}_m \to \mathcal{R}_2)$. Then the scheme of tests is given by

$$\varXi = AF + N,$$

where the j-th column of N defines the error of j-th test measure, $j = 1, \ldots, m$. For the error matrix N that defined by the distribution of possibilities of its values by the form $\pi^N(X) = \mu_0(\|Z\|_2^2)$, where $\mu_0(\cdot) : \mathcal{R}_+ \to [0, 1]$ is a monotonically decreasing function, $\mu_0(0) = 1$, $\mu_0(+\infty) = 0$.

Let the rank of F is equal to two. The signal f is measured according to the scheme $\xi = Af + \nu$. The result $x \in \mathcal{R}_2$ of this measurement is known. The distribution is $\pi^\nu(z) = \mu_0(\|z\|^2)$ of possibility of fuzzy vector $\nu \in \mathcal{R}_2$ for measurement error Af. It is required to determine the reduction of the vector ξ to the form which would be a measurement of the signal f by the instrument $I \in (\mathcal{R}_2 \to \mathcal{R}_2)$.

Let us write the problem of calculating the reduction as a minimax problem

$$\min_{A,f}(\max \|Af - x\|^2, \|AF - X\|_2^2).$$

Minimum of $J_2(A) = \|A_* F - X\|_2^2$ is achieved on a single matrix $\widehat{A}_0 = XF^-$, since the rank of F is equal to two and therefore holds $(I - FF^-) = 0$. If this matrix is nonsingular, then $J_1(\widehat{A}_0, \widehat{f}(\widehat{A}_0)) = 0 \le J_2(\widehat{A}_0)$ and $(A_*, f_*) = (XF^-, (XF^-)^{-1}x)$, a result of the reduction is $f_* = (XF^-)^{-1}x$.

Example 2. Let the unknown operator A each the number of f associates a two-dimensional vector $Af = \begin{pmatrix} a_1 f \\ a_2 f \end{pmatrix}$, i.e. A is defined by the matrix of size

1×2, $A = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$; is given m of test scalar signals $f_1, \dots, f_m$, forming a matrix $F \in \mathcal{R}_m \to \mathcal{R}_1$, consisting of a single line, and test results are given as vectors $x_1 = \begin{pmatrix} x_{11} \\ x_{21} \end{pmatrix}, \dots, x_m = \begin{pmatrix} x_{1m} \\ x_{2m} \end{pmatrix}$, forming columns of $X \in \mathcal{R}_m \to \mathcal{R}_2$. Scheme and model test and measurement reducible are the same as in previous Example 1.

Let us consider the minimax problem

$$\min_{A,f}(\max \|Af - x\|^2, \|AF - X\|_2^2). \tag{5}$$

In this case, the operator A such that its value space is the dimensional linear subspace of $\mathcal{R}_2$, containing the results of test and reduction measurement. The location of the points x and $x_1, \dots, x_m$ such that inequality $\|(I - \widehat{A}_0\widehat{A}_0^-)x\|^2 \leq \|X(I - F^-F)\|_2^2$ is not satisfied and the point $(\widehat{A}_0, \widehat{f}_0)$ is not the point at which is achieved the minimax in (5). To specify the value space of the operator A from the geometric point view means to specify the line through the origin of coordinates. For any such line value of the vector a, which determines the action operator A with a given space of values, given length vector a along a given line.

To achieve the minimax we have to change the direction of the vector a, specifying the space of values of the operator A. Calculating $J_1(A, f)$ as the square of the distance from x to the line with direction vector a, and $J_2(A)$ are making such disposition of the space values of A, at which the equality $J_1(A, f) = J_2(A)$. The length of the projection of x on a one-dimensional subspace divided by the length of a vector a, yields a reduction of measurement x. This ensures a compromise between being able to test and reduction measurements.

Conclusions. In this paper we consider a method of empirical reconstruction and reduction of measurements for fuzzy model in restrictions on Euclidean norms of signals and the operator of the model. The information about the model is contained in a series of test experiments and reduction measurements. This work was supported by the Russian Foundation for Basic Research (project no. 11-07-00338-a, 09-01-96508 and 09-07-00505-a).

References

1. T.A. Kopit, A.I. Chulichkov, D.M. Ustinin, Empirical reconstruction of fuzzy model of the experiment and the reduction of measurements in the Euclidean metric, Computational Methods and Programming, vol.12, pp. 220–226, 2011.[in Russian].
2. L. A. Zadeh, Fuzzy Sets as a Basis for a Theory of Possibility, Fuzzy Sets Syst., no. 1, pp. 3-28, 1978.
3. Yu. P. Pytev, Possibility as an Alternative of Probability, Fizmatlit, Moscow, 2007. [in Russian].
4. Yu. P. Pytev, Methods of Mathematical Simulation of Measuring-Computing Systems, Fizmatlit, Moscow, 2002. [in Russian].

SVM Based Offline Handwritten Gurmukhi Character Recognition

Munish Kumar[1], M. K. Jindal[2], R. K. Sharma[3]

[1]Assistant Professor, Computer Science Department, GGS College for Women, Chandigarh, INDIA

[2]Associate Professor, Department of Computer Science and Applications, Panjab University Regional Centre, Muktsar, INDIA

[3]Professor, School of Mathematics & Computer Applications, Thapa r University, Patiala, INDIA

munishcse@gmail.com, manishphd@rediffmail.com, rksharma@thapar.edu

Abstract. Support Vector Machines (SVMs) have successfully been used in recognizing printed characters. In the present work, we have used this classification technique to recognize handwritten characters. Recognition of handwritten characters is a difficult task owing to various writing styles of individuals. A scheme for offline handwritten Gurmukhi character recognition based on SVMs is presented in this paper. The system first prepares a skeleton of the character, so that feature information about the character is extracted. Features of a character have been computed based on statistical measures of distribution of points on the bitmap image of character. SVM based approach has been used to classify a character based on the extracted features. In this work, we have taken the samples of offline handwritten Gurmukhi characters from one hundred different writers. The partition strategy for selecting the training and testing patterns has also been experimented in this work. We have used in all 3500 images of Gurmukhi characters for the purpose of training and testing. We have used diagonal and; intersection and open end points feature extraction techniques in order to find the feature sets for a given character. The proposed system achieves a maximum recognition accuracy of 94.29% with 90% training data and 10% testing data using intersection and open end points as features and SVM with polynomial kernel.

Keywords: Handwritten character recognition, Feature extraction, Diagonal features, Intersection and open end points features, SVM.

1. Introduction

Most of the published work on Indian scripts recognition deals with printed documents and very few articles deal with handwritten script problem. This has motivated us to consider the handwritten script recognition for Gurmukhi script. Handwritten Character Recognition, usually abbreviated as HCR, is the process of converting handwritten text into machine processable format. HCR is the field of research in pattern recognition and artificial intelligence. HCR can be online or offline. In online handwriting recognition, data are captured during the writing process with the help of a special pen and an electronic surface. Offline documents are scanned images of prewritten text, generally on a sheet of paper. Offline handwriting recognition is significantly different from online handwriting recognition, because here, stroke information is not available [1, 2]. In this work, we have proposed a recognition system for offline handwritten Gurmukhi characters. A recognition system consists of the activities, namely, digitization, preprocessing, features extraction and classification. These activities in such a system have a close proximity with printed characters recognition system. A good number of researchers have already worked on the recognition problem of offline printed characters. For example, a printed Gurmukhi script recognition system has been proposed by Lehal and Singh [3]. Wen *et al.* [4] have proposed handwritten Bangla numerals recognition system for automatic letter sorting machine. Swethalakshmi *et al.* [5] have proposed handwritten Devanagri and Telugu character recognition system using SVM. The input to their recognition system consists of features of the stroke information in each character and SVM based stroke information module has been considered for generalization capability. Pal *et al.* [6, 7] have presented a technique for off-line Bangla handwritten compound characters recognition. They have used modified quadratic discriminant function for feature extraction. Pal *et al.* [8] have also used curvature feature for recognizing Oriya characters. Hanmandlu *et al.* [9] have reported grid based features for handwritten Hindi numerals. They have divided the input image into 24 zones. After that, they have computed the vector distance for each pixel position in the grid from the bottom left corner and normalized these distances to [0, 1] in order to obtain the features.

2. Gurmukhi script and data collection

Gurmukhi script is the script used for writing Punjabi language and is derived from the old Punjabi term "Guramukhi", which means "from the mouth of the Guru". Gurmukhi script has three vowel bearers, thirty two consonants, six additional consonants, nine vowel modifiers, three auxiliary signs and three half characters. Gurmukhi script is 12th most widely used script in the world. Writing style of Gurmukhi script is from top to bottom and left to right. In Gurmukhi script, there is no case sensitivity. The character set of Gurmukhi script is given in Figure 1. In Gurmukhi script, most of the characters have a horizontal line at the upper part called headline and characters are connected with each other through this line.

The Consonants

ਸ ਹ ਕ ਖ ਗ ਘ ਙ ਚ ਛ ਜ ਝ ਞ ਟ ਠ ਡ ਢ ਣ ਤ ਥ ਦ ਧ ਨ ਪ ਫ ਬ ਭ ਮ ਯ ਰ ਲ

The Vowel Bearers

ਵ ੜ

The Additional Consonants (Multi Component Characters)

ੳ ਅ ੲ

The Vowel Modifiers

ਸ਼ ਜ਼ ਖ਼ ਫ਼ ਗ਼ ◌ਲ਼

Auxiliary Signs

◌ੋ ◌ੌ ◌ੇ ◌ੈ ਿ◌ ◌ੀ ◌ਾ ◌ੁ ◌ੂ

The Half Characters

◌ੱ ◌ੰ ◌ਂ

◌੍ਹ ◌੍ਰ ◌੍ਵ

Figure 1: Gurmukhi script character set.

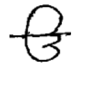
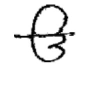
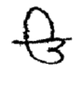
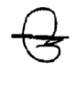
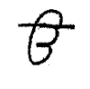
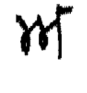
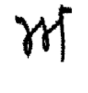
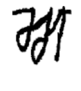
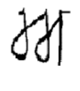
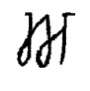
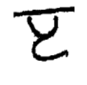
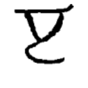
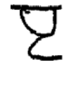
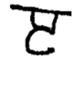
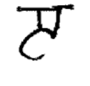
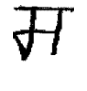
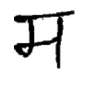
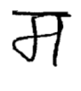
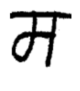
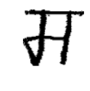
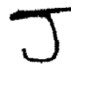
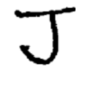
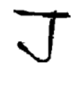
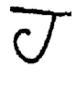
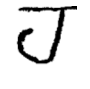
Script Character	W1	W2	W3	W4	W5
ੳ					
ਅ					
ੲ					
ਸ					
ਹ					

Figure 2: Samples of handwritten Gurmukhi characters.

For this work, a sample of 100 writers was selected from schools, colleges, government offices and other places. These writers were requested to write each Gurmukhi character. A sample of five handwritten Gurmukhi characters by five different writers (W1, W2, …, W5) is given in Figure 2.

3. The proposed recognition system

The proposed recognition system consists of the phases, namely, digitization, preprocessing, feature extraction and classification. The block diagram of proposed recognition system is given in Figure 3.

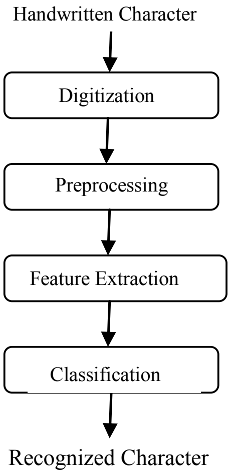


Figure 3: Block diagram of handwritten character recognition system.

3.1 Digitization

Digitization is the process of converting the paper based handwritten document into electronic form. The electronic conversion is accomplished using a process whereby a document is scanned and an electronic representation of the original document, in the form of a bitmap image, is produced. Digitization produces the digital image, which is fed to the pre-processing phase.

3.2 Preprocessing

Preprocessing is a series of operations performed on the digital image. Preprocessing is the initial stage of character recognition. In this phase, the character image is normalized into a window of size 100×100. After normalization, we produce bitmap image of normalized image. Now, the bitmap image is transformed into a contour image.

3.3 Feature extraction

The feature extraction stage analyzes a handwritten character image and selects a set of features that can be used for uniquely classifying the character. In this phase, the features of input characters are extracted. The performance of recognition system greatly depends on features that are being extracted. The extracted features should be able to classify each character uniquely. We have used diagonal and intersection and open end points features for recognition of offline handwritten Gurmukhi characters.

3.3.1 Diagonal feature extraction

Diagonal features are playing an important role in order to achieve higher accuracy of the recognition system. Here, the skeletonized image of a character is divided into *n* (=100) zones. Now, diagonal features are extracted from the pixels of each zone by moving along its diagonals as shown in Figure 4. The steps that have been used to extract these features are given below.

Step I: Divide the skeletonized image into *n* (=100) number of zones, each of size 10×10 pixels.
Step II: Each zone has 19 diagonals; foreground pixels present along each diagonal is summed up in order to get a single sub-feature.
Step III: These 19 sub-feature values are averaged to form a single value and placed in corresponding zone as its feature.
Step IV: Corresponding to the zones whose diagonals do not have a foreground pixel, the feature value is taken as zero.

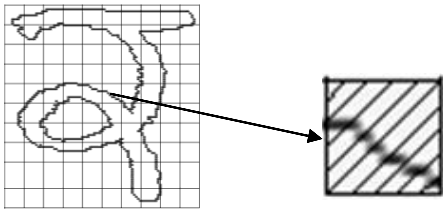


Figure 4: Diagonal feature extraction.

These steps will give a feature set with *n* elements.

3.3.2 Intersection and open end points feature extraction

We have also extracted intersection and open end points for a character. An intersection point is the pixel that has more than one pixel in its neighborhood and an open end point is the pixel that has only one pixel in its neighborhood. Following steps have been implemented for extracting these features.

Step I: Divide the skeletonized image of a character into *n* (=100) zones, each of size 10×10 pixels (Figure 5).
Step II: Calculate number of intersection and open end points for each zone.

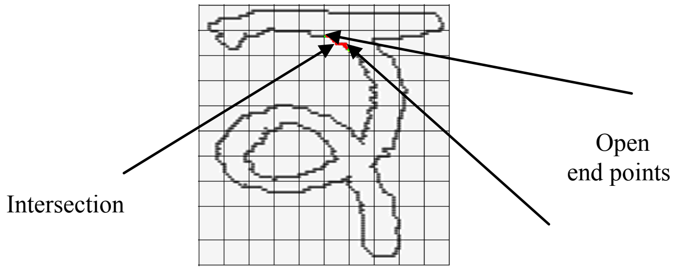


Figure 5: Intersection and open end point feature extraction.

This will give us 2*n* features for a character image.

3.4 Classification

Classification phase is the decision making phase of an HCR engine. This phase uses the features extracted in the previous stage for deciding the class membership. In this work, we have used Support Vector Machine (SVM) classifier for recognition. The SVM is a very useful technique for data classification. The SVM is a learning machine, which has been widely applied in pattern recognition. SVMs are based on statistical learning theory that uses supervised learning. In supervised learning, a machine is trained instead of programmed to perform a given task on a number of inputs/outputs pairs.

4. Experimental results and discussion

In this section, the results of recognition system for offline handwritten Gurmukhi characters are presented. The results are based on two feature extraction techniques, namely, diagonal and; intersection and open end point features. As stated earlier, we have also experimented some partitioning strategies while using the SVM as a classifier. We have divided the data set using five partitioning strategies. In the first strategy (strategy *a*), we have taken 50% data in training set and other 50% data in the testing set. In the second strategy (strategy *b*), we have considered 60% data in training set and remaining 40% data in the testing set. Strategy *c* has 70% data in training set and 30% data in testing set. Similarly, strategy *d* has 80% data in training set and 20% in testing set. Strategy *e* is formulated by taking 90% data in training set and remaining 10% data in testing set. SVM classifier has also been considered with three different kernels, namely, linear kernel, polynomial kernel and RBF kernel.

Feature-wise experimental results of testing are presented in the following sub-sections.

4.1 Diagonal features

In this section, the diagonal features have been considered to be taken as input to three types of SVM classifier, namely, SVM with linear kernel, SVM with polynomial kernel and SVM with RBF kernel.

4.1.1 Recognition accuracy using SVM with linear kernel

In this sub-section, we have presented recognition results of five partitioning strategies (*a*, *b*, *c*, *d* and *e*) based on the diagonal features using SVM with linear kernel. Using this approach, *i.e*., diagonal features and SVM with linear kernel, we achieved an accuracy of 81.83% when we use strategy *a* and achieved an accuracy of 90.29% when we used the strategy *e*. These results are depicted in Figure 6.

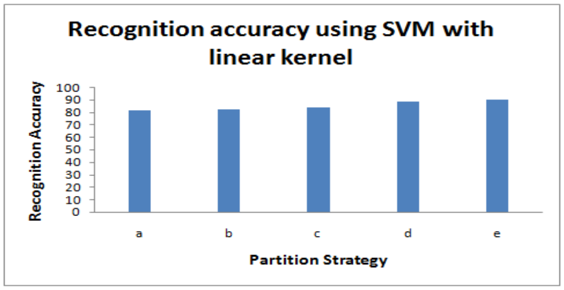


Figure 6: Recognition accuracy using SVM with linear kernel.

4.1.2 Recognition accuracy using SVM with polynomial kernel

When we use SVM with polynomial kernel, the results are not that encouraging. In partitioning strategy *a*, the accuracy that could be achieved was minimum at 43.6% and in strategy *e*, the accuracy achieved was maximum at 60.29%. These results are given in Figure 7.

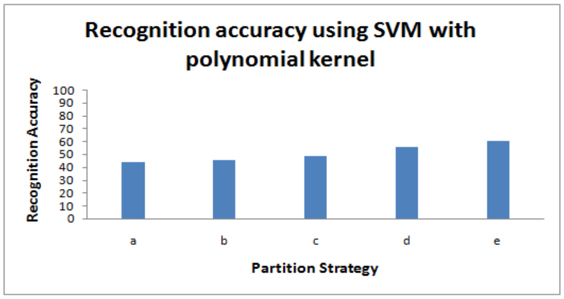


Figure 7: Recognition accuracy using SVM with polynomial kernel.

4.1.3 Recognition accuracy using SVM with RBF kernel

In this sub-section, recognition results using five partitioning strategies and based on the diagonal features using SVM with RBF kernel are presented. Here, partitioning strategy *a* gives the minimum accuracy (71.14%) and partitioning strategy *e* gives the maximum accuracy (84.29%). These results are given in Figure 8.

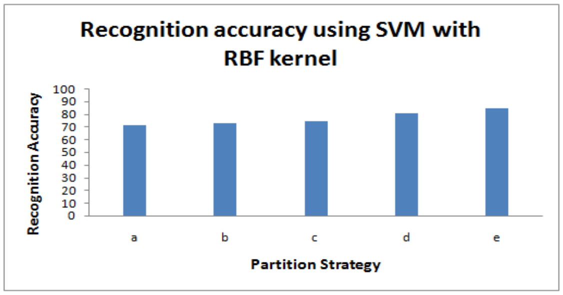


Figure 8: Recognition accuracy using SVM with RBF kernel.

4.2 Intersection and open end points features

In this subsection, the intersection and open end points features have been considered for inputting the classifier. Again three types of SVM as taken in 4.1 have been considered with these features.

4.2.1 Recognition accuracy using SVM with linear kernel

For the features under consideration and the SVM classifier with linear kernel, the minimum accuracy achieved is 81.26% in partitioning strategy *a* and the maximum accuracy achieved is 91.43% in partitioning strategy *e*. The results for this case are depicted in Figure 9.

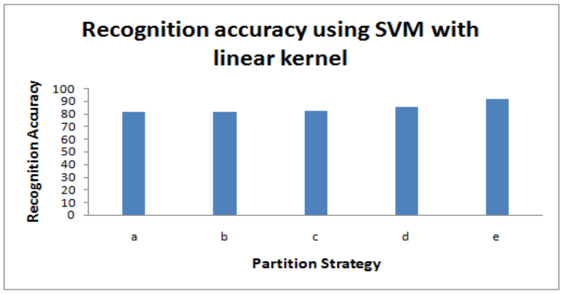


Figure 9: Recognition accuracy using SVM with linear kernel.

4.2.2 Recognition accuracy using SVM with polynomial kernel

In this sub-section, recognition results of five strategies and the SVM with polynomial kernel are presented. Again, the minimum accuracy is achieved when we use strategy *a* and the accuracy achieved is 82.69%. Maximum accuracy is again achieved when we use strategy *e* and the maximum accuracy achieved is 94.29%. These results are depicted in Figure 10.

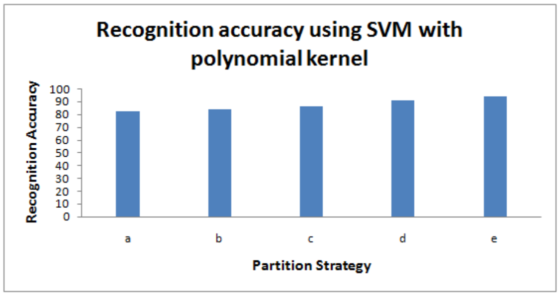


Figure 10: Recognition accuracy using SVM with polynomial kernel.

4.2.3 Recognition accuracy using SVM with RBF kernel

In this sub-section, recognition results of five partitioning strategies using SVM with RBF kernel are presented. Minimum accuracy achieved is 6% while using strategy *d* and maximum accuracy achieved is 22.23% while using strategy *a*. These results are depicted in Figure 11.

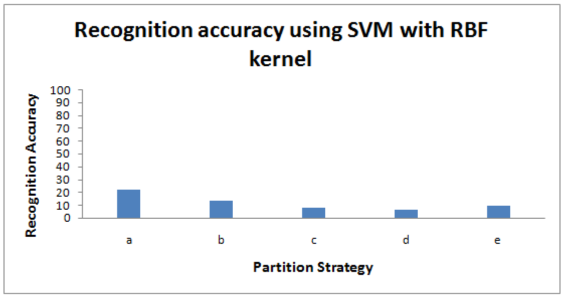


Figure 11: Recognition accuracy using SVM with RBF kernel.

4.3 Diagonal and intersection & open end points features

In this subsection, the diagonal and; intersection and open end points features simultaneously have been considered for inputting the classifier. Here, also again three types of SVM as taken in 4.1 have been considered with these features.

4.3.1 Recognition accuracy using SVM with linear kernel

In this sub-section, we have presented recognition results of five partitioning strategies based on the two features taken from 4.1 and 4.2 simultaneously using SVM with linear kernel. Using this approach, *i.e.*, SVM with linear kernel, we achieved an accuracy of 81.26% when we use strategy *a* and achieved an accuracy of 91.43% when we used the strategy *e*. These results are depicted in Figure 12.

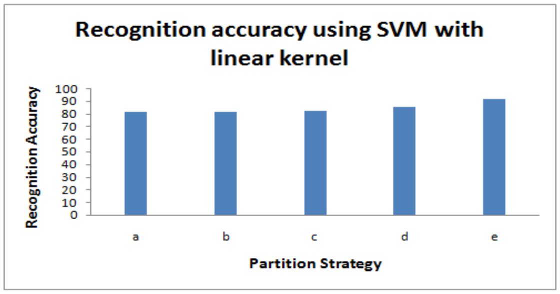


Figure 12: Recognition accuracy using SVM with linear kernel.

4.3.2 Recognition accuracy using SVM with polynomial kernel

In this sub-section, recognition results of five partitioning strategies and the SVM with polynomial kernel are presented. In partitioning strategy *a*, the accuracy that could be achieved was minimum at 82.69% and in strategy *e*, the accuracy achieved was maximum at 94.29%. These results are given in Figure 13.

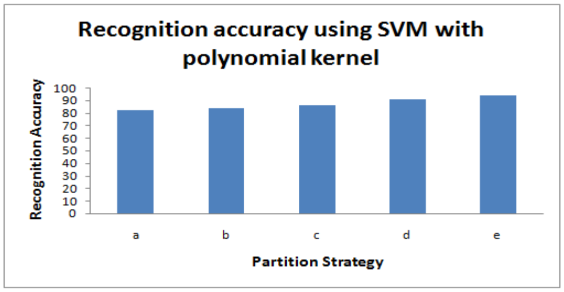


Figure 13: Recognition accuracy using SVM with polynomial kernel.

4.3.3 Recognition accuracy using SVM with RBF kernel

In this sub-section, recognition results of five partitioning strategies using SVM with RBF kernel are presented. Minimum accuracy achieved is 3.29% while using strategy *d* and maximum accuracy achieved is 19.37% while using strategy *a*. The results are depicted in Figure 14.

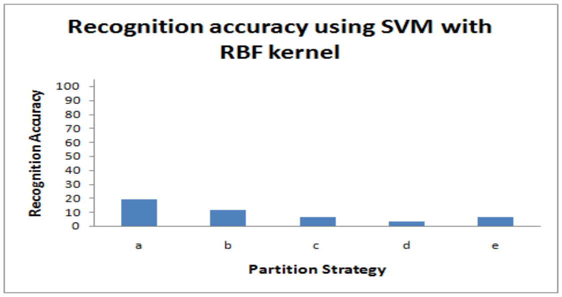


Figure 14: Recognition accuracy using SVM with RBF kernel.

5. Conclusion

The work presented in this paper proposes an offline handwritten Gurmukhi character recognition system. The features of a character that have been considered in this work include diagonal features and; intersection and open end points features. The classifier that has been employed in this work is SVM with three flavors, *i.e.*, SVM with linear kernel, SVM with polynomial kernel and SVM with RBF kernel. The features have been inputted to the classifiers individually and have also been inputted simultaneously. The maximum recognition accuracy of 94.29% is achieved in this work for the case when we input the two features simultaneously to the SVM classifier with polynomial kernel. This accuracy can probably be increased by considering a larger data set while training the classifier. This work can also be extended for offline handwritten character recognition of other Indian scripts.

