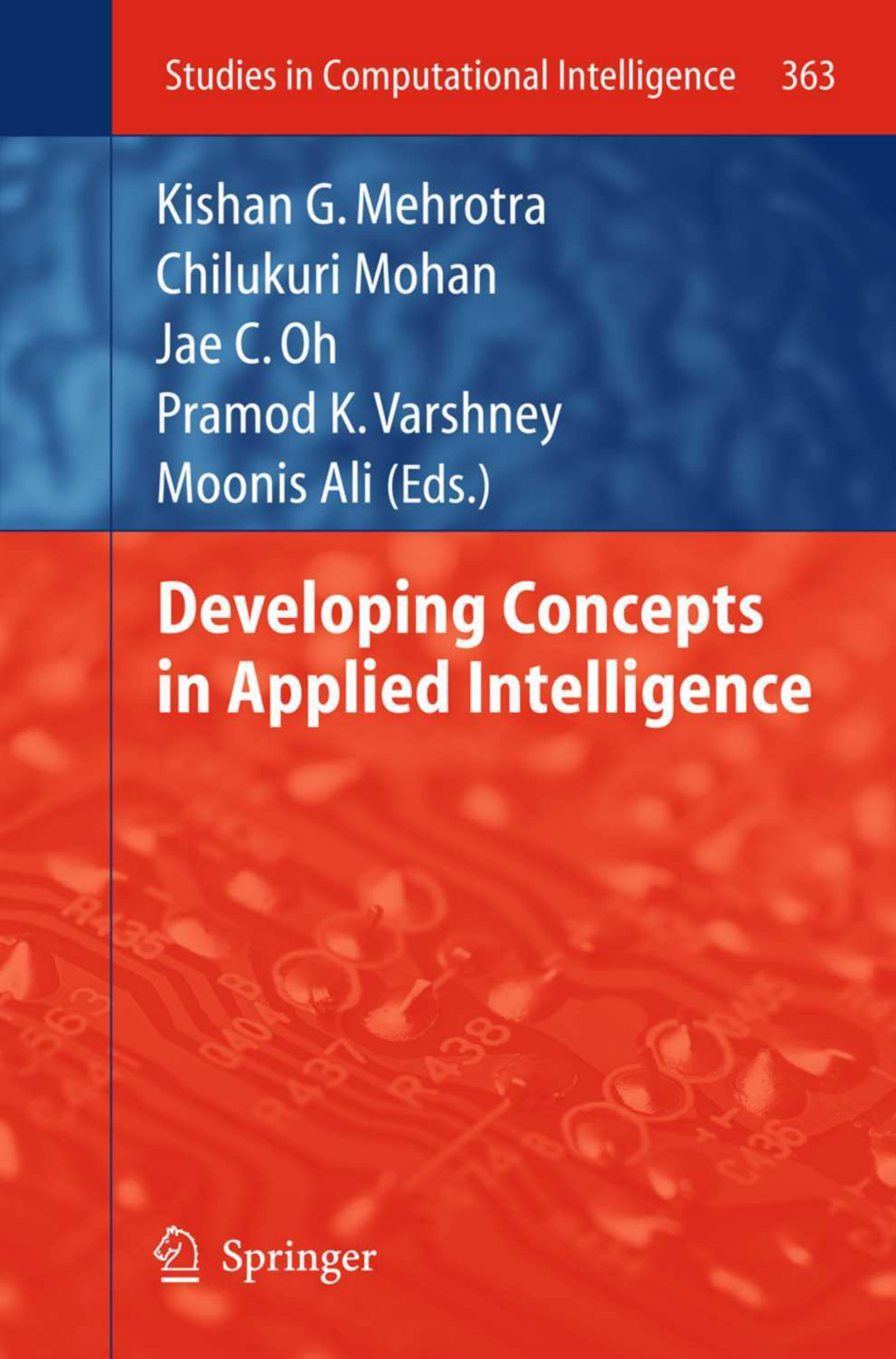
Studies in Computational Intelligence 363

Kishan G. Mehrotra
Chilukuri Mohan
Jae C. Oh
Pramod K. Varshney
Moonis Ali (Eds.)

Developing Concepts in Applied Intelligence



Springer

Kishan G. Mehrotra, Chilukuri Mohan, Jae C. Oh, Pramod K. Varshney,
and Moonis Ali (Eds.)

Developing Concepts in Applied Intelligence

Studies in Computational Intelligence, Volume 363

Further volumes of this series can be found on our homepage: springer.com

Vol. 344. Atilla Elçi, Mamadou Tadiou Koné, and Mehmet A. Orgun (Eds.)
Semantic Agent Systems, 2011
ISBN 978-3-642-18307-2

Vol. 345. Shi Yu, Léon-Charles Tranchevent, Bart De Moor, and Yves Moreau
Kernel-based Data Fusion for Machine Learning, 2011
ISBN 978-3-642-19405-4

Vol. 346. Weisi Lin, Dacheng Tao, Janusz Kacprzyk, Zhu Li, Ebroul Izquierdo, and Haohong Wang (Eds.)
Multimedia Analysis, Processing and Communications, 2011
ISBN 978-3-642-19550-1

Vol. 347. Sven Helmer, Alexandra Poulovassilis, and Fatos Xhafa
Reasoning in Event-Based Distributed Systems, 2011
ISBN 978-3-642-19723-9

Vol. 348. Beniamino Murgante, Giuseppe Borruso, and Alessandra Lapucci (Eds.)
Geocomputation, Sustainability and Environmental Planning, 2011
ISBN 978-3-642-19732-1

Vol. 349. Vitor R. Carvalho
Modeling Intention in Email, 2011
ISBN 978-3-642-19955-4

Vol. 350. Thanasis Daradoumis, Santi Caballé, Angel A. Juan, and Fatos Xhafa (Eds.)
Technology-Enhanced Systems and Tools for Collaborative Learning Scaffolding, 2011
ISBN 978-3-642-19813-7

Vol. 351. Ngoc Thanh Nguyen, Bogdan Trawiński, and Jason J. Jung (Eds.)
New Challenges for Intelligent Information and Database Systems, 2011
ISBN 978-3-642-19952-3

Vol. 352. Nik Bessis and Fatos Xhafa (Eds.)
Next Generation Data Technologies for Collective Computational Intelligence, 2011
ISBN 978-3-642-20343-5

Vol. 353. Igor Aizenberg
Complex-Valued Neural Networks with Multi-Valued Neurons, 2011
ISBN 978-3-642-20352-7

Vol. 354. Ljupco Kocarev and Shiguo Lian (Eds.)
Chaos-Based Cryptography, 2011
ISBN 978-3-642-20541-5

Vol. 355. Yan Meng and Yaochu Jin (Eds.)
Bio-Inspired Self-Organizing Robotic Systems, 2011
ISBN 978-3-642-20759-4

Vol. 356. Slawomir Koziel and Xin-She Yang (Eds.)
Computational Optimization, Methods and Algorithms, 2011
ISBN 978-3-642-20858-4

Vol. 357. Nadia Nedjah, Leandro Santos Coelho, Viviana Cocco Mariani, and Luiza de Macedo Mourelle (Eds.)
Innovative Computing Methods and their Applications to Engineering Problems, 2011
ISBN 978-3-642-20957-4

Vol. 358. Norbert Jankowski, Włodzisław Duch, and Krzysztof Grąbczewski (Eds.)
Meta-Learning in Computational Intelligence, 2011
ISBN 978-3-642-20979-6

Vol. 359. Xin-She Yang, and Slawomir Koziel (Eds.)
Computational Optimization and Applications in Engineering and Industry, 2011
ISBN 978-3-642-20985-7

Vol. 360. Mikhail Moshkov and Beata Zielosko
Combinatorial Machine Learning, 2011
ISBN 978-3-642-20994-9

Vol. 361. Vincenzo Pallotta, Alessandro Soro, and Eloisa Vargiu (Eds.)
Advances in Distributed Agent-Based Retrieval Tools, 2011
ISBN 978-3-642-21383-0

Vol. 362. Pascal Bouvry, Horacio González-Vélez, and Joanna Kolodziej (Eds.)
Intelligent Decision Systems in Large-Scale Distributed Environments, 2011
ISBN 978-3-642-21270-3

Vol. 363. Kishan G. Mehrotra, Chilukuri Mohan, Jae C. Oh, Pramod K. Varshney, and Moonis Ali (Eds.)
Developing Concepts in Applied Intelligence, 2011
ISBN 978-3-642-21331-1

Kishan G. Mehrotra, Chilukuri Mohan, Jae C. Oh,
Pramod K. Varshney, and Moonis Ali (Eds.)

Developing Concepts in Applied Intelligence

Springer

Editors

Dr. Kishan G. Mehrotra
Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse
E-mail: mehrotra@syr.edu

Dr. Chilukuri Mohan
Department of Electrical Engineering and Computer Science,
Syracuse University,
Syracuse
E-mail: ckmohan@syr.edu

Dr. Jae C. Oh
Department of Electrical Engineering and Computer Science,
Syracuse University,
Syracuse
E-mail: jcoh@ecs.syr.edu

Dr. Pramod K. Varshney
Department of Electrical Engineering and Computer Science,
Syracuse University,
Syracuse
E-mail: varshney@syr.edu

Dr. Moonis Ali
Texas State University
Department of Computer Science
San Marcos, TX 78666
USA
E-mail: ma04@txstate.edu

ISBN 978-3-642-21331-1 e-ISBN 978-3-642-21332-8

DOI 10.1007/978-3-642-21332-8

Studies in Computational Intelligence ISSN 1860-949X

Library of Congress Control Number: 2011928062

Typeset & *Cover Design:* Scientific Publishing Services Pvt. Ltd., Chennai, India.

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Preface

"Artificial Intelligence" is an area of computer science that aims to develop important concepts for simulation of human intelligence whereas "Applied Intelligence" focuses on issues involving solutions of real-life problems, arising in defense, government, management, manufacturing and many other areas which are too complex to solve through conventional approaches. The objective of the International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems (IEA/AIE) is to promote and disseminate recent research developments in Applied Intelligence. It accepts quality papers on a wide range of topics in all disciplines including science, social science, art, engineering, and finance. Held yearly, the conference brings together scientists, engineers and practitioners, who work on designing and developing applications that use intelligent techniques and apply them to variety of application domains.

This book is published in the "Studies in Computational Intelligence" series by Springer Verlag. It is comprised of eighteen chapters providing up-to-date and state-of-the-art research on the applications of Applied Intelligence techniques. Each chapter contains a short paper authored by a participant of the 24th International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems (lEA/AlE-2011). Each chapter was reviewed by at least two anonymous referees to assure the high quality.

The book covers a wide range of topics including Adaptive Learning, Belief Management, Multi-agent Systems, Genetic Algorithms, Neural Networks, and Fuzzy Modeling. This book provides useful reference values and readers will find opportunities as well as recognize challenges in the development and applications of intelligent systems.

We would like to thank Springer Verlag and Professor Janusz Kacprzyk, the editor of this series, for their help in publishing the book. We would also like to thank the reviewers for their hard work in assuring the high quality of the book. Finally, we cordially thank all the authors, who did important contributions to the book. Without their efforts, this book cannot be successfully carried out.

April 2011

Kishan G. Mehrotra
Chilukuri Mohan
Jae C. Oh
Pramod K. Varshney
Moonis Ali

Contents

An Asymmetric Criterion for Cluster Validation

Hosein Alizadeh, Behrouz Minaei, Hamid Parvin, Mohsen Moshki

School of Computer Engineering, Iran University of Science and Technology (IUST), Tehran, Iran
{halizadeh,b_minaei,parvin,moshki}@iust.ac.ir

Abstract. Many stability measures to validate a cluster have been proposed such as Normalized Mutual Information. The drawback of the common approach is discussed in this paper and then a new asymmetric criterion is proposed to assess the association between a cluster and a partition which is called Alizadeh-Parvin-Moshki-Minaei criterion, APMM. The APMM criterion compensates the drawback of the common Normalized Mutual Information (NMI) measure. Also, a clustering ensemble method is proposed which is based on aggregating a subset of primary clusters. This method uses the Average APMM as fitness measure to select a number of clusters. The clusters which satisfy a predefined threshold of the mentioned measure are selected to participate in the clustering ensemble. To combine the chosen clusters, a co-association based consensus function is employed. Since the Evidence Accumulation Clustering, EAC, method cannot derive the co-association matrix from a subset of clusters, a new EAC based method which is called Extended EAC, EEAC, is employed to construct the co-association matrix from the chosen subset of clusters. The empirical studies show that the proposed method outperforms other ones.

Keywords: Clustering Ensemble, Stability Measure, Improved Stability, Evidence Accumulation, Extended EAC, Co-association Matrix, Cluster Evaluation.

1 Introduction

Data clustering or unsupervised learning is an important and very challenging problem. The objective of clustering is to partition a set of unlabeled objects into homogeneous groups or clusters [10]. There are many applications which use clustering techniques to discover structures in data, such as data mining [10], information retrieval, image segmentation, and machine learning [12]. In real-world problems, the clusters can appear with different shapes, sizes, data sparseness, and degrees of separation. Clustering techniques require the definition of a similarity measure between patterns. On account of lack of any prior knowledge about cluster shapes, choosing a specialized clustering method is not easy [24]. Studies in the last few years have tended to combinational methods. Cluster ensemble methods attempt to find a better and more robust clustering solution by fusing information from several primary data partitionings [1].

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 1–14.
springerlink.com

Generally, there are two main steps in clustering ensemble: (a) the creation of some weak partitionings, (b) the aggregation of the obtained primary partitioning. The first step is the creation of some weak partitionings. Because every primary partitioning reveals a hidden aspect of a data, their ensemble can cover their individual drawbacks. So, the primary results are needed to be as diverse as possible to give more information about the underlying patterns in the data. Many methods have been suggested to create the necessary diversity for the primary results. To do this, using different clustering algorithms is the simplest way. Some other methods are choosing different initialization, different algorithm parameters, subset of features, mapping the data to other feature spaces [1], resampling of the data [20]. In this paper the resampling, different base algorithms, different initialization and different parameters are used to provide the necessary diversity for the primary results.

The second main step in clustering ensemble is to combine the primary partitionings obtained in the first step. The co-association matrix based aggregator is one of the most common methods to combine the primary partitionings which is employed in this paper too. EAC which is first proposed by Fred and Jain maps the individual data partitions in a clustering ensemble into a new between-patterns similarity measure, summarizing inter-pattern structures perceived from these clusterings. The final data partition is obtained by applying the single-link method to this new similarity matrix [13].