6. References

[1] Lorigo, L. M., and Govindaraju, V.: Offline Arabic handwriting recognition: a survey. IEEE Transactions on PAMI, 28, 5 (2006) 712-724

[2] Plamondon, R. and Srihari, S. N.: On-line and off- line handwritten character recognition: A comprehensive survey, IEEE Transactions on PAMI, 22, 1 (2000), 63-84

[3] Lehal, G. S. and Singh, C.: A Gurmukhi script recognition system, In Proceedings of 15th ICPR, 2 (2000), 557-560

[4] Wen, Y., Lu, Y. and Shi, P.: Handwritten Bangla numeral recognition system and its application to postal automation, Pattern Recognition, 40 (2007), 99-107

[5] Swethalakshmi, H., Jayaraman, A., Chakravarthy, V. S. and Sekhar, C. C.: Online handwritten character recognition of Devanagari and Telugu characters using support vector machine, In Proceedings of 10th IWFHR, (2006), 367-372

[6] Pal, U., Wakabayashi, T. and Kimura, F.: A system for off-line Oriya handwritten character recognition using curvature feature, In Proceedings of 10th ICIT, (2007), 227-229

[7] Hanmandlu, M., Grover, J., Madasu, V. K. and Vasikarla, S.: Input fuzzy for the recognition of handwritten Hindi numeral, In Proceedings of ITNG, (2007), 208-213

[8] Rajashekararadhya, S. V. and Ranjan, S. V.: Zone based Feature Extraction algorithm for Handwritten Numeral Recognition of Kannada Script, In Proceedings of IACC, (2009), 525-528

[9] Tripathy, J.: Reconstruction of Oriya alphabets using Zernike Moments, International Journal of Computer Applications, 8,8 (2010), 26-32

[10] Jindal, M. K.: Degraded Text Recognition of Gurmukhi Script", PhD Thesis, Thapar University, Patiala, India, 2008

Obtaining the Minimal Polygonal Representation of a Curve by Means of a Fuzzy Clustering

Alexander Lepskiy

Higher School of Economics, Moscow, Russia

Abstract. The problem of obtaining of a minimal polygonal representation of a plane digital curve is treated. Means of a fuzzy clustering method are used. The fuzzy clustering is realized by relations of similarity and dissimilarity that are defined on a planar digital curve.

1 Introduction

As a rule the some set of features is extracted in image to analyse and recognize an object in the image. We will distinguish between low- and high-level features in the image. The low-level features are features that may be extracted without information about a special location of certain parts of the image object [14]. In the contrary the information about special location of certain parts of the image object is used to detect high-level features. The edges of image object, the curvature of a curve on the image are the main low-level features in the image. The low-level features are aggregated for receive a compact representation of an image object. As a result we will receive a high-level representation of image object, for example, a curve. The compact curve representation is necessary for image compression, image vectorization etc. In general digital curve $\varGamma$ depend on a set of parameters so the number of parameters may be equal to the number of points of digital curve. In this case a problem of representation is to find the curve $\varGamma'$ that depends from a smaller set of parameters and saves a main information about the curve $\varGamma$. There are many methods of solving of this problem, which may be divided into two groups – the group of approximate methods and the group of interpolative methods. The methods of first group are based on the replacement of digital curve $\varGamma$ by a such curve of some fixed class that satisfies to some conditions "nearness". The methods based on the Bezier curves and on the B-spline are the most popular approximate methods of finding the curve representation [15], [13]. The application of those methods requires a prior detection of knots of spline and this task is equal to a general task of a curve representation. The methods of second group assumes the choice of some set of points on $\varGamma$ and replacement of every part of curve between the two neighbor points by the other curve from some fixed class (for example, class of segments, circles, algebraic curves of some order etc.) in agreement with the defined optimal conditions. The straight-line interpolation is called by polynomial representation of curve. There are two main approaches of solving the task of polynomial representation of curve: heuristic and optimization. The algorithms based on extraction of dominant points, on the using of split-junction procedure for a side of polygon (for example, Douglas-Peucker algorithm), genetic algorithms [6], algorithms of multiple smoothing [24], algorithms on the fuzzy logic [11] etc. are referred to the algorithms of the first approach. These algorithms are rapid but not optimal as a rule. Algorithms of the second approach assume to find such approximate polygonal line which satisfied to a defined optimal condition. The conditions which are considered as an optimal criterion are following: 1) the polygon with fixed number of vertex must have minimal perimeter [23]; 2) the maximal distance between the points of the curve and segments of the polygon must be a minimal [18]; 3) the number of segment of polygon with approximation error must be a minimal [4]; 4) the area of a symmetric difference between the set bounded by a closed curve and set bounded by the polygon must be a minimal [28]; 5) the approximation error of polygon with a fixed length of a segment must be a minimal [21]. Thus the algorithms of second approach solve tasks of nonlinear optimization with boundary conditions. The majority of algorithms mentioned above are suboptimal. Almost all algorithms of finding of compact polygonal representation assumes the preliminary finding of basis set of curve points which must be optimized with a respect to the chosen criteria. The set of points with a high curvature is chosen as a basis set. At this paper we will consider the approach to find polygonal representations of curve with a help of fuzzy clustering methods. The main idea of this approach bases on a fact that the quantitative low-level local features of a curve at the given point may be considered as a degree of membership of this point to polygonal representation. The curve itself is considered as a fuzzy set. Then a problem of finding of a minimal representation of a fuzzy set may be formulated as a solution of a task of a fuzzy clustering.

2 Statement of Problem

We will considered the plane discrete curve $\mathit{\Gamma} = (\mathbf{g}_k)_{k=0}^{n-1}$, $\mathbf{g}_k = x_k\mathbf{i} + y_k\mathbf{j}$. The points $\mathbf{g}_k$, $k = 0, ..., n-1$, belongs to $\mathbb{Z}^2$ and they satisfy to a condition of a connectivity in the initial contour which will be used in an image processing. We will consider the set of points of curve $\mathit{\Gamma}$ as an ordering set. We want to extract some subset $B = \{\mathbf{g}_{i_1}, ..., \mathbf{g}_{i_l}\}$ of points of a curve $\mathit{\Gamma}$ that will be a “good” representation of $\mathit{\Gamma}$.

The minimal polygonal representations of curve must consist of such points $\mathbf{g}$ of curve $\mathit{\Gamma}$ which have a great information capacity relatively to a given set of features $\{\omega_i\}_{i\in I}$. We will consider only local features: low-level features of curve. We may be consider those features as some functions of points of curve: $\omega_i(\mathbf{g})$, $\mathbf{g} \in \mathit{\Gamma}$, $i \in I$. It will be assumed that $\omega_i(\mathbf{g}) \in [0, 1]$ for all $\mathbf{g} \in \mathit{\Gamma}$, $i \in I$ and $\omega_i(\mathbf{g}) \leq \omega_i(\mathbf{h})$, if the point $\mathbf{h} \in \mathit{\Gamma}$ is more informative than point $\mathbf{g} \in \mathit{\Gamma}$ relatively of feature ω_i. The normalized estimation of curvature and the normalized variation of contour length after deletion of point $\mathbf{g}$ are by examples of such features functions [2]. The function $\omega_i(\mathbf{g})$ characterizes the degree of membership of point $\mathbf{g}$ to set informative points of curve Γ relatively i-th feature. Therefore the set of informative points of curve Γ relatively i-th feature may be considered as a fuzzy set $\{(\mathbf{g}, \omega_i(\mathbf{g})),\ \mathbf{g} \in \Gamma\}$ with membership function ω_i. If we consider the information capacity of points of curve Γ relatively to the set of features $\{\omega_i\}_{i \in I}$ set Γ can be considered in terms of a fuzzy set with membership function $\omega(\mathbf{g}) = T(\omega_i(\mathbf{g}))$, where $T(\cdot)$ – t-norm on $[0,1]^I$ [9]. For example, $T(\omega_i) = \min\limits_i \omega_i$ or $T(\omega_i) = \prod_{i \in I} \omega_i$. In general case we can use some nonnegative function from feature $\mu_\Gamma(\mathbf{g}) = f(\omega(\mathbf{g}))$ as a membership function. Then we may formulated the task of finding of such minimal fuzzy subset B of set Γ for which the set $\{\omega(\mathbf{g})\}_{\mathbf{g} \in \mathrm{B}}$ will be an optimal representation of $\{\omega(\mathbf{g})\}_{\mathbf{g} \in \Gamma}$. Let's specify a statement of problem. Let's consider α-cut $\mathrm{B}_\alpha = \{\mathbf{g} \in \Gamma : \omega(\mathbf{g}) \geq \alpha\{$ of fuzzy set Γ for some fixed value $\alpha \in [0,1]$. The set B_α is a some representation of a contour Γ. It is necessary to find such value of parameter $\alpha \in [0,1]$ that the representation B_α will be minimal on the one hand and will be optimal on other hand. The finding of minimal representation of a fuzzy set is a task of fuzzy clustering. The main ways to solve a fuzzy clustering task were considered in the works [19],[20], [5], [1] etc. The review of fuzzy clustering methods may be found in [29]. The modern state of problem may is reviewed in [12]. One approach to fuzzy clustering consists to definition of some functionals on the set of all representations, which then are optimized to receive a desired clustering.

3 The Using of Similarity Relation

Let us consider representation B_α of contour Γ, $\alpha \in [0,1]$ with membership function $\mu_\alpha^\omega(\mathbf{g}) = \begin{cases} \omega\ (\mathbf{g})|\mathrm{B}_\alpha|, & \mathbf{g} \in \mathrm{B}_\alpha, \\ 0, & \mathbf{g} \notin \mathrm{B}_\alpha. \end{cases}$ We are introduced into consideration so called fuzzy similarity relation $r(\mathbf{g}, \mathbf{h})$ on Γ that is reflexive, symmetric fuzzy relation satisfying to inequality $|r(\mathbf{g}, \mathbf{h}) - r(\mathbf{g}, \mathbf{e})| \leq 1 - r(\mathbf{h}, \mathbf{e})$ for all $\mathbf{e}, \mathbf{g}, \mathbf{h} \in \Gamma$ for construction of identifying functional. The last inequality is an equivalent to condition for strongly Δ -transitive relation (respect to t-norm $a\Delta b = \max\{a + b - 1, 0\}$) [7]. The equivalence of strongly Δ-similarity (that is reflexive, symmetric, strongly Δ-transitive relation) and Δ-similarity was proved in [7]. The coherent nearness relation [3] is weak. By analogy with E.H.Ruspini we called set B_α by fuzzy r-representation of set Γ if the inequality

$$\sum_{\mathbf{h} \in \Gamma} r(\mathbf{g}, \mathbf{h}) \mu_\alpha^\omega(\mathbf{h}) \geq \mu_\Gamma(\mathbf{g}) \qquad (1)$$

is holds for all $\mathbf{g} \in \Gamma$. The efficiency of such clustering depends on a fuzzy similarity relation $r(\mathbf{g}, \mathbf{h})$. The choice of this relation is defined by classification features. In particular, the function $r(\mathbf{g}, \mathbf{h}) = 1 - \ n^{-1} \sum_{i=1}^n \rho_i(\omega_i(\mathbf{g}), \omega_i(\mathbf{h}))$ is the similarity relation, where $\omega_i(\mathbf{g})$ is an informativity function of the i-th feature of point $\mathbf{g}$, ρ_i is such metric in R^1 that $\rho_i(a, b) \leq 1$ for all $a, b \in [0,1]$. We will consider the similarity relation $r(\mathbf{g}, \mathbf{h}) = 1 - \ |\omega(\mathbf{g}) - \omega(\mathbf{h})|$ below. Then (1) take the form

$$|\mathrm{B}_\alpha| \sum_{\mathbf{h}\in \mathrm{B}_\alpha} \left(1 - |\omega(\mathbf{g}) - \omega(\mathbf{h})|\right) \omega(\mathbf{h}) \geq \omega(\mathbf{g})|\mathit{\Gamma}| \qquad (2)$$

for all $\mathbf{g} \in \mathit{\Gamma}$. It is obvious to see that $\mathrm{B}_\alpha = \mathit{\Gamma}$ povided (2) is valid. Thus the task consists of a maximal reduction of a cardinality of B_α (with increased α) until (2) remains valid. The set B_α of minimum cardinality for which (2) is valid we will call by minimal r-representation of set $\mathit{\Gamma}$ and will denote by $\underline{\mathrm{B}}_\alpha$. The following inequality may be get from (2) if we considered that $\omega(\mathbf{g}) < \alpha$ if $\mathbf{g} \in \mathit{\Gamma}\backslash\mathrm{B}_\alpha$:

$$\sum_{\mathbf{h}\in \mathrm{B}_\alpha} (1 - \omega(\mathbf{h}))\omega(\mathbf{h}) \geq \left(\frac{|\mathit{\Gamma}|}{|\mathrm{B}_\alpha|} - \sum_{\mathbf{h}\in \mathrm{B}_\alpha} \omega(\mathbf{h}) \right) \max_{\mathbf{g}\in \mathit{\Gamma}\backslash \mathrm{B}_\alpha} \omega(\mathbf{g}). \qquad (3)$$

Thus we proved the validity of the following proposition.

Proposition 1. *If set* B_α *is a fuzzy* r*-representation of set* $\mathit{\Gamma}$ *then (3) is correct.*

The contrary of this statement may be is not true in general. The algorithm of finding of minimal representation $\underline{\mathrm{B}}_\alpha$ consists of two steps: 1) to find the set $\mathrm{B}_\alpha^{(1)}$ of minimum cardinality for which (3) is valid; 2) to add (if it is necessary) the set $\mathrm{B}_\alpha^{(1)}$ by such points $\mathbf{h} \in \mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}$ that (2) is correct. Let $\widehat{\mathit{\Gamma}} = \{\mathbf{h}_i\}_{i=1}^{|\mathit{\Gamma}|}$ be a set of points of contour $\mathit{\Gamma}$ ordered by decreasing of weights $\omega(\mathbf{h})$, $\mathbf{h} \in \mathit{\Gamma}$. Calculate the function

$$Q(p) := \sum\nolimits_{i=1}^{p} (1 - \omega(\mathbf{h}_i))\omega(\mathbf{h}_i)$$

and let

$$R(p) := \left(\frac{|\mathit{\Gamma}|}{p} - \sum\nolimits_{i=1}^{p} \omega(\mathbf{h}_i) \right) \max_{p+1\leq j\leq |\mathit{\Gamma}|} \omega(\mathbf{g}_j)$$

for $p = 1, 2, ..., |\mathit{\Gamma}|$. The minimum p for which $Q(p) \geq R(p)$ will be define a boundary of partition of set $\widehat{\mathit{\Gamma}}$ on two classes $\mathrm{B}_\alpha^{(1)} := \left\{ \mathbf{h}_i \in \widehat{\mathit{\Gamma}} : \ i = 1, 2, ..., p \right\}$ and $\mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}$ as a consequence from (3). On the second step are to find such point $\mathbf{h} \in \mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}$ for which $\left(1 - |\omega(\mathbf{g}) - \omega(\mathbf{h})|\right)\omega(\mathbf{h}) \to \max$ for all $\mathbf{g} \in \mathit{\Gamma}\backslash \left(\mathrm{B}_\alpha^{(1)} \cup \{\mathbf{h}\}\right)$. We will check the validity of condition (2) for set $\mathrm{B}_\alpha^{(2)} = \mathrm{B}_\alpha^{(1)} \cup \{\mathbf{h}\}$. If (2) is not correct then we add the new point from $\mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(2)}$ to the set $\mathrm{B}_\alpha^{(2)}$ etc. We have a question. Will we get a minimal fuzzy r-representation of curve $\mathit{\Gamma}$ with help of suggested algorithm indeed? The following proposition gives us the condition when we will get the minimal fuzzy r-representation after the first step.

Proposition 2. *If we get after the first step of algorithm such a representation* $\mathrm{B} = \mathrm{B}_\alpha^{(1)}$ *that*

$$\sum_{\mathbf{h}\in \mathrm{B}} (1 - \omega(\mathbf{h}))^2 \leq 1 + |\mathrm{B}| - \frac{|\mathit{\Gamma}|}{\mathrm{B} + 1} \qquad (4)$$

and $|\mathit{\Gamma}| \max_{\mathbf{g}\in \mathit{\Gamma}} \omega(\mathbf{g}) \leq \alpha^2 |\mathrm{B}|^2$ *then* $\mathrm{B}_\alpha^{(1)}$ *will be a minimal fuzzy* r*-representation of a curve* $\mathit{\Gamma}$.

Proof. Firstly we show that the representation $\mathrm{B}_\alpha^{(1)}$ formed on the first step of algorithm is a minimal representation for which (3) is satisfied. To show this we consider the set function

$$\phi(\mathrm{B}) = \sum_{\mathbf{h}\in\mathrm{B}} (1-\omega(\mathbf{h}))\omega(\mathbf{h}) \Bigg/ \left(\frac{|\mathit{\Gamma}|}{|\mathrm{B}|} - \sum_{\mathbf{h}\in\mathrm{B}} \omega(\mathbf{h})\right),$$

where $\mathrm{B} \subseteq \mathit{\Gamma}$ such that $\sum_{\mathbf{h}\in\mathrm{B}} \omega(\mathbf{h}) < \frac{|\mathit{\Gamma}|}{|\mathrm{B}|}$. Let $\phi(\emptyset) = 0$. Let us show that ϕ is monotone set function. Let $S_i = \sum_{\mathbf{h}\in\mathrm{B}} \omega^i(\mathbf{h})$, $\delta_i = \frac{|\mathit{\Gamma}|}{|\mathrm{B}|+i-1}$, $i = 1, 2$. Then $\phi(\mathrm{B}) = \frac{S_1 - S_2}{\delta_1 - S_1}$. We have

$$\psi(\omega(\mathbf{g})) = \phi(\mathrm{B} \cup \{\mathbf{g}\}) = \frac{S_1 - S_2 + \omega(\mathbf{g}) - \omega^2(\mathbf{g})}{\delta_2 - S_1 - \omega(\mathbf{g})}$$

for such every $\mathbf{g} \in \mathit{\Gamma}\backslash\mathrm{B}$ that $\omega(\mathbf{g}) + \sum_{\mathbf{h}\in\mathrm{B}} \omega(\mathbf{h}) < \frac{|\mathit{\Gamma}|}{|\mathrm{B}|+1}$. Then $\phi(\mathrm{B} \cup \{\mathbf{g}\}) - \phi(\mathrm{B}) = \frac{\omega(\mathbf{g}) - \omega^2(\mathbf{g})}{\delta_2 - S_1 - \omega(\mathbf{g})} + \frac{\psi(S_1 - S_2)(\delta_1 - \delta_2 + \omega(\mathbf{g}))}{(\delta_2 - S_1 - \omega(\mathbf{g}))(\delta_2 - S_1)} \geq 0$ such as $S_1 \geq S_2$, $\delta_1 \geq \delta_2$. Therefore ϕ is a monotone set function. In addition $\psi\,(x)$ is a monotone function on $[0, 1]$. Indeed, we have $\psi'(x) = \frac{x^2 - 2x(\delta_2 - S_1) + \delta_2 - S_2}{(\delta_2 - S_1 - x)^2}$. Two cases are possible. At the first case the minimal value of numerator of derivative $\psi'(x)$ up to $x = 1$ if $\delta_2 - S_1 > 1$. In this case we have $\psi'(x) \leq 0$ if $\delta_2 \geq 1 + 2S_1 - S_2 \Leftrightarrow$ (4). At the other case $0 \leq \delta_2 - S_1 \leq 1$ and the minimal value of numerator of derivative $\psi'(x)$ up to $x = \delta_2 - S_1$ and $\psi'(x) \geq 0$ if $\delta_2 - S_2 - (\delta_2 - S_1)^2 \geq 0$. The last inequality is true because $\delta_2 - S_2 - (\delta_2 - S_1)^2 \geq \delta_2 - S_1 - (\delta_2 - S_1)^2 \geq 0 = (\delta_2 - S_1)\,(1 - (\delta_2 - S_1)) \geq 0$. Therefore ϕ is a monotone set function and $\phi(\mathrm{B} \cup \{\mathbf{g}'\}) \geq \phi(\mathrm{B} \cup \{\mathbf{g}''\})$ if $\omega(\mathbf{g}') \geq \omega(\mathbf{g}'')$. We may get two cases when we form the set $\mathrm{B}_\alpha^{(1)}$: 1) $\sum_{\mathbf{h}\in\mathrm{B}_\alpha^{(1)}} \omega(\mathbf{h}) < \frac{|\mathit{\Gamma}|}{|\mathrm{B}_\alpha^{(1)}|}$; 2) there exist such point $\mathbf{g} \in \mathrm{B}_\alpha^{(1)}$ that $\frac{|\mathit{\Gamma}|}{|\mathrm{B}_\alpha^{(1)}|} - \omega(\mathbf{g}) < \sum_{\mathbf{h}\in\mathrm{B}_\alpha^{(1)}\backslash\{\mathbf{g}\}} \omega(\mathbf{h}) < \frac{|\mathit{\Gamma}|}{|\mathrm{B}_\alpha^{(1)}|}$. The set $\mathrm{B}_\alpha^{(1)}$ will be by set with minimal cardinality which satisfied to (3) since the set $\mathrm{B}_\alpha^{(1)}$ in the first case and the set $\mathrm{B}_\alpha^{(1)}\backslash\{\mathbf{g}\}$ in the second case formed from the points $\mathbf{h} \in \mathit{\Gamma}$ with maximal value of feature $\omega(\mathbf{h})$. We will show the validity of inequality (2) for set $\mathrm{B}_\alpha^{(1)}$ if the condition of proposition obeys to completing of proof of proposition. If $\mathbf{g} \in \mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}$, an inequality (2) will lead to inequality (3). Consequently it will be correct. Let $\mathbf{g} \in \mathrm{B}_\alpha^{(1)}$. Then $|\omega(\mathbf{g}) - \omega(\mathbf{h})| \leq 1 - \alpha$ for any $\mathbf{h} \in \mathrm{B}_\alpha^{(1)}$. Then we have

$$\sum_{\mathbf{h}\in\mathrm{B}_\alpha^{(1)}} (1 - |\omega(\mathbf{g}) - \omega(\mathbf{h})|)\,\omega(\mathbf{h}) \geq \alpha \sum_{\mathbf{h}\in\mathrm{B}_\alpha^{(1)}} \omega(\mathbf{h}) \geq$$

$$\alpha^2 \left|\mathrm{B}_\alpha^{(1)}\right| \geq \frac{|\mathit{\Gamma}|}{|\mathrm{B}_\alpha^{(1)}|} \max_{\mathbf{g}\in\mathit{\Gamma}} \omega(\mathbf{g}) \geq \frac{|\mathit{\Gamma}|}{|\mathrm{B}_\alpha^{(1)}|} \omega(\mathbf{g}).$$

The proposition is thus proved.

If we get on the first step of algorithm the set $\mathrm{B}_\alpha^{(1)}$ for which conditions of proposition aren't satisfied we must carry out the second step of algorithm. In this case we may to get the set B_α near to minimal in general.

Remark 1. The function $r_s(\mathbf{g},\mathbf{h}) = 1 - |\omega(\mathbf{g}) - \omega(\mathbf{h})|^s$, $s \in (0,1]$, may be used in the capacity of similarity relation too. This function satisfies to all conditions of similarity relation because the inequality $(a+b)^s \le a^s + b^s$ is correct for $a,b \ge 0$, $0 \le s \le 1$. It is obvious that the inclusion $\underline{\mathrm{B}}_\alpha(s_1) \supseteq \underline{\mathrm{B}}_\alpha(s_2)$ is correct for minimal r_s-representation $\underline{\mathrm{B}}_\alpha(s)$ if $0 < s_1 \le s_2 \le 1$ because $r_{s_1} \le r_{s_2}$ in this case.

4 The Using of Dissimilarity Relation

Other relation may be used in task of fuzzy clustering without similarity relation. For example, the points of minimal polygonal representation must be located far from each other on a curve Γ. We may introduce the fuzzy dissimilarity relation regarding these conditions. This relation must be antireflexive, symmetric fuzzy relation $\tau(\mathbf{g},\mathbf{h})$ and obeying to an inequality $|\tau(\mathbf{g},\mathbf{h}) - \tau(\mathbf{g},\mathbf{e})| \le \tau(\mathbf{h},\mathbf{e})$ for all $\mathbf{e},\mathbf{g},\mathbf{h} \in \Gamma$. Note that definition of fuzzy dissimilarity relation is coordinated with a fuzzy similarity relation that is introduced above. Let $f(\mathbf{g})$ be membership function of point $\mathbf{g} \in \Gamma$ to the set of informative points. Again we will call the set $\mathrm{B}_\beta = \left\{\mathbf{g} \in \Gamma : \mu_\beta^f(\mathbf{g}) \ge \beta\right\}$ with membership function $\mu_\beta^f(\mathbf{g})$ by fuzzy τ-representation of set Γ if the inequality

$$\sum_{\mathbf{h}\in\Gamma} (1 - \tau(\mathbf{g},\mathbf{h}))\left(1 - \mu_\beta^f(\mathbf{h})\right) \ge 1 - f(\mathbf{g}) \tag{5}$$

is correct for all $\mathbf{g} \in \Gamma$. We will considered that condition (5) is valid if $\mathrm{B}_\beta = \emptyset$ and isn't valid if $\mathrm{B}_\beta = \Gamma$. Thus the task is to increase maximally the cardinality of B_β (with decreased β) until (5) remains is valid. The set B_β of maximum cardinality for which (5) is valid we will call a maximal τ-representation of set Γ and will denote by $\bar{\mathrm{B}}_\beta$.

We will use the function $\tau(\mathbf{g},\mathbf{h}) = l(\mathbf{g},\mathbf{h})$ as a dissimilarity relation. Here $l(\mathbf{g},\mathbf{h})$ is a minimal length of arc of the curve Γ located between the points $\mathbf{g},\mathbf{h} \in \Gamma$, that normed by length of all curve Γ. We will use also the functions $f(\mathbf{g}) = \omega(\mathbf{g})|\Gamma|$ and $\mu_\beta^f(\mathbf{g}) = |\Gamma \backslash \mathrm{B}_\beta| \begin{cases} 1 & , \ \mathbf{g} \in \mathrm{B}_\beta, \\ f(\mathbf{g}), & \mathbf{g} \notin \mathrm{B}_\beta \end{cases}$ as a membership function of curve Γ and set B_β correspondingly. Then inequality (5) can be rewritten:

$$|\Gamma \setminus \mathrm{B}_\beta| \sum_{\mathbf{h}\in\Gamma\setminus\mathrm{B}_\beta} (1 - l(\mathbf{g},\mathbf{h}))\,(1 - f(\mathbf{h})) \ge |\Gamma|\,(1 - f(\mathbf{g})) \tag{6}$$

for all $\mathbf{g} \in \Gamma$. Then the new formulation of task about seach of (r,τ)-representation of curve Γ follows from (2) and (6). It is necessary to find such set B for which the system of inequalities

$$\sum_{\mathbf{h}\in\mathrm{B}} (1 - |\omega(\mathbf{g}) - \omega(\mathbf{h})|)\,\omega(\mathbf{h}) \ge \frac{|\Gamma|}{|\mathrm{B}|}\omega(\mathbf{g}),$$

$$\sum_{\mathbf{h}\in\Gamma\setminus\mathrm{B}} (1 - l(\mathbf{g},\mathbf{h}))\,(1 - \omega(\mathbf{h})) \ge \frac{|\Gamma|}{|\Gamma \setminus \mathrm{B}|}\,(1 - \omega(\mathbf{g}))$$

are holds for all $\mathbf{g} \in \mathit{\Gamma}$. We have a question: in what case does the algorithm give us the minimal (r, τ)-representation of curve $\mathit{\Gamma}$? The next statement follows from proposition 2.