In this paper a new clustering ensemble method is proposed which uses a subset of primary clusters. A new validity measure which is called Improved Stability, IStability, is suggested to evaluate the cluster goodness. Each cluster that satisfies a threshold of IStability can be considered to participate in constructing the co-association matrix. A new method named Extended Evidence Accumulation Clustering, EEAC, is proposed to construct this matrix. Finally, a hierarchical method is applied over the obtained matrix to extract the final partition.

2 Background

The clustering ensemble which is based on a subset of selected primary clusters or partitions has a main problem which is the manner of evaluating clusters or partitions. As the data clustering is an unsupervised problem, its validation process is the most troublesome task. Baumgartner et al. in [2] have presented a resampling based technique to validate the results of exploratory fuzzy clustering analysis. Since the concept of cluster stability is introduced as a means to assess the validity of data partitionings, it has been incrementally used in the literature [14]. This idea which is based on resampling method is initially described in [5] and later generalized in different ways in [16]. Roth et al. in [24] have proposed a resampling based technique to validate a cluster. The basic element in their method which is a complementary version of the past methods is cluster stability. The stability measures the association between obtained partitions from two individual clustering algorithms. The great values of the stability measure mean that applying the clustering algorithm several times on a data set probably yields the fixed results [22]. Roth and Lange [23] have presented a new algorithm for data clustering which is based on feature selection. In their method the resampling based stability measure is used to set the algorithm

parameters. There are several cluster validation methods which are based on stability concept [19]. Ben-Hur et al. [3] have proposed a technique to exploit the stability measurements of the clustering solutions obtained by perturbing a data set. In their approach, the stability is characterized by the distribution of the pairwise similarities between clusterings obtained from sub samples of the data. First, the co-association matrix is acquired using the resampling method. Then, Jaccard coefficient is extracted from this matrix as the stability measure. Also, Estivill-Castro and Yang in [9] have offered a method by which Support Vector Machines are used to evaluate the separation of the clustering results. By filtering noise and outliers, this method can identify the robust and potentially meaningful clustering result.

Moller and Radke [21] have introduced an approach to validate a clustering results based on partition stability. This method uses a perturbation which is produced by adding some noise to the data. An empirical study robustly indicates that the perturbation usually outperforms bootstrapping and subsampling. Whereas the empirical choice of the subsampling size is often difficult [7], the choosing of the perturbation strength is not so crucial. This method uses a Nearest Neighbor Resampling approach (NNR) that offers a solution to both problems of information loss and empirical control of the change degree made to the original data. The NNR techniques were first used for time series analysis [4]. Inokuchi et al. [17] have proposed a kernelized validity measures where a kernel means the kernel function used in support vector machines. Two measures are considered in this measure. One is the sum of the traces of the fuzzy covariances within clusters and the second is a kernelized Xie-Beni's measure [26]. This validity measure is applied to the determination of the number of clusters and also the evaluation of robustness of different partitionings. Das and Sil [6] have proposed a method to determine the number of clusters which validates the clusters using splitting and merging technique in order to obtain optimal set of clusters.

Fern and Lin [11] have suggested a clustering ensemble approach which selects a subset of solutions to form a smaller but better performing cluster ensemble than using all primary solutions. The ensemble selection method is designed based on quality and diversity, the two factors that have been shown to influence the cluster ensemble performance. This method attempts to select a subset of primary partitions that simultaneously has both the highest quality and diversity. The Sum of Normalized Mutual Information, SNMI [25], [12] and [13], is used to measure the quality of individual partition with respect to other ones. Also, the Normalized Mutual Information, NMI, is employed to measure the diversity between partitions. Although the ensemble size in their method is relatively small, this method can achieve a significant performance improvement over full ensembles. Law et al. propose a multi objective data clustering method based on the selection of individual clusters produced by several clustering algorithms, through an optimization procedure [18]. This technique chooses the best set of objective functions for different parts of the feature space from the results of base clustering algorithms. Fred and Jain [14] have offered a new clustering ensemble method that learns the pairwise similarity between points in order to facilitate a proper partitioning of the data without the a priori knowledge of the number of clusters and of the shape of the clusters. This method

which is based on cluster stability evaluates the primary clustering results instead of final clustering.

3 Proposed Clustering Ensemble

In this section, first the proposed clustering ensemble method is briefly outlined, and then its phases are described in the subsequent subsections in more detail.

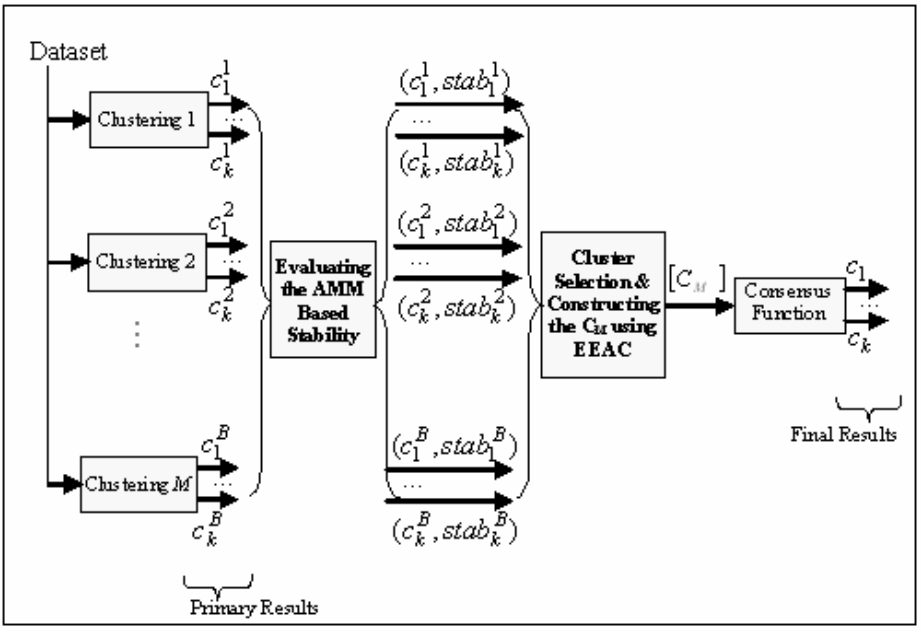


Fig. 1. The proposed clustering ensemble method.

The main idea of the proposed clustering ensemble method is to utilize a subset of the best performing primary clusters in the ensemble instead of all of them. It seems that every cluster does not have a good quality. So, in this method just those clusters which satisfy enough stability to participate in the combination are chosen. The cluster selection is done based on cluster stability which is defined according to Normalized Mutual Information, NMI.

Fig. 1 depicts the proposed clustering ensemble procedure. First, a set of B primary partitions is provided using K-means and Linkage methods to create the necessary diversity for an ensemble. Then, the Improved Stability, IStability, is computed for all clusters of each obtained partitions. The manner of computing IStability is described in next sections in detail. After that, a subset of the most stable clusters is selected to participate in the final decision committee. This is done simply by applying a threshold over the IStability value of any cluster. In the next step, the selected clusters construct the co-association matrix. Several methods have been proposed how to

combine the primary results [25], [13] and [15]. Here, the difference is the absence of some clusters in primary partitions. Since the original EAC method [13] cannot truly identify the pairwise similarity while there is just a subset of clusters, in this paper a new method to construct the co-association matrix is presented which hereafter called: Extended Evidence Accumulation Clustering method, EEAC. Finally, the single-link method is employed to extract the final clusters from this matrix.

3.1 Cluster Evaluation Using APMM Criteria

A stable cluster is the one that has a high likelihood of recurrence across multiple applications of the clustering method. The stable clusters are usually preferable, since they are robust with respect to minor changes in the dataset [18].

Now assume that we want to compute the stability of cluster C_i. In this method first a set of partitionings over resampled datasets is provided which is called the reference set. In this notation D is resampled data and $P(D)$ is a partitioning over D. Now, the problem is: "How many times is the cluster C_i repeated in the reference partitions?" Denote by NMI(C_i,$P(D)$), the Normalized Mutual Information between the cluster C_i and a reference partition $P(D)$. The most previous works only compare a *partition with another partition* [25]. However, the stability used in [18] evaluates the similarity between a *cluster and a partition* by transforming the cluster C_i to a partition and employing common *partition to partition* methods. To illustrate this method let $P_1 = P^a = \{C_i, D/C_i\}$ be a partition with two clusters, where D/C_i denotes the set of data points in D that are not in C_i.

Then we may compute a second partition $P_2 = P^b = \{C^*, D/C^*\}$, where C^* denotes the union of all "positive" clusters in $P(D)$ and others are in D/C^*. A cluster C_j in $P(D)$ is positive if more than half of its data points are in C_i.

Now, define NMI(C_i,$P(D)$) by NMI(P^a,P^b) which is calculated as [12]:

$$NMI(P^a, P^b) = \frac{-2\sum_{i=1}^{k_a}\sum_{j=1}^{k_b} n_{ij}^{ab} \log\left(\frac{n_{ij}^{ab}.n}{n_i^a.n_j^b}\right)}{\sum_{i=1}^{k_a} n_i^a \log\left(\frac{n_i^a}{n}\right) + \sum_{j=1}^{k_b} n_j^b \log\left(\frac{n_j^b}{n}\right)} \tag{1}$$

where n is the total number of samples and n_{ij}^{ab} denotes the number of shared patterns between clusters $C^a{}_i \in P^a$ and $C^a{}_j \in P^a$.

This is done between the cluster C_i and all partitions available in the reference set. Fig. 2 shows this method.

NMI_i in Fig. 2 shows the stability of cluster C_i with respect to the i-th partition in reference set. The total stability of cluster C_i is defined as:

$$Stability(C_i) = \frac{1}{M}\sum_{i=1}^{M} NMI_i \tag{2}$$

where M is the number of partitions available in reference set. This procedure is applied to each cluster of every primary partition.

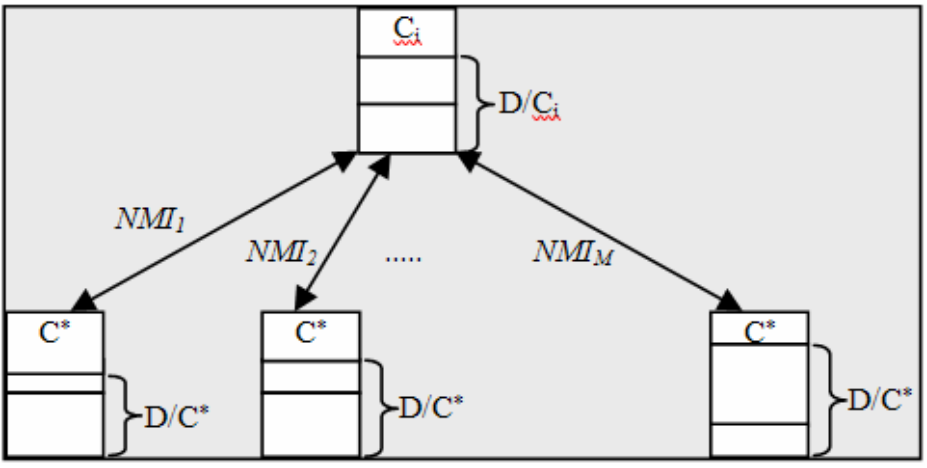


Fig. 2. Computing the Stability of Cluster C_i.

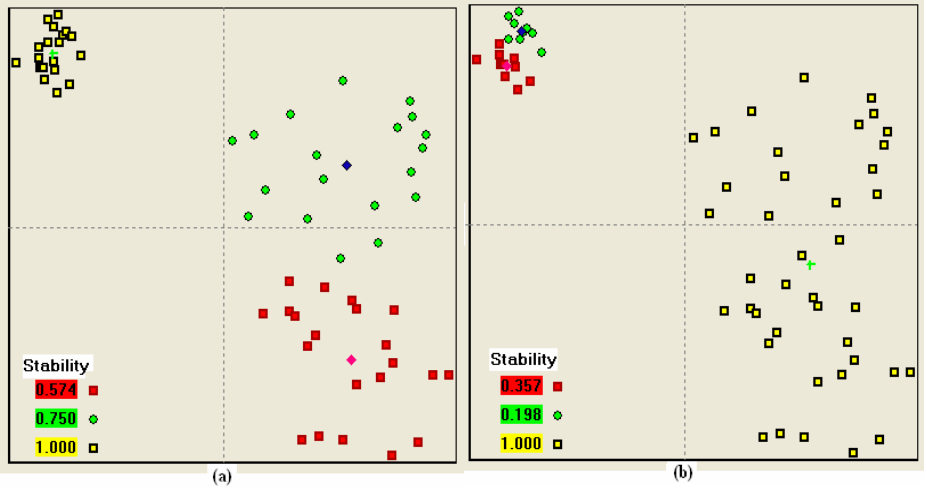


Fig. 3. Two primary partitions with k=3. (a) True clustering. (b) Spurious clustering.

This approach to evaluate the stability has some drawbacks which can appear under some circumstances. Here, a drawback of this procedure is introduced. Then an approach which is based on a novel asymmetric measure named Alizadeh-Parvin-Moshki-Minaei, APMM criteria, is proposed.

Fig. 3 shows two primary partitions with the evaluated stability of each cluster and the process of its computation. In this example K-means is applied as the base clustering algorithm and the true number of clusters, k=3, is forced. For this example the number of all partitions in reference set is 40. In 36 partitions the result is relatively similar to Fig 3.a and in 4 times the top left cluster is divided into two

clusters, like Fig 3.b. Fig. 3a shows a true clustering. Since the well separated cluster in the top left corner is repeated several times (90% repetition) in partitionings of the reference set, it has to acquire a great stability value (but not equal to 1), however it acquires the stability value of 1. On account of the two clusters in right hand of Fig. 3a are relatively jointed and sometimes they are not recognized in reference set as well, they acquire less stability value. Fig. 3.b shows a spurious clustering which the two right clusters are incorrectly merged. Since a fixed number of clusters are forced to the base algorithm, consequently the top left cluster is divided into two clusters. Here the drawback of the stability measure is significantly appeared. Although it is obvious that this partition and correspondingly the right big cluster is rarely appeared in reference set (10% repetition), the stability of the right big cluster is evaluated equal to 1. Since the NMI is a symmetric equation, the stability of the top left cluster in Fig 3.a is exactly equal to the big right cluster in Fig 3.b; however they are repeated 90% and 10% respectively. In other word, when two clusters are complement of each other, their stabilities are always equal. This drawback occurs when the number of positive clusters in the considered partition of reference set is greater than 1. It means when the cluster C* is obtained by merging two or more clusters, the unwelcome results in stability value is occurred.

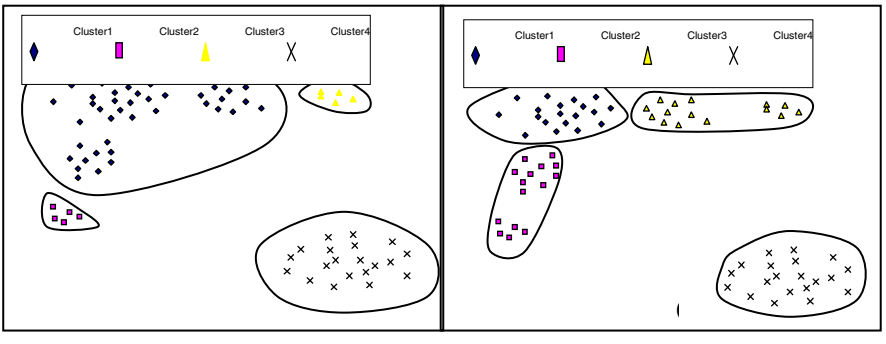


Fig. 4. Evaluating the APMM criterion for cluster C_1 from clustering (a) with respect to clustering (b), with k=4.

Here, a new criterion is proposed which can solve this problem. Assume that the problem is evaluating the APMM criterion for cluster C_1 in Fig. 4a with respect to clustering obtained in Fig. 4b.

The main idea in this method is to eliminate the symmetricalness which exists in NMI equation. In this approach, except the cluster C_1 all other clusters in P^a are taken out. Also, all clusters in P^b which are not included the samples of this cluster are eliminated. In the next step, the other samples which are not in C_1 of P^a, are removed from clusters in P^b (from the clusters which include some of these samples). This process is depicted in Fig. 5.

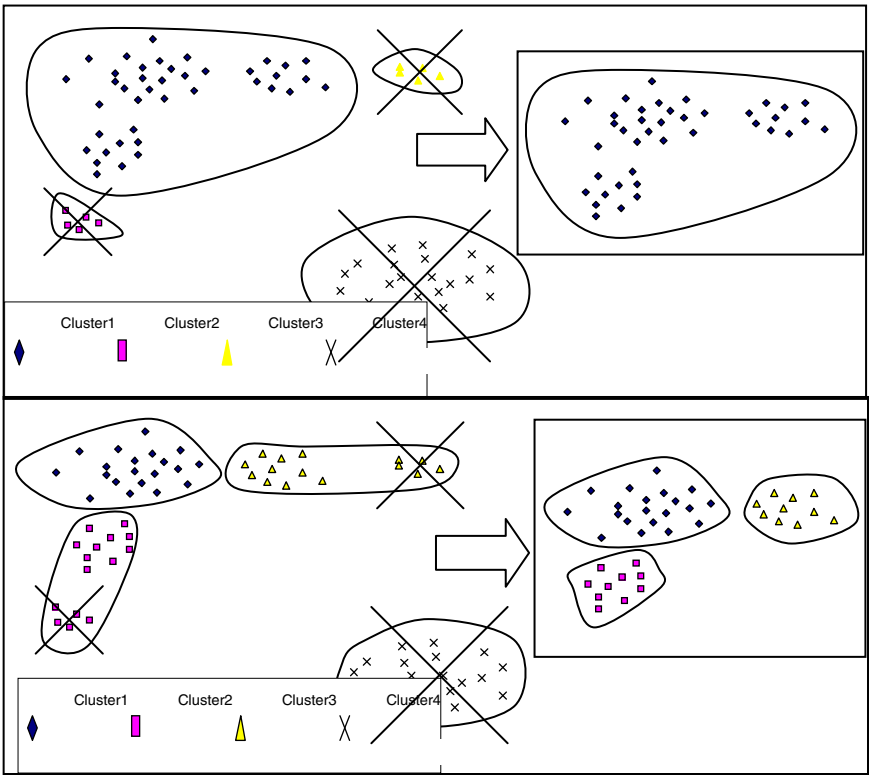


Fig. 5. Providing data for evaluating the APMM criterion. (a) Deleting all other clusters except C_1 from P^a. (b) deriving P^{b*}, the corresponding samples of C_1 in P^b.

Now, the entropy between remained clusters in two partitions P^a and P^b is computed (see Fig. 6). On account of the other involved samples are eliminated, this criterion is not symmetric.

All the previous works are based on the NMI definition as equation 1. Even for evaluating the occurrence of a cluster in a partition, the problem is modified in some way to become the comparing problem between two partitions and then the NMI equation is used. In this paper, the problem is not changed according to definition of NMI; instead, the NMI equation is modified so that the occurrence of a cluster in a partition is computed. It is done by evaluating the entropy between the considered cluster and other pseudo clusters in the corresponding partition. In this paper the Alizadeh-Parvin-Moshki-Minaei criterion, APMM, is defined between a cluster C_i from P^a and the partition P^{b*} from P^b, as below equation:

$$APMM(C_i^a, P^{b*}) = \frac{-2\log\left(\frac{n}{n_i^a}\right)\sum_{j=1}^{k_{b*}} n_j^{b*}}{n_i^a \log\left(\frac{n_i^a}{n}\right) + \sum_{j=1}^{k_{b*}} n_j^{b*} \log\left(\frac{n_j^{b*}}{n}\right)} \tag{3}$$

where n is number of samples available in the cluster C_i and n_{ij}^{ab*} denotes the number of shared samples between the clusters $C_i^a \in P^a$ and $C_j^{b*} \in P^{b*}$. Also k_{b*} is the number of clusters in P^{b*}.

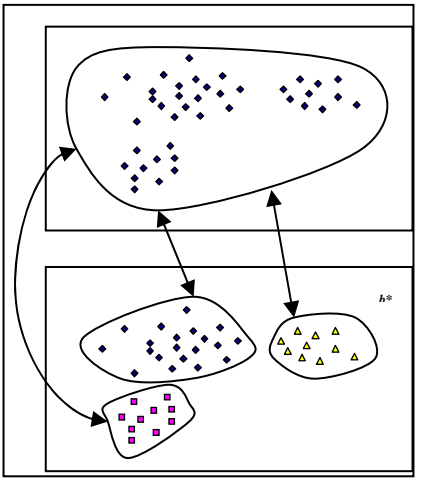


Fig. 6. Computing the entropy between the cluster C_1 from P^a and P^{b*} from P^b.

Here, the Average APMM, AAPMM is proposed as a measure of stability of a primary cluster C_i with respect to the partitions available in the reference set as equation 4:

$$AAPMM(C_i) = \frac{1}{M}\sum_{j=1}^{M} APMM(C_i^a, P_j^{b*}) \tag{4}$$

where P_j^{b*} is from j-th partition of the reference set.

3.2 Extended EAC (EEAC)

In this step, the selected clusters have to construct the co-association matrix. In EAC method the m primary results from resampled data are accumulated in an $n \times n$ co-association matrix. Each entry in this matrix is computed from equation 5.

$$C(i,j) = \frac{n_{i,j}}{m_{i,j}} \tag{5}$$

Where n_{ij} counts the number of clusters shared by objects with indices i and j in the partitions over the B clusterings. Also m_{ij} is the number of partitions where this pair of objects is simultaneously present. There are only a fraction of all primary clusters available, after Thresholding. So, the common EAC method cannot truly recognize

the pairwise similarity for erecting the co-association matrix. Each entry of the co-association matrix according to definition of the novel method which is called Extended Evidence Accumulation Clustering method, EEAC is as equation 6.

$$C(i,j) = \frac{n_{i,j}}{\max(n_i, nj)} \tag{6}$$

Where n_i and n_j are the number of presence in selected clusters for the i-th and j-th objects, respectively. Also, n_{ij} counts the number of selected clusters which are shared by objects with indices i and j.

To more explain consider the following example. Assume that 5 samples according to Fig. 7a, which 4 primary clusterings P_1 to P_4 are applied on this data (according to Fig. 7b).

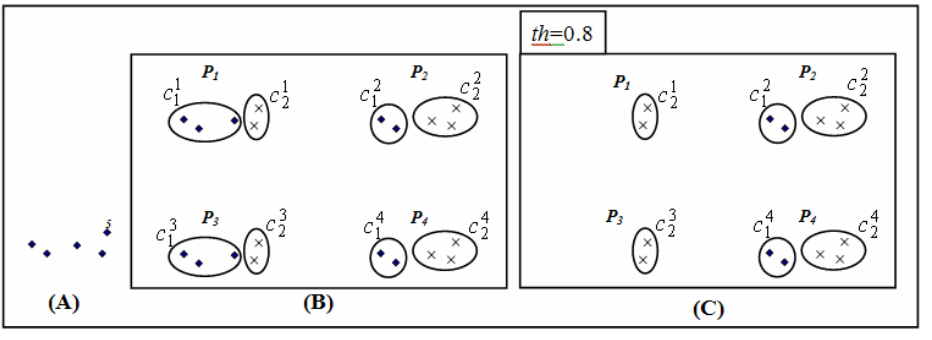


Fig. 7. Computing the co-association matrix by EEAC method. (A) Data samples. (B) 4 primary clusterings. (C) Reminded clusters after applying threshold, *th*=0.8.

Also, consider that that stability of the clusters of Fig. 7b is as below:

$$Stability(c_2^1) = Stability(c_2^3) = 1$$
$$Stability(c_1^2) = Stability(c_1^4) = 1$$
$$Stability(c_2^2) = Stability(c_2^4) = 0.82$$
$$Stability(c_1^1) = Stability(c_1^3) = 0.55$$

By choosing *th*=0.8 the first clusters from P_1 and P_3 are deleted (Fig. 7c). According to equation 6, each entry of co-association matrix is

$$C(1,2) = \frac{2}{\max(2,2)} = \frac{2}{2} = 1$$
$$C(1,3) = C(2,3) = \frac{0}{\max(2,2)} = \frac{0}{2} = 0$$
$$C(3,4) = C(3,5) = \frac{2}{\max(2,4)} = \frac{2}{4} = 0.5$$
$$C(4,5) = \frac{4}{\max(4,4)} = \frac{4}{4} = 1$$

Before and after applying threshold, the co-association matrix is equal to equation 7 and 8, respectively

$$C_{before} = \begin{bmatrix} 1 & 1 & 0.5 & 0 & 0 \\ 1 & 1 & 0.5 & 0 & 0 \\ 0.5 & 0.5 & 1 & 0.5 & 0.5 \\ 0 & 0 & 0.5 & 1 & 1 \\ 0 & 0 & 0.5 & 1 & 1 \end{bmatrix} \quad (7)$$

In this matrix the 3rd object can be considered as both clusters with an equal probability 50%. The stability measure adds some info to this matrix by applying the threshold.

$$C_{after} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.5 & 0.5 \\ 0 & 0 & 0.5 & 1 & 1 \\ 0 & 0 & 0.5 & 1 & 1 \end{bmatrix} \quad (7)$$

By comparing these two matrices and also considering the stability values, it can be apprehended that deletion of unstable clusters improves the co-association matrix. In other hand, eliminating the unstable cluster with samples {1,2,3} which is spuriously created by primary clusterings, enlightens the matrix.

After erecting the co-association matrix by EEAC method, a consensus function is employed to extract the final clusters from the matrix. Here, the single-link method is used for this task.

4 Experimental Results

This section reports and discusses the empirical studies. The proposed method is examined over 5 different standard datasets. It is tried for datasets to be diverse in their number of true classes, features and samples. A large variety in used datasets can more validate the obtained results. Brief information about the used datasets is available in Table 1.

Table 1. Brief information about the used datasets.

	Class	Features	Samples
Glass	6	9	214
Breast-C	2	9	683
Wine	3	13	178
Bupa	2	6	345
Yeast	10	8	1484

All experiments are done over the normalized features. It means each feature is normalized with mean of 0 and variance of 1, N(0, 1). All of them are reported over means of 10 independent runs of algorithm. The final performance of the clustering algorithms is evaluated by re-labeling between obtained clusters and the ground truth labels and then counting the percentage of the true classified samples. Table 2 shows the performance of the proposed method comparing with most common base and ensemble methods.

Table 2. Experimental results.

	Simple Methods (%)				Ensemble Methods (%)			
Dataset	Single Linkage	Average Linkage	Complete Linkage	Kmeans	Kmeans Ensemble	Full Ensemble	Cluster Selection by NMI Method	Cluster Selection by max Method
Wine	37.64	38.76	83.71	96.63	96.63	97.08	97.75	**98.31**
Breast-C	65.15	70.13	94.73	95.37	95.46	95.10	95.75	**98.33**
Yeast	34.38	35.11	38.91	40.20	45.46	47.17	**47.17**	**47.17**
Glass	36.45	37.85	40.65	45.28	47.01	47.83	48.13	**50.47**
Bupa	57.68	57.10	55.94	54.64	54.49	55.83	58.09	**58.40**

The four first columns of Tab. 2 are the results of some base clustering algorithms. The results show that although each of these algorithms can obtain a good result over a specific data set, it does not perform well over other data sets. For example, according to Tab. 2 the *K*-means algorithm has a good clustering result over Wine data set in comparison with linkage methods. But, it has lower performance in comparison to linkage methods in the case of Bupa data set. Also, the complete linkage has a good performance in Breast-Cancer data set in comparison with others; however it is not in the case of all data sets. The four last columns show the performance of some ensemble methods in comparison with the proposed one. Taking a glance on the last four columns in comparison with the first four columns shows that the ensemble methods do better than simple base algorithms in the case of performance and robustness against different data sets. The first column of the ensemble methods is the results of an ensemble of 100 *K*-means which is fused by EAC method. The 90% sampling from data set is used for creating diversity in primary results. The sub-sampling (without replacement) is used as the sampling method. Also the random initialization of the seed points of *K*-means algorithm helps them to be more diverse. The single linkage algorithm is applied as consensus function for deriving the final clusters from co-association matrix. The second column from ensemble methods is the full ensemble which uses several clustering algorithms for generating the primary results. Here, 70 *K*-means with the above mentioned parameters in addition to 30 linkage methods provide the primary results. Since different runs of a specific linkage method always yield to the same result, there is a limitation for using them as the base clustering algorithms. Here, forcing a set of different number of clusters, $K\pm2$, is used to create diversity, which K is the true number of clusters. The used linkage algorithms and their distance criterion are shown in Tab. 3. The detailed information about each of these linkage algorithms can be found in [8].