Proposition 3. *If after the first step of algorithm we get such representation* $\mathrm{B}_\alpha^{(1)}$ *that (4) is true and* $|\mathit{\Gamma}| \max_{\mathbf{g}\in\mathit{\Gamma}} \omega_I(\mathbf{g}) \leq \alpha^2 \left|\mathrm{B}_\alpha^{(1)}\right|^2$, $|\mathit{\Gamma}| \min_{\mathbf{g}\in\mathit{\Gamma}} \omega_I(\mathbf{g}) \geq |\mathit{\Gamma}| - 0.5(1-\alpha)\left|\mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}\right|^2$ *then* $\mathrm{B}_\alpha^{(1)}$ *will be a minimal fuzzy* (r,τ)*-representation of closed digital curve* $\mathit{\Gamma}$*.*

Proof. If the conditions of proposition are satisfied, then the set $\mathrm{B}_\alpha^{(1)}$ will be a minimal fuzzy r-representation as was shown in proposition 2. Now we should proof that the set $\mathrm{B}_\alpha^{(1)}$ will be fuzzy τ-representation too. To show this, it's noticed that $\sum_{\mathbf{h}\in A} l(\mathbf{g},\mathbf{h}) \leq 0.5\,|A|$ for closed curve and any point $\mathbf{g} \in \mathit{\Gamma}$, $A \in 2^{\mathit{\Gamma}}$. Then we have

$$\sum_{\mathbf{h}\in\mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}} (1 - l(\mathbf{g},\mathbf{h}))(1-\omega(\mathbf{h})) \geq (1-\alpha) \sum_{\mathbf{h}\in\mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}} (1 - l(\mathbf{g},\mathbf{h})) \geq$$

$$0.5(1-\alpha)\left|\mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}\right| \geq \frac{|\mathit{\Gamma}|}{\left|\mathit{\Gamma}\backslash\mathrm{B}_\alpha^{(1)}\right|}\left(1 - \min_{\mathbf{g}\in\mathit{\Gamma}} \omega(\mathbf{g})\right).$$

The proposition is thus proved.

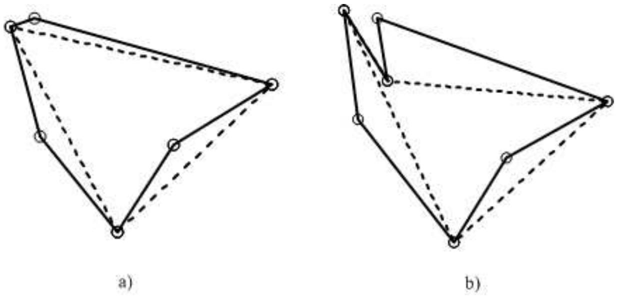


Fig. 1. The initial contour and the minimal polygonal representation of contour found by fuzzy clustering method

The results of the algorithm of a research of minimal polygonal representation of contour are shown in Fig.1. On the Fig.1.a the representation was found by fuzzy clustering method with help of similarity and dissimilarity relations separately. On the Fig.1.b the representation was found by fuzzy clustering method with help of combined using of similarity and dissimilarity relations. We used normalized estimation of curvature in the capacity of feature function $\omega(\mathbf{g})$ (see [10]). Note that the quality of algorithm work may be improved if we will use the fuzzy clustering for the few features.

5 Summary and Conclusion

In this paper we have considered two clusters in a polygonal representation of a curve. The first cluster consists of points that belong to the polygonal representation. The second cluster consists of points that not belong to the polygonal representation. In case of the crisp clustering distance within one cluster is small, whereas clusters are sparse, so two objects from different clusters are distant. The notion of distance at this paper was replaced by similarity and dissimilarity fuzzy relation. We have received the fuzzy clustering method for polygonal representation. The quality of this representation depends on a similarity and dissimilarity fuzzy relation.

6 Acknowledgement

This work was supported by the grant 11-07-00591 of RFBR (Russian Foundation for Basic Research). The study was undertaken in the framework of the Programme of Fundamental Studies of the Higher School of Economics in 2011.

References

[1] Bezdek, J.C.: Pattern recognition with fuzzy objective finction algorithms. Plenum Press, New York, (1981).

[2] Bronevich, A., Lepskiy, A.: Geometrical fuzzy measures in image processing and pattern recognition. Proc. of the 10th IFSA World Congress. Istanbul, Turkey (2003) 151–154

[3] Dobrakovova, J.: Pseudometrics and fuzzy relations. Aplimat – J. Appl. Math. **2** (1) (2009) 89–95

[4] Dunham, J.G.: Optimum uniform piecewise linear approximation of planar curves. IEEE Trans. Pattern Anal. Mach. Intell. **8** (1986) 67–75

[5] Dunn, J.C.: A fuzzy relative of the ISODATA process and its use in detecting compact, well-separated clusters. J. Cybernetics **3** (1974) 32–57

[6] Huang, S.-C., Sun, Y.-N.: Polygonal approximation using genetic algorithms. Pattern Recognition **32** (1999) 1409–1420

[7] Kreinovich, V.: Strongly transitive fuzzy relations: an alternative way to describe similarity. Int. J. of Intel. Sys. **10** (1995) 1061–1076

[8] Kurozumi, Y., Davis, W.A.: Polygonal approximation of the minimax method. Comput. Graph Image Process **19** (1982) 248–264

[9] Klement, E.P., Mesiar, R. Pap, E.: Triangular norms. Kluwer, Dordrecht (2000)

[10] Lepskii, A.E.: On stability of the center of masses of the vector representation in one probabilistic model of noiseness of an image contour. Automation and Remote Control, (2007), **68**(1), 75–84

[11] Li, L., Chen, W.: Corner detection and interpretation on planar curves using fuzzy reasoning. IEEE Trans. PAMI **21**(11), (1999) 1204–1210

[12] Miyamoto, S., Ichihashi, H., Honda, K.: Algorithms for fuzzy clustering: methods in c-means clustering with applications. Studies in fuzziness and soft computing. Springer-Verlag, (2008) 247pp.

[13] Medioni, G., Yasumoto, Y.: Corner detection and curve representation using cubic B-splines. Comput. Vision. Graph. Image Process. **39** (1987) 267–278

[14] [NA1] Nixon, M.S., Aguado, A.S.: Feature extraction and image processing. Newnes, Oxford (2002)

[15] Pavlidis, T.: Algorithms for graphics and image processing. Computer Science Press, Rockville, Maryland (1982) 416 pp.

[16] Pavlidis, T., Horowitz, S.L.: Segmentation of plane curves. IEEE Trans. Comput., **23** (1974) 860–870

[17] Pei, S.-C., Horng, J.-H.: Optimum approximation of digital planar curves using circular arcs. Pattern Recognition **29** (3), (1996) 383–388

[18] Ramer, U.: An iterative procedure for the polygonal approximation of plane closed curves. Computer Graphics Image Processing **1** (1972) 244–256

[19] Ruspini, E.H.: New experimental results in fuzzy clustering. Information Sciences **6** (1973) 273–284

[20] Ruspini, E.H.: A new approach to clustering. Information and Control **15** (1969) 22–32

[21] Rannou, F., Gregor, J.: Equilateral polygon approximation of closed contours. Pattern Recognition **29** (1996) 1105–1115

[22] Ray, B.K., Ray, K.S.: Determination of optimal polygon from digital curve using L1 norm. Pattern Recognition **26** (1993) 505–509

[23] Sklansky, J., Chazin, R.L., Hansen, B.J.: Minimum-perimeter polygons of digitized silhouettes. IEEE Trans. Comput. **21** (1972) 260–268

[24] Saint-Marc, P., Chen, J.-S., Medioni G.: Adaptive smoothing: A general tool for early vision. IEEE Trans. Pattern Anal. Machine Intell. **13** (6), (1991) 514–529

[25] Sklansky, J., Gonzalez, V.: Fast polygonal approximation of digitized curves. Pattern Recognition **12** (1980) 327–331

[26] Williams, C.M.: An efficient algorithm for the piecewise linear approximation of planar curves. Comput. Graph Image Process **8** (1978) 282–293

[27] Wall, K., Danielson, P.-E.: A fast sequential method for polygonal approximation of digitized curves. Comput. Vis. Graph. Image Process **28** (1984) 220–227

[28] Wu, J.-S., Leou, J.-J.: New polygonal approximation schemes for object shape representation. Pattern Recognition **26** (1993) 471–484

[29] Yang, M.-S.: A survey of fuzzy clustering. Mathl. Comput. Modelling Vol. **18** (11), (1993) 1–16

KDDClus: A Simple Method for Multi-Density Clustering

Sushmita Mitra[1] and Jay Nandy[1]

[1]Machine Intelligence Unit, Indian Statistical Institute, Kolkata 700 108, INDIA,
sushmita@isical.ac.in

Abstract. Automated clustering of multi-density spatial data is developed. The algorithm KDDClus serves as an enhancement to the well-known DBSCAN. Averaging the distances of a pattern to all k of its nearest neighbours allows a smoothing out of noise while automatically detecting the "knees" from the k-distance plot. The use of the KD-tree data structure enables efficient computation of the k-nearest neighbours (k-NN) of a pattern point, particularly for large data. Experimental results on synthetic data, involving nested multiple densities of different shapes, demonstrates the superiority of KDDClus.

Keywords: Density-based clustering, DBSCAN.

1 Introduction

Often we come across spatial data consisting of a mixture of pattern distributions involving different densities, which may or may not be nested within each other, in the presence of background noise. Clusters of different densities can, therefore, be modeled as belonging to point processes having different intensities. Clustering of such data is a challenging problem in data mining [6]. It becomes imperative to detect the number of point processes (or cluster type of a certain density) while also assigning the patterns to these different clusters. One needs to estimate a number of thresholds in order to discriminate between these different density distributions. Automatic estimation of such parameters is a difficult task. The complexity of searching the neighbourhood is also large in high-dimensions.

The density-based approach addresses this issue, while detecting clusters of differing densities having arbitrary shape and size. It is non-parametric, and requires no prior information regarding the number of clusters or their underlying density. The algorithms detect the difference in densities among regions of contiguous patterns in a spatial database, and accordingly assign them to different clusters. Noise and outliers are treated as low-density regions, and are removed in terms of certain density criteria. The earliest research in this direction was reported in Refs. [4, 9]. Some of the interesting studies of efficient density-based clustering, in the context of databases and large datasets, are DBSCAN [3], OPTICS [2], DENCLUE [5], CLIQUE [1] and WaveCluster [8].

Typically these algorithms require user-specification of certain parameters, related to density-level thresholds, to be provided as input. Often this becomes all the more difficult when clusters in different regions of the feature space have considerably different densities or clusters with different density levels are nested. In such cases the partitioning might not be proper with one single density threshold.

In this article we describe a new and simple algorithm KDDClus which clusters multiple pattern distributions of different densities in the presence of noise. It is able to distinguish between different density regions, which may or may not be nested and are generally of non-convex shape. The algorithm automatically estimates a number of thresholds to optimally identify the different density regions, without any prior knowledge about the data. While conventional density-based clustering algorithms like DBSCAN typically resort to visual determination of a single threshold to distinguish between two density regions, algorithm KDDClus may be considered as an enhancement to it. The space-partitioning KD-tree data structure [7] is utilized to efficiently determine the k-nearest neighbours (k-NN) of a pattern for large data. The sorted average k-NN distances for the patterns is clustered (i) for the purpose of smoothing out the noise and (ii) automatically determining the optimal number of density regions while minimizing a validity index. The algorithm is computationally inexpensive. The experimental results on a synthetic dataset, consisting of clusters of different densities, demonstrates the effectiveness of the algorithm.

2 KDDClus: An Algorithm Enhancing DBSCAN

Clustering patterns involving different densities and noise, coexisting in the same spatial dataset, requires the determination of a number of thresholds. Automatic estimation of such parameters, particularly in varying densities of multiple point processes, is a difficult task.

Algorithm DBSCAN requires proper estimation of two global parameters ϵ and $MinPts$. This is highly data-dependent, and can be overestimated or underestimated by the visual and/or interactive procedure used. It may automatically lead to misplacement of patterns and even misidentification of clusters. Moreover, the algorithm does not consider the handling of a simultaneous presence of different densities, originating from different point processes in the data. Note that no single set of ϵ and $MinPts$ can properly cluster such a dataset. The complexity of searching the neighborhood is large in high-dimensions, thereby leading to the difficulty in determining a proper distance estimate.

We present here a new and simple way to automatically identify the number of point processes (or clusters of different densities) including noise. The algorithm utilizes the KD-tree data structure for efficient processing in high dimensions. It can simultaneously estimate the different density parameters without any prior knowledge about the data. It is also not expensive.

We compute the average of the distances of a pattern to all k of its nearest neighbors. This is unlike DBSCAN, where only the kth nearest neighbor is considered during the distance computation. The use of the KD-tree data structure [7] enables efficient computation of k-nearest neighbors (k-NN) of a point, particularly for large data. The averaging allows a smoothing of the curve towards noise removal, for subsequent easier automated detection of density-thresholds. We plot these averaged k-distances in an ascending order, to help identify noise with relative ease. Note that patterns corresponding to noise are expected to have larger k-distance values. The aim is to determine the "knees" for estimating the set of ϵ parameters.

A knee corresponds to a threshold where a sharp change of gradient occurs along the k-distance curve. This represents a change in density distribution amongst patterns. Any value less than this density-threshold ϵ estimate can efficiently cluster patterns whose average k-NN distances is lower than that, implying patterns belonging to a certain density. Analogously all knees in the smoothed graph can collectively estimate a set of ϵ's for identifying all the clusters having different density distributions. The knee regions are detected in KDDClus by clustering the sorted k-NN distances. We determine the optimal number c_0 of such segments, by using c-means while optimizing a clustering validity index.

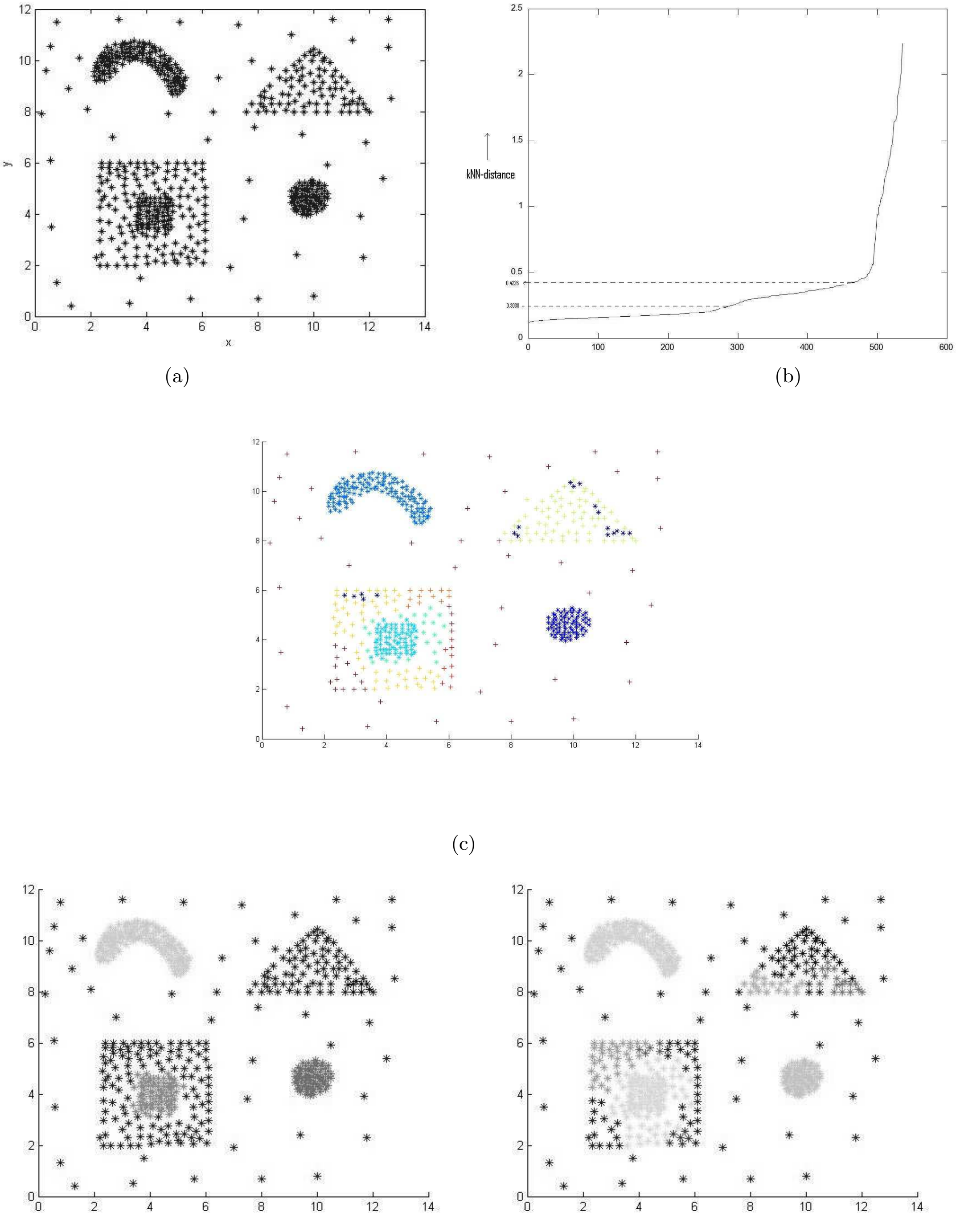
Starting from the lowest value in the sorted k-NN distance graph, we sequentially execute DBSCAN for each of the c_0 estimated ϵ's considered in ascending order. The first estimate obviously corresponds to the most dense cluster. Tagging the patterns in the already detected clusters as "visited", we proceed towards larger values of k-distance while allowing DBSCAN to work on the still-unvisited patterns only. In this manner we are able to effectively determine all clusters in a multi-density framework, in a decreasing order of density, with noise being modelled as the sparsest region.

3 Experimental Results

We have implemented the proposed KDDClus algorithm on a synthetic pattern set. There exist five clusters with three different densities for dataset *Decode* in Fig. 1(a). The semi-circular region on the top-left, inner quadrilateral, and circular region on bottom-right of the figure constitute the most dense clusters. The outer quadrilateral and triangular region form the medium-density clusters. The background is least dense and consists of noise.

We find from part (d) of the figure that DBSCAN, with the lower threshold of $\epsilon = 0.3038$, could correctly identify only (i) the smaller quadrilateral inscribed within the larger one, (ii) the circle, and (iii) the semi-circular region. These are indicated by different shades of gray in the figure. Using DBSCAN with the higher ϵ-value of 0.4226 resulted in the output map in part (e) of the figure. In this case it is noticed that the smaller dense quadrilateral along with some surrounding points from the outer medium-density quadrilateral get merged into one cluster.

This adverse effect is eliminated with algorithm KDDClus, as observed from part (c) of the figure. After correctly detecting the smaller quadrilateral, the circle and the semi-circular region with $\epsilon = 0.3038$, the algorithm marks the



(a)

(b)

(c)

(d)

(e)

Fig. 1. Pattern *Decode*. (a) Original pattern, with (b) sorted average k-distance plot and (c) KDDClus clustering; DBSCAN clustering with (e) $\epsilon = 0.3038$, (f) $\epsilon = 0.4226$.

constituent patterns from the dataset as visited. In the next step DBSCAN needs to work only on the unvisited patterns with $\epsilon = 0.4226$. Now it is able to correctly distinguish the larger quadrilateral and the triangle as the second lower-density level, within the background noise.

4 Conclusions

The algorithm KDDClus is an enhancement to DBSCAN, in terms of automatically estimating the various density-based parameters for optimal clustering. Unlike DBSCAN, where only the kth nearest neighbour is considered during the distance computation, here we calculate the average of the distances of a pattern to all k of its nearest neighbours. Such averaging allows a smoothing of the curve for subsequent easier automated detection of the "knees" amongst the background noise. The use of the KD-tree data structure enables efficient computation of the k-nearest neighbours (k-NN) of a pattern point, particularly for large data. Comparative study has been made on three sets of synthetic data to establish the superiority of the proposed algorithm.

References

1. Agrawal, R., Gehrke, J., Gunopulos, D., Raghavan, P.: Automatic sub-space clustering of high dimensional data for data mining applications. In: Proceedings of 1998 ACM-SIGMOD International Conference on Management of Data (SIGMOD'98). pp. 94–105. Seattle, USA (June 1998)
2. Ankerst, M., Breunig, M., Kriegel, H.P., Sander, J.: OPTICS: Ordering points to identify the clustering structure. In: Proceedings of 1999 ACM-SIGMOD International Conference on Management of Data (SIGMOD'99). pp. 49–60. Philadelphia, USA (June 1999)
3. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases. In: Proceedings of 1996 International Conference on Knowledge Discovery and Data Mining (KDD'96). pp. 226–231. Portland, USA (August 1996)
4. Hartigan, J.A.: Clustering Algorithms. John Wiley & Sons (1975)
5. Hinneburg, A., Keim, D.A.: An efficient approach to clustering in large multimedia databases with noise. In: Proceedings of 1998 International Conference on Knowledge Discovery and Data Mining (KDD'98). pp. 58–65. New York, USA (August 1998)
6. Mitra, S., Acharya, T.: Data Mining: Multimedia, Soft Computing, and Bioinformatics. John Wiley, New York (2003)
7. Moore, A.: A tutorial on KD-trees. Computer Laboratory Technical Report # 209, University of Cambridge, http://www.cs.cmu.edu/∼awm/papers.html (1991)
8. Sheikholeslami, G., Chatterjee, S., Zhang, A.: WaveCluster: A multi-resolution clustering approach for very large spatial databases. In: Proceedings of 1998 International Conference on Very Large Data Bases (VLDB'98). pp. 428–439. New York, USA (August 1998)
9. Wishart, D.: Mode analysis: A generalization of nearest neighbor which reduces chaining effects. Numerical Taxonomy (1969)

Intelligent Data Mining for Turbo-Generator Predictive Maintenance: An Approach in Real-World

Alexandre Pellicel[1], Gonçalo Cássio[1], Marco Aurélio A. Lopes[1],
Luiz Eduardo Borges da Silva[2], Erik Leandro Bonaldi[2],
Levy Ely de Lacerda de Oliveira[2], Jonas Guedes Borges da Silva[2],
Germano Lambert-Torres[2], and Pierre Rodrigues[3]

[1.] TermoNorte Energy Thermal Power Plant Co., Porto Velho, Brazil
{alexandre.pellicel, goncalo.cassio, marco.lopes}@termonorte.com.br
[2.] Institute Gnarus, Itajuba, Brazil
{levy.oliveira, erik.bonaldi, jonas.borges, germanoltorres, leborgess}@gmail.com
[3.] Jordão Engineering Co., Rio de Janeiro, Brasil
pierre.rodrigues@jordaoengenharia.com.br

Abstract. This paper presents the development of a supervision system for predictive maintenance and diagnosis of turbo-generators. The aim of the developed system is to verify the degradation conditions of TermoNorte generators. Initially, a system for extracting features of the turbo-generator operational database has been developed to detect possible problems that cause premature fails. The system has been divided in two parts. The first one is a data acquisition system directly connected to the generator in order to sample some operational variables. The second part concerns an intelligent data mining, based on Rough Sets Theory, into the database involving the supervision system variables, to use the existing historic data to perform analysis of the problems and possible causes.

Keywords: Electrical measurements, signal processing, rough sets, data mining, intelligent systems, turbo-generators.

1 Introduction

The generators are the most important equipments in the energy generation process. The power system reliability, power system supply and power system stability are indexes directly affects for the generator operational conditions. For this reason, protection and monitoring equipment are increasingly employed in order to prevent fails [1].

One of the technologies that can be employed within the purpose of predicting failures is the electric signature analysis (ESA) [2], which consists of a set of methods and techniques that monitor the condition of electric machines by identifying patterns and deviations. It is detected by processing and analysis of voltage and current signals acquired machinery under monitoring.

These techniques based on electrical signatures can be applied from the generator and primary source until the motor and load coupled. They may be based on: (a) invasive methods, such as the electric circuit analysis (with static analysis and non-energized machine, also referred to as offline analysis and therefore invasive), or (b) non-invasive methods, such as ESA (dynamic analysis, i.e. with the machine in operation, also referred to as online analysis) [2].

For a more comprehensive monitoring of the generator, it is important to the application of invasive and non-invasive methods, based not only on the signature electric as well as other monitoring techniques such as vibration analysis. It is recommended the application of invasive techniques in shutdowns, while non-invasive techniques should be applied periodically during the operating cycle of the machine.

This project aims to develop a methodology for the detection and dynamic analyses of online monitoring of the condition of turbo-generators based on acquisition, processing and analysis of voltage and current signals. The main fails such as short-circuit in stator and rotor windings, fails in excitement system, misalignment and eccentricity of spinning field have been studied by electric signature analysis.

The paper presents the developed system and some practical results in a TermoNorte Thermal Power-Plant, located in Porto Velho, northwest part of Brazil.

2 Electric Signature Analysis

Electric Signature Analysis (ESA) is the term used for all evaluations of voltage and current signals of electric machines. The most common analysis transforms the voltage and current signal to the frequency domain where they are analyzed. The analysis is based on two fundamental assumptions: (a) the signature of a machine with failure is different from the signature of a machine in perfect state of operation and (b) the failures are repeated with regular patterns, causing failure patterns, which can be identified and related parts of the machine.

These techniques can be applied in electric motors and generators. It is important to note that the Voltage Signature Analysis (VSA) is related to an upstream analysis, i.e. toward the generator; and the Current Signature Analysis (CSA) is related to a downstream, i.e. toward the motor. In this project, CSA and the Extended Park Vector Approach (EPVA) are the methods used in this development because they have more features applicable to electric generators. Also these methods have been applied in electric motors, but not to generators yet [2].

2.1 Overview of Current and Voltage Signature Analysis

CSA or VSA techniques are used to generate analyses and trend of electric machines. They aim to detect predictive failures in a plant, such as: problems in the stator winding, rotor problems, problems on coupling load, efficiency and loading of the system; bearing problems, among others.

It might be of surprise, but electrical signals (voltage and\or current) can carry additional information about electrical and mechanical problems of generating equipment, but the machine works as a transducer for mechanical failures, allowing the electrical signals (voltage and/or current) can carry information of electrical and mechanical problems. The signals of current and/or voltage of one (or three) phases of the machine produce, after examination, the *signature of* machine, i.e., its operating pattern. This signature is composed of frequency magnitudes of each individual component extracted from their signals of current or voltage. This fact allows the monitoring of the evolution of the frequency magnitudes, which can denote some sort of evolution of the operational conditions of the machinery.

The response that the user wants is to know whether your machine is "healthy" or not, and what part of machine is in failure.