The two last columns are the results of ensemble methods which uses only a subset of primary clusters. The first one shows the results of an ensemble which uses the traditional NMI based stability for cluster validation. The second one is the proposed clustering ensemble method which uses the APMM criterion for cluster validation. Accumulating the selected clusters in co-association matrix in both methods is done using the proposed EEAC method. Finally, the single linkage algorithm is applied over co-association matrix to extract final clusters. The primary clustering results are provided similar to the full ensemble. In these methods, the threshold which is used to cluster selection is determined adaptively. In other words, first it is adjusted to 95%. If less than 10% of samples are absent in the selected clusters, the threshold is reduced to 90%. This procedure (reducing 5% from threshold value) continues until the selected clusters include more than 90% of samples. The results of the two last columns show that although these approaches use a subset of the primary clusters, they usually outperform the full ensemble. Also, comparing the last two columns shows the power of APMM based stability in comparison with the NMI based stability. Examinations by 10 independent run over different data sets robustly show the quality of the APMM criterion with respect to NMI.

5 Conclusions

In this paper a new clustering ensemble method is proposed which is based on a subset of total primary spurious clusters. Since the quality of the primary clusters are not equal and presence of some of them can even yield to lower performance, here a method to select a subset of more effective clusters is proposed. A common cluster validity criterion which is needed to derive this subset is based on normalized mutual information. In this paper some drawbacks of this criterion is discussed and an alternative criterion is suggested which is named Alizadeh-Parvin-Moshki-Minaei, APMM. The experiments show that the APMM criterion does slightly better than NMI criterion generally; however it significantly outperforms the NMI criterion in the case of synthetic data sets. Because of the symmetry which is concealed in NMI criterion and also in NMI based stability, it yields to lower performance whenever symmetry is also appeared in the data set. Another innovation of this paper is a method for constructing the co-association matrix where some of clusters and respectively some of samples do not exist in partitions. This new method is called Extended Evidence Accumulation Clustering, EEAC. The empirical studies over several data sets robustly show that the quality of the proposed method is usually better than other ones.

References

1. Ayad, H.G., Kamel, M.S.: Cumulative Voting Consensus Method for Partitions with a Variable Number of Clusters. IEEE Trans. on Pattern Analysis and Machine Intelligence 30(1), 160–173 (2008)
2. Baumgartner, R., Somorjai, R., Summers, R., Richter, W., Ryner, L., Jarmasz, M.: Resampling as a Cluster Validation Technique in fMRI. Journal of Magnetic Resonance Imaging 11, 228–231 (2000)
3. Ben-Hur, A., Elisseeff, A., Guyon, I.: A stability based method for discovering structure in clustered data. Pasific Symposium on Biocomputing 7, 6–17 (2002)

4. Brandsma, T., Buishand, T.A.: Simulation of extreme precipitation in the Rhine basin by nearest-neighbour resampling. Hydrology and Earth System Sciences 2, 195–209 (1998)
5. Breckenridge, J.: Replicating cluster analysis: Method, consistency and validity. Multivariate Behavioral Research (1989)
6. Das, A.K., Sil, J.: Cluster Validation using Splitting and Merging Technique. In: Int. Conf. on Computational Intelligence and Multimedia Applications, ICCIMA (2007)
7. Davison, A.C., Hinkley, D.V., Young, G.A.: Recent developments in bootstrap methodology. Statistical Science 18, 141–157 (2003)
8. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley, Chichester (2001)
9. Estivill-Castro, V., Yang, J.: Cluster Validity Using Support Vector Machines. In: Kambayashi, Y., Mohania, M., Wöß, W. (eds.) DaWaK 2003. LNCS, vol. 2737, pp. 244–256. Springer, Heidelberg (2003)
10. Faceli, K., Marcilio, C.P., Souto, D.: Multi-objective Clustering Ensemble. In: Proceedings of the Sixth International Conference on Hybrid Intelligent Systems (2006)
11. Fern, X.Z., Lin, W.: Cluster Ensemble Selection. In: SIAM International Conference on Data Mining (2008)
12. Fred, A., Jain, A.K.: Combining Multiple Clusterings Using Evidence Accumulation. IEEE Trans. on Pattern Analysis and Machine Intelligence 27(6), 835–850 (2005)
13. Fred, A., Jain, A.K.: Data Clustering Using Evidence Accumulation. In: Intl. Conf. on Pattern Recognition, ICPR 2002, Quebec City, pp. 276–280 (2002)
14. Fred, A., Jain, A.K.: Learning Pairwise Similarity for Data Clustering. In: Int. Conf. on Pattern Recognition (2006)
15. Fred, A., Lourenco, A.: Cluster Ensemble Methods: from Single Clusterings to Combined Solutions. SCI, vol. 126, pp. 3–30 (2008)
16. Fridlyand, J., Dudoit, S.: Applications of resampling methods to estimate the number of clusters and to improve the accuracy of a clustering method. Stat. Berkeley Tech. Report No. 600 (2001)
17. Inokuchi, R., Nakamura, T., Miyamoto, S.: Kernelized Cluster Validity Measures and Application to Evaluation of Different Clustering Algorithms. In: IEEE Int. Conf. on Fuzzy Systems, Canada, July 16-21 (2006)
18. Law, M.H.C., Topchy, A.P., Jain, A.K.: Multiobjective data clustering. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 424–430 (2004)
19. Lange, T., Roth, V., Braun, M.L., Buhmann, J.M.: Stability-based validation of clustering solutions. Neural Computation 16(6), 1299–1323 (2004)
20. Minaei-Bidgoli, B., Topchy, A., Punch, W.F.: Ensembles of Partitions via Data Resampling. In: Intl. Conf. on Information Technology, ITCC 2004, Las Vegas (2004)
21. Möller, U., Radke, D.: Performance of data resampling methods based on clustering. Intelligent Data Analysis 10(2) (2006)
22. Rakhlin, A., Caponnetto, A.: Stability of k-means clustering. In: Advances in Neural Information Processing Systems, vol. 19. MIT Press, Cambridge (2007)
23. Roth, V., Lange, T.: Feature Selection in Clustering Problems. In: Advances in Neural Information Processing Systems (2004)
24. Roth, V., Lange, T., Braun, M., Buhmann, J.: A Resampling Approach to Cluster Validation. In: Intl. Conf. on Computational Statistics, COMPSTAT (2002)
25. Strehl, A., Ghosh, J.: Cluster ensembles - a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research 3, 583–617 (2002)
26. Xie, X.L., Beni, G.: A Validity measure for Fuzzy Clustering. IEEE Trans. on Pattern Analysis and Machine Intelligence 13(4), 841–846 (1991)

Discovery of Domain Values for Data Quality Assurance

Łukasz Ciszak

Institute of Computer Science, Warsaw University of Technology,
ul. Nowowiejska 15/19, 00-665 Warszawa, Poland
l.ciszak@ii.pw.edu.pl

Abstract. Data profiling is a crucial step in the data quality process as it provides the current data quality rules. In this paper we present experimental results comparing our *DOMAIN* method for the discovery of domain constraint values to the commercially available Oracle Warehouse Builder (OWB). The experimental results prove that the effectiveness of our approach in the discovery of domain values for textual data affected by data quality problems is greater than that offered by the OWB.

Keywords: information quality, data quality, data profiling, domain discovery.

1 Introduction

Business intelligence systems whose core is a data warehouse require high quality data as the input for analytical processes used to draw conclusions regarding the future of a given enterprise and support the decision-making process. In this paper we concentrate on the experimental comparison of the effectiveness of our *DOMAIN* method for the discovery of domain constraint values (data quality rules) to the Oracle Warehouse Builder profiler offering similar functionality. The details of *DOMAIN* are covered by [3].

Data quality in an information system is defined as 'the fitness of the data for use' [10]; ontologically, high quality data represent the Universe of Discourse (UoD) in the way it allows us to draw conclusions about the future states of the UoD [11]. Real-world data are always affected by various data quality problems resulting from the imperfection of data input methods (misspelling, mistypings), data transport (encoding problems), and finally the evolution of the data source (stale and inaccurate metadata). Domain constraints assure that only the states that represent valid states of the UoD are allowed in the information system. Data quality assessment process discovers the data quality rules and assesses the current quality of data in the system. The results are subsequently consumed by the data cleansing process that improves the quality of data.

The data quality improvement is researched in the domain of constraint repair and consistent query answering [1,2] and record linkage and duplicate elimination [13]. The profiling methods are studied in terms of the discovery of

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 15–20.
springerlink.com

functional,approximate and conditional dependencies [5,6,7], and by data warehousing literature [8,9,10]. Commercial tools for the support of profiling activities (e.g. *Informatica Data Quality*, *Informatica Data Explorer*) do not offer domain discovery capabilities; in the ETL tool area, domain discovery options are available in the *Oracle Warehouse Builder (OWB)*. To our knowledge, apart from ours [3] there has been no research specifically concentrated on the discovery of domain constraints.

2 DOMAIN Method

In [3] we presented a *DOMAIN (DOmain Mining and repAIr in uNclean data)* method for the discovery of domain values from textual data sets heavily affected by various data quality issues. In the remainder of this section we present briefly the concepts and the pseudocode of *DOMAIN*. For the details the reader should refer to [3].

2.1 Fundamental Concepts

Let $\hat{r}$ be a relation in scheme $R = \{A_1, A_2, \ldots, A_n\}$ representing certain collection of UoD objects in the way that there are no objects without information system representation and each legit state of each object is represented unambiguously by a value from a domain for attribute from R. We will refer to $\hat{r}$ as the *ideal relation* having the highest possible ontological quality of data. Let r be another relation in R such that $\forall A_i \in R\ dom(A_i) \supset \hat{D}_i$.

We assume that the set of values for each attribute in r is defined as $D_i = \hat{D}_i \cup E_i \cup N_i$, where E_i denotes the set of values that are the 'damaged' version of correct domain values introduced by the imperfection of entry methods or other factors, and N_i is the set of *noise*, that is, meaningless and random values. In the remainder of this paper we will focus on a single attribute A and skip the subscript i. We will be considering a multiset $S_D = (D, m)$ where $m : D \rightarrow \mathbf{N}$ is a multiplicity function such that $m(d) = |\{t \in r : t(A) = d\}|$.

Let $\rightsquigarrow$ be *a similarity relation* on S_D such that

$d_k \rightsquigarrow d_l$ iff $(sim_{kl} \geq \varepsilon \wedge ratio_{kl} \geq \alpha)$, where sim_{kl} is the *Jaro-Winkler string similarity measure* [12] equal 0 for two completely different strings and equal 1 for two identical strings, ε is a *textual similarity threshold*, $ratio_{kl} = \max\left(\frac{m(d_k)}{m(d_l)}, \frac{m(d_l)}{m(d_k)}\right)$ is the *multiplicity ratio*, and α is *multiplicity ratio threshold*. Informally, element d_k is similar to element d_l in terms of the $\rightsquigarrow$ relation if they are textually similar and element is d_l is significantly more frequent in the relation r than d_k. The set D and its subsets have the following features:

$\forall e \in E\ \exists \hat{d} \in \hat{D} : e \rightsquigarrow \hat{d}$ and $\nexists \hat{d} \in \hat{D} : \forall n \in N\ n \rightsquigarrow \hat{d}$.

The multiset S_D and the relation $\rightsquigarrow$ may be represented as a directed weighed graph, where the nodes of the graph represent the elements from S_D and the arcs of the graph represent the relation $\rightsquigarrow$. In this graph we can distinguish two classes of nodes – *sinks*, that is, nodes having only incoming arcs and *isolated nodes*, having neither incoming nor outgoing arcs.

2.2 DOMAIN Definition

The *DOMAIN* algorithm is presented in Fig.1; in Fig. 2 we present the behavior of the algorithm.

The input for the method is the examined set of values for a given attribute provided as a multiset S_D; the output is the set D' of the discovered domain values and ideally we expect $D' \Delta \hat{D} = \emptyset$. The behavior of the algorithm is driven by two input parameters: ε – the textual similarity threshold and α – the multiplicity ratio threshold.

```
Procedure calculateSimilarities(SD,ε,α)
for all di∈SD do
 for all dj∈SD do
  s←sim(di,dj)
  r ← ratio(di,dj)
  if (s≥ε)∧(m(di)<m(dj))∧(r≥ α)
  then
   similarities[di,dj]←(s,r)
  end if
 end for
end for
return similarities

Procedure getSink(d1,d2,parents,ε)
if ((parents[d2]= NULL)
    ∨(parents[d2]!= NULL
    ∧sim(d1,parents[d2]) ≤ ε)
then
 return d2
else
 return getSink(d1,parents[d2],parents,ε)
end if

Procedure graphToTree(SD, similarities)
for all di∈SD do
 parent[di]←NULL
end for
for all di∈SD do
 parent[di]←arg maxdj similarities[di,dj]
end for
return parent

Procedure streamline(SD,parents,ε)
for all di ∈SD do
 if (parents[di]!= NULL)
     ∧(!∃dj : parents[dj]=di)
 then
  parents[di]←getSink(di,parents[di],
     parents,ε)
 end if
end for
for all di ∈SD do
 if (parents[di]!=NULL)
    ∧(∃dj : parents[dj]=di)
 then
  parents[di]← NULL
 end if
end for
return parents

Procedure DOMAIN(SD,ε,α)
similarities←calculateSimilarities(SD,ε,α)
parent←graphToTree(SD,similarities)
parent←streamline(SD,parent)
for all di∈SD' do
 if parent[di]=NULL then
  D'← D'∪ parent[di]
 end if
end for
return D'
```

Fig. 1. The pseudocode of the *DOMAIN* method.

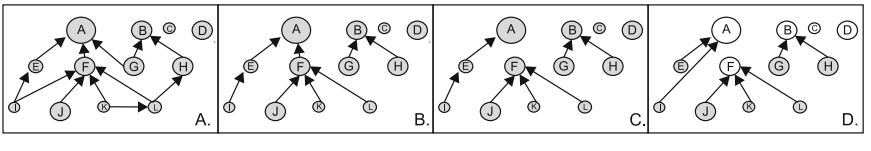


Fig. 2. A fragment of the graph structure after the application of the steps of the algorithm. The sizes of nodes represent the number of value occurrences. In fig.A., we created the graph with all the arcs satisfying the conditions of the $\rightsquigarrow$ relationship. In Fig.B. we retained only the arcs representing the greatest similarity between the values. In Fig.C., the node *F.* has been transformed into a sink node as the relation $\rightsquigarrow$ between *A* and values *J,K,L* did not occur. In Fig.D. we streamlined the graph and identified values *A, B, C* and *D* as the domain values.