This analysis (diagnosis) is not easily done because it involves a set of comparisons with previously stored patterns and own "history" of the machine under analysis. At this moment, usually an expert is called to produce the final diagnosis, generating command when stopping the machine. Thus, the system developed in this project for automatic diagnosis combines the history of turbo-generator, expert knowledge and failures patterns and it can be very useful for a power company.

2.2 Extended Park Vector Approach

The EPVA technique should be used to verify electric stator imbalances. However, it can only be used if the signals of voltage and/or current have been demodulated [3]. The central idea of this technique is checking failures by the distortion of Park´s circle, i.e., more distortion in the Park´s circle more is the unbalance of the machine. The current components of the Park vector are described by i_D and i_Q:

$$i_D = \left(\frac{\sqrt{2}}{\sqrt{3}}\right) i_A - \left(\frac{1}{\sqrt{6}}\right) i_B - \left(\frac{1}{\sqrt{6}}\right) i_C \quad \textbf{(1)}$$

$$i_Q = \left(\frac{1}{\sqrt{2}}\right) i_B - \left(\frac{1}{\sqrt{2}}\right) i_C \quad \textbf{(2)}$$

Where the currents i_A, i_B and i_C are the three phases. In ideal conditions:

$$i_D = \left(\frac{\sqrt{6}}{2}\right) i_M \cos(\omega t - \alpha) \quad \textbf{(3)}$$

$$i_Q = \left(\frac{\sqrt{6}}{2}\right) i_M \, sin(\omega t - \alpha) \quad \textbf{(4)}$$

For normal conditions, Park circle is centered at the origin of coordinates.

The Park circle has distortions when there are abnormal conditions of operation or when mechanical or electric failures occur. However, these distortions in the Park circle are not easy to be seen or measured, hence the proposition of the Extended PVA (EPVA), observing the spectrum module of Park vector.

The EPVA technique combines the robustness analysis of Park circle and the flexibility of spectral analysis [4]. An important feature of the Park transformation process is the fundamental component of analyzed signals is erased [5]. This fact allows the component characteristics of failure to appear with greater prominence. And more, to be a method that covers the three-phase simultaneously electric stator imbalances are also covered by this method [6].

3 Electrical Signal Processing

For the characteristic extraction of digital signals, there is a pre-conditioning process and then some paths to compute the values of each variable. Different parameters are obtained in the time domain and frequency domain. In Fig. 1, a flowchart of the used techniques is shown.

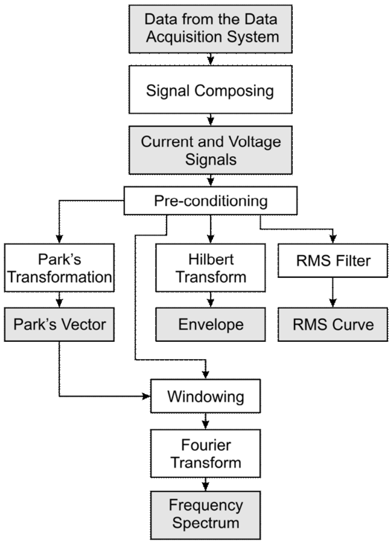


Fig. 1. Block diagram of the algorithms of signal conditioning and processing.

The grayish background blocks represent a processed signal that can be viewed or used for the characteristic extraction. The blank blocks represent an algorithm or a processing applied to digital signal. Below a brief commentary on each of the processing blocks is presented:

- Signal Composition: it converts the data from the data acquisition system to digital signals whose amplitudes represent actual values for current and voltage.

- Pre-conditioning: process that eliminates the initial part of the signal to avoid samples obtained during the transients of the filters. Then the average value of each signal whose nature is alternating is deleted.
- Park Transformation: when there is a three-phase electrical system composed by three currents (I_A, I_B and I_C) and three voltages (V_{AB}, V_{BC} and V_{CA}), the Park transformation is applied to obtain the Park vector, composed by components I_Q, I_D and I_0. In some cases, it is also used the spectrum of this vector module for electric system imbalance.
- Hilbert transformation: when applied to a signal, it returns the magnitude (envelope) and instantaneous phase.
- RMS Filter: knowing the fundamental frequency of the signal, the RMS filter returns the instantaneous RMS value of the signal during the sampling period, resulting in the so-called RMS Curve.
- Windowing: filter applied to a signal in time to reduce the effect of "leakage" in the frequency spectrum. There are several types of windowing (Blackman, Hamming, Hanning, etc.), the Blackmann windowing has been used. This window allows the identification of peaks as lobes slightly wider and less "leakage" on their side bands than other Windows.
- Fourier Transform: used to transform a time domain signal into the frequency domain, the discrete Fourier transform (DFT) returns a vector with the spectrum amplitudes and their phases. To accelerate the achievement of the DFT, we used an algorithm called FFT (Fast Fourier Transform).

For each acquired electrical signals, various parameters are computed. These parameters are used for the evaluation process and for the extraction of new features, and they are listed below.

- Average amplitude: the average value of the signal in the period under review;
- RMS amplitude: also called effective value or mean square;
- Minimum and maximum amplitude values: maximum and minimum values of amplitude in the period under review;
- Amplitude, phase and fundamental frequency: value of amplitude and frequency in Hz of the fundamental component of signal (electrical system to signal fundamental frequency is 60 Hz);
- Fundamental harmonics: multiples of the fundamental component.
- Harmonic distortion index (HDI): it indicates the significance of harmonic content when compared to the fundamental component of the signal.

4 Description of the Data Mining Algorithm

This section introduces expeditiously the data mining algorithm used to perform comparisons between the processed signals, the database signals and failures patterns. The used technique was based on the Rough Set Theory [7]. This technique aims to extract a set of rules (or conditions) from a database through two hyper-sets, called upper approximation set and lower approximation set. The set of rules contains the lower approximation set and is contained by the upper approximation set. The central idea of the algorithm is to reduce the number of elements in the upper approximation set and to increase the number of elements in the lower approximation set. In an ideal condition, these two sets would become only one set that would be the required set. This set is represented by the set of production rules.

The used algorithm [8] has six main steps, they are:

1. initialization;
2. remove equal examples;
3. remove of dispensable attributes;
4. compute the core set;
5. compute the reduce set; and
6. merge rules.

In the first step (initialization), ranks (classes) of each attribute (input or output variables) are defined, and each interval receives an identification label. This division creates a cross-linked sample space and the next step may apply (remove equal examples). All examples within a same hyper-cube are grouped into only one.

The next step is to check equal attributes or unnecessary classification (dispensable attributes). This is done in the first case by mere inspection and, in the second case, by the removal of each of them and subsequent verification of the inclusion of issues of classification.

Then, with only the essential attributes, the core of each rule set is computed. The core is formed by those attributes indispensable for that rule, and those sets of examples. Next step of forming core set consists of computing reduce set, which contain only of core attributes augmented by attributes qualifying exactly a rule. Finally, the rules are similar and the set of production rules for the classification of input signals.

In the context of the developed software for turbo-generator predictive maintenance, the algorithm and mathematical structures described were implemented and serve to extract knowledge of diagnostic data. Thus, the system is able to diagnose new cases on the basis of the knowledge extracted from previous cases. It is important to note that the procedure is transparent to the user, that is, it occurs within the computational package developed, activated by a button command in the program window itself and providing the user with the proper classification.

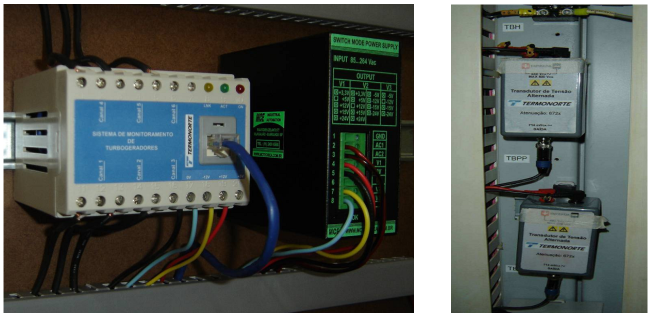
5 Illustrative Example of the Data Mining Process for Turbo-Generator Feature Extractions

The current development has been applied in the TermoNorte Thermal Power Plant, located in the Brazilian north region close to Amazon jungle. This power plant is composed by two plant in the same area. The TermoNorte I has a total generation capacity equal to 64 MW, from 4 Diesel Wärtsilä motor-generators, each one with 16

MW. The TermoNorte II has a total generation capacity equal to 340 MW, from 3 GE gas turbines.

5.1 Some Features about the Data Acquisition

The installed data acquisition system is composed by current and voltage transducers, a pre-processing acquisition module and a data acquisition module, shown in Fig. 2. This system has been entirely developed to TermoNorte Co. and very-well adapted to the severe conditions of Amazon jungle humidity. The current signals are taken from the secondary of panel CTs and the voltage signal from the panel PTs.


SWITCH MODE POWER SUPPLY
SISTEMA DE MONITORAMENTO DE TURBOGERADORES
TERMONORTE
TBH
Transdutor de Tensão Alternada
TERMONORTE
TBPP

(a) (b)

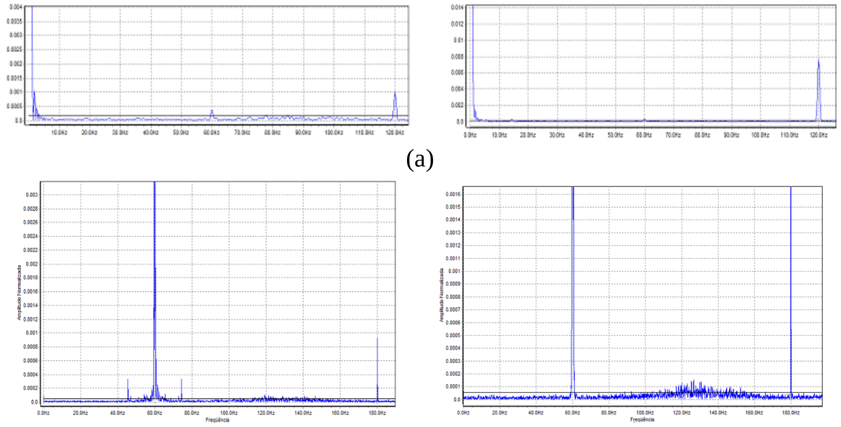
Fig. 2 Data acquisition system: (a) data acquisition pre-processing acquisition modules, and (b) voltage transducers.

5.2 Computational Package

The developed computational package is composed to two main parts: (a) data acquisition control and (b) feature extraction and signal processing. The first part contains parameters for data acquisition process. The main signal acquisitions are usually made through three acquisitions:

- Acquisition 1: it aims to collect both the signs of current and voltage (phase) of the turbine, so the EPVA techniques and energy quality are applied;
- Acquisition 2: acquisition of voltages, with the goal of applying the technique VSA (Voltage Signature Analysis);
- Acquisition 3: it does the acquisition of one of the stages for the application of the technique CSA (Current Signature Analysis).

Examples of signals acquire by the developed system are shown in Fig. 3.



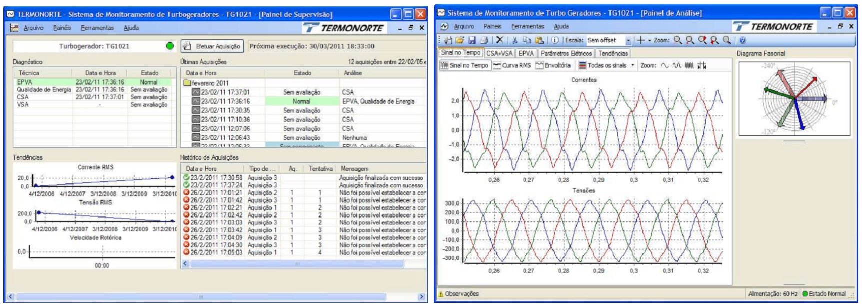
(a)

(b)

(c)

Fig. 3 Acquired Signals by: (a) EPVA to current and voltage, respectively, (b) CSA signal, and (c) VSA signal.

Special window interfaces have been developed to transfer all system control to the operator. Examples of this interface are shown in Fig. 4. These interfaces are in Portuguese language. The first figure shows an example of the supervision interface with the data acquisition control information; and the second figure is one of the analysis procedures.



Turbogerador: TG1021

Próxima execução: 30/03/2011 18:33:00

12 aquisições entre 22/02/05

Técnica	Data e Hora	Estado
EPVA	23/02/11 17:36:16	Normal
Qualidade de Energia	23/02/11 17:36:16	Sem avaliação
CSA	23/02/11 17:37:01	Sem avaliação
VSA	-	Sem avaliação

Data e Hora	Estado	Análise
fevereiro 2011		
23/02/11 17:37:01	Sem avaliação	CSA
23/02/11 17:36:16	Normal	EPVA, Qualidade de Energia
23/02/11 17:30:35	Sem avaliação	CSA
23/02/11 17:10:36	Sem avaliação	CSA
23/02/11 12:07:06	Sem avaliação	CSA
23/02/11 12:06:43	Sem avaliação	Nenhuma

Data e Hora	Tipo de ...	Aq.	Tentativa	Mensagem
23/2/2011 17:30:58	Aquisição 3			Aquisição finalizada com sucesso
23/2/2011 17:37:24	Aquisição 3			Aquisição finalizada com sucesso
26/2/2011 17:01:21	Aquisição 2	1	1	Não foi possível estabelecer a cor
26/2/2011 17:01:42	Aquisição 3	1	1	Não foi possível estabelecer a cor
26/2/2011 17:02:21	Aquisição 1	1	2	Não foi possível estabelecer a cor
26/2/2011 17:02:42	Aquisição 2	1	2	Não foi possível estabelecer a cor
26/2/2011 17:03:03	Aquisição 3	1	2	Não foi possível estabelecer a cor
26/2/2011 17:03:42	Aquisição 1	1	3	Não foi possível estabelecer a cor
26/2/2011 17:04:09	Aquisição 2	1	3	Não foi possível estabelecer a cor
26/2/2011 17:04:30	Aquisição 3	1	3	Não foi possível estabelecer a cor
26/2/2011 17:05:03	Aquisição 1	1	4	Não foi possível estabelecer a cor

Escala: Sem offset

Alimentação: 60 Hz

Estado Normal

Fig. 4. Examples of user interface of the computational package.

5.3 Feature Extraction – Data Mining Process

The described algorithm in Section 4 has been also implemented in the computational package (second part of the package in our description – for users this division of the package doesn´t exist). The signals shown in Fig. 3 are expressed by their main features, such as frequencies, amplitudes, phases, and merge to turbo-generator parameters itself. This set of data is the input data, and must be related to a type of previous operation condition: normal, abnormal, failure #1, failure #2, and so on. An example of the input signal database is shown in Table 1.

With the database the data mining process starts with the definition of labels (classes or ranks) for each attribute (input variable). The program contains a pre-set of labels for each attribute. This pre-set has been adjusted during the test phase of the prototype in the power plant. However, if the user would change the interval of these labels it is possible. However, in the daily operation, this pre-set of labels remains constant.

Table 1. Partial example of the signal acquisition database.

Acquisition	Sample Frequency (Hz)	Number of Samples	Time of Sample (s)	Spectral Definition (Hz)	Total Time (s)
1	8193	21845	2,7	0,3704	27
	1638		13,3	0,0752	37
2	8193	131072	16	0,0625	40
	1638		80	0,0125	104
3	8193	131072	16	0,0625	40
	1638		80	0,0125	104

Internally, the program merges the equal examples, verifies dispensable attributes, computes the core and the reduce sets, and finally produces the final set of rules. And academic example of this process is presented for a small database (part of the real database). Table 2 shows the data after the application of labels. Ten examples are shown with the following input attributes: frequency, amplitude, TDH (harmonic distortion level), and distortion (from Park Vector circle). The possible outputs are “normal”, “warning”, and “danger”.

Table 2. Partial example of the signal acquisition database.

Example	Frequency	Amplitude	TDH	Distortion	Output
1	Low	Normal	Normal	Normal	Normal
2	Low	Medium	Medium	Normal	Normal
3	Low	Medium	Normal	High	Normal
4	Medium	Medium	Normal	Medium	Warning
5	Medium	Medium	Normal	High	Warning
6	Medium	High	Normal	High	Danger
7	Low	Medium	Medium	Normal	Normal
8	Medium	Medium	Normal	Medium	Warning
9	High	High	Medium	Medium	Danger
10	Medium	High	Normal	Medium	Danger

After the transformation from numbers in labels of the attribute values, the second step of the algorithm can be performed - to remove equal examples. In this case, examples 2 and 7 are equal, and 4 and 8 also. Then one of them can be removed without any type of information lack, resulting in Table 3.

Table 3. Original table of examples without repetitions.

Example	Frequency	Amplitude	TDH	Distortion	Output
1	Low	Normal	Normal	Normal	Normal
2	Low	Medium	Medium	Normal	Normal
3	Low	Medium	Normal	High	Normal
4	Medium	Medium	Normal	Medium	Warning
5	Medium	Medium	Normal	High	Warning
6	Medium	High	Normal	High	Danger
9	High	High	Medium	Medium	Danger
10	Medium	High	Normal	Medium	Danger

In order to verify possible dispensable attributes, each attribute is removed and a verification of possible mistake classification is performed. In this case, for instance, the attribute “frequency” is not dispensable because without it examples 2 and 5 present two different outputs for the same input. The same occurs with the attributes “amplitude” and “distortion”. However, the attribute “TDH” is dispensable in this case because without this attribute the table remains consistent (Table 4).

Table 4. Original table of examples without repetitions and with the dispensable attributes.

Example	Frequency	Amplitude	Distortion	Output
1	Low	Normal	Normal	Normal
2	Low	Medium	Normal	Normal
3	Low	Medium	High	Normal
4	Medium	Medium	Medium	Warning
5	Medium	Medium	High	Warning
6	Medium	High	High	Danger
9	High	High	Medium	Danger
10	Medium	High	Medium	Danger

At this moment, the database is ready to compute the core set. Removing each value of each example and verifying the mistake in the classification, it is possible to computer each element of the core set. If the lack of the element creates a mistake, this element takes part of the core set, otherwise not. Table 5 presents the core set of the illustrative database example.

Table 5. Core set.

Example	Frequency	Amplitude	Distortion	Output
1	-	-	-	Normal
2	-	-	-	Normal
3	Low	-	-	Normal
4	-	Medium	Medium	Warning
5	Medium	Medium	-	Warning
6	-	High	-	Danger
9	-	-	-	Danger
10	-	High	-	Danger

Then the reduce set can be computed. It is made including the minimum number of attributes with the core to represent the example. In this case, it results in 11 examples (rules). Finally, the final set of rules is composed by 7 different rules, shown in Table 6.

Table 6. Reduce set and final set of rules.

Rule	Example	Frequency	Amplitude	Distortion	Output
1	1	Low	-	-	Normal
2	1	-	Normal	-	Normal
1	2	Low	-	-	Normal
3	2	-	Medium	Normal	Normal
1	3	Low	-	-	Normal
4	4	-	Medium	Medium	Warning
5	5	Medium	Medium	-	Warning
6	6	-	High	-	Danger
6	9	-	High	-	Danger
7	9	High	-	-	Danger
6	10	-	High	-	Danger

An example of the produced rule of the developed system is:

If $I_{PV} \geq$ -26db (0.05) then output = “warning” and failure = “stator current unbalance”.

In English language:

If the current Park Vector component is equal to or bigger than -26 db (0.05) then the operational condition is “warning” and the possible failure is “stator current unbalance”.

Rule-extraction algorithm is usually run once a quarter. The most important part of the process to the users is the analysis of the current signals, it means, the operational condition of the machine at this moment. For acquired current signals pass by the rule set and a condition of the generator is presented to the operator. The major part of the time the answer of the program is “Normal”; however, when a abnormal situation is

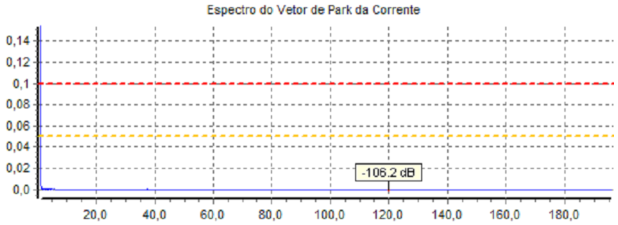


Fig. 5. Example of abnormal situation and warning and danger levels.

detected a failure pattern is shown to the operator. An example of this is presented in Fig. 5. This figure shows this abnormal situation with two pre-set lines. The yellow line expresses the warning level and the red line express a danger level. In this cases, -26 db and -20 db, respectively.

6 Conclusions

This paper shows a complete development of a supervision system with an intelligent data signal processing based on feature extraction using Rough Set Theory. The feature extraction relates processed current and voltage signals from the turbo-generator by VSA, CSA and EPVA techniques, turbo-generator electrical and mechanical parameters, and typical types of failures existing in this kind of machine.

Hardware and software have been developed to acquire and treat the electrical signals in a non-invasive process. It means, the operational condition of the generator is verified without any type of disturbance in the machine or in its control. The electrical signals are taken out of the machine, more specifically in the secondary of instrument transformers (CT and PT) in the panel control.

This system is currently in full operation at TermoNorte Thermal Power Plant, in Brazil.

References

1. Bonaldi, E.L., Oliveira, L.E.L., Lambert-Torres, G., Borges da Silva, L.E.: Proposing a Procedure for the Application of Motor Current Signature Analysis on Predictive Maintenance of Induction Motors. In: 20th Int. Cong. Exh. Condition Monitoring and Diagnosis Engineering Management, COMADEM 2007, Faro, Portugal (2007).
2. Bonaldi, E.L.: Failure Predictive Diagnostic in Three-Phase Induction Motors with MCSA and Rough Set Theory. Ph.D. Thesis, Itajuba Federal School of Engineering, Itajuba – Brazil (2006) - in Portuguese.
3. Cardoso, A.J.M.: Failure Diagnostic in Three-Phase Induction Motors. Coimbra Editora, Coimbra – Portugal (1991) - in Portuguese.
4. Benbouzid, M.H.: A Review of Induction Motors Signature Analysis as a Medium for Faults Detection. IEEE Trans. Industrial Eletronics 47, 984--993 (2000).
5. Cruz, S.M.A., Cardoso, A.J.M.: Diagnosis of the Multiple Induction Motor Faults Using Extended Park's vector Approach. Int. J. Comadem 1, 19--25 (2001).
6. Cruz, S.M.A., Cardoso, A.J.M.: Diagnosis of Stator Inter-Turn Short Circuits in DTC Induction Motor Drives. IEEE Trans. Industry Applications 40, 1349--1360 (2004).
7. Pawlak, Z.: Rough Sets. Int. J. Information and Computer Sciences .11, 341--356 (1982).
8. Rissino, S., Lambert-Torres, G.: Rough Set Theory – Fundamental Concepts, Principals, Data Extraction, and Applications. In: Ponce, J., Karahoca, A. (eds.) Data Mining and Knowledge in Real Life Applications., pp. 35--58, ISBN 978-3-902613-53-0, In-Tech Press (2009).

Fuzzy Predicting Models in “Structure – Property” Problem

Eugeny Prokhorov, Ludmila Ponomareva, Eugeny Permyakov, and Mikhail Kumskov

Department of Computational Mathematics, Faculty of Mechanics and Mathematics, Lomonosov Moscow State University **

Abstract. A new approach for analyzing the moleculedescriptor matrix for the QSAR problem (Quantitative StructureActivity Relationship) based on a fuzzy cluster structure of the learning sample is presented. The ways for generating fast rules for refusing prediction and searching the spikes in the learning sample are described. For this purpose, a special space of descriptors, simple for calculation, is introduced. The ways for optimizing the discriminant function according to fuzzy clustering parameters are examined. Highly predictive models based on the presented approach have been generated. The models are compared, and the efficiency of the described methods is revealed.

1 Introduction

The solution of the QSAR problem consists of two stages: the stage of description and the stage of discriminant function generation [3]. Very often the learning sample is separated into clusters, and each cluster is processed separately. In fact the cluster analysis of the learning sample determines the discriminant function generation. The method presented makes it possible to optimize the discriminant function with respect to clustering parameters. For screening a large database of compounds, it is extremely important to generate the rules for refusing prediction, and the rules should be fast in terms of computation. Fuzzy clustering makes it possible to remove the main disadvantages intrinsic to classical methods and to choose the discriminant function in wider, generic classes.

2 Problem statement

Detailed problem statement is given in [3]. It will specify the problem of constructing fast rejection rules. Let the alphabet of descriptors consists of M elements [2]. Feature vectors of the molecular graph G is called a vector $x = (x_1, \dots x_M) \in R^M$, where x_i – the value of the i-th descriptor computed for G. Describing the mapping $D : G \to R^M$ is called a map that assigns M–graph to his feature vector. Space R in this case is called the space of descriptors. We

** The work is supported by the grant RFFI 10-07-00694

call classifying function $F : R^M \to \{C_i\}_{i=1}^H$ that receives as an argument to a feature vector $x = (x_1, \dots x_M)$ of an arbitrary molecular graph G, and assigns the corresponding M-graph to the to one of the classes of activity C_i. Sometimes it is convenient to classifying function was defined on the set of molecular graphs. When an argument F to specify an M-graph, one should realize that F is computed on the corresponding feature vector. We set by definition $F(G_i) = F(D(G_i))$, where D - describing the mapping from the set of M-graphs in the space of descriptors.

Let the fixed algorithm Alg for constructing classifying function F on the training sample $\{(G_i, C_i)\}_{i=1}^N$, predictive model is called the set of training set $\{(G_i, C_i)\}_{i=1}^N$ and algorithm Alg construct classifying function $F = Alg(\{(G_i, C_i)\}_{i=1}^N)$.

To assess the predictive ability of models used the coefficient of cross-validation [1, 3]. Now we formulate the problem of constructing rejection rules: Rejection rule is called one or more functions $g : R^M \to \{0, 1\}$ with the following interpretation: $g(G_i) = 1$ will constitute a reject of prediction activity of this molecular graph, otherwise the prognosis can be made. Let $g(G_i) := g(D(G_i))$, where D - describing the mapping from the set of M-graphs in the space of descriptors R^M. Is called a molecular graph G_i is admissible if in accordance with accepted rejection rules, it belongs to the range of admissible argument for the classifying function F. I.e. $g(G_i) = 0$. Let $O = \{(G_i, C_i)\}_{i=1}^N$ – training sample, denoted

$$\tilde{O} = \{(G_i, C_i)\}_{i=1}^N \setminus \{(G_j, C_j) | g(G_j) = 1, j = 1, \dots N\}$$

– a sample composed only of admissible M-graphs learning sample O. We call rejection rule strong if it satisfies the inequality $R^2_{cv}(O, Alg) < R^2_{cv}(\tilde{O}, Alg)$.