3 Experimental Results

We performed the experiments on two datasets – a synthetic 'American cities' dataset and a real-life 'Polish cities' dataset. The latter data set consists of the names of Polish cities extracted from an Internet survey and contains 2199 distinct values (28429 in total) of which 1862 were domain values (valid Polish city names), covering 27279 elements of the data set. The synthetic 'American cities' data set was created using the names of 50 of the largest American cities as the domain values. The incorrect variants of these values were generated according to the distribution of the textual errors in the former data set. After these operations, there were 1350 distinct values (24749 total values) and the domain values covered 22516 values from the data set. The purpose of the experiments was to compare the domain discovery effectiveness of *DOMAIN* to the effectiveness offered by the OWB.

The effectiveness of domain discovery was assessed using two measures. The first measure p_D is the ratio of discovered domain values to the all domain values in the given dataset; the second measure p_{nD} is the ratio of the non-domain values classified as domain values to the all non-domain values in the given data set. Both measures are also expressed in an absolute form, where we take into consideration the number of occurrences of the value within the data set; defined respectively as P_D and P_{nD}.

$$p_D = \frac{|D' \cap \hat{D}|}{|\hat{D}|} \quad P_D = \frac{|\{t \in r \mid t(A) \in (D' \cap \hat{D})\}|}{|\{t \in r \mid t(A) \in \hat{D}\}|}$$

$$p_{nD} = \frac{|D' \cap (N \cup E)|}{|N \cup E|} \quad P_{nD} = \frac{|\{t \in r \mid t(A) \in (D' \cap (N \cup E))\}|}{|\{t \in r \mid t(A) \in (N \cup E)\}|}$$

3.1 Results

We performed the experiments on the two aforementioned data sets for $\varepsilon = 0.87$ and $\alpha = 2$ obtained using the results described in [4] and yielding optimal results in terms of the effectiveness measures. For comparison, we ran the *Oracle Warehouse Builder* profiler on the same data sets and chose the optimal results obtained using this tool. In our experiments, we chose two sets of the driving parameters, denoted respectively by OWB(1) and OWB(2). In both sets, we set the limit for the maximal number of discovered values to 10000 to assure the discovery of all the domain values. In case of OWB(1) the lower limits were set to 0 to allow the discovery of infrequent values; in case of OWB(2), we set the lower limits to 1 hence limiting the discovered values only to those whose frequency of appearance in the data set was greater than 1%.

The results of the experiments are presented in Fig. 3.

The experimental results prove that our approach is capable of effective discovery of domain values for textual attributes heavily affected (10%) by typographic errors. *DOMAIN* managed to discover all the domain values in the synthetic data

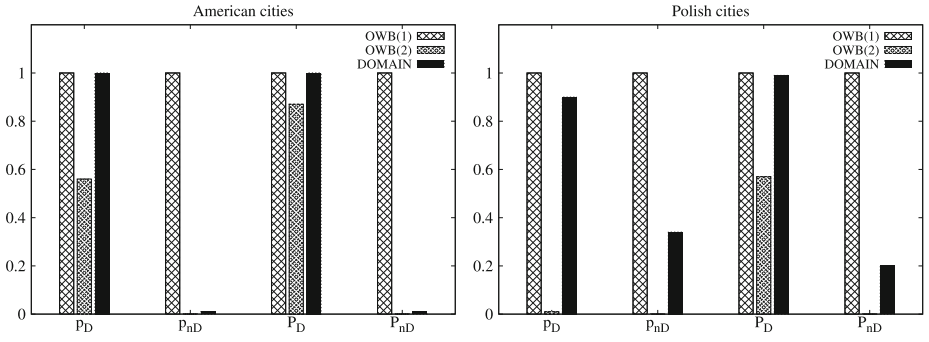


Fig. 3. The results of the domain discovery executed using Oracle Warehouse Builder (OWB) and our approach on two data sets – synthetic 'American cities (left frame) and real-world 'Polish cities' (right frame).

set and 90% of the domain values in the real-world data set while maintaining low level of the ratio of the non-domain values identified as domain values – 1.5% of non-domain values that covered 0.8% of relation tuples in case of synthetic data and 34% of non-domain values covering less than 20% relation tuples in case of the real-world data.

The OWB was less effective than *DOMAIN* in terms of the effectiveness measures for both data sets. In the case of the synthetic data set and the OWB(2) parameter set, the OWB managed to identify 28 of the 50 domain values that covered 87% of the relation tuples, whereas our method identified all the domain values. The OWB managed to avoid false positives for the cost of missing almost a half of the valid domain values. When applied to the real-world data set, the OWB profiler using the OWB(2) parameter set identified only 12 out of 1862 domain values which is 0.64% of all the domain values which covered 57% of the relation tuples. Similarly to the synthetic data set, the ratio of false positives was 0. In contrast, our algorithm identified 90% of the domain values covering the 99% of tuples. The number of false positives produced by *DOMAIN* covered less than 20% of the tuples.

In case of the OWB(1) parameter set, the OWB marks all the values from the set as domain members producing 100% ratio of false positives.

Our results show than the *DOMAIN* method us can be used for the effective discovery of domain values in high cardinality data sets heavily affected by data quality issues. In contrast, the OWB profiler is more effective false positive-wise in situations where the cardinality is low and only takes into consideration the coverage of the examined relation by the potential domain values as in case of high cardinality data set it can either discover either a fraction of the domain values or denote all the values as the domain members.

4 Conclusions

In this paper, we have presented the results of experiments performed using the *DOMAIN* method applied to synthetic and real-world data. The results show that *DOMAIN* can effectively discover domain values with high accuracy and a low ratio of false positives in data sets suffering from data quality issues. The method has been compared to the commercial solution offered by the Oracle Warehouse Builder and showed greater effectiveness while also maintaining a higher level of flexibility. The presented method can be used to support the data profiling stage of the data quality assessment and improvement process via the automatization of work needed to obtain the up-to-date metadata describing a given dataset.

References

1. Arieli, O., Denecker, M., Bruynooghe, M.: Distance Semantics for Database Repair. Annals of Mathematics and Artificial Intelligence 50(3-4), 389–415 (2007)
2. Ceri, S., Giunta, F.D., Lanzi, P.L.: Mining Constraint Violations. ACM Transactions on Database Systems 32(1), 1–32 (2007)
3. Ciszak, Ł.: A method for automatic discovery of reference data. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS, vol. 5579, pp. 797–805. Springer, Heidelberg (2009)
4. Ciszak, Ł.: Experimental Comparison of String Similarity Measures for Data Cleaning. In: Proceedings of the 3rd National Scientific Conference on Data Processing Techniques, KKNTPD (2010)
5. Engle, J.T., Robertson, E.L.: HLS: Tunable Mining of Approximate Functional Dependencies. In: Gray, A., Jeffery, K., Shao, J. (eds.) BNCOD 2008. LNCS, vol. 5071, pp. 28–39. Springer, Heidelberg (2008)
6. Fan, W., Geerts, F., Jia, X., Kementsietsidis, A.: Conditional Functional Dependencies for Capturing Data Inconsistencies. ACM Transactions on Database Systems 33(2), 1–48 (2008)
7. Huhtala, Y., Porkka, P., Toivonen, H.: TANE: An Efficient Algorithm for Discovering Functional and Approximate Dependencies. The Computer Journal 42(2), 100–111 (1999)
8. Kimball, R., Caserta, J.: The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, 1st edn. Wiley, Chichester (2004)
9. Lindsey, E.: Three-Dimensional Analysis. Data Profiling Techniques. Data Profiling LLC (2008)
10. Maydanchik, A.: Data Quality Assessment. Technics Publications, LLC (2007)
11. Wand, Y., Yang, R.D.: Anchoring data quality dimensions in ontological foundations. Communications of the ACM 39, 86–95 (1996)
12. Winkler, W.E.: String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. In: Proceedings of the Section on Survey Research Methods (1990)
13. Winkler, W.E.: Overview of Record Linkage and Current Research Directions. Tech. rep., Statistical Research Division U.S. Census Bureau (2006)

Dynamic Time Warping-Based K-Means Clustering for Accelerometer-Based Handwriting Recognition

Minsu Jang[1], Mun-Sung Han[1], Jae-hong Kim[1], and Hyun-Seung Yang[2]

[1] Electronics and Telecommunications Research Institute
Gajeong-dong 161, Yuseong-gu, Daejeon-si, 305-700, South Korea
{minsu,msh,jhkim504}@etri.re.kr
[2] Korea Advanced Institute of Science and Technology
291 Daehak-ro, Yuseong-gu, Daejeon-si, 305-701, South Korea
hsyang@kaist.ac.kr

Abstract. Dynamic time warping(DTW) is widely used for accelerometer-based gesture recognition. The basic learning strategy applied with DTW in most cases is instance-based learning, where all the feature vectors extracted from labeled training patterns are stored as reference patterns for pattern matching. With the brute-force instance-based learning, the number of reference patterns for a class increases easily to a big number. A smart strategy for generating a small number of good reference patterns is needed. We propose to use DTW-based K-Means clustering algorithm for the purpose. Initial training is performed by brute-force instance-based learning, and then we apply the clustering algorithm over the reference patterns per class so that each class is represented by $5 \sim 10$ reference patterns each of which corresponds to the cluster centroid. Experiments were performed on 5200 sample patterns of 26 English uppercase alphabets collected from 40 personals using a handheld device having a 3-d accelerometer inside. Results showed that reducing the number of reference patterns by more than 90% decreased the recognition rate only by 5%, while obtaining more than 10-times faster classification speed.

Keywords: Dynamic Time Warping, K-Means, Accelerometer, Gesture Recognition.

1 Introduction

Nowadays, most mobile devices, including game console controllers and smart mobile phones, are designed to include inertial sensors for improved natural user experience. The most popular inertial sensor is the MEMS-based accelerometer. With accelerometer, it is possible to capture and recognize the posture as well as the motion of devices. Several research efforts tried to develop gesture recognition systems based on acceleration signals from mobile devices [3–5].

In many of such efforts, template matching-based 1-nearest neighbor classification combined with dynamic time warping(DTW) has been successfully applied,

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 21–26.
springerlink.com

where brute-force instance-based learning is used for training in most cases. With brute-force instance-based learning, every training pattern is added to the set of reference patterns, which can make the size of reference patterns very large. Large number of reference patterns incur the problem of low performance and aggressive resource consumption that limits the practical value of the system.

In this paper, we propose a DTW-based clustering algorithm to optimize the number of reference patterns in a systematic way. In our method, training a classifier is performed in two phases. In the first phase, a classifier is trained according to the brute-force instance-based learning. In the second phase, the DTW-based K-Means clustering algorithm is run on each gesture class over all the reference patterns. The algorithm partitions the reference patterns into a small number of clusters according to the similarity measured by DTW and extracts the centroid of each cluster. The centroids of the clusters become the new optimized reference patterns for each gesture class. One important issue in applying K-Means with DTW is that the reference patterns are different in length. We cannot simply estimate the centroid of non-uniform length patterns as they are in different hyperspaces. The solution we propose is to warp the reference patterns along the warping path determined by DTW. By repeating pairwise warping across all the patterns in a cluster, it is possible to estimate the centroid of the cluster. We describe the details of the algorithm in section 2.

To show the effectiveness of the algorithm, we applied the algorithm to the problem of recognizing 26 English uppercase alphabets written using a handheld device having a 3-d accelerometer. The results from the experiment are presented in section 3.

2 DTW-Based Temporal Pattern Clustering

In this section, we provide detailed description of how patterns of non-uniform length can be clustered based on DTW measure. The basic idea is to average two patterns along *the warping path* generated by DTW. Let us assume that P and Q are two patterns, each of which is respectively represented as $n \times k$ and $n \times l$ matrices. By applying DTW, we can get a warping path W, which is an ordered list of index pairs. Each index pair (i, j) corresponds to the best matching element pairs from P and Q respectively. With W, we can calculate the centroid M of P and Q as follows.

$$M_i = \frac{1}{2}(P_i + \frac{1}{|F|}\sum_{j=1}^{|F|} F_j),\ 1 \le i \le k$$
$$F = \{Q_{W_{m,1}} | W_{m,0} = i, 1 \le m \le |W|\}$$

m is the index of the warping path W. The length of the warping path is $|W|$. $W_{m,0}$ is the index of P's element that lies in the m-th place in the warping path, while $W_{m,1}$ is the index for Q. Consequently, F represents the set of Q's elements that are matched to P_i. Extending the above formula to N patterns , we get the algorithm 2. In the algorithm, P is a list of N patterns, and WarpingPathByDTW$(n, M, |M|, P_i, |P_i|)$ is assumed to calculate the warping path W of two input patterns M and P_i using DTW.

Algorithm 1. DTW-MeanForNPatterns(n, P, N, M)

$M \leftarrow P_0$
for $i = 2$ **to** N **do**
  $W \leftarrow$WarpingPathByDTW$(n, M, |M|, P_i, |P_i|)$
  for $j = 1$ **to** $|M|$ **do**
    $Q \leftarrow \{P_{i,W_{x,1}} | W_{x,0} = j, 1 \leq x \leq |W|\}$
    $M_j \leftarrow M_j + \frac{1}{|Q|} \sum_{k=1}^{|Q|} Q_k$
  end for
end for
$M \leftarrow \frac{1}{N} M$

The key to the algorithm 2 is the for-loop where input vectors are incrementally added along the warping path W. After adding all the N patterns, the centroid pattern is calculated simply by dividing M by N. With the algorithm 2, it is possible to apply K-Means clustering on non-uniform length patterns to optimize the reference patterns for each gesture class.

3 Experimental Result

We evaluated the DTW-based K-Means clustering algorithm by applying it on a problem of recognizing gesture patterns.

We used Nintendo^{TM}Wii remote to collect gesture pattern data. Wii remote is a handheld device that embeds a 3-axis accelerometer. Users performed gestures while holding the device by hand. The embedded accelerometer captures the hand movements as a series of acceleration changes across the 3-axes of (x, y, z).

The target patterns for this experiment are uppercase English alphabets, each stroke of which is defined as in Figure 1. 40 adult people participated to perform gestures. Each person played every alphabet 5 times, so each alphabet has 200 pattern samples. The total number of patterns for all the 26 alphabets are 5200.