3 Solution method

The idea is to use the cluster structure of the original training set for building rejection rules for this compound. This is not only the ejections that simply do not fall into one cluster, but also about the compounds, predict the activity of which should not be on the more complex reasons based on the cluster structure. For example, molecules that belong equally to two clusters, the models which predict its activity in different ways. In addition, the important point is the need to determine the admissibility of the molecular graph with minimal computational cost. It is therefore proposed to compute rejection rules on special space of descriptors, much smaller dimension than the original, for example, only topological. Thus, we construct 2 of the space of descriptors, one – to build a rejection rules, another – to do the classification and predict of activity. It arises naturally reduced (special) the matrix molecule - descriptor, whose rows are the vectors in a special space of descriptors. Regarding the fuzzy classifying function, the approach is as follows. Apply a fuzzy clustering algorithm (c-means fuzzy, or any other) [2]. Fuzzy clustering techniques, in contrast to the clear methods allow the same object simultaneously belong to multiple clusters, but with varying degrees [2, 7]. Fuzzy clustering in many situations, the more "natural" than the clear, for example, for facilities located on the border of the clusters. Fuzzy clusters describe the following matrix of the fuzzy partition:

$$S = [\mu_{ij}], \quad \mu_{ij} \in [0,1], \quad i \in \{1, \dots N\}, \quad j \in \{1, \dots k\}$$

in which the i-th row contains the degree of membership of an object to clusters $S_1, \dots S_K$.

The only difference between the matrix of the fuzzy partition and the corresponding matrix of clearly partition that, when the fuzzy partition the degree of membership of the object to the cluster takes values from the interval [0, 1], and when the clearly - from the two-element set 0, 1. Now, with the partition of the original space on the fuzzy clusters, within each construct local predictive model (we assume that this is linear regression, but can be used any other algorithm) [3, 4]. Suppose, for simplicity, we have 2 possible values of activity: active / inactive, denote their respective numerical values 1 and -1. For the new compound $\tilde{x} = (\tilde{x}_1, \dots \tilde{x}_M)$ we have k predictions of activity in accordance with the number of clusters (models). Let the i-th model gave a prediction R_i, then we can calculate the resulting prediction from the formula:

$$\tilde{y} = \frac{\sum_{i=1}^{k} R_i \mu_i}{k},$$

where μ_i – the degree of membership of the molecule to the i-th cluster. You can specify the scope of the normalization of response $\tilde{y}$, for example

$$\tilde{y} < -0.5 \Rightarrow \tilde{y} = -1,$$

$$\tilde{y} > 0.5 \Rightarrow \tilde{y} = 1,$$

otherwise $\tilde{y} = 0$ – the rejection of predict.

We now consider the optimization of fuzzy classifying function in the parameters of fuzzy clustering. In [5] described several methods for constructing the cluster structure of the training sample. We are interested in "fuzzy" generalization of the cluster structure for the application of this approach. Let discovered cluster structure of the original training set taking into account the removal of ejections. Suppose, as before, the number of clusters k, and is known for a clear partition of the matrix: $S = [\nu_{ij}], \quad \nu_{ij} \in \{0,1\}, \quad i \in \{1, \dots N\}, \quad j \in \{1, \dots k\}, \quad \sum_{j=1}^{k} \nu_{ij} = 1, i \in \{1, \dots N\}, \quad 0 < \sum_{i=1}^{N} \nu_{ij} < N, \quad j \in \{1, \dots k\}$. in which the i-th row contains information about an object $x_i = (x_{i1}, \dots x_{iM})$ belonging to one of the clusters $S_1, \dots S_k$. Assume also that each cluster is given by its center $Z_i = \{c_i 1, \dots c_{iq}\}$ – a subset of the points of the cluster S_i, center point is called the cores of the cluster, and the radius is called $r_i = max_{x_j \in S_i} \rho(x_j, Z_i)$. We construct a matrix of fuzzy partition $\tilde{S} = [\mu_{ij}]$, in which the i-th row contains the degree of membership of an object $(x_{i1}, \dots x_{iM})$ to the clusters $\tilde{S}_1, \dots \tilde{S}_k$. Optimization parameters will be $\lambda_1, \lambda_2 \in R^M, \quad \lambda_1 \leq 1 \leq \lambda_2$. We define small and large cluster $\tilde{S}_i$ radius as $r_i^1 = \lambda_1 r_i$ and $r_i^2 = \lambda_2 r_i$ respectively. Then the elements of the matrix $\tilde{S} = [\mu_{ij}]$ calculated by the formula:

$$\mu_{ij} = 1, \qquad if \quad \rho(x_i, Z_j) < r_i^1,$$

$$\mu_{ij} = 0, \qquad if \quad \rho(x_i, Z_j) > r_i^2,$$
$$\mu_{ij} = \frac{r_j^2 - \rho(x_i, Z_j)}{r_j^2 - r_j^1} \qquad otherwise.$$

Membership function of the point to cluster can also be nonlinear and contain additional optimization options. This approach allows us to meaningfully use the cluster structure of the sample is not limited in this case beyond a single cluster.

4 Results

This algorithm was implemented and applied to three samples – amber odorants, glycosides, and toxic compounds. In constructing the models used algorithm of evolutionary selection of descriptors [4], we used a set of standard clustering algorithms such as hierarchical clustering, k-means, etc Method showed significant improvement in prediction quality in the construction of simple local models. Optimization of the fuzzy classifier function by an average of 5 % improves the prognosis. In all three samples were obtained for models with high predictive ability. Along with the fast rejection rules our models can be used for subsequent screening of databases of chemical compounds [5] in order to identify compounds having the property under consideration.

5 Conclusion

These results demonstrate the practical significance of the proposed in paper approach. New methods have yielded predictive models of high quality. In most cases, a significant improvement in prediction quality as compared with classical methods. Interests are other parameterizations of fuzzy cluster structure of the training set and optimization of fuzzy classifying function for the new parameters. Continuations of the work are also testing fast rejection rules and a fuzzy classifying function in the screening of large databases of compounds with unknown activity.

References

[1] Devetyarov D.A., Grigorieva S.S., Permyakov E.A. Kumskov M.I., Ponamoreva L.A., Svitanko I.V. — Solution to the problem “structure – property” for molecules with multiple spatial conformations. // System of predicting the properties of chemical compounds: Algorithms and Models: Collected Works, Ed. MI Kumskov. Moscow: MAKS Press, 2008 (in Russian).

[2] Shtovba S.D. Introduction to the theory of fuzzy sets and fuzzy logic. Vinnitsa: Publishing the Vinnitsa State Technical University, 2001. - 198. (in Russian).

[3] Prokhorov E.I., Perevoznikov A.V., Voropaev I.D., Kumskov M.I., Ponomareva L.A. - Search representations of molecules and methods of prediction activity in the problem of “structure – property” // Reports of the 14th All-Russian Conference ”Mathematical Methods for Pattern Recognition” MMRO-2009. — Moscow: MAX Press. — 2009. — S. 589-591 (in Russian).

[4] Devetyarov D.A., Kumskov M.I. Apryshko G.N., Noseevich F.M. et al — Comparative analysis of fuzzy descriptors in solving the "structure-property" problem for a sample of glycosides // 14-th All–Russia. Conf. MMRO-14. — M.: MAKSPress, 2009. — C. 575-578 (in Russian).

[5] Prokhorov E.I., Perevoznikov A.V., Ponomarev L.A., Kumskov M.I. Neural network as a tool to implement a piecewise linear classifier for mass screening of molecules in "structure – property" problem. // Neurocomputers: development, application. — 3. — S. 39-45 (in Russian).

[6] M.I. Kumskov, E.A. Smolensky, L.A. Ponomareva, D.F. Mityushev, N.S. Zefirov. Systems of Structural Descriptors for QSAR Problem Solving// Proceedings of the Academy of Science – 1994. — Vol. 336, No. 1. — P. 64-66 (in Russian).

[7] L.A. Zadeh. Fuzzy Sets. Information and Control. — 1965. — P. 338–353.

[8] Sergei O. Kuznetsov, Mikhail V. Samokhin: Learning Closed Sets of Labeled Graphs for Chemical Applications. ILP 2005: 190-208

Handwritten Script Identification from a Bi-Script Document at Line Level using Gabor Filters

G.G. Rajput[1] and Anita H.B.[2]

[1] Department of Computer Science, Gulbarga University,
Gulbarga 585106, Karnataka, India
ggrajput@yahoo.co.in, anitahb@yahoo.com

Abstract. In a country like India where more number of scripts are in use, automatic identification of printed and handwritten script facilitates many important applications including sorting of document images and searching online archives of document images. In this paper, a Gabor feature based approach is presented to identify different Indian scripts from handwritten document images. Eight popular Indian scripts are considered here. Features are extracted from pre-processed images, consisting of portion of a line extracted manually from a handwritten document, using Gabor filters. Script classification performance is analyzed using the k-nearest neighbor classifier (KNN). Experiments are performed using five-fold cross validation method. Excellent recognition rate of 100% is achieved for data set size of 100 images for each script.

Keywords: handwritten script, multilingual documents, Gabor filters, KNN classifier.

1 Introduction

In present information technology era, document processing has become an inherent part of office automation process. Many of the documents in Indian environment are multi-script in nature. A document containing text information in more than one script is called a multi-script document. Many of the Indian documents contain two scripts, namely, the state's official language (local script) and English. An automatic script identification technique is useful to sort document images, select appropriate script-specific OCRs and search online archives of document images for those containing a particular script. Handwritten script identification is a complex task due to following reasons; complexity in pre-processing, complexity in feature extraction and classification, sensitivity of the scheme to the variation in handwritten text in document (font style, font size and document skew) and performance of the scheme. Existing script identification techniques mainly depend on various features extracted from document images at block, line or word level. Block level script identification identifies the script of the given document in a mixture of various script documents. In line based Script identification, a document image can contain more than one script but it requires the same script on a single line. Word level script identification allows the document to contain more than one script and the script of every word is identified. A brief description of the existing pieces of work at line level is given below.

To discriminate between printed text lines in Arabic and English, three techniques are presented in [1]. Firstly, an approach based on detecting the peaks in the horizontal projection profile is considered. Secondly, another approach based on the moments of the profiles using neural networks for classification is presented. Finally, approach based on classifying run length histogram using neural networks is described. An automatic scheme to identify text lines of different Indian scripts from a printed document is attempted in [2]. Features based on water reservoir principle, contour tracing, profile etc. are employed to identify the scripts. Twelve Indian scripts have been explored to develop an automatic script recognizer at text line level in [3,4]. Script recognizer has been designed to classify using the characteristics and shape based features of the script. Devanagari was discriminated through the headline feature and structural shapes were designed to discriminate English from the other Indian script. Further this has been extended with Water Reservoirs to accommodate more scripts rather than triplets. Using the combination of shape, statistical and Water Reservoirs, an automatic line-wise script identification scheme from printed documents containing five most popular scripts in the world, namely Roman, Chinese, Arabic, Devnagari and Bangla has been introduced [5]. This has been further extended to accommodate 12 different Indian scripts in the same document instead of assuming the document to contain three scripts (triplets). Here various structural features, horizontal projection profiles, Water reservoirs (top, bottom, left and right reservoirs), Contour tracing (left and right profiles) were employed as features with a decision tree classifier for script identification. In [6], a model to identify the script type of a trilingual document printed in Kannada, Hindi and English scripts is proposed. The distinct characteristic features of these scripts are thoroughly studied from the nature of the top and bottom profiles and the model is trained to learn thoroughly the distinct features of each script. Some background information about the past researches on both global based approach as well as local based approach for script identification in document images is reported in [7]. Thus, all the reported studies, accomplishing script recognition at the line level, work for printed documents. Script identification from handwritten documents is a challenging task due to large variation in handwriting as compared to printed documents. Some pieces of work of handwritten script identification of Indian scripts at block and word level can be found in the literature [8-11]. To the best of our knowledge, script identification at line level for Indian handwritten scripts has not been reported in the literature as compared to non Indian scripts [12]. This motivated us to design a robust system for Indian script identification from handwritten documents at line level for bilingual scripts. The method proposed in this paper employs analysis of portion of a line comprising at least two words, for script identification, extracted manually from the scanned document images. Consequently, the script classification task is simplified and performed faster as compared to the analysis of the entire line extracted from the handwritten document. Gabor filter bank is used for feature extraction and classification is done using KNN classifier.

2 Properties of Scripts

A brief description of the properties of the scripts considered in our study is given below. All these scripts are written from left to right.

English Script. The modern English (Roman) alphabet is a Latin-based alphabet consisting of 26 letters each of upper and lower case characters. In addition, there are some special symbols and numerals. The letters A, E, I, O, U are considered vowel letters and the remaining letters are considered consonant letters. The structure of the English alphabet contains more vertical and slant strokes.

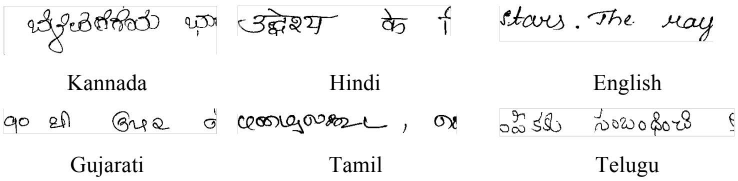
Indian Scripts. The scripts considered in this paper are Devanagari, Kannada, Tamil, Bangla, Telugu, Punjabi, and Malayalam. All the Indian languages do not have the unique scripts. Some of them use the same script. Devanagari script is used to write the languages Hindi, Bhojpuri, Marathi, Mundari, Nepali, Pali, Sanskrit, Sindhi and many more. Devanagari is recognizable by a distinctive horizontal line running along the tops of the letters that links them together. Assamese and Bangla languages are written using the Bangla script; Urdu and Kashmiri are written using the same script and Telugu and Kannada use the same script. Like Kannada and Telagu, Tamil and Malayalam belong to southern group of Dravidian languages. The Gujarati script is derived from the Devanagari script. The major difference between Gujarati and Devanagari is the lack of the top horizontal bar in Gujarati. The Gurmukhi (Punjabi) alphabet is modeled on the Landa alphabet. Similar to Devanagari script, in Gurumukhi most of the characters have a horizontal lines at the upper part called headline and primarily the characters of words in these scripts are connected by a these headlines. The image blocks of these scripts are shown in Fig.1. The details about these scripts can be found elsewhere [http://en.wikipedia.org/wiki/Languages_of_India].

3 The Proposed Method

The proposed method is inspired by the observation that in Indian context, handwritten script identification from multilingual/multi-script documents images is very promising and is still in emerging status.

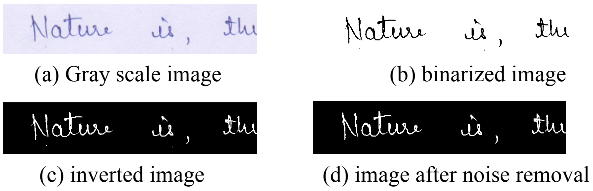
3.1 Data collection and Preprocessing

At present, standard database of handwritten Indian scripts are not available. Hence, we created our own database of handwritten documents. The document pages for the database have been collected by different persons on request under our supervision. The writers were asked to write few text lines inside A-4 size pages. Restrictions were not imposed regarding the content of the text and use of pen. Handwritten documents were written in English, Devanagari, Kannada, Tamil, Bangla, Telugu, Punjabi, and Malayalam scripts by persons belonging to different professions. The document pages were scanned at 300 dpi resolution and stored as gray scale images. The scanned image is then deskewed using the method defined in [13]. Noise is removed by applying median filter. The portion of lines of width 512 pixels and height equal to that of the height of the largest character appearing in that line were then manually cropped out from different areas of the document image, and stored as data set (Fig. 1). It should be noted that the handwritten text line (actually, portion of the line arbitrarily chosen) may contain two or more words with variable spaces between words and characters. Numerals that may appear in the text were not considered. It is ensured that at least 50% of the cropped text line contains text. A sample of line images representing different scripts is shown in Fig. 1. These lines, representing a small segment of the handwritten document images are then binarized using well known Otsu's global thresholding approach [14] (Fig. 2(b)). The binary images are then inverted so that text pixels represent value 1 and background pixels represents value 0(Fig. 2(c)). The salt and pepper noise around the boundary is removed using morphological opening. This operation also removes discontinuity at pixel level (Fig. 2(d)). However, we do not try to eliminate dots and punctuation marks appearing in the text line, since these contribute to the features of respective scripts. A total of 800 handwritten line images containing text are created, with 100 lines per scripts.


Kannada Hindi English

Gujarati Tamil Telugu

Fig. 1. Sample handwritten line images in different scripts (display in binary form).


(a) Gray scale image (b) binarized image

(c) inverted image (d) image after noise removal

Fig. 2. Pipeline process of pre-processing

3.2 Feature Extraction

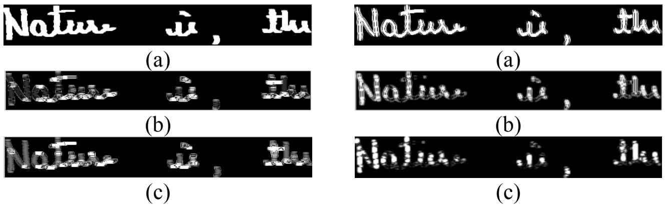
Features are the representative measures of a signal which distinguish it from other signals. A bank of Gabor filters are chosen for the task under consideration. The features are extracted by using two dimensional Gabor functions by transforming the image in time domain to the image in frequency domain. The frequency information of image is needed to see information that is not obvious in time-domain. Inherent advantages offered by Gabor function include (i) it is the only function for which the lower bound of space bandwidth product is achieved, (ii) the shapes of Gabor filters resemble the receptive field profiles of the simple cells in the visual pathway, and (iii) they are direction specific band-pass filters. Gabor filters are formed by modulating a complex sinusoid by a Gaussian function with different frequencies and orientations. A two dimensional Gabor function consists of a sinusoidal plane wave of some frequency

$$g(x,y)=\left(\frac{1}{2\pi\sigma_x\sigma_y}\right)\exp\left(-\frac{1}{2}\left(\frac{x'^2}{\sigma_x^2}+\frac{y'^2}{\sigma_y^2}\right)\right)\exp\left(2\pi jWx'\right) \qquad \mathbf{(1)}$$

$$x' = \quad x\cos\theta + y\sin\theta$$
$$y' = -x\sin\theta + y\cos\theta$$

where σ_x^2 and σ_y^2 control the spatial extent of the filter, θ is the orientation of the filter and w is the frequency of the sinusoid.

We employ two dimensional Gabor filters to extract the features from input text line image to identify the script type from a bi-script document. The preprocessed input binary image is convolved with Gabor filters considering six different orientations (0º, 30º, 60º, 90º, 120º, and 150º) and three different frequencies (a=0.125, b=0.25, c=0.5) with $\sigma_x = 2$ and $\sigma_y = 4$. The values of these parameters are fixed empirically. From the 18 output images we compute the features of dimension 54. These features are then fed to the K-NN classifier to identify the script. The feature extraction algorithm is given below (Algorithm-1). An example of Gabor filtered images for 0^0 and 30^0 degree orientations and frequencies a, b, and c is shown in Fig. 3.



(a) (a)

(b) (b)

(c) (c)

Fig. 3. Gabor filtered images for 0^0 and 30^0 degree orientations and frequencies a, b, and c, respectively

Algorithm-1

Input: Image in gray scale at line level.

Output: Feature vector

Method:

1. Apply median filter to remove noise (Fig. 2(a)).
2. Binarize the image using Otsu's method and invert the image to yield text representing binary 1 and background binary 0 (Fig. 2(b)).
3. Remove small objects around the boundary using morphological opening (Fig. 2(c)).
4. Crop the image by placing bounding box over the portion of line. And apply thinning operation (Fig. 2(d))
5. Create Gabor filter bank by considering six different orientations and three different frequencies to obtain 18 filters.
6. Convolve the input image with the created Gabor filter Bank (Fig. 3).

7. For each output image of step 6 (out of total 18), perform following steps.
   a. Extract cosine part and compute the standard deviation (18 features).
   b Extract sine part and compute the standard deviation(18 features).
   c. Compute the standard deviation of the entire output image (18 features).
   This forms feature vector of length 54.

3.3 Script Recognition

The KNN classifier is adopted for recognition purpose. This method is well-known non-parametric classifier, where posterior probability is estimated from the frequency of nearest neighbors of the unknown pattern. The key idea behind k-nearest neighbor classification is that similar observations belong to similar classes. The test image feature vector is classified to a class, to which its k-nearest neighbor belongs to. Feature vectors stored priori are used to decide the nearest neighbor of the given feature vector. The recognition process is described below.

During the training phase, features are extracted from the training set by performing feature extraction algorithm given in the feature extraction section. These features are input to KNN classifier to form a knowledge base that is subsequently used to classify the test images. During test phase, the test image, which is to be recognized, is processed in a similar way and features are computed as per the algorithm described in feature extraction section. The classifier computes the Euclidean distances between the test feature vector with that of the stored features and identifies the k-nearest neighbor. Finally, the classifier assigns the test image to a class that has the minimum distance with voting majority. The corresponding script is declared as recognized script.

3.4 Experimental Results

We evaluated the performance of the proposed bi-script identification system on a dataset of 800 pre-processed images obtained as described in section 3.1. Each bi-script document contains one Indian script and an English script. Further, we have assumed that the bi-script document contains uniscript text in the portion of line extracted for experimentation. Samples of one script are input to our system and performance is noted in terms of recognition accuracy. For each data set of 100 line images of a particular script, 60 images are used for training and remaining 40 images are used for testing. Identification of the test script is done using KNN classifier. The results were found to be optimal for k=1 as compared to other values of k. To test the robustness of the proposed method k-fold cross validation was carried out with k=5. The proposed method is implemented using Matlab 6.1 software. The recognition results of all the scripts are tabulated in Table 1 and Table 2. The results clearly shows that features extracted by using Gabor function yield very good results. The recognition accuracy of 100% (nearly) is achieved demonstrating the fact that Gabor filters provide good features for the text images at line level as compared to other methods found in the literature. However, the results obtained have certain limitation as explained below. Firstly, the process of extracting the portion of a line, ensuring that it consists of at least two words, is manual. Secondly, we have assumed that the extracted portion of the line is uniscript in text. Thirdly, as with the many other researchers, we have assumed that the documents are text only. Lastly, we need to validate our proposed system on a larger database. Experimentation is underway to take care of these limitations and propose the system in general to recognize the script at word level.

4 Conclusion

In this paper, a robust algorithm for script identification from multi script handwritten documents is presented. Gabor filters are used for feature extraction. Experiments are performed at line level by considering only a portion of the line. KNN classifier is used in recognition phase. Recognition rate of 100% is achieved. The proposed method is independent of style of hand writing. The novelty of the proposed method lies in the use of Gabor features on a portion of the line for script recognition, instead of entire line. We have assumed that such a portion of line contains uniscript text. Though this assumption is valid for many of the multi-lingual documents, in a general case we need to recognize the script at word level. Hence, our further study involves extending the proposed method for the remaining Indian scripts and also for script type identification at word level.

Table 1. Handwritten script recognition performance of Gabor filter based technique on bi-script documents

Script	% of recognition
Kannada	100%
Malayalam	100%
Punjabi	100%
Tamil	100%
Gujarati	99.92%
Telugu	100%
Hindi	99.98%

Table 2. The average recognition results in the form of confusion matrix with k-fold (k=5) cross validation for Hindi-English and Gujarati-English

Script	Hindi	English
Hindi	99.98%	0.02%
English	0%	100%

Script	Gujarati	English
Gujarati	99.92%	0.08%
English	0%	100%

Acknowledgement. The authors thank the reviewers for their helpful comments. Also, the authors are very grateful to Dr. P. S. Hiremath, Professor, Department of Computer Science, Gulbarga University, Gulbarga and Dr. Peeta Basa Pati, Bangalore, for their valuable suggestions during this work.

5 References

1. Elgammmal. A. M and Ismail. M.A, "Techniques for Language Identification for Hybrid Arabic-English Document Images", Proc. Sixth Int'l Conf. Document Analysis and Recognition, pp. 1100-1104, (2001).
2. U. Pal, S. Sinha, B. B. Chaudhuri, "Multi-Script Line identification from Indian Documents", ICDAR, Seventh International Conference on Document Analysis and Recognition (ICDAR'03) – vol. 2, pp.880 (2003).
3. Pal. U and Chaudhury.B.B, "Identification of Different Script Lines from Multi-Script Documents", Image and Vision Computing, vol. 20, no. 13-14, pp. 945-954 (2002).
4. Pal. U and B.B. Chaudhuri, "Script Line Separation From Indian Multi-Script Documents," 5th ICDAR, pp.406- 409(1999).
5. Pal U. and Chaudhuri. B. B, "Automatic identification of English, Chinese, Arabic, Devnagari and Bangla script line", Proc. 6th Intl. Conf: Document Analysis and Recognition (ICDAR'OI), pp 790-794(2001).
6. M. C. Padma and P. A. Vijaya, Script Identification From Trilingual Documents Using Profile Based Features, International Journal of Computer Science and Applications, Technomathematics Research Foundation, Vol. 7 No. 4, pp. 16 - 33 (2010).
7. S. Abirami, Dr. D. Manjula, "A Survey of Script Identification techniques for Multi-Script Document Images", International Journal of Recent Trends in Engineering, Vol. 1, No. 2, May 2009.
8. K. Roy, A. Banerjee and U. Pal, "A System for Wordwise Handwritten Script Identification for Indian Postal Automation", In Proc. IEEE India Annual Conference 2004,(INDICON-04), pp. 266-271 (2004).
9. Ram Sarkar, Nibaran Das, Subhadip Basu, Mahantapas Kundu,Mita Nasipuri and Dipak Kumar Basu, "Word level Script Identification from Bangla and Devanagri Handwritten Texts mixed with Roman Script, Journal of Computing", volume 2, Issue 2, February 2010, ISSN 2151-9617 (2010).
10. B. V. Dhandra and Mallikarjun Hangarge. Article: "Offline Handwritten Script Identification in Document Images. International Journal of Computer Applications", 4(5): 1–5, July 2010.
11. G. G. Rajput and Anita H B., "Handwritten Script Recognition using DCT and Wavelet Features at Block Level", IJCA, Special Issue on RTIPPR (3):158–163 (2010).
12. Judith Hochberg, Kevin Bowers, Michael Cannon and Patrick Keely, "Script and language identification for handwritten document images," *IJDAR,* vol.2, pp. 45-52 (1999).
13. G. G. Rajput, Anita H. B., "A Two Step Approach for Deskewing Handwritten and Machine Printed Document Images using Histograms and Geometric features", Proc. of Second Intl. Conf. on Signal and Image Processing(ICSIP-2009), Editors: D. S. Guru and T. Vasudev, pp 414-417(2009).
14. N. Otsu, "A Threshold Selection Method from Gray-Level Histogram", IEEE Transaction Systems, Man and Cybernetics, Vol 9, no.1, pp.62-66 (1979).