The signal preprocessing stage for feature extraction is shown in Figure 2. The input to the preprocessing is a train of 3-axis acceleration signal sampled

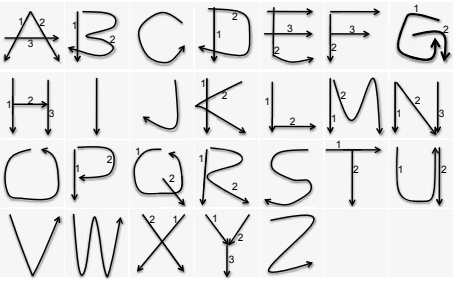


Fig. 1. 26 capitalized English alphabet gestures are collected.

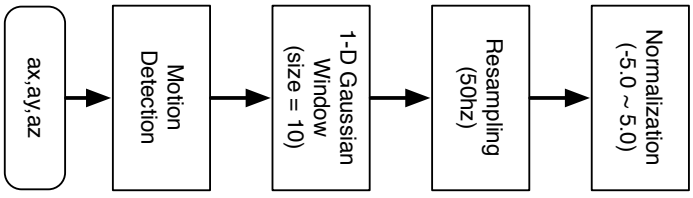


Fig. 2. 26 capitalized English alphabet gestures are collected.

in 100hz. In the first phase, motion detection is performed for detecting the duration in which a gesture is performed. The detection algorithm depends on the variation of motion energy inside a sliding window. The details of the algorithm is out of the scope of this paper. The output of the motion detection phase is a segment of acceleration signals. It is then smoothed for noise reduction using 1-D Gaussian filter, resampled in 50hz for reducing the number of samples, and finally normalized to the values between -5.0 $\sim$ 5.0.

In our experiment, we used a 1-NN classifier with DTW as the distance measure. Training is in two-phases: pure instance-based learning in the first phase and then DTW-based K-Means clustering for optimizing the reference patterns in the second phase. As we have 5200 sample patterns, applying 10-fold cross validation produces a 1-NN classifier with 4680 reference patterns and 520 test patterns. Each class has 180 reference patterns after the first phase. We measured the performance of the classifier with all the 180 reference patterns, and then we optimized the number of reference patterns by applying the DTW-based clustering algorithm and compared the performance against the original classification performance. We tested with varying number of clusters and the result is shown in table 1.

Table 1. Recall and precision according to different number of clusters.

No. of clusters per class	10	20	30	40	50	180
Recall(%)	88.1	89.6	90.4	90.8	91.2	92.5
Precision(%)	88.1	89.7	90.5	90.9	91.2	92.6

The recall rate constantly decreases as the number of reference patterns is reduced, but the degree of decrease is not so significant. With 10 reference patterns per class, which is 5% of the non-clustered 180 reference patterns per class, the recall rate stays only 4.8% less than that of the non-clustered case.

Table 2 shows how long it does take to classify a test pattern with different number of reference patterns. It is noticeable that in the case of 10 reference patterns per class, the classification took only 8% of the time for non-clustered case with 180 reference patterns.

Table 2. The relation of performance with the number of reference patterns.

No. of clusters per class	10	20	30	40	50	180
Time(msec)	51.123	96.730	140.985	184.173	226.155	657.220

K-Means clustering stops when no cluster center does move anymore. In our experiment, about 10% of the clustering cases did not converge to the stable status. In such cases, we need another criteria for stopping the clustering process. We used two criteria. First, we limited the number of iteration under 100. If a clustering process does not stop until it iterates 100 times, the process is forced to stop and the final cluster centers are considered to be the results. Second, we put a threshold on the minimum distance of center movement. A cluster center is flagged as has been moved only when the distance of the movement is greater than the threshold.

4 Related Work

Leong et al. experimented with a very small number of reference patterns to see how DTW performs on accelerometer-based gesture patterns, and showed that DTW performed very well with over 90% recognition accuracy [3]. The experiment is limited to a set of small number, 10, of very discriminative and simple gestures under user-dependent cases only though, and a systematic way of choosing good reference patterns was not given.

Liu et al. proposed an adaptation algorithm to limit the number of reference patterns [4]. Their method initially assigns only two reference patterns per gesture class. On receiving a new training sample, an evaluation is performed on previously selected reference patterns as well as on the new training sample, and one of them is discarded.

Bong et al. utilized K-Means clustering to partition reference patterns for each gesture class into a small number of clusters and use the centroid of each cluster as the reference patterns, which is very similar to our method. One major difference is that Bong et al.'s K-Means clustering estimates the centroid of a cluster by resizing reference patterns into an average length so that the patterns become of a uniform length.

Chen et al. proposed a method to utilize DTW for fuzzy clustering. The use of DTW is limited to calculate pairwise distances of input patterns to build a distance matrix as an input to fuzzy clustering algorithm. DTW is not used at runtime for classifying test patterns. Their specific algorithm runs in the complexity of $O(2^n - 1)$ [1]. Our method is different in that we use DTW both as a distance measure and as a tool for deriving cluster centers. Also, at runtime, DTW-based simple 1-NN algorithm is enough for classifying test patterns.

Oates et al. introduced time series clustering methods that are based on DTW and HMM. They utilized DTW not only as a distance measure but also as a method for finding cluster centers. To find the center of a cluster, their algorithm first selects a pattern that minimizes distance to all other patterns in the cluster.

Then non-selected patterns are warped into the selected pattern. Then a simple averaging is used to derive the center pattern [2]. Their approach is similar to ours in that DTW is utilized for calculating cluster centers. But they did not clearly show the details of their method. We present details of our method and algorithms for deriving cluster centers using DTW.

5 Conclusion

Clustering temporal patterns is not easy because temporal patterns are non-uniform in length and discriminating features are not temporally aligned. We proposed an algorithm for calculating centers of temporal patterns based on DTW, which enables us to implement K-Means clustering for temporal patterns. We validated the performance of the algorithm by utilizing it for optimizing the number of reference patterns of a 1-NN classifier that learns by instance-based learning. We observed, in our experiment, that reducing the number of reference patterns down to 5% using the proposed clustering algorithm did deteriorate the recognition performance of the classifier only marginally. On the other hand, after reducing the number of reference patterns, the classifier performed more than 10 times faster, which is a critical benefit to utilize such classifiers in resource-bounded platforms.

Acknowledgment

The IT R&D program of MKE/IITA supported this work. [KI001836, Development of Mediated Interface Technology for HRI].

References

1. Chen, G., Wei, Q., Zhang, H.: Discovering similar time-series patterns with fuzzy clustering and DTW methods. In: The Proceedings of IFSA World Congress and 20th NAFIPS International Conference, pp. 2160–2164 (2001)
2. Oates, T., Firoiu, L., Cohen, P.R.: Clustering time series with hidden Markov models and dynamic time warping. In: Proceedings of the IJCAI 1999 Workshop on Neural, Symbolic, and Reinforcement Learning Methods for Sequence Learning (1999)
3. Leong, T.S., Lai, J., Panza, J., Pong, P., Hong, J.: Wii Want to Write: An Accelerometer Based Gesture Recognition System (2009)
4. Liu, J., Wang, Z., Zhon, L., Wickramasuriya, J., Vasudevan, V.: uWave: Accelerometer-based Personalized Gesture Recognition, TR0630-08, Rice University and Motorola Labs (June 2008)
5. Kela, J., Korpipää, P., Mntyjärvi, J., Kallio, S., Savino, G., Jozzo, L., Marca, D.: Accelerometer-based gesture control for a design environment. Personal and Ubiquitous Computing 10(5), 285–299 (2006)
6. BongWhan, C., Jun-Ki, M., Sung-Bae, C.: Online gesture recognition for user interface on accelerometer built-in mobile phones. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010. LNCS, vol. 6444, pp. 650–657. Springer, Heidelberg (2010)

On Segmentation of Interaction Values

Nguyen Chi Lam[1], Hiep Xuan Huynh[1], and Fabrice Guillet[2]

[1] College of Information and Communication Technology Cantho University, Vietnam
[2] Polytechnic School of Nantes University, France

Abstract. Post-processing of association rules with interestingness measures is considered as one of the most difficult and interesting task in the research domain of Knowledge Discovery from Databases (KDD). In this paper, we propose a new approach to discover the behaviors of interestingness measures by modeling the interaction between them. The interaction values are calculated based on the capacity function (also called fuzzy measure) and then are segmented to discover the interaction's trends of clusters of interestingness measures.

Keywords: Knowledge Discovery from Databases, association rules, interestingness measures, correlation graph, capacity function, Sugeno measure, interaction value, segmentation.

1 Introduction

The analysis of the relations between interestingness measures based on the calculations of interestingness values from a data set structured by rules, especially association rules [1], is a challenge in Knowledge Discovery in Databases (KDD). Each interestingness measure is often considered as representing an aspect or a property on the data set studied. The process of determining the approriated interestingness measures for a data set always a complex problem to resolve. In addition, the use of only one interestingness measure to reflect the whole data set is very difficult. In reality, each data set has itself many aspects that are not express obviously.

In this paper, we deal with a new approach based on the interaction between interestingness measures[3] [4] [7] , using the a graph model called interaction graph. After the determination of clusters of interestingness measures and the selection of the approriate interestingness measures that are strongtly represent the specific aspects of the data set, we can easily rank the association rules generated from the data set according to the order of the common interestingness measure obtained from the group of interestingness measures chosen.

2 Interestingness Measures

Suppose that we have a data set D contains the transactions in a supermarket [1]. The set of items (itemset) $I = \{i_1, i_2, ..., i_n\}$ corresponds with a set of available articles in a supermarket. The set X, contains the articles appeared in a transaction. In fact, the transaction expresses the bought articles of custormers.

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 27–32.
springerlink.com

An association rule is represented in the form $X \rightarrow Y (X \cap Y = \emptyset)$ [1] with X itemset and Y itemset are generated by the algorithm APRIORI [1].

The interestingness value of an association rule is calculated via a function m with the parameters $n, n_X, n_y, n_{X\overline{Y}}$ (see Fig.1). In this case, the funtion m is called an interestingness measure $m(X \rightarrow Y) = f(n, n_X, n_Y, n_{X\overline{Y}})$ on association rules. The importance of an association rule depends strongly on the interestingness value that it can obtain from the corresponding interestingness measure.

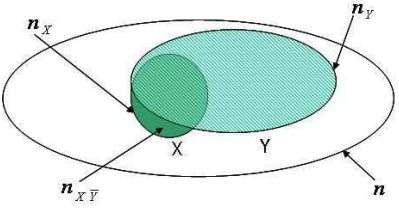


Fig. 1. Venn diagram for an association rule$X \rightarrow Y$.

3 Interaction between Interestingness Measures

3.1 Interaction Value

The interaction value between two interestingness measures is calculated by a capacity function (i.e., a fuzzy measure, a measure of capacity) [2] [3] [4]. The capacity function μ on a set Θ of interestingness measures $\mu : 2^{\mathcal{Q}} \rightarrow [0, 1]$ with the following axioms [2] [9]:

- $\mu(\emptyset) = 0$ and $\mu(\mathcal{Q}) = 1$ (normalization).
- $\mu(A) < \mu(B)$ when $A \subseteq B \subseteq \mathcal{Q}$ (monotonicity)

The value $\mu(A)$ demonstrates the capacity of a subset A of interestingness measures. A capacity function Sugeno [9] or called a capacity with the parameter λ (λ-measure), if exists a value $\lambda > -1$ so that :

$$\mu(A \cup B) = \mu(A) + \mu(B) + \lambda\mu(A)\mu(B) \tag{1}$$

with $A \subseteq \mathcal{Q} and A \subseteq \mathcal{Q}$.

The capacity function Sugeno can be determined by the method "Singleton Fuzzy Measure Ratio Standard" [10]. When the function μ does not carry on the additive property ($\mu(A \cup B) \neq \mu(A) + \mu(B)$) we have :

$$1 + \lambda = \prod_{i=1}^{q} (1 + \lambda\mu(x_i)) \tag{2}$$

with : $q = |\mathcal{Q}|$.

3.2 Interaction Matrix

Each interestingness measure is studied the interaction relations with the $q-1$ remaining interestingness measures. In our work, we only focus on the interaction relations from each couple of interestingness measures m_i, m_j so that the interaction value ξ_{ij} between two interestingness measures is peformed by the formula [7][11]:

$$\xi_{ij} = \mu(\{m_i, m_j\}) - \mu(\{m_i\}) - \mu(\{m_j\}) \quad (3)$$

A remark is that the interaction valueξ_{ij} between to interestingness measures m_i and m_j has the symmetric property $\xi_{ij} = \xi_{ji}$. After calculating the interaction values between each couple of interestingness measures, we have a matrix of interaction values ξ (i.e., interaction matrix):

$$\xi = \begin{bmatrix} \xi_{11} & \xi_{12} & \dots & \xi_{1q} \\ \xi_{21} & \xi_{22} & \dots & \xi_{2q} \\ \dots & \dots & \dots & \dots \\ \xi_{q1} & \xi_{q2} & \dots & \xi_{qq} \end{bmatrix}$$

3.3 Interaction Graph

For an intuitive observation of the interactive relations obtained from an interaction matrix, we have proposed a graph model (i.e., interaction graph). The interaction graph is an extension of the correlation graph [5] [11].

In the interaction graph, $G = (V, E)$

- V : each vertex represents an interestingness measure.
- E : each edge is the interaction value between the two vertices

An interaction graph with q interestingness measures will have $\frac{(q-1)q}{2}$ edges. It is difficult to observe and to evaluate the interaction relations, especially the strongly interaction relations. The interaction relation exists when the interaction value is greater or equal a predetermined threshold τ , called τ-interaction.

3.4 Tool for Interaction Graph

The model of interaction graph is implemented in the INTERACTION module of the ARQAT tool [6]. This module also provide the important means to calculate the interaction values between the interestingness measures via the capacity function Sugeno. The module performs in the following steps :

1. Determine the first initialized values of the capacity function for each interestingness measure from the data set or the user's points of view by the submodule CAPACITYINITIALISATION.
2. Calculate the values of the capacity function with the algorithm "Singleton Fuzzy Measure Ratio Standard" [10] by using the submodule CAPACITY-COMPUTATION.

3. Perform the calculation of the interaction values and establish the interaction matrix by the submodule INTERACTIONCOMPUTATION.
4. Build and visualize the interaction graph by the submodule INTERACTIONVIEW.

4 Experiments

4.1 Values of Capacity Function

The initial values of capacity function of each interestingness measure is determined by the ratio between the number of different interestingness values obtained and the total number of association of a rule set. (see table 1).

Table 1. Initially capacity values of interestingness measures.

Interestingness measures	Initially capacity values		Interestingness measures	Initially capacity values
Conviction	0.54		Confidence	0.002
EII ($\alpha = 1$)	0.604		Dependency	0.062
EII ($\alpha = 2$)	0.536		Phi-Coefficient	0.014
F-measure	0.016		Rule Interest	0.438
Implication index	0.704		Support	0.086
...	...		...	...

With the initially capacity values in the table 1, we can calculate the capacity value of Sugeno for each interestingness measure by the method "Singleton Fuzzy Measure Ratio Standard" [10] (see table 2). Table 2 describes in detail the values of the Sugeno's capacity function obtained from the rule set.

Table 2. The capacity value of Sugeno for each interestingness measure.

Interestingness measures	Sugeno's capacity values		Interestingness measures	Sugeno's capacity values
Causal Confidence	0.0002		J-measure	0.0384
Causal Confirm	0.0020		Kappa	0.0017
Collective Strength	0.0082		Least Contradiction	0.0037
Confidence	0.0002		Lerman	0.0037
...	...		...	...

4.2 Interaction Values

A part of interaction values obtained from interestingness measures on the rule set is represented in table 3. We can see that the measures EII and Implication Index has the highest value of interaction (the interaction value is 1).

Table 3. The capacity value of Sugeno for each interestingness measure.

Interestingness measures 1	Interestingness measures 2	Interaction values
EII	Implication Index	1
Conviction	Implication Index	0.892268317
EII 2	Implication Index	0.885549175
Conviction	EII	0.763155019
EII	EII 2	0.757408143
Implication index	Rule Interest	0.721446156
...	...	...

4.3 Interaction Graph

We can observe rank of interaction on an interaction matrix based on the determination of different values of the threshold -interaction. For instance, to observe the different levels of strong interaction, we can start from the couple of interestingness measures having the the strongly interaction values and the other couples of interestingness measures corresponding to the decrease of interaction values (Fig. 2,).

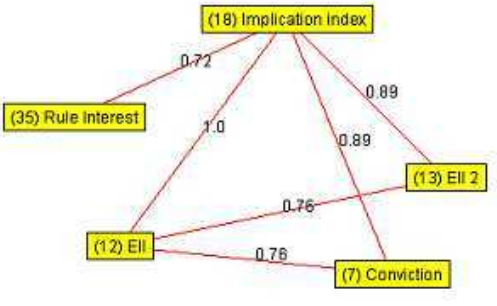


Fig. 2. An interaction graph.

On Fig. 2, we can observe the affect the measure Implication Index with regard to the other interestingness measures in a cluster of 5 interstingness measures Implication Index, EII, Conviction, EII 2, Rule Interest. The number of interestingness measures in a cluster increases according to the decrease of the value of τ-interaction.

5 Conclusion

We can evaluate the behaviors of interestingness measures by segmenting the set of interestingness measures in the interaction graph. By using the capacity function of Sugeno (corresponding to the parameter value of $\lambda = 0.5$), at the first step, we have modeled the interaction between 40 interestingness measures via the interaction matrix and the interaction graph. The observation of the interaction between interestingness measures can be performed via the decrease of interaction values (i.e., the threshold τ-interaction) or the segmentation of interaction values. Based on the clusters of interstingness measures found, the user can select the appropriated clusters of interestingness measures to evaluate the quality of knwoledge represented in the form of association rules.

Acknowledgements

This publication was made possible through support provided by the IRD-DSF.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proceedings of 20th International Conference on Very Large Data Bases, VLDB 1994, pp. 487–499 (1994)
2. Choquet, G.: Theory of capacities. Annales de l'Institut Fourier 5, 131–295 (1953)
3. Grabisch, M.: The application of fuzzy integrals in multicriteria decision making. European Journal of Operational Research 89(3), 445–456 (1996)
4. Grabisch, M., Roubens, M.: An axiomatic approach to the concept of interaction among players in cooperative games. International Journal of Game Theory (28), 547–565 (1999)
5. Huynh, X.-H., Guillet, F., Briand, H.: Evaluating interestingness measures with linear correlation graph. In: Ali, M., Dapoigny, R. (eds.) IEA/AIE 2006. LNCS (LNAI), vol. 4031, pp. 312–321. Springer, Heidelberg (2006)
6. Huynh, H.X.: Interestingness measures for association rules in a KDD process: postprocessing of rules with ARQAT tool, PhD Thesis, Nantes University (2006)
7. Marichal, J.L.: Aggregation of interacting criteria by means of the discrete Choquet integral, Aggregation Operators: New Trends and Applications. Studies in Fuzziness and Soft Computing, vol. 97, pp. 224–244 (2002)
8. Murofushi, T., Soneda, S.: Techniques for reading fuzzy measures (iii): Interaction index. In: Proceedings of the 9th Fuzzy Systems Symposium, pp. 693–696 (1993)
9. Sugeno, M.: Theory of fuzzy integrals and its applications, PhD Thesis, Tokyo Institute of Technology (1974)
10. Takahagi, E.: On identification methods of λ-fuzzy measure identification using weights and λ. Japanese Journal of Fuzzy Sets and Systems 12(5), 665–676 (2000)
11. Lam, N.C., Huynh, H.X., Guillet, F.: On interestingness measures interaction. In: IEEE RIVF 2008: Proceedings of the 6th IEEE International Conference on Computer Sciences: Research & Innovation - Vision for the Future, Ho-chi-minh Ville, Vietnam, pp. 161–166 (2008)

SI-APRENDE: An Intelligent Learning System Based on SCORM Learning Objects for Training Power Systems Operators

Liliana Argotte[1], G. Arroyo-Figueroa[1], and Julieta Noguez[2]

[1] Instituto de Investigaciones Electricas
Cuernavaca, Morelos, 62490, Mexico
{largotte, garroyo}@iie.org.mx
[2] Tecnologico de Monterrey, Campus Ciudad de Mexico,
Mexico, D.F., 14380, Mexico
jnoguez@itesm.mx

Abstract. This paper presents the architecture of the Intelligent Learning Systems for training of power systems operators and describes one of its components: the tutor module. Operators can acquire knowledge in different ways or with different paths of learning, also called models of sequence. The tutor model is an adaptive intelligent system that selects the sequence of the learning material presented to each operator. The adaptive sequence is represented as a decision network that selects the best pedagogical action for each specific operator. The decision network represents information about the current state of the tutor, their possible actions, the state resulting from the action of the tutor and the usefulness of the resulting state. The model was evaluated using graduate students with good results. Based on the adaptive model, we developed an Intelligent Learning System called as SI-Aprende. The SI-Aprende system manages, spreads and promotes the knowledge by mean of the search and recovery of SCORM Learning Objects.

Keywords: adaptive learning, intelligent environment, learning objects, SCORM, sequencing model.

1 Introduction

Training of operators is an important problem faced by electrical power plants. The process of learning how to control, maintain and diagnose in a complex industrial environment takes years of practice and training. An operator must comprehend the physical operation of the process and must be skilled in handling a number of abnormal operating problems and emergencies. The problem increases when the complexity of the system obstructs the efficiency, reliability and safe operation. Therefore, the novice and experienced operators need continuous training in order to deal reliably with uncommon situations.

Most of the learning proposals of intelligent systems in education are based on Intelligent Tutoring Systems (ITS) [1], which are interactive learning environments that have the ability to adapt to a specific student during the teaching process.

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 33–38.
springerlink.com

This paper describes the architecture and development of an intelligent environment (IE) for training of power systems operators. The IE takes elements of artificial intelligence such as student modeling, intelligent tutors, intelligent sequence, concept maps and reusable Learning Objects (LO) to build a tool that allows meeting the requirements for learning and training of power system operators. In contrast with a traditional training system, the main goal of the intelligent environment is to certify operators in knowledge, skills, expertise, abilities and attitudes for operation of power systems.

2 Architecture of the Intelligent Learning System SI-Aprende

The architecture of the IE is based on the generation of dynamic courses generating systems proposed by Brusilovsky [2]. The IE is composed of four main components (see Figure 1): the domain knowledge module, the tutor, the operator model, and the learning management system (LMS).

The first module contains the knowledge of the domain as LOs and concept structure maps. The tutor module is the component that generates the intelligent sequence of learning objects to be presented to the operator as a course. Taking as a basis the concept map, the pedagogical component and the operator model, the sequencing model generates a specific course for each operator. The operator model is used to adapt the IE to each specific operator and this model is divided in three subcomponents: cognitive component, operator profile and affective component. Finally, the LMS controls the interactions with the operator, including the dialogue and the screen layout.

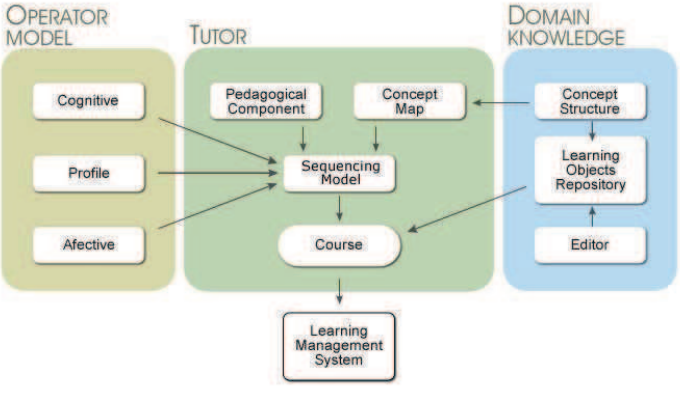


Fig. 1. Architecture of the intelligent environment (adapted of [3]).

This paper describes in detail one of the components of the IE, the Tutor Model. Detail description of the Operator Model, Domain Knowledge Model and LMS are already described in a previous work [4].

3 Tutor Module

The main component of the tutor module, also known as instructional or educational model, is the sequencing model that contains the sequence of LOs to be presented to an operator as a self instructional nontraditional course. The adaptive sequence is represented as a decision network that selects the best pedagogical action for each specific operator. The goal is to present to the operator the learning materials that better fits his learning needs.

A decision network or influence diagram represents information about the current state of the tutor, their possible actions, the state resulting from the action of the tutor and the usefulness of the resulting state. It can be seen as an extension to Bayesian Networks with random nodes, incorporating decision nodes and utility nodes. The sequencing model consists of the following random variables (see Fig. 3): knowledge of LO, satisfaction of the objective, progress of the objective, progress of the activity, score of LO, quiz, project, task, practice.

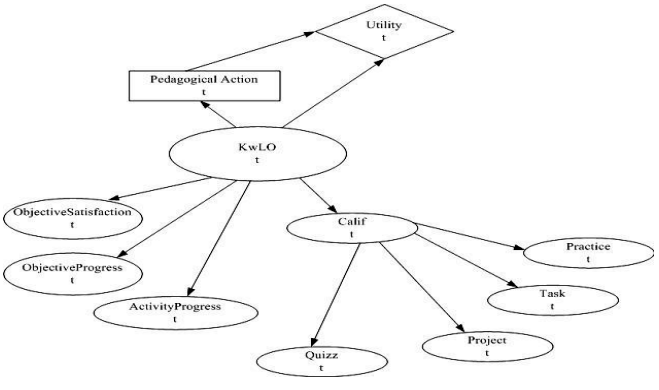


Fig. 2. Decision Network.

As shown in the figure above relations between the nodes are causal. KwLO random node has four values: Very Good, Good, Enough and Not Enough. The decision node considers the pedagogical actions that will be evaluated according to the utility function to select the best one, and in this work four possible pedagogical actions are identified: LO challenge, Next LO, Repeat refined and Repetition of LO. The calibration of the decision network is given by the experts in the domain. The utility table entries are set based on the teacher's experience about the best over all possible pedagogical actions according to the given knowledge state and the student's interaction. Adding the evidence of the previous Knowledge of the LO's random node, as well as the previous pedagogical action the DN was turned into a dynamic decision network (DDN).

4 Model Validation

A model validation is been carried out in two stages; the first one was evaluated in an academic environment, and the second one is running in a business environment proposed in this work. We have not finished last, because the training course of the Electrical Power Plants operators is too long and the operators are still using the Intelligent Environment.

Following, the first evaluation process will be described it. In the first stage, three groups of summer students were selected at the graduate degrees from Tecnológico de Monterrey, Mexico City campus. The courses were taught by different instructors, with a total population of 58 students (N = 58) distributed in Table 1 as follows:

Table 1. Groups of Students.

Course	# Students	
	Focus Group	Control Group
Math II	15	15
Electricity and Magnetism	9	9
Introduction to Physics	5	5
	NFocus= 29	NControl= 29

The Focus Group of students used the learning objects of IE while the Control Group used the traditional planning course. The LOs were designed as additional support for the students to have more tasks and exercises that they can perform anywhere, anytime. Three focus groups were heterogeneous in the level of subject knowledge, skills and even interest, since they were formed by students from different disciplines and professional careers. The time period in which the system was used was 15 days, having unlimited attempts at each learning activity and varied resources, from static HTML pages or text documents to simulations and videos.

The starting point was a diagnostic evaluation (PreTest) of the topic to all students of the course on a scale of 0-100. After applying the pre-test, the instructors made the separation of focus and control groups, giving appropriate details in both cases, indicating the time that their learning activities would take. The material prepared by instructors consisted of tasks, projects, practices and self-assessments of a specific topic of the course. The learning objective was the same for both groups, but the way of reaching the goal was different for each group.

The instructors created four LOs about a specific topic of the course and all students in the focus group started from the same LO. If the student received very good grades (or equivalent numeric 9 <= 10 on a scale of 1 to 10) in the first LO, it was designed a greater challenge or a higher level of difficulty for that student, seeking to have material that fits the needs and learning pace of each student. If a student got a good grade (or its equivalent numeric 7 <= 9 on a scale of 1 to 10) it was designed a LO with the same level of difficulty of the first LO. If the student had enough grade (or equivalent number 6 <= 7 on a scale of 1 to 10) it was designed a LO with lower level of difficulty that the student may reach the desired learning objective or could even repeat the same LO. If the rating was not enough for the student (or its numeric equivalent 1 <= 6 on a scale of 1 to 10) the student got the less

difficult LO. This design was established by the instructors but the pedagogical action that takes the system is the result of the propagation of a dynamic decision network that considers the knowledge of the LO and other variables, such as the progress of the objective, progress of learning activity and the satisfaction of the objectives for each student.