Image Recognition Using Kullback-Leibler Information Discrimination

Andrey Savchenko,

National Research University Higher School of Economics, B. Pecherskaya St. 25/12,
603155 Nizhniy Novgorod, Russian Federation
avsavchenko@hse.ru

Abstract. The problem of automatic image recognition based on the minimum information discrimination principle is formulated and solved. Color histograms comparison in the Kullback–Leibler information metric is proposed. It's combined with method of directed enumeration alternatives as opposed to complete enumeration of competing hypotheses. Results of an experimental study of the Kullback-Leibler discrimination in the problem of face recognition with a large database are presented. It is shown that the proposed algorithm is characterized by increased accuracy and reliability of image recognition.

Keywords: Image recognition, method of directed enumeration alternatives, Kullback-Leibler information discrimination

1 Introduction

The well-known challenging problem of the image recognition [1, 2] is processing of large image databases [3, 4, 5]. Traditional pattern recognition methods [6] based on exhaustive search can't be implemented in real-time applications. For this case a method of directed enumeration alternatives (NDEA) [3] has been proposed as opposed to the traditional method of complete enumeration of competing hypotheses [6]. MDEA can be exploited with different metrics, but the efficiency of that method highly depends on the quality of applied metric in terms of given image database [3]. Hence the choice of metric to compare images becomes quite significant [8].

One of the most perspective methods of image recognition is based on image color histogram comparison [9], [10]. The histogram-based methods are very suitable for color image recognition, because such methods are unaffected by geometrical characteristics of the images, such as translation and rotation [10]. It's shown [11] that histogram-based methods could provide quality comparable with special methods based on specific information (i.e. methods for faces recognition [12]).

In this paper, we propose a novel histogram-based method which applies minimum discrimination information (MDI) principle [13]. It is known to be an effective tool for solving various problems of pattern recognition. Meanwhile, its capabilities have not been fully exploited. In particular, almost no studies have addressed the advantages of the MDI principle over traditional methods and approaches in problems of automatic image recognition, especially, for half-tone images as one of the most complex cases in the theory and practice of pattern recognition [2, 3]. The present paper seeks to fill this gap.

The rest of the paper is organized as follows: Section 1 introduces image recognition problem and the usage of minimum discrimination information principle and Kullback-Leibler discrimination [14] to compare color histograms. In Section 3, we present metric properties of proposed decision statistics. In Section 4, we introduce new modification of MDEA [3], which can be used to reduce the amount of calculations needed for recognition. In Section 5, we present the experimental results and analyze the proposed method in the acute problem of face images recognition. Finally, concluding comments are presented in Section 6.

2. Minimum Information Discrimination Criterion

Let a set of *R>1* half-tone images $X_r = \left\| x_{uv}^r \right\|$, ($u = \overline{1,U}, v = \overline{1,V}$) be specified. Here *U* and *V* are the image height and width, respectively, $x_{uv}^r \in \left\{0,1,\ldots,x_{\max}\right\}$ is the intensity of an image point with coordinates (*u, v*); *r* is the reference number (*r = 1,…,R*), and $x_{\max}$ is the maximum intensity. It is assumed that the templates X_r define some classes of images, for example, as a method of protection against noise. Furthermore, the objects belonging to each class have some common features or similar characteristics. The common feature that unites objects in a class is called a pattern. It is required to assign a new input image $X = \left\| x_{uv} \right\|$ to one of the *R* classes. The procedures for constructing decision rules for the problem are developed in accordance with some deterministic or statistical approach. The deterministic approach is currently the most widely used. This approach seeks to determine a certain distance (measure of similarity) between any pairs of objects. For image recognition, one often uses the criterion based on the standard metric l_1:

$$\rho_1(X / X_r) = \frac{1}{U \cdot V} \sum_{u=1}^{U} \sum_{v=1}^{V} \left| x_{uv} - x_{uv}^r \right| \to \min \qquad \mathbf{(1)}$$

However, this approach does not always provide satisfactory results. The first reason is well-known [15] variability of visual patterns. The second reason is the presence of noise in the input image *X*, such as unknown intensity of light sources or simply random distortions of some points of the image. In the deterministic approach, the indicated problems are usually solved by adding new images to database, which, in turn, leads to a dramatic increase in its size. The above-mentioned difficulties are overcome by invoking a statistical approach [7, 8].

Let's consider a random variable - template X_r color of a point. It's distribution $H_r = \left[h_1^r, h_2^r, \ldots, h_{x_{\max}}^r \right]$ could be evaluated based on matrix $\left\| x_{uv}^r \right\|$

$$h_x^r = \frac{1}{U \cdot V} \sum_{u=1}^{U} \sum_{v=1}^{V} \delta\left(x_{uv} - x\right) \qquad \textbf{(2)}$$

Here $\delta(x) = \begin{cases} 1, & x = 0, \\ 0, & otherwise \end{cases}$ -discrete Dirac delta-function. H_r is often called “color histogram” [10], [11] of image X_r. The same procedure for color histogram H definition is applied for the input image *X*.

Color histograms’ comparison has widely been using in the image recognition problem [9], [10]. The common way to compare histograms is “merged histogram method” [10]

$$\sum_{x=1}^{x_{\max}} \min\left\{h_x^r, h_x\right\} \to \max_r \qquad \textbf{(3)}$$

Unfortunately, images with the same visual information, but with shifted color intensity, may significantly degrade if the conventional method of direct comparison of histograms is used. Actually, if the input image *X* is one of the images X_r from database but all pixels are decolourized, its color histogram *H* will be quite different from H_r. The common approach for solving the problem is shifting histograms [11] after their evaluation using (2). This procedure may cause the increase of decision time. Hence in this paper we use normalization of images’ intensities [16] which could be done really efficient and doesn’t influence recognition’s average time

According to statistical approach, recognizing image *X* is supposed to be a received signal of noisy communication channel where transmitted signal is one of $\{X_r\}$ image. Hence the problem is to minimize the mutual-entropy (or Kullback-Leibler information discrimination) [13] between color distributions of template image and recognizing image.

Based on this approach we assume that histogram H_r stands for the distribution of discrete certain random variable – image color. This interpretation seems justified considering the common properties of discrete distribution are valid for Hr: $\sum_{x=1}^{x_{\max}} h_x^r = 1$ (normalization condition) and $h_x^r \geq 0, x = \overline{1, x_{\max}}$ (regularization condition [13]).

Based on such image probability model, it is required to verify *R* hypotheses on the distribution H_r, $r = \overline{1, R}$ of the input image signal *X*. It’s well-known [8], that the optimal decision of the problem of statistical check of hypotheses about distribution of discrete stochastic variable in Bayesian terms is equivalent to the minimum discrimination information principle and the optimal decision rule

$$\rho_{KL}\left(X / X_r\right) = \sum_{x=1}^{x_{\max}} h_x \ln\left(h_x / h_x^r\right) \tag{4}$$

Here the statistic $\rho_{KL}\left(X / X_r\right)$ defines the Kullback–Leibler information discrimination [14] between the observed image X and its rth template from set $\{X_r\}$.

Thus, the image recognition procedure in this case involves a multichannel processing scheme in which the number of channels R is given by images' database size. Decision making is based on the statistic minimum criterion from expressions (1) or (3) for traditional image recognition methods [1] or from expression (4) when using proposed criterion and the Kullback-Leibler discrimination [3], [14].

3. Metric Properties of Information Discrimination Decision Statistics

We consider the most important and difficult case *R>>1*, where the image recognition problem is solved with a template images' set containing hundreds and thousands of images. For the specified conditions, practical implementation of the optimal decision rule (4) by the *R*-channel processing scheme encounters the obvious problem of its computational complexity and even feasibility, especially considering the labor-consuming procedure of image equalization in accordance with multiple parameters: size, color, project view, etc. The present work seeks to develop methods other than complete enumeration of reference images' set to solve the above problem. We first note the metric properties of the Kullback-Leibler discrimination decision statistic $\rho_{KL}(X / X_r) \geq 0$, which is equal to zero only in the ideal case of coincident input and template signals. Therefore, we first transform the minimum discrimination information criterion (4) to a simplified form, suitable for practical implementation [3]:

$$W_\nu(X):\ \rho_{KL}(X / X_\nu) < \rho_0 = const \tag{5}$$

Here ρ_0 is the threshold for the admissible information discrimination on the set of similarly-named images due to their known variability. The value of this threshold is easy to find experimentally by fixation of the beta error probability.

$$\beta = P\left\{\rho_{KL}(X / X_\nu) < \rho_0 \,\middle|\, \overline{W}_\nu\right\}$$

It's known [13], [17], that if recognizing image distribution is the same as for template X_ν then $2UV \cdot \rho_{KL}(X / X_\nu)$ is distributed according to the chi-square distribution with $x_{\max} - 1$ degrees of freedom. For other classes $X_r, r \neq \nu$, random variable $2UV \cdot \rho_{KL}(X / X_\nu)$ is distributed according to the noncentral chi-square distribution with $x_{\max} - 1$ degrees of freedom and noncentral parameter $\lambda = 2UV \cdot \rho_{KL}(X_r / X_\nu)$. Thus, in practice ($U \cdot V >> 1$), threshold could be determined from the following expression

$$\beta = \frac{1}{R} \sum_{r=1}^{R} H\left(\rho_0 - \min_{\nu \neq r} \rho_{KL}(X_r / X_\nu)\right) \qquad \textbf{(6)}$$

where $H(x) = \begin{cases} 0, x < 0 \\ 1, x \geq 0 \end{cases}$ is a Heaviside step function.

In fact, expression (5) defines the termination condition for the enumeration procedure using the MDI criterion (4). Thus, in decision-making process based on the minimum discrimination information principle (4), instead of looking through all templates, one needs to calculate the value of Kullback-Leibler divergence only until it becomes smaller than a certain threshold level. It is easy to see that this circumstance should reduce the amount of enumeration by 50% in average A natural development of this idea is the MDEA described below, which fully exploits the metric properties of the Kullback-Leibler decision statistic (4).

4. Method of Directed Enumeration Alternatives

Following the general computation scheme (2), (4), we reduce the image *X* recognition problem to a check of the first *N* variants $X_1, ..., X_N$ from the specified database $\{X_r\}$ subject to the condition *N*<<*R*. If, at least, one of them, namely $X_\nu, \nu \leq N$, meets the stopping requirement (5), the enumeration of the optimal solution by the minimum information discrimination criterion (4) will end with it. However, it can generally be assumed that none of the first *N* alternatives passes the check (5) in the first step. Then, it is possible to check the second group from *N* template images within the set $\{X_r\}$, then the third group, etc., until condition (5) is satisfied. There is also another, more rational, method to solve the problem in question.

Following the definition of information discrimination (4), we generate an $(R \times R)$ matrix $\mathrm{P} = \left\| \rho_{ij} \right\|$ of values

$$\rho_{ij} = \rho_{KL}(X_i / X_j), \quad i, j \leq R$$

This computationally complex operation needs to be made only once: in the preliminary computation step and for each concrete set of alternatives.

Let us arrange the images of the first control sample $X_1, ..., X_N$ in decreasing order of their information discriminations

$$\rho_{KL}(X / X_n),\ n = \overline{1, N}$$

As a result, we have an ordered (ranged) sequence of template images

$$\left\{X_{i_j}\right\} = \left\{X_{i_1}, X_{i_2} \ldots X_{i_N}\right\}, i_j \le N$$

The corresponding sequence $\{\rho_j\}$ of their discriminations $\rho_j = \rho_{KL}\left(X / X_{i_j}\right), i_j \le N,$ is a monotonically decreasing dependence $\rho_1 \ge \rho_2 \ge \ldots \ge \rho_N \ge 0$

This procedure is used to obtain a number of data $\left\{X_{i_j}\right\}$ ranged by the MDI criterion (4), and ultimately find the first local optimum X_{i_N}. The first computation step ends with this. In the second step, for the distinguished template image X_{i_N} from the matrix P, we find the set of *M*<*R* images $X_{i_N}^{(M)} = \left\{X_{i_{N+1}}, \ldots X_{i_{N+M}}\right\}$, $i_j \le R$

that are separated from the image X_{i_N} by the distance (4) not exceeding the threshold value $\rho_N = \rho\left(X / X_N\right)$:

$$\left(\forall X_i \notin X_{i_N}^{(M)}\right)\left(\forall X_j \in X_{i_N}^{(M)}\right)\ \Delta\rho\left(X_i\right) \ge \Delta\rho\left(X_j\right) \qquad \textbf{(7)}$$

Here

$$\Delta\rho(X_j) = \left|\rho\left(X_j / X_i\right) - \rho_i\right| \qquad \textbf{(8)}$$

is the deviation of the information discrimination $\rho\left(X_j / X_i\right)$ relative to the discrimination between the pair of images *X* and X_{i_N}. To this set we add one more *(M+1)*-th element $X_{i_{N+M+1}}$ that did not fall in the control sample in the previous computation step. This brings some randomness to the search procedure as a method of attaining a global optimum in a finite number of steps (computation steps). For the analysis we obtain the second control sample of template images

$$\left\{X_{i_1}, ,\ldots, X_{i_N}, \ldots X_{i_{N+M+1}}\right\}, i_j \le R$$

Next, all computations of the first step are repeated cyclically until, in some *K*th step, an element $X^{*} = X_{i_N}$ satisfies the termination condition:

$$\rho_{KL}\left(X / X^{*}\right) < \rho_0 \qquad \mathbf{(9)}$$

At this moment, the input image is within the set of the control points of the last computation step. In this case, a decision is made in favour of the closest pattern X^{*} or, at worst, after enumeration of all alternatives from the set $\{X_r\}$ but in the absence of a solution from (9), the conclusion is drawn that the input image *X* cannot be assigned to any class from template images' set and that it is necessary to switch to the decision feedback mode.

Generally, there may be a considerable gain in the total number $N + M \cdot K \le R$ of checks carried out according to (7) compared to the size of the used set of alternatives. It's explained by the fact that probability p of desired image X^{*} containing in $X_{i_N}^{(M)}$, usually exceeds the probability of belonging X^{*} to *M* alternatives for random choice

$$p = P\left\{ X^{*} \in X_{i_N}^{(M)} \right\} >> p_0 = \frac{M}{R} \qquad \mathbf{(10)}$$

This is the effect of the directed enumeration. The difference in the steps' count *K* can be explained by the depending probability p from applied metric and properties of input image and give images' set.

Thus, the system of expressions (2), (4)–(9) defines the proposed MDEA modification in the image recognition problem. This modification differs from the original method [3] in the way of evaluation $X_{i_N}^{(M)}$ set (6). In the initial version discrimination extrapolation based on autoregressive model [20] was used causing increase in the amount of calculations.

5. Experimental Results

The experiment deals with the problem of face images recognition. We use a real large database of photographs of people [21]. The photos were preliminary processed to detect faces using OpenCV library. Then detected faces were spited into 16 (4x4) parts for information discrimination computation. Such fragmentation is used to take into account heterogeneous illumination of images. The discrimination between images was calculated as a sum of discriminations (4) between these parts.

The 6000 photographs of 400 different people were selected as templates *R=900* of the most different images using standard clusterization [7]. Thus, the number of elements involved in search procedures at the next stages of the algorithms was decreased in 5 times.

In addition, 1000 more photographs of the same people were used in the test. These test images were modified by reducing the intensity of all of their points (blanking) to demonstrate the illumination's influence. In each case, the problem of image *X* recognition was solved.

In the first case the metric (1) was used and the following method parameters were chosen: *N=9* and *M=64*. The threshold $\rho_0 = 30$ was chosen experimentally. Here the exact solution X^* (the same person photograph as from input image *X*) was obtained in 90.5% of the cases. For each solution, it was required to check, in average, 51% of the total size of images database *R*.

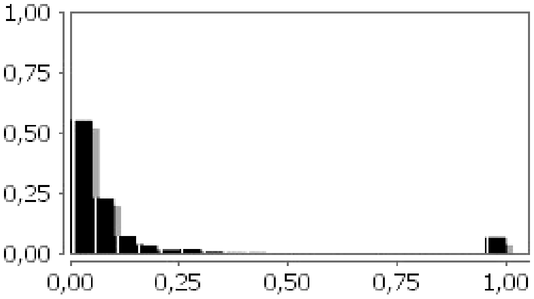


Fig. 1. Histogram of the Number of Checks per Template Images' Database Size

In the second case illuminated test images and the Kullback–Leibler metric (4) were used. Threshold $\rho_0 = 0.019$ was determined from (5) by fixation beta-error probability to 5%. Using the MDEA (2), (4)–(9) with the parameters *N, M* from the previous experiment, we obtained an average number of checks equal to 13% of *R*. A histogram of the number of checks carried out by MDEA for this case is shown in Fig. 1. With a probability of 90%, the number of template images to check does not exceed 15% of *R*. In this case, condition (4) was not satisfied for any template from the given database for 7.1% of the initial images; therefore, all *R* alternatives were checked. In 96% of the cases, the exact solution was obtained.

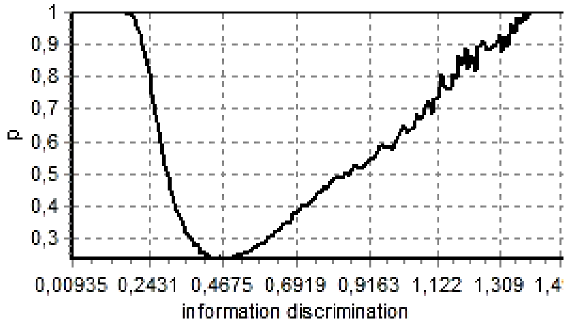


Fig. 2. Dependence of Probability $p = P\left\{X^{*} \in X_{i_N}^{(M)}\right\}$ on $\rho_{KL}\left(X / X_{i_N}\right)$.

The probability of desired image X^{*} containing in $X_{i_N}^{(M)}$ from (10) for Kullback-Leibler discrimination was equal to *p=33%* which is quite greater then $p_0 = \frac{M}{R} = \frac{64}{900} = 7\%$. Dependence of probability p on the discrimination $\rho_{KL}\left(X / X_{i_N}\right)$ is shown at Fig.2.

There is an unquestionable advantage in using the Kullback–Leibler discrimination in this problem and this is due to the fact that, in our example, the input images were artificially distorted (blanked). Let us now consider the case where all images are equally illuminated. Use of the metric l_1 from expression (1) and the directed enumeration method with a threshold $\rho_0 = 15$ gives much better results then it was for the first experiment. The error probability reduces to 6%, and the average number of checks decreases to 21% of *R*. However, the Kullback–Leibler metric (3) seems to be more efficient even in that case of equal illumination. Those metric and proposed methods show practically the same result as for the previous experiment. The error probability reduces to 3,2 with 12% of average number of checks.

For comparison, merged histogram method (3) was used. The error probability for the second experiment (non-illuminated images) was estimated to 3,1% with 11% average number of checks according to MDEA. However, in the first experiment with illuminated test images the quality of (3) was much worse – 6,5% error with 20% of average distance (3) calculation.

6. Conclusion

The problem of increasing the computation speed has attracted considerable interest of experts in both the theory and practice of image recognition. Despite huge number of approaches, most of the algorithms compare an input image with each template image, and unavoidably cannot be implemented in real-time mode for large databases. For solution of that problem directed enumeration method [3] may be used. This method is based on an information theoretical approach which uses the metric properties of the Kullback-Leibler decision statistic [3], [13] and possesses wide functional capabilities and high operational properties. The point of fundamental importance in this enumeration method is the termination criterion (4). Even in the most unprofitable case, the method almost halves the amount of computations. The use of the proposed method in the formulation (2)–(8) reduces the computational complexity by 10–20%. If proposed discrimination is combined with clustering [7] of the database, the performance increases in 40-50 times.

It's shown that efficiency of proposed method depends on the quality of applied metric in terms of given database. Thus, the metric choice becomes very important. In this paper, the new histogram-based method using minimum discrimination information principle is proposed for histogram-based image recognition. It's based on the dominant colors in images. The method is very suitable for color image recognition because it is unaffected by geometrical changes in images, such as translation and rotation. However, images with similar visual information but with shifted color intensity may result in a significant degradation in the similarity level, if the conventional histogram intersection method is used. To solve the problem, the histogram method in Kullback-Leibler metric was proposed. Our experimental results show that histogram methods (3), (4) of image recognition have significantly higher recognition effectiveness than the standard methods using l_1. metric for pixels comparison.

Our method of directed enumeration alternatives is more straightforward then traditional recognition methods used with large databases [4]. It doesn't have special requirements or extra restrictions for images to recognize. Meanwhile, the quality of the solution reached by the method is comparable to that obtained by continuous enumeration of given database [12] using special methods of faces recognition. The main advantage is that we reach the same quality with 50-times reduce of the computational complexity using proposed method.

Our further work on image recognition will continue in the following directions:

- investigating more effective halftone image models (for example, models based on gradient direction features),
- finding metric thresholds to increase recognition accuracy,
- presenting experiments with recognition for most popular image databases (FERET, Yale, AT&T, etc.).

References

1. Rui Y., Huang T., Chang S.F.: Image retrieval: current techniques, promising directions and open issues, Visual Communication and Image Representation 10, 39–62 (1999)
2. Flickner M., et al.: Query by image and video content: The QBIC system. IEEE Computer 28(9), 23–32 (1995)

3. A. V. Savchenko: Method of directed enumeration of alternatives in the problem of automatic recognition of half-tone images, Optoelectronics, Instrumentation and Data Processing 45(3), 83–91 (2009).
4. Jia Z., Amselang L., Gros P.: Content-based image retrieval from a large image database, Pattern Recognition 11(5), 1479-1495 (2008).
5. Huet B., Hancock E.R.: Shape recognition from large image libraries by inexact graph matching, Pattern Recognition Letters 20(11-13), 1259-1269 (1999).
6. Cover T.M., Hart P.E.: Nearest Neighbor Pattern Classification, IEEE Trans. Information Theory 13, 1968, 21-27
7. Theodoridis S., Koutroumbas C.: Pattern Recognition, 4th edn. Elsevier Amsterdam (2009)
8 Eickeler S., Jabs M., Rigoll G.: Comparison of Confidence Measures for Face Recognition, Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), 257-263 (2000),
9 Min R., Cheng H.D.: Effective image retrieval using dominant color descriptor and fuzzy support vector machine, Pattern Recognition 42(1), 147-157 (2009).
10. Wong K.M., Cheung C.H., Po L.M.: Dominant Color Image Retrieval using Merged Histogram. Proc. the 2003 Int. Symposium, (2003).
11. Yoo G.-H., Kim B.K., You K.S.: Content-based image retrieval using shifted histogram, ICCS, LNCS 4489, 894–897 (2007).
12. Foon N. H., Jin A. T. B., Ling D. N. C.: Face Recognition Using Wavelet Transform and Non-negative Matrix Factorization, Proc. 7th Australian Joint Conference on Artificial Intelligence, Cairns, 192-202 (2004)
13. Kullback S.: Information Theory and Statistics, Dover Pub., New York (1978).
14. Kullback S., Leibler R.A.: On information and sufficiency, Annals of Mathematical. Statistics 22, 79–86 (1951).
15. Oppenheim A. V.: Discrete-time signal processing, Prentice-Hall, (1989).
16. Zhao W., Chellappa R. ed. Face Processing: Advanced Modeling and Methods Elsevier/Academic Press, (2005)
17. Kupperman M.: Further applications of information theory to multivariate analysis and statistical inference, Dissertation, Graduate Council of George Washington University, (1957).
18. Gonsalvesh M., Papa J., Zhang B. etc: A genetic programming framework for content-based image retrieval, Pattern Recognition 42(2), 283-292 (2009).
19. Voskoboinikov Yu.E., Litvinov L.A.: Choosing the stopping iteration in iterative algorithms of image and signal reconstruction, Optoelectronics, Instrumentation and Data Processing 40(4), 3–9 (2004).
20. Marpl S L (Jr): Digital Spectral Analysis with Applications, Englewood Cliffs, N.J.: Prentice-Hall, (1987)
21. Essex Faces database, http://cswww.essex.ac.uk/mv/allfaces/index.html

Beyond Analytical Modeling, Gathering Data to Predict Real Agents' Strategic Interaction

Rustam Tagiew (tagiew@informatik.tu-freiberg.de)

Institute for Computer Science of TU Bergakademie Freiberg, Germany

Abstract. This paper presents research proposals on the interdisciplinary research infrastructure for understanding human reasoning in game-theoretic terms. Strategic reasoning impacts human decision making in social, economical and competitive interactions. The provided introduction summarizes concepts from AI, game theory and psychology. First result is a concept of interdisciplinary game description language as a part of the focused interdisciplinary research infrastructure. The need of this domain-specific language is motivated and is aimed to accelerate the current developments. As second result, the paper provides a summary of ongoing research and its significance.

1 Introduction

Different scientific disciplines predict outcomes of human social, economical and competitive interactions on the granularity level of individual decisions [1, p.4]. Autonomous intelligent systems, which perceive, decide and act in an environment according to their preferences, are called *agents* in AI, MAS and sociology. People and implemented artificial agents are called as *real agents.*

A *rational* agent makes always decisions, whose execution has according to his subjective estimation the most preferred consequences for him [2,3]. Level of intelligence impacts the quality of the subjective estimation. Rationality justifies agents' decisions and predictions of other agents' decisions. In *strategic interactions*, agents are *rational* and apply mutually and even recursively the concept *rationality.* Game theory predicts outcomes of strategic interactions [3]. Further, *game* is a notion [4] for the formal structure of a concrete strategic interaction. Agents involved in a game are called *players.*

A game in *normal form* consists of *strategies* (sets of decisions) and players' preferences over outcomes [3]. *Finite normal form games* contain a finite number of outcomes. Game theory is there to provide *equilibria* id est to solve games. An equilibrium is an irrevocable combination of players' strategies – none of the players can improve his outcome by altering his chosen strategy. If players' preferences are defined by their *payoff functions*, the equilibria of finite normal form games are guaranteed [5]. A payoff or also utility function assigns a numeric value as magnitude of preference to every outcome. Calculation of equilibria is generally NP-hard [6]. *GAMBIT* is a software library for this task [7].

In the case of players, who are intelligent enough to solve their game, an equilibrium should (immediately) occur independently of players' builds or even real identities. This is fundamentally different to the common AI approach (referenced further as *game playing*), where programs compete the last decades in playing chess e.g. still without achieving any irrevocable solution [2, chap.5]. Although the existence of an equilibrium is proven for chess, analytical abilities of none of the present game theorists suffices to show at least, whether the equilibrium implicates the win of the whites. If at least one of the players is not able to solve the game, an equilibrium is not guaranteed to occur.

Although the modern game-theoretic textbooks contain mostly fictive scenarios, the reader is asked to resist the conclusion that real-life cases can not be formalized as games. Contrary to parlor games, a concrete strategic interaction with lacking explicit rules is to be formalized as a game. Manual formalization is not guaranteed to be accurate and mostly results in far too simplified *"toy games"* as criticized in AI literature [8]. Scaling down the space of games can enable automatic formalization [9]. A player faces this problem as well as a strict outsider like a game theorist. He has to answer the question correctly, which interacting agents really exist in the environment and in what kind of game he is involved in. It is called *incomplete information* in game-theoretic terms [3], if this question can not be clarified.

Incomplete information transforms to *imperfect information*, if a probability distribution over all possible formalizations of a game can be provided [10]. Poker e.g. is a game of imperfect information, where every player is supposed to be aware about the probability distribution over the possible hands, but does not know the current hands. It is not obligatory that a game becomes a *common knowledge* (CK) among its players – it is also possible that a player bears his own variant of the game in mind, he misconceives to be involved in. As a reminder, CK is that, what everybody knows and everybody knows that everybody knows it and infinitely prepending "everybody knows that" [3].

CK of the game would not exist during a poker round, if Alice hides cards in her sleeve unknown to her opponent Bob. Alice considers that Bob's actions should conform the equilibria calculated in the original game and she calculates the equilibria of a *global game* consisting of the original game and the "cheated" game [11]. Players may have an intractable number of mutual nested believes about the details of their game. For instance, Alice could reckon to some extent that Bob also cheats, Bob could suspect Alice of considering him as a cheater and so on. An intractable number of mutual nested believes results in an intractable size of the global game, where no game-theoretic solution is guaranteed to be calculated or even to exist in the case of infinity. Therefore, cases are concentrated on, where a game of imperfect information is its players' CK.

This paper concentrates on the relevance of strategic reasoning in real agents' decision making, which causes an (immediately) occurring equilibrium, if the unbounded intelligence is assumed. In the cases of absent strategic reasoning, game theory can still predict the direction of convergence id est towards an equilibrium. This is proposed by Price [12] for prediction of stochastic processes, more precisely populations in biological terms, and is called *evolutionary game theory*

[13]. A convergence is also observed in the case of using *reinforcement learning* instead of strategic reasoning [14, e.g.]. The assumption of this paper is that the usage of the concept strategic reasoning can be extended by newer methods. This assumption does not negate the existence of reinforcement learning e.g. in real agents' strategic interactions and is discussed next in the case of people.

Although the real human preferences are a subject of philosophical discussions [15], the application of strategic reasoning assumes that they can be captured in concrete interactions as required for modeling rationality. The consideration of people as rational agents is disputed at least in psychology [16, pp.527–530], where even a scientifically accessible argumentation exposes the existence of stable and consistent human preferences as a myth [17]. Since the last six decades nevertheless, the common scientific standards for *econometric* experiments are that subjects' preferences over outcomes can be insured by paying differing amounts of money [18]. However, insuring preferences by money is criticized by the term *homo economicus* as well.

The ability of identifying other agents and of modeling their reasoning corresponds with the psychological term *ToM* (Theory of Mind) [19], which lacks almost only in the cases of autism. For application of strategic reasoning, subjects as well as researchers, who both are supposed to be non-autistic people, may be then able of modeling of others' strategic reasoning too. In Wason task at least, subjects' reasoning does not match the researchers' one though [20]. People may use no logic at all [21], but also mistake seriously in the calculus of probabilities [22].

The data of econometric experiments does not match the equilibria of games according to which they are conducted [23,24]. That means that the strategic reasoning according to the global (researchers' point of view) game does not arise among the subjects due to a set of reasons, which should be clarified. There is a need for more than only the pure analytical game theory, because even people familiar with game theory are observed to deviate from equilibria in multiple cases [24]. It is gathering and analyzing of data from experiments and field studies. This paper steps further – it proves the potential of making the interdisciplinary research on real agents' strategic interactions more efficient. Like in bio-informatics [25], it is supposed to be done by an *interdisciplinary research infrastructure* – domain specific languages and common tools.

The paper is organized as follow. Next section summarizes the main concepts for the interdisciplinary research infrastructure. Then, the section 3 presents detailed the hitherto research. At the end, the results are concluded in order to figure out the remaining construction sites.

2 Conceptualizing Interdisciplinary Research Infrastructure

A conceptualization of the already partially existing (interdisciplinary) research infrastructure for real agents' strategic interactions follows. It aims to provide an elaborate overview and an exhaustive motivation. Artificial agents are also included into consideration, although this paper concentrates mostly on people. People can be replaced by artificial agents in order to simulate or to intervene human strategic interactions. Whether simulation or intervention – in both cases, artificial agents can cut costs, allow a direct control of their builds, can be numerously deployed and are almost unlimited in period of use.

In order to avoid incomplete information, which can make the application of strategic reasoning intractable, an explicit formalization is to be used. Because a formalization of a concrete strategic interaction can be inaccurate, it is reasonable to execute it inversely. A concrete strategic interaction has to be created out of an already existing game. This is *game realization* and a software-based game realization is a *game implementation* [1, p.108]. Game realization is the almost always unmentioned action after *mechanism design* [2, p.632]. Mechanism design is inverse to game solving – adjusting of a game to already predetermined desired equilibria. If game realization is impossible, mechanism design is futile.

In real-life cases, games can be realized by physical conditions, by non-participating agents, by participating agents themselves or by a subset of the 3 previous instances [26]. If participating agents are responsible for a (partial) game realization, they should prefer the compliance with rules over the advantages from "cheating". A non-participating agent responsible for game realization can be modeled as rational too. For instance, the attorney in *prisoner's dilemma* keeps his word in order to not risk his reputation, where prisoner's dilemma is supposed to be familiar to the reader.

In the case of game implementation, the software can be divided into fractions: *game management*, game-solving algorithms, game-playing algorithms and auxiliary algorithms. Game management is the part of software, which executes the rules and calculates outcomes [27]. It may also record in order to gather data [28]. Human-computer interfaces aka *proxy agents* [29] are examples of auxiliary algorithms. Additionally, game implementation minus game-solving/playing algorithms is called *game infrastructure* [1, p.53].

In the case of explicit rules, the question about the form of games arises. The normal form is one of the two most general forms to express games in *non-co-operative game theory*. In *co-operative game theory* or also *coalitional game theory* [30], players may covenant and group into *coalitions*. The process of negotiating and the way of ensuring the agreements themselves are not issues of co-operative game theory. From non-co-operative game-theoretic point of view, disadvantages should arise for the players, who break the agreements. From both points of view, a player makes a rational decision, whether it is his own behavior or an agreement about a co-operative behavior. Both points of view are considered to be equal [3]. Due to the fact that the pure analytical game theory does not suffice, the interest is focused a more detailed form – the *extensive form* – the second most general to express games in non-co-operative game theory. In contrast to others, the extensive form captures separately the actions' sequences and their alternating subsequences in a representation known as *game tree* in AI. Therefore, the actions' sequences can be called as *root paths*. The normal form and the coalitional formalization are skipped as argued because of their higher abstraction. Also skipped is the consideration of continuous games [31], where equilibra can be acquired by solving of differential equations and no general form exit.

The theoretical appropriateness of the extensive form does not result a computational one – the problem is the inappropriate size of a games represented in extensive form. For instance, one can consider that the rules of chess are sent as a game tree via network. It is rather an *Interdisciplinary Game Description Language* (IGDL) having expressive power of the extensive form, which is needed for 4 rough categories of considered instances within and beyond game implementations. These categories are **A**) game-solving algorithms, **B**) game-playing algorithms, **C**) non-solving/playing algorithms (game management, mechanism design, auxiliaries etc.) and **D**) scientific human users (sociologists e.g.).

Less important issues are skipped out of this consideration for IGDL – issues like adjustments for *evolutionary mechanism design* [32] and for non-scientific human users. Reducing the size compared with an equivalent representation in extensive form is called further *compactness* [1, p.65]. Compactness is not the only criterion for IGDL. For the categories A–D, one can summarize the partial criteria as following:

1. **Computational speed-up.** Regularities like symmetries can be used in order to reduce the computation time of equilibria [33]. This feature is captured by formalisms called *succinct games* and should be also provided by IGDL. Reduction of computation time by using compact representation applies also to game-playing algorithms [34, e.g.] and can be considered in the case of game management.
2. **Re-usability & comparability.** A language for games forces the re-usability and also comparability of game-playing algorithms as suggested by Pell [35]. This applies also for game-solving algorithms [7] and for non-solving/playing algorithms algorithms [36].
3. **Interdisciplinary human usability.** IGDL should prevent the scientific manual game-theoretic formalization from resulting in "toy games" as criticized in AI literature [8]. A graphical representation of a game may improve the usability even more [37].
4. **Decidability.** This feature should be provided for IGDL in order to ensure that the calculation of outcomes definitely terminates [38]. This also important for game-solving/playing algorithms to be able to calculate the consequences of actions.
5. **General compact interchange format.** For interfacing instances of the categories A–D, IGDL should satisfy the need of a compact interchange format. At same time, IGDL should be as general as possible, where the facility to express n-person games of imperfect information is most general. Finally, instances of all the mentioned categories should be at least theoretically IGDL-compatible in order to skip reformatting, id est to facilitate their efficient mutual integration.
6. **Time.** Time remains the issue disregarded by the extensive form. As one can conceive by comparing game playing in fast chess and in normal chess,

time given for making decisions impacts them. Therefore, time is needed to be included in IGDL in order to ascertain the time dependent details by the explicit rules. Otherwise, the durations of actions' sequences e.g. may depend on the current game implementation and not be given explicitly in advance conjoined with the game.

Some examples for usage of IGDL can be provided. As 1st example, IGDL-compatible chess playing algorithms can be incorporated into system, which compete in playing other chess-like games described in IGDL. As 2nd example, a IGDL-based game editor can be used to allow non-computer scientists to set-up their own experiments. As 3rd example, data gathered in experiments conducted according to a game described in IGDL can be compared with the equilibria calculated by IGDL-based game-solving algorithms for the same game. As 4th example, game described in IGDL can be easily forwarded to the IGDL-based game-playing algorithms for an approximative solution, if IGDL-based game-solving algorithms fail to output timely. As 5th example (proposed by [39]), the data of the experiments conducted based on IGDL can be better compared or even stored in a central web database like the state of art in bio-informatics. Games described in a special language with and without conjoined time dependent details are called further *game descriptions*.

3 Summarizing Ongoing Research

IGDL is the desired domain-specific language, which has already some precursors and these precursors are summarized in this section. It is also possible that the ongoing research will bear different concurrent versions of IGDL. In order to assess the hitherto approaches better, a rough categorization of the used formal means is provided. Such categories are functions, graphs, logic, Petri-nets and so on. Tab.1 contains the regarded approaches and their rough categories. The ability to describe *simultaneous moves* (id est actions) is subset to more general imperfect information, because other players' actions can be unobserved only during a simultaneous execution. In *deterministic* games, it is impossible to describe a probability distribution over possible subsequences of actions.

In discrete non-co-operative game theory, there are different approaches for compact game forms, whose aim is computational speed-up [33]. The most important of them chronologically ordered are *congestion games* [40], *sequential form* [8], *graph games* [41], *local effect games* [42] and *action graph games* [43]. The software *GAMUT* can generate random games of these and other kinds, where the extensive form is included [52]. Computational speed-up of sequential form in solving 2-person-games of imperfect information is used in GAMBIT [7].

GAme LAnguage (Gala) [44] is developed in order to provide an interface to the game-solving algorithms. Factually, Gala affords Prolog-based game descriptions, where a game of extensive form or of normal form can be generated. The generated game-theoretic representation can be forwarded to GAMBIT or other game-solving algorithms. The main improvement of Gala compared to these representations is the game descriptions' compactness as perceived at least at the

Table 1. Overview of the precursors for IGDL [1, p.69]. Numbers in the **Crit.**-subrow are the satisfied criteria from the section 2, The **Used**-subrow shows the categories of used software. The **Means**-row includes the rough categories of the means used for describing games. "perfect" concerning information is to be inferred in case of missing "imperfect" and "simultaneous moves".

Approach **Citation**	**Crit.** **Used**	**Means**	**Class of games**
congestion games [40]	1,4 A,C	functions	subset of n-person games, simultaneous moves
sequential form [8]	1,4 A,C	matrices	2-person games of imperfect information
graph games [41]	1,4 A,C	graphs, functions	n-person games, simultaneous moves
local effect games [42]	1,4 A,C	functions	subset of n-person games, simultaneous moves
action graph games [43]	1,4 A,C	graphs, functions	n-person games, simultaneous moves
Gala [44]	2 A	logic	n-person games of imperfect information
MAID [45]	1,4 A,B	Bayes-nets	n-person games of imperfect information
continuous games [46]	6 A	functions	subset of 2-person games of imperfect information
timed games [47,48]	6 B	functions	2-person games
GDL [38]	1,2,4 B,C	logic	deterministic n-person games, simultaneous moves
GDL-II [49]	1,2,4,5 B,C	logic	n-person games of imperfect information
game Petri-nets [50]	1 A	Petri-nets	deterministic n-person games, simultaneous moves
PNSI [51]	2,4–6 A–C	Petri-nets	n-person games of imperfect information
SIDL2.0 [1, p.98]	2,5,6 C	logic	n-person games of imperfect information
z-Tree-language [36]	2,3,5,6 C,D	imperative language	n-person games of imperfect information

examples from Gala's software package. Due to full-scale Prolog, Gala does not provide decidability. Therefore, the generation of extensive form games from Gala game descriptions must not terminate.

There is a subset of game-playing algorithms, which is not only aimed to play games, but also to simulate human (strategic) reasoning in them. Simulating human reasoning falls in the subject of *cognitive science*. There are currently two different approaches, where the human strategic reasoning has to be expressed in a general language in order to facilitate an efficient comparability of the models. The first is based on *cognitive architectures* [53][54], which are languages for models of general human reasoning. The second is based on *Multi-Agent Influence Diagrams* (MAID) [45]. The second factually conforms the game-theoretic point of view on strategic interactions and provides the alternative language MAID for describing games. MAID are shown to be expressive enough to represent n-person games of imperfect information in the algorithms for game playing. MAID can be also transformed to extensive form in order to solve them [55].

The previously discussed literature does not mention the inclusion of time dependent details in game descriptions, what some theoretical approaches from game theory, *concurrency theory* [56] and *control engineering* [48] aim. A current work [46] in game theory extends extensive form games of perfect information and continuous time [57] to such of imperfect information – *continuous games*. In [57][46], only a subset of such games is regarded – the set of player's actions is always the same. Generally, a point of time is assigned to every action and time grows strictly over a sequence of actions.

Game Description Language (GDL) succeeded in sparking an international programing competition on general game playing [38]. A concrete game description in GDL is sent to an artificial game player and never to its human programmer – the programmer knows only the structure of GDL. This satisfies the criterion 2. For the criterion 5, generality of IGDL's precursors is a trend – either it will be possible to describe n-person games of imperfect information in IGDL or IGDL should be extended to facilitate that. This trend caused the development of GDL-II [49], which is an extension of GDL for n-person games of imperfect information. Like GDL, GDL-II is based on Datalog, which is a version of Prolog [58]. Datalog guarantees decidability by banning functions, limiting variables' ranges and restricting recursion. Due to the decidability, the existing game management algorithm based on GDL-II is guaranteed to terminate. However, the ban of functions worsen compactness. For instance, if arithmetic addition is needed to describe actions' consequences in a game, the result for every required summands' combination must be separately defined in the game description.

There are no time dependent details included explicitly in GDL-II game descriptions. GDL and GDL-II describe games in a way *STRIPS-like* [59] languages do. In languages for planning tasks, STRIPS-like descriptions can be replaced by descriptions based on Petri nets [60]. *Petri Nets for Strategic Interaction* (PNSI) is a game description language, which is proposed chronologically between GDL and GDL-II [37]. PNSI uses basic Petri nets instead of logic. Petri nets are also known being used in game theory to describe subclasses of games [50]. PNSI provides decidability [1, p.89]. The advantages of PNSI compared with GDL-II are the graphical representation of Petri nets and the ability to describe games of *equidistant time*. Equidistant time means that the game management algorithm for PNSI pauses exactly for one *chronon* between two *game states* [51], where a game state is also a node of its game tree. In this context, a chronon is a constant period of time, which is explicitly known to players. During a chronon, players' actions can be submitted. The game management algorithm for GDL-II is not of equidistant time, because the next state is calculated exactly after the submission of the last action, if it is inside the allowed time period. The time point of the last submission may vary depending on players. Of cause, GDL-II is supposed be also able to describe games of equidistant time, if its game management algorithm is modified as proposed for PNSI.

For PNSI, there exist an algorithm that can generate games of extensive form from game descriptions as in the case of Gala [51]. Therefore, PNSI provides an interface to game solving algorithms. A game of extensive form generated based on a PNSI game description is a slightly modified *state transition system* of the underlying basic Petri net. In this context, a state transition system is an oriented graph consisting of game states, where every edge represents a progression in time. There is still no algorithm for GDL-II to generate games of extensive form. A state transition system can be also generated for GDL-II game descriptions [49]. Therefore, generation of extensive form games is supposed to be also possible for GDL-II game description.

PNSI suffers of insufficient compactness as well as GDL-II but in a different way. The arithmetic addition and subtraction are banned in Datalog, but inherently supported by basic Petri nets. On the other hand, basic Petri nets require every state to be decoded as a vector of natural numbers and do not have other operations than the addition and the subtraction. The game description of the parlor game *Nim* needs in PNSI much less space than in GDL-II [1]. The opposite for chess, a GDL-II chess description needs less.

Dropping the criterion of decidability may dramatically improve compactness. *Strategic Interaction Definition Language* (SIDL) is based on ISO-Prolog, does not provide decidability and attains for example games a higher compactness [1, p.98]. Of cause, the widely used game description in an imperative language can be also mentioned. However, the game management part of software aka *game server* is then required to send its own code in order to provide explicit rules [1, p.54].

For scientific human users beyond computer science, there is an ongoing development of user-friendly software for experiments [61]. *RatImage* [62] and *TEEC* [61] are examples of the first generation of such software. They are libraries, which facilitate programming. The second generation provides already domain-specific languages for the game management and the layout of human-computer interfaces. These are *ComLabGames* [63] and *z-Tree* [36]. z-Tree is the most used [39]. There is a z-Tree-language, which is actually an imperative language, in which the game and also the human-computer interfaces can be described. This language does not provide decidability. It has no relations to game solving or playing algorithms.

4 Conclusion

A large-scale view on the problem of understanding human strategic reasoning is presented. The elaborated solution is the development of the interdisciplinary research infrastructure. This research infrastructure is proposed to make the interdisciplinary research more efficient, as it is already observed in similar interdisciplinary problems. As an underline of the large-scale view, there are matters chained from different sources, which have been never cited together before. For instance, GDL-II and z-Tree are such matters.

The elaborated concept is the domain-specific language IGDL. A summary of its precursors shows that none of these is developed enough to satisfy IGDL's full outline. There is still no language, which incorporates the graphical representation like PNSI, compactness improvements like GDL-II and a proven human usability like z-Tree-language.

References

1. R. Tagiew, Strategische Interaktion realer Agenten: Ganzheitliche Konzeptualisierung und Softwarekomponenten einer interdisziplinären Forschungsinfrastruktur, Ph.D. thesis, TU Bergakademie Freiberg (2011).
2. S. Russel, P. Norvig, Artificial Intelligence, Pearson Education, 2003.
3. M. J. Osborne, A. Rubinstein, A course in game theory, MIT Press, 1994.
4. O. Morgenstern, J. von Neumann, Theory of Games and Economic Behavior, Princeton University Press, 1944.
5. J. Nash, Non-cooperative games, Annals of Mathematics (54) (1951) 286 – 295.
6. I. Gilboa, E. Zemel, Nash and correlated equilibria, Games and Economic Behavior 1 (1989) 80–93.
7. R. D. McKelvey, A. M. McLennan, T. L. Turocy, Gambit: Software tools for game theory, version 0.2007.01.30, gambit-project.org (2007).
8. D. Koller, N. Megiddo, B. von Stengel, Fast algorithms for finding randomized strategies in game trees, in: STOC, 1994, pp. 750–759.
9. A. Barbu, S. Narayanaswamy, J. M. Siskind, Learning physically-instantiated game play through visual observation, in: ICRA, IEEE, 2010, pp. 1879–1886.
10. J. C. Harsanyi, Games with incomplete information played by bayesian players, Management Science 14 (1967) 59–182, 320–334, 486–502.
11. S. Morris, H. S. Shin, Global games: theory and applications, in: M. Dewatripont, L. Hansen, S. J. Turnovsky (Eds.), Advances in Economics and Econometrics, Cambridge University Press, 2003, Ch. 3.
12. J. Maynard Smith, G. R. Price, The logic of animal conflict, Nature 246 (1973) 15–18.
13. J. Hofbauer, K. Sigmund, The Theory of Evolution and Dynamical Systems, Cambridge University Press, 1988.
14. J. B. Pollack, A. D. Blair, Co-evolution in the successful learning of backgammon strategy, Machine Learning 32 (3) (1998) 225–240.

15. L. Stevenson, D. L. Haberman, Ten Theories of Human Nature, OUP USA, 2004.
16. M. W. Eysenck, M. T. Keane, Cognitive Psychology: A Student's Handbook, Psychology Press, 2005.
17. M. H. Bazerman, D. Malhotra, Economics wins, psychology loses, and society pays, in: D. De Cremer, M. Zeelenberg, J. K. Murnighan (Eds.), Social Psychology and Economics, Lawrence Erlbaum Associates, 2006, Ch. 14, pp. 263–280.
18. E. H. Chamberlin, An experimental imperfect market, Journal of Political Economy 56 (1948) 95–108.
19. R. Verbrugge, L. Mol, Learning to apply theory of mind, Journal of Logic, Language and Information 17 (2008) 489–511.
20. P. C. Wason, Reasoning, in: B. M. Foss (Ed.), New horizons in psychology, Penguin Books, 1966, pp. 135–151.
21. M. Oaksford, N. Chater, The probabilistic approach to human reasoning, Trends in Cognitive Sciences 5 (2001) 349–357.
22. D. Kahneman, P. Slovic, A. Tversky, Judgment Under Uncertainty: Heuristics and Biases, Cambridge University Press, 1982.
23. R. Pool, Putting game theory to the test, Science 267 (1995) 1591–1593.
24. C. F. Camerer, Behavioral Game Theory, Princeton University Press, 2003.
25. H. Nakamura, S. Date, H. Matsuda, S. Shimojo, A challenge towards next-generation research infrastructure for advanced life science, New Generation Computing 22 (2004) 157–166.
26. R. Tagiew, General game management agent, CoRR abs/0903.0353.
27. M. R. Genesereth, N. Love, B. Pell, General game playing: Overview of the aaai competition, AI Magazine 26 (2) (2005) 62–72.
28. R. Tagiew, Towards a framework for management of strategic interaction, INSTICC Press, 2009, pp. 587–590.
29. Y. Gal, A. Pfeffer, Modeling reciprocal behavior in human bilateral negotiation, in: AAAI, AAAI Press, 2007, pp. 815–820.
30. D. Ray, A Game-Theoretic Perspective on Coalition Formation, Oxford University Press, 2007.
31. R. Isaacs, Differential Games, John Wiley and Sons, 1965.
32. S. G. Phelps, Evolutionary mechanism design, Ph.D. thesis, University of Liverpool (2007).
33. C. H. Papadimitriou, The complexity of finding nash equilibria, Ch. 2, pp. 29–50.
34. S. Schiffel, M. Thielscher, Fluxplayer: A successful general game player, in: AAAI, AAAI Press, 2007, pp. 1191–1196.
35. B. Pell, Metagame: a new challenge for games and learning, in: H. van den Herik, L. Allis (Eds.), Heuristic programming in artificial intelligence 3–the third computer olympiad, Ellis-Horwood, 1992.
36. U. Fischbacher, z-Tree: Zurich toolbox for ready-made economic experiments, Experimental Economics 10 (2007) 171–178.
37. R. Tagiew, Multi-agent petri-games, in: CIMCA/IAWTIC/ISE, IEEE, 2008, pp. 130–135.
38. N. Love, T. Hinrichs, D. Haley, E. Schkufza, M. R. Genesereth, General game playing: Overview of the aaai competition, Tech. rep., Stanford University (2008).
39. S. Gächter, Improvements and future challenges for the research infrastructure in the field 'experimental economics', SSRN eLibrary Working Paper No.56.
40. R. Rosenthal, A class of games possessing pure-strategy nash equilibria, International Journal of Game Theory 2 (1973) 65–67.
41. M. Kearns, M. Littman, S. Singh, Graphical models for game theory, in: Proceedings of UAI, 2001.

42. K. Leyton-Brown, M. Tennenholtz, Local-effect games, in: IJCAI, Morgan Kaufmann, 2003.
43. N. Bhat, K. Leyton-Brown, Computing nash equilibria of action-graph games, in: Proceedings of UAI, 2004.
44. D. Koller, A. Pfeffer, Representations and solutions for game-theoretic problems, Artificial Intelligence 94 (1997) 167–215.
45. D. Koller, B. Milch, Multi-agent influence diagrams for representing and solving games, in: IJCAI, Morgan Kaufmann, 2001, pp. 1027–1034.
46. Y. Sannikov, Games with imperfectly observable actions in continuous time, Econometrica 75 (2007) 1285–1329.
47. R. Alur, D. L. Dill, A theory of timed automata, Theoretical Computer Science 126 (1994) 183–235.
48. C. J. Tomlin, J. Lygeros, S. S. Shankar, A game theoretic approach to controller design for hybrid systems, Proceedings of the IEEE 88 (2000) 949–970.
49. M. Thielscher, A general game description language for incomplete information games, in: AAAI, AAAI Press, 2010, pp. 994–999.
50. J. Clempner, Modeling shortest path games with petri nets, Internaltional Journal of Applied Mathematics and Computer Science 16 (2006) 387–397.
51. R. Tagiew, On multi-agent petri net models for computing extensive finite games, in: N. T. Nguen, R. Katarzyniak, A. Janiak (Eds.), ICCCI (SCI Volume), Vol. 244, Springer, 2009, pp. 243–254.
52. E. Nudelman, J. Wortman, Y. Shoham, K. Leyton-Brown, Run the gamut: A comprehensive approach to evaluating game-theoretic algorithms, in: AAMAS, IEEE, 2004, pp. 880–887.
53. F. E. Ritter, D. P. Wallach, Models of two-person games in ACT-R and Soar, in: ECCM, Nottingham University Press, 1998, pp. 202–203.
54. R. L. West, C. Lebiere, D. J. Bothell, Cognition and Multi-Agent Interaction, Cambridge University Press, 2006, Ch. 5.
55. D. Koller, B. Milch, Multi-agent influence diagrams for representing and solving games, Games and Economic Behavior 45 (2003) 181–221.
56. L. de Alfaro, M. Faella, T. A. Henzinger, R. Majumdar, M. Stoelinga, The element of surprise in timed games, in: CONCUR, Springer, 2003, pp. 144–158.
57. L. K. Simon, M. B. Stinchcombe, Extensive form games in continuous time: Pure strategies, Econometrica 57 (1989) 1171–1214.
58. E. Dantsin, T. Eiter, G. Gottlob, A. Voronkov, Complexity and expressive power of logic programming., in: Conference on Computational Complexity, IEEE, 1997, pp. 82–101.
59. R. Fikes, N. Nilsson, Strips: a new approach to the application of theorem proving to problem solving, Artificial Intelligence 2 (1971) 189–208.
60. D. Zhang, Planning with petri nets, in: Robotics and Automation, IEEE, 1991, pp. 769–775.
61. S. Geisler, E. Ponick, Teec: Ein java toolkit für ökonomische experimente, in: M. H. Breitner, B. Bruns, F. Lehner (Eds.), Neue Trends im E-Learning, Physica-Verlag HD, 2007, pp. 93–106.
62. K. Abbink, A. Sadrieh, Ratimage - research assistance toolbox for computer-aided human behavior experiments, Discussion Paper Serie B 325, University of Bonn, Germany (1995).
63. P. Kese, R. A. Miller, V. Prasnikar, D. Zupanic, Comlabgames, comlabgames.com (2004).

Construction and Analysis of Enzyme Centric Network of *A. thaliana* using Graph Theory

Kasthuribai Viswanathan[1], Nita Parekh[1]
[1]Center for Computational Natural Science and Bioinformatics,
International Institute of Information Technology, Hyderabad - 500032, India
kasthuribai.vpg08@research.iiit.ac.in, nita@iiit.ac.in

Abstract. Graph comparisons, quantitative characterizations, computation of topological indices, clustering and partitioning are some of the major computations of graph that have yielded valuable results in various disciplines. Motivated by the potential benefits of graph theory application on biological data, we discuss the reconstruction and analysis of enzyme centric network of *Arabidopsis thaliana* using graph theory concepts. We had earlier constructed the metabolite network of *Arabidopsis thaliana* and witnessed the scale free and small world nature of the network. Compared to metabolites, the enzymes are more conserved in and across many pathways. So the aim of constructing the enzyme centric network is to see if the network follows similar network properties of the metabolite network and to look for additional details that a metabolite network cannot reveal. The enzyme flat file from KEGG FTP is used as the data set for the reconstruction of the enzyme centric network. We examined the network to find the relationship between topological connections among enzymes and their functions during evolution. The enzyme sequences of high degree and high betweenness enzymes belonged to ancient fold class and ancestry value showing evidence that they evolved very slowly.

Keywords: metabolic pathways; graph theory; enzyme network; modularity; centrality measures.

1 Introduction

Metabolism is one of the most complex cellular processes. Connections between the biochemical reactions are represented as series of metabolic reactions which constitute the metabolic pathways. These pathways are used by researchers for the molecular evolution studies. Most studies of molecular evolution are focused on individual genes and proteins. The most interesting challenge in systems biology is constructing such biological networks and trying to interpret the hidden evolutionary details. The correlations that the researchers draw out in linear pathways are not highly visible and can emerge when analyzed as a whole network. We had previously developed a metabolite network from *Arabidopsis thaliana* pathway data using reaction files from the KEGG FTP. The metabolite network exhibited the small world, scale free nature and showed hierarchical organization. But the metabolite network is not very suitable for evolutionary analysis due to reasons like the metabolites are less conserved compared to the enzymes in pathways. Also, the number of enzymes is smaller compared to the number of metabolites enabling a closer look of the interactions.

The enzyme centric network is constructed with enzymes catalyzing the reactions as the vertices of the graph and the edge drawn between them if they one or more metabolites. Thus, an edge is drawn from an enzyme E1 to an enzyme E2 if E1 catalyzes a reaction in which compound A is the product and E2 consumes A, that is, it is a reactant in the second reaction [1]. For simplification of the network analysis, we assume that reactions are reversible and therefore each link or enzyme-enzyme relation in the network is undirected (bidirectional). For reconstruction of the enzyme centric network, the Enzyme flat file in the KEGG FTP is used as raw data. In order to include only functional relationships in the calculation of the enzyme connectivity, we excluded the 18 highly connected metabolites and co-factors. The enzyme network is then analyzed using Pajek, the network analysis tool, and various network properties such as degree, betweenness, etc. have been computed using this tool [2].

The degree of a node in a network is the number of connections or edges the node has with other nodes. The degree distribution of the *A.thaliana* enzyme network construction shows that a few nodes have high degree and most of the nodes have low degree revealing the scale free nature. It would be interesting to investigate role of high-degree nodes in the evolution of the organism. The enzyme centric network exhibits modular architecture suggesting clusters of interacting enzymes. The enzymes having high betweenness values are observed to be relating different pathways and an analysis of the corresponding gene/protein sequences can help in assessing its age and conservation across species.

2 Materials and Methods

2.1 Dataset

The KEGG database is designed in a way to facilitate understanding of higher-order protein and cellular functions using genomic and molecular information. The main reason for choosing the pathways information from KEGG is because it is open source and free to academic users and the enzyme data is available as a single flat file format for easy processing. The enzyme flat file used for the analysis has been downloaded from KEGG FTP [3]. Each enzyme has details on the genes, organisms, reactants, products, references, etc.

2.2 Construction of Enzyme Centric Network

The flowchart in Figure 1 explains the steps in the reconstruction of enzyme network starting with the enzyme flat file obtained from KEGG FTP. The enzyme flat file is a complete listing of all the known enzymes with an enzyme commission name (EC) and is delimited with "///" (Fig. 1). A total of 5391 enzymes are listed in this file. It is not an organism specific file, hence *A. thaliana* specific enzymes need to be extracted by parsing this file. This was done by first splitting this single file into 5391 separate files with enzyme name as the file name. Each of these files consisted of all the relevant information such as the enzyme name, synonyms, class, reactants, products, organisms and gene for each enzyme and references. This format was chosen to simplify computational processing and to minimize data duplication and extract only *A. thaliana* specific enzymes. Next only those enzyme files which have “ath” in the organism names list were extracted; a total of 3277 enzymes were identified to be *A. thaliana* specific and used for constructing the network.

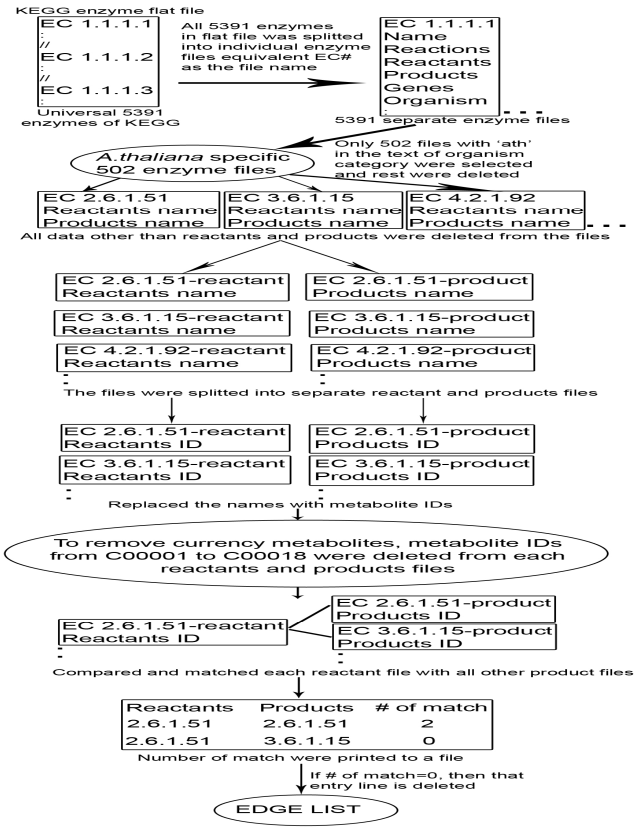

KEGG enzyme flat file
EC 1.1.1.1
//
EC 1.1.1.2
//
EC 1.1.1.3
Universal 5391 enzymes of KEGG
All 5391 enzymes in flat file was splitted into individual enzyme files equivalent EC# as the file name
EC 1.1.1.1
Name
Reactions
Reactants
Products
Genes
Organism
5391 separate enzyme files
A.thaliana specific 502 enzyme files
Only 502 files with 'ath' in the text of organism category were selected and rest were deleted
EC 2.6.1.51 Reactants name Products name
EC 3.6.1.15 Reactants name Products name
EC 4.2.1.92 Reactants name Products name
All data other than reactants and products were deleted from the files
EC 2.6.1.51-reactant Reactants name
EC 2.6.1.51-product Products name
EC 3.6.1.15-reactant Reactants name
EC 3.6.1.15-product Products name
EC 4.2.1.92-reactant Reactants name
EC 4.2.1.92-product Products name
The files were splitted into separate reactant and products files
EC 2.6.1.51-reactant Reactants ID
EC 2.6.1.51-product Products ID
EC 3.6.1.15-reactant Reactants ID
EC 3.6.1.15-product Products ID
Replaced the names with metabolite IDs
To remove currency metabolites, metabolite IDs from C00001 to C00018 were deleted from each reactants and products files
EC 2.6.1.51-product Products ID
EC 2.6.1.51-reactant Reactants ID
EC 3.6.1.15-product Products ID
Compared and matched each reactant file with all other product files
Reactants Products # of match
2.6.1.51 2.6.1.51 2
2.6.1.51 3.6.1.15 0
Number of match were printed to a file
If # of match=0, then that entry line is deleted
EDGE LIST

Fig. 1. Flow chart for the construction of enzyme-centric network.

The next step was to extract reactant and product information corresponding to every reaction catalyzed by an enzyme. Scripts were developed to automate the extraction process from each file, and a number of problems were addressed, such as removing incomplete and redundant reactants and products. The metabolite name had many synonyms, for example, NADH-glyoxylate reductase, glyoxylic acid reductase, and NADH-dependent glyoxylate reductase are all synonyms of glyoxylate reductase. So we replaced all the metabolite names with the KEGG metabolite ID and so the synonyms of the metabolite are not considered as a new metabolite to the list Fig. 1). Removing the most abundant substrates, called “currency metabolites” was the second processing step in the resonstruction of the network. Differentiating the currency metabolite from the primary metabolite is troublesome. For example, Glutamate (GLU) is a current metabolite for transferring amino groups in many reactions, but in the following reaction it is defined as a primary metabolite:

$$AKG + NH3 + NADPH = GLU + NADP^{+} + H2O$$

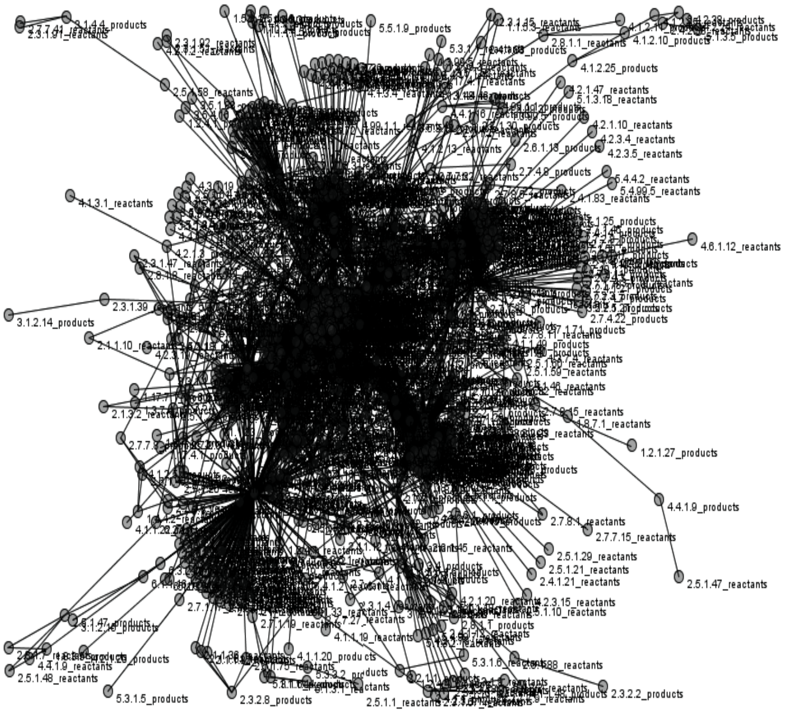


Fig. 2. The enzyme centric network of *Arabidopsis thaliana* visualized in Pajek. The circular balls represent the nodes (enzymes) and the lines represent the edges (metabolites catalyzed by the enzyme). The distribution of the connection is heterogeneous with few nodes very densely connected and few with only a single connection.

Jeong *et al.* (2000) ranked the metabolites according to their connection degree. Based on their analysis, the metabolites C00001 to C00018 (ATP, H, H20, ADP, H202, pyrophosphate, orthophosphate, CO2, NAD, glutamate, NADP, NADH, NADPH, AMP, NH3, and CoA) were labelled as currency metabolites that occurred multiple times in most of the reactions [4]. We removed these metabolites during our construction of the enzyme network. To compare enzymes that catalyze the reactants with the enzymes that catalyze products they share, we wrote perl scripts that split each enzyme file into two files, listing reactants and products respectively. and enzyme product. For example, 1.1.1.1_reactant file contains all the reactants that take part in the reaction catalyzed by enzyme 1.1.1.1. The file 1.1.1.1_product contains the products of the reaction catalyzed by 1.1.1.1. Now each enzyme product file is compared with all the enzyme reactant files to find the number of reactants and products that exactly matched.

A file containing the pair of enzymes and the number of shared reactants and products between them is listed and those that never shared any reactants or products were removed because product in the enzyme product file, say, EPi, is present in the enzyme reactant file as product, ERj, then a link is formed between the enzymes catalyzing the two reactions, Ei and Ej [5]. The edge list of the interacting enzymes is thus constructed. Each node is labeled with an EC number. The enzyme network of *Arabidopsis thaliana* thus constructed is shown in Fig. 2. From the 5391 enzyme files, the number of files drops to 507 Arabidopsis *thaliana* specific enzyme files. The enzyme network has 502 enzyme nodes. There are 4950 metabolites that utilize these enzymes in the network.

3 Results and Discussion

The analysis of the enzyme centric network constructed as discussed in the above section was carried out. It is clear from Fig. 2 that the enzyme network of *A. thaliana* is not a random network and a clear clustering of nodes is observed. In Table 1 is summarized the global properties of the network. The diameter of the network, which is the largest distance between two nodes, is 8. The low diameter reveals that the information flow between the enzymes is faster which means that the reactions are very closely connected. The average path length is 2.9. The low value of average path length means every node is close to all of the others in the network, they can reach others quickly without going through too many intermediaries.

Table 1. Global Properties of Enzyme Network

Number of Nodes	502
Number of Edges	4950
Clustering Coefficient	0.47
Diameter	8
Average path length	2.9

3.1 Scale free Nature of *Arabidopsis thaliana* Enzyme Centric Network

The degree distribution $P(k)$ gives the fraction of nodes that have degree k and is obtained by counting the number of nodes N (k) that have k = 1, 2, 3,… edges and dividing it by the total number of nodes *N*. From the degree distribution graph in Fig. 3, it is clear that the *A. thaliana* enzyme network exhibits power law behavior (Fig. 3(a)), revealed by the straight line on a logarithmic plot (Fig. 3(b)). This indicates the ‘scale free nature’ of the enzyme network [6]. That is, a high heterogeneity in the degree of the nodes is observed, critically influencing the topological properties of the network. Following Barabasi-Albert algorithm of preferential growth model, the power-law behavior of enzyme network proposes that, during evolution, new nodes tend to attach preferentially to well connect network nodes. As a consequence, the nodes having high degree are likely to be ancient enzymes.

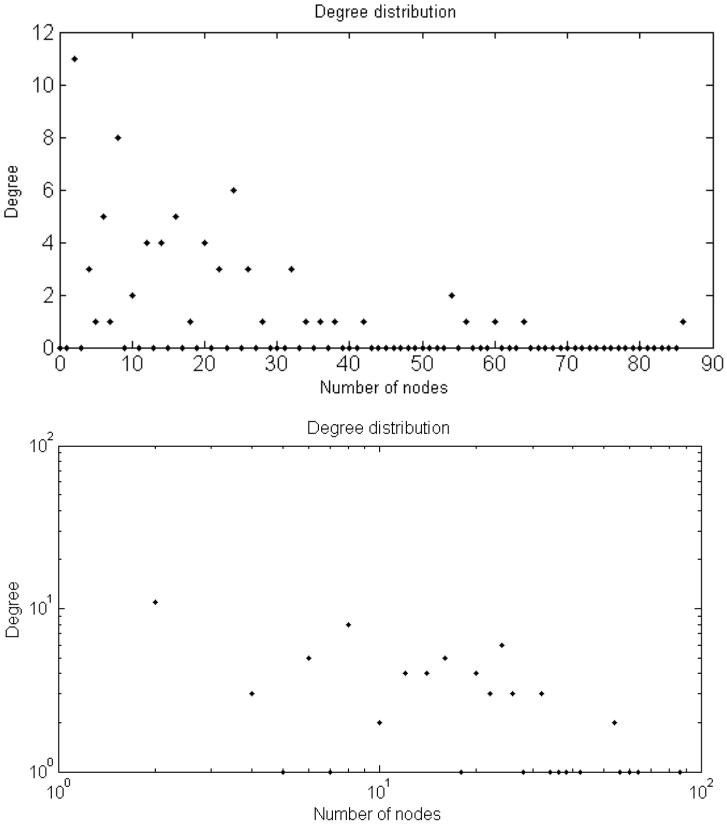


Fig. 3. Degree distribution of the enzyme network plotted both on linear and logarithmic scale. The degree distribution of the scale-free network follows the power law P (k) = Ak$^{-1.26}$, which appears as a straight line on a logarithmic plot.

3.2 Analysis of High Betweenness Enzymes

The betweenness of a node *v* is defined as the number of shortest paths going through that node:

$$C_B(v) = \frac{1}{(|V|-1)(|V|-2)} \sum_{s\neq v\neq t\in V} g_{st}(v)/g_{st}$$

where *V* is the set of nodes and |*V*| represents the number of nodes in *V*; g_{st} is the number of shortest paths from node *s* to node *t*; $g_{st}(v)$ is the number of shortest paths from node *s* to node *t* lying on node *v*. The nodes connecting pathways are critical and their removal can have a deleterious effect on the stability of the network. Such nodes can be identified by an analysis of their betweenness value. Enzymes that connect pathways do have relatively high betweenness value. This reflects the central role that such enzymes play in relaying metabolites from one enzymatic reaction to another. The differences in the amino acids of the encoded protein (nonsynonymous changes) and some, because of the degeneracy of the genetic code, leave the amino acid unchanged (synonymous or silent changes). Counting up the number of each gives us a measure of the amount of change of the sequence. Chiaoquan Qi has shown that a negative correlation between K_a/K_s *(*non-synonymous/synonymous substitutions) and betweenness of an enzyme, providing clear evidence that high-betweenness enzymes evolve slowly [7] .

In Table 2 are listed top 10 enzymes with high betweenness values. These are highly central in the network as most of the reactions are catalyzed by these enzymes. These high betweenness enzymes belong to transferases that help in transfer of a functional group, oxidoreductases that catalyzes the transfer of electrons from one molecule, and hydrolases which hydrolysis a chemical bond. Theoretically these enzymes are more important as all the pathways involve reactions that help in transferring functional groups by breaking bonds.

Table 2. Top Ten Enzymes with High Betweeness values in the Network

Enzyme	Enzyme Name	Betweenness Values	Enzyme Class
2.3.3.8	ATP citrate synthase	48571.2	Transferase
2.4.2.17	ATP phosphoribosyltransferase	47192.5	Transferase
2.3.3.8	ATP citrate synthase	46350.1	Transferase
1.2.1.41	Glutamate-5-semialdehyde dehydrogenase	34695.7	Oxidoreductases
3.6.1.1	Hydrolases	22370	Hydrolases
3.3.1.1	Adenosylhomocysteinase	18708.4	Hydrolases
3.1.3.16	Phosphoprotein phosphatase	18333.2	Hydrolases
1.11.1.6	Catalase	18316.4	Oxidoreductases
1.7.7.1	Oxidoreductases	16426.6	Oxidoreductases

3.3 Enzymes with degree one

In the enzyme centric network of *A. thaliana*, there are 136 nodes with degree one. These enzymes are catalysis just a single reaction. These are referred to as choke points. A “chokepoint reaction” is a reaction that either uniquely consumes a specific substrate or uniquely produces a specific product. Enzymes associated with high damage are involved in the production of compounds of small connectivity that connect important parts of the metabolism. Inactivation of choke points may lead to an organism's failure to produce or consume particular metabolites which could cause serious problems for fitness or survival of the organism. We listed ten degree one enzymes in the *Arabidopsis* enzyme network is shown in Table 3. We show only top ten of them for convenience and comprehensiveness.

Table 3. List of Degree One Enzymes in *Arabidopsis thaliana* Enzyme Network

Enzyme ID	Enzyme Name
1.10.2.2	ubiquinol---cytochrome-c reductase
1.1.1.1	alcohol dehydrogenase
1.15.1.1	superoxide dismutase
1.1.5.3	glycerol-3-phosphate dehydrogenase
1.2.1.27	methylmalonate-semialdehyde dehydrogenase
1.2.4.1	Pyruvate dehydrogenase
1.2.4.4	3-methyl-2-oxobutanoate dehydrogenase
1.3.3.3	coproporphyrinogen oxidase
1.3.7.4	phytochromobilin:ferredoxin Oxidoreductases
1.3.99.3	acyl-CoA dehydrogenase

3.4 Enzyme Evolution

The earliest enzymes were probably weakly catalytic and multifunctional and specific new enzymes should have evolved through gene duplication, mutation and divergence. As enzymatic pathways became more complicated, new enzymatic function could have been generated by recruitment of individual enzymes from the same or different pathways. The age of metabolic enzymes and the evolution of their metabolism with phylogenetic analysis are interesting. Since the network is scale free, it follows preferential attachment of nodes. The preferential attachment means that new nodes attach to a growing network by connecting to nodes with existing high connectivity. Nodes with high connectivity are therefore often those that have been in the network for a very long time [8]. Thus, enzymes appearing in the early stages of evolution tend to be found more frequently in different organisms (e.g. those involved in glycolysis) and that much of the metabolism of current species is based on the products of those enzymes.

This implies that evolutionarily early enzymes tend to have more connectivity to other enzymes or metabolites. The high degree enzymes in the enzyme network should therefore be highly conserved and should belong to primitive class of enzymes. In Molecular Ancestry Network database (MANET) a method the evolutionary scale of enzymes is computed using information in the Structural Classification of Proteins (SCOP) database[9]. MANET traces evolution of protein architecture in bimolecular networks by linking information in the Structural Classification of Proteins (SCOP), the Kyoto Encyclopedia of Genes and Genomes (KEGG), and phylogenetic reconstructions depicting the evolution of protein fold architecture. In MANET, the phylogenetic tree drawn for all the enzymes in KEGG and the phylogenetic distance is defined as the ancestry value. Table 4 shows the list of top ten high degree enzymes and their ancestory value predicted by MANET from a scale of (0-1), where the values closer to zero means they are older enzymes. It also shows the enzyme fold class and the PDB id of the enzyme structure. Most of the enzymes with high degree have an ancestory value closer to zero and belong to ancient fold class, proving that they are highly conserved and are older enzymes.

Table 4. Enzyme Ancestry value for the Top Ten Degree Enzymes in the Enzyme Network

Enzyme	Degree	Ancestry Value	Fold class	PDB
1.11.1.6 Catalase	113	0.0314	c.23.16.3	1SY7
2.3.3.8 ATP citrate synthase	111	0.0188	c.1.12	NA
3.5.1.5 Urease	106	0.0188	c.1.9.2	4UBP
4.2.1.52 Dihydrodipicolinate synthase	102	0.0188	c.1.10.1	1S5W
1.2.1.41 Glutamate-5-semialdehyde dehydrogenase	101	0.213	c.82.1.1	1VLU
4.1.1.31 Phosphoenolpyruvate carboxylase	101	0.0188	c.1.12.3	1QB4
4.2.1.11 Phosphopyruvate hydratase	101	0.0188	c.1.11.1	7ENL
1.9.3.1 Cytochrome-c oxidase	101	0.088	a.118.11.1	2OCC
3.1.3.16 Phosphoprotein phosphatase	101	0.088	a.118.8.1	1A17
4.2.1.9 Dihydroxy-acid dehydratase	100	0.0188	c.37.1	NA
3.6.1.1 inorganic diphosphatase	93	0.044	b.40.5.1	8PRK
2.4.2.17 ATP phosphoribosyltransferase	67	0.0125	d.58.5.3	1Q1K
3.3.1.1 Adenosylhomocysteinase	66	0.025	c.2.1.4	1V8B

4 Conclusion

Here we have reconstructed and analyzed the enzyme centric network of *Arabidopsis thaliana* using information available in the KEGG metabolic pathway data. The scale-free behavior of the enzyme network of *A. thaliana* suggests that suggests that during evolution new nodes tend to attach preferentially to a few ancient nodes. This possibly indicates that enzymes (nodes) with very high-degree are likely to be very ancient, which is further confirmed by the analysis of high degree nodes. Our preliminary analysis of betweenness centrality measure suggests that apart from high-degree nodes, nodes having high betweenness values are also very crucial for the stability of the network, and these are typically enzymes catalyzing reactions that connect pathways, or are involved in catalyzing important reactions such as those transferases that help in transfer of a functional group, oxidoreductases that catalyzes the transfer of electrons from one molecule, etc.

5 References

[1] C. R. Yang, An enzyme-centric approach for modelling non-linear biological complexity, *BMC Syst Biol*, vol. 2, p. 70, (2008).

[2] W. d. Nooy, *et al.*, *Exploratory social network analysis with Pajek*. New York: Cambridge University Press, (2005).

[3] ftp://ftp.genome.jp/pub/kegg/ligand/enzyme/enzyme, Kyoto Encyclopedia of Genes and Genomes

[4] H. Jeong, *et al.*, The large-scale organization of metabolic networks, *Nature,* vol. 407, pp. 651-4, Oct 5 (2000).

[5] S. A. Rahman and D. Schomburg, Observing local and global properties of metabolic pathways: 'load points' and 'choke points' in the metabolic networks, *Bioinformatics*, vol. 22, pp. 1767-74, Jul 15 (2006).

[6] A. L. Barabasi and E. Bonabeau, Scale-free networks, *Sci Am,* vol. 288, pp. 60-9, May (2003).

[7] Chiaoquan Qi, Genes encoding hub and bottleneck enzymes of the Arabidopsis metabolic network preferentially retain homeologs through whole genome duplication , *BMC Evolutionary Biology,* **10**:145, (2010).

[8] B. S. Hartley, Evolution of enzyme structure, *Proc R Soc Lond B Biol Sci,* vol. 205, pp. 443-52, Sep 21 (1979).

[9] H. S. Kim, *et al.*, MANET: tracing evolution of protein architecture in metabolic networks, *BMC Bioinformatics,* vol. 7, p. 351, (2006).

Author Index