After two weeks of using the system, the instructors applied a final evaluation (PostTest) to both groups on the subject of pre-test, obtaining the necessary data to calculate the learning gain in both groups. According to Hake [5], the following formulas were applied to determine the learning gain in both groups:

- Average pre-Test

$$< Pre >= \frac{\sum_i Pre_i}{N} \quad (1)$$

- Average post-Test

$$< Post >= \frac{\sum_i Post_i}{N} \quad (2)$$

- Average Relative Gain

$$Grel = \frac{< Post > - < Pre >}{100 - < Pre >} \quad (3)$$

- Simple Gain

$$G = < Post > - < Pre > \quad (4)$$

We summarize our results in Table 2, where we show the average values for the pre-Test, the post-Test, the Relative gain and the Simple gain, for the focus and control groups of each course.

Table 2. Results of Control and Focus Group.

Group	Pre-Test (1)	Post-Test (2)	G (4)	Grel (3)
Math II Focus	17.5	90	72.5	0.88
Math II Control	10.5	71.63	61.13	0.68
Electricity and Magnetism Focus	8.33	74.67	66.34	0.72
Electricity and Magnetism Control	10	69.11	59.11	0.66
Introduction to Physics Focus	31.25	79	47.75	0.69
Introduction to Physics Control	27.5	55.33	27.83	0.37

As it was shown in the tables, students using the IE had higher relative and simple learning gains as compared to those students that did not used it.

Although the electrical operators' evaluation process has not finished yet, the qualitative opinions show that SI-APRENDE provides a more dynamic and interactive training to the Operators of Power Plants, where the employees really experience the acquisition and transfer of skills and knowledge.

5 Implementation of the Intelligent Learning System SI-Aprende

In order to have an environment with a suitable training to certify personnel in knowledge, skill, expertise, abilities and attitudes for operation of power systems, we developed an Intelligent Learning System called SI-Aprende [6].

The SI-Aprende system manages, spreads and promotes the knowledge of the power utility, by mean of the search and recovery of SCORM LOs. SI-Aprende composed by computer based training components labeled as reusable learning objects; a learning object repository; a tutor module based on decision networks to select the best pedagogical action for each operator.

6 Conclusions and Future Work

In this paper we have presented the architecture of an Intelligent Environment for training of Power Systems Operators. One of the components was described: the Tutor Module, which contains the sequencing model. An empirical evaluation of the sequencing model is presented using graduate students with good results. Based on the conceptual model of the IE, we developed an Intelligent Learning System called as SI-Aprende. SI-Aprende composed by computer based training components labeled as reusable learning objects; a learning object repository; a tutor module based on decision networks to select the best pedagogical action for each operator. The intelligent tutor module will be a dynamic course generating system interacting with the Operator Model to produce specific training material for each operator. Currently SI-Aprende has been loaded with 130 learning objects for electric power operators training.

References

1. Peachey, D.R., McCalla, G.I.: Using planning techniques in intelligent tutoring systems. Int. Journal Man-Machines Studies 24, 77–98 (1986)
2. Brusilovsky, P., Vassileva, J.: Course sequencing techniques for large-scale web based education. Int. Journal Cont. Engineering Education and Lifelong Learning 13(1/2), 75–94 (2003)
3. Hernández, Y., Noguez, J., Sucar, E., Arroyo-Figueroa, G.: A probabilistic model of affective behavior for intelligent tutoring systems. In: Gelbukh, A., de Albornoz, Á., Terashima-Marín, H. (eds.) MICAI 2005. LNCS (LNAI), vol. 3789, pp. 1175–1184. Springer, Heidelberg (2005)
4. Arroyo-Figueroa, G., Hernández, Y., Reyes, A., Enrique Sucar, L.: Intelligent Environment for training of power systems operators. In: CERMA 2008, pp. 27–31 (2008)
5. Hake, R.: Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. Am. J. Phys. 66(1), 64–74 (1998)
6. Argotte, L.: Intelligent E- Learning model for adaptive sequence of learning objects. Msc Thesis, Instituto Tecnológico y de Estudios Superiores de Monterrey, Campus Ciudad de México (2010) (in Spanish)

Least Squares Support Vector Machines for Channel Prediction in the MIMO System

Jerzy Martyna

Institute of Computer Science, Jagiellonian University,
ul. Prof. S. Lojasiewicza 6, 30-348 Cracow, Poland

Abstract. A new LS-SVM method for a multiple-input multiple-output (MIMO) channel prediction is presented. A least squares support vector machine (LS-SVM) is proposed as a prediction technique. The LS-SVM has nice properties in that the algorithm implements nonlinear decision regions, converges to minimum mean squared error solutions, and can be implemented adaptively. We also formulate a recursive implementation of the LS-SVM for channel prediction in the MIMO system. The performance of the new method is shown by a simulation of the bit error rate in the given environment.

Keywords: channel estimation, multiple-input multiple-output (MIMO) channel, least squares support vector machine (LS-SVM).

1 Introduction

During the last several years support vector machines (SVM) and kernel methods [1], [10] have generated considerable interest and have been used in many applications from image processing to analyzing DNA data. These methods belong to one of the preeminent machine learning statistical models for classification and regression developed over the past decade.

The multiple-input multiple-output (MIMO) communication systems have recently drawn considerable attention in the area of wireless communications. Multiple antenna systems are one of the key technologies for the next generation wireless communications, especially in WCDMA-based 3G systems [5]. This technique has attracted much attention due to its advantage in capacity as well as the ability to support multiple users simultaneously [3]. In the MIMO system it is usually assumed that the channel state information (CSI) is known at the receiver, but not at the transmitter, and the transmitter has no knowledge about the channel coefficients. Although the increasing number of antennas improves the capacity of the system, it increases its complexity. However, these methods waste time while learning the channel when meaningful data can be sent.

The use of the SVM method and recently developed the least squares (LS) SVM method [7] have been proposed to solve a number of digital communication problems. Among others, signal equalization and detection for a multicarrier (MC)-CDMA system based on SVM linear classification has been investigated by Rahman et al. [4]. In the paper by Sánchez-Fernández et al. [6] SVM techniques

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 39–44.
springerlink.com

have been used for a nonlinear channel estimation for multiple-input multiple-output systems. The SVM technique for a robust channel estimation in the OFDM data structure was proposed by M.J. Fernández-Getino Garcia et al. in the paper [2]. However, none of these solutions can be used easily in on-line adaptive algorithms.

In this paper, we propose a LS-SVM solution for channel prediction. There are two novelties in our proposal. We use the LS-SVM for channel prediction in a MIMO system in which inequality constraints are replaced by equality constraints and a quadratic error criterion is used.

This paper is organized as follows. Section 2 presents the system description and the problem formulation. Section 3 introduces a recursive least squares algorithm that can be implemented on-line. In section 4, the proposed algorithm is then evaluated by means of computer simulations. Section 5 presents some final conclusions.

2 The System Description and Problem Formulation

In this section, the MIMO received model for an un-coded and a beam-forming system is introduced.

In the first case, we assume a MIMO wireless flat fading communication system with N_r receive antennas and N_t transmit antennas modelled by

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{1}$$

where $\mathbf{y}$ is the $N_r \times 1$ received vector, $\mathbf{x}$ is the $N_t \times 1$ transmitted symbol vector with each x_i belonging to constelation C with symbol energy E_s, and $\mathbf{n}$ is the while noise vector of size $N_r \times 1$ with $n_i \sim \mathcal{CN}(0, N_0)$. The channel state matrix $\mathbf{H} = \{h_{mn}\}$ gives a complex channel gain between the m-th receiver and the n-th transmit antenna.

In the second case, the received symbols are expressed in two scenarios, when the transmitter and the receiver have full channel information (CSI) and when they have the prediction matrix. Then, the channel matrix $\mathbf{H} = \mathbf{U} \cdot \mathbf{D} \cdot \mathbf{V^H}$ is a singular value decomposition (SVD) where $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices corresponding to the i-th non-zero singular value $\sigma_H(i)$, $(\sigma_H(1) \leq \ldots \leq \sigma_H(M))$ and $M = rank(\mathbf{H})$. Assuming that $\tilde{x} = v_1 \cdot \mathbf{x}$ we can obtain from Eq. (1) the received symbols u_1^H as follows:

$$u_1^H \mathbf{y} = \sigma_H(1)x + u_1^H \mathbf{n} \tag{2}$$

Assuming $\tilde{u} = \mathbf{u}_1^H \mathbf{n}$ we can obtain

$$E \mid \tilde{n} \mid^2 = N_r \cdot N_0 \tag{3}$$

Thus, the channel within a MIMO system is time-varying and can be expressed in a matrix notation as $\mathbf{H} = \hat{H} + E$. Therefore, the received symbols are as follows:

$$\hat{u}_1^H \mathbf{y} = (\sigma_H(1) + \hat{u}_1^H E \hat{v}_1)\mathbf{x} + u_1^H \mathbf{n} \tag{4}$$

In the general case the doppler spread of the signal is greater than the pulse bandwidth. In a typical environment the MIMO channel is fast-fading. Assuming a MIMO flat fast-fading transmission each sub-channel can be formulated as follows [11]:

$$h_{mn}(t) = h_{mn}(k) + j h_{mn}(k) \tag{5}$$

where in-phase component is represent as

$$h_{mn}^I(k) = \sqrt{\frac{2}{M}} \sum_{n=1}^{M} \cos(2\pi f_d k \sin(\alpha_n) + \Phi_n) \tag{6}$$

and the quadrature component can be written as

$$h_{mn}^Q(k) = \sqrt{\frac{2}{M}} \sum_{n=1}^{M} \cos(2\pi f_d k \sin(\alpha_n) + \Psi_n) \tag{7}$$

where $\alpha_n = \frac{2\pi n - \pi + \Theta}{4M}$ and Φ_n, Ψ_n, Θ are $U[-\pi, \pi]$.

3 The Multidimensional Recurrent LS-SVM

The recurrent LS-SVM based on the sum squared error (SSE) to deal with the function approximation and prediction has been proposed by Suykens and Vandewalle [8]. However, the so defined recurrent LS-SVM will not be adequate for the channel prediction in the MIMO system. It is caused by the lack of the high-dimensional reconstructed embedding phase space.

In order to extend the recurrent least squares vector machine to a multidimensional recurrent LS-SVM we introduce scalar time series $\{s_1, s_2, \ldots, s_T\}$ in the form

$$\hat{s}_k = f(\hat{s}_{k-1}), \quad k = m', m'+1, \ldots, N'+m'-1 \tag{8}$$

where m', N' are referred to the embedding dimension and the number of training data, respectively.

The function approximation is given by

$$\hat{s}_k = \mathbf{w}^T \phi_i(\hat{s}_{k-1}) + b, \quad k = m', m'+1, \ldots, N'+m'-1, \quad i = 1, 2, \ldots, m' \tag{9}$$

where $\mathbf{w} = [w_1, w_2, \ldots, w_{m'}]$ is the output weight vector, $b \in R$ is the bias term $\phi(.)$ is the nonlinear mapping function estimated by means of using training data.

The recurrent LS-SVM can be formulated as the quadratic optimization problem:

$$\min_{w_i, b_i, e_{k,i}} \mathcal{J}(w_i, b_i, e_{k,i}) = \frac{1}{2} \sum_{i=1}^{m'} w_i^T w_i + \frac{\gamma}{2} \sum_{k=m'+1}^{N'+m'-1} \sum_{i=1}^{m'} e_{k,i}^2 \tag{10}$$

subject to the following constraints:

$$\begin{cases} s_k^{(1)} - e_{k,1} &= w_1^T \phi_1(s_{k-1} - e_{k-1}) + b_1 \\ s_k^{(2)} - e_{k,2} &= w_2^T \phi_2(s_{k-1} - e_{k-1}) + b_2 \\ & \vdots \\ s_k^{m'} - e_{k,m'} &= w_{m'}^T \phi_{\prime}(s_{k-1} - e_{k-1}) + b_{m'} \end{cases} \tag{11}$$

where $k = m' + 1, m' = 2, \ldots, N' + m' - 1$, $e_k = s_k - \hat{s}_k$. Generally, the error term here is defined as

$$e_k = x_k - f(x_{k-1}) \tag{12}$$

4 Experimental Results

In order to test the performance of the MIMO channel prediction, we used the received signal-to-noise ratio (SNR) in the general form

$$\rho = \frac{\sigma_x^2}{\sigma_e^2 + \sigma_n^2} \tag{13}$$

where σ_x^2 is the average received signal power, σ_e^2 is the predictive error, σ_n^2 is the average noise variance. Thus, after some algebraic manipulations for the un-coded system we can obtain

$$\rho_{uc} \stackrel{\triangle}{=} \frac{E \parallel \hat{\mathbf{H}}\mathbf{x} \parallel_2^2}{E \parallel \mathbf{E}\mathbf{x} \parallel_2^2 + E \parallel \mathbf{n} \parallel_2^2} \tag{14}$$

and after several manipulations

$$\rho_{uc} = \frac{\sum_{i=1}^{M} E[\sigma_{\mathbf{H}}^2(i)]}{\sum_{i=1}^{N} E[\sigma_{\mathbf{E}}^2(i)] + \frac{N_r N_0}{E_s}} \tag{15}$$

where $N = rank(\mathbf{E})$, $\sigma_{\hat{\mathbf{H}}}(i)$ and $\sigma_{\mathbf{E}}(i)$ are the i-th non-zero singular values of $\hat{\mathbf{H}}$ and $\mathbf{E}$, respectively.

The MIMO beam-forming can be formulated as follows

$$\hat{\mathbf{u}}_1^{\mathbf{H}}\mathbf{y} = (\hat{\sigma}_1^2 + \hat{u}_1^{\mathbf{H}})x + \tilde{n} \tag{16}$$

Then, after a similar technique as the one used before, we can state the received SNR for the MIMO beam-forming system

$$\rho_{bf} = \frac{E[\sigma_1^2]}{E \mid \hat{\mathbf{u}}_1^{\mathbf{H}}\mathbf{U}\mathbf{D}\mathbf{V}^{\mathbf{H}}\hat{v}_1 - \hat{\sigma}_{max} \mid^2 + \frac{N_r N_0}{E_s}} \tag{17}$$

Thus, comparing the above equation with the Eq. (19) we get the value of the prediction error σ_e^2 for the beam-forming prediction, namely

$$\sigma_e^2 = E \parallel \mathbf{E}\mathbf{x} \parallel_2^2 = E \parallel \hat{\mathbf{U}}^{\mathbf{H}}\mathbf{E}\hat{\mathbf{V}}\mathbf{x} \parallel_2^2 \tag{18}$$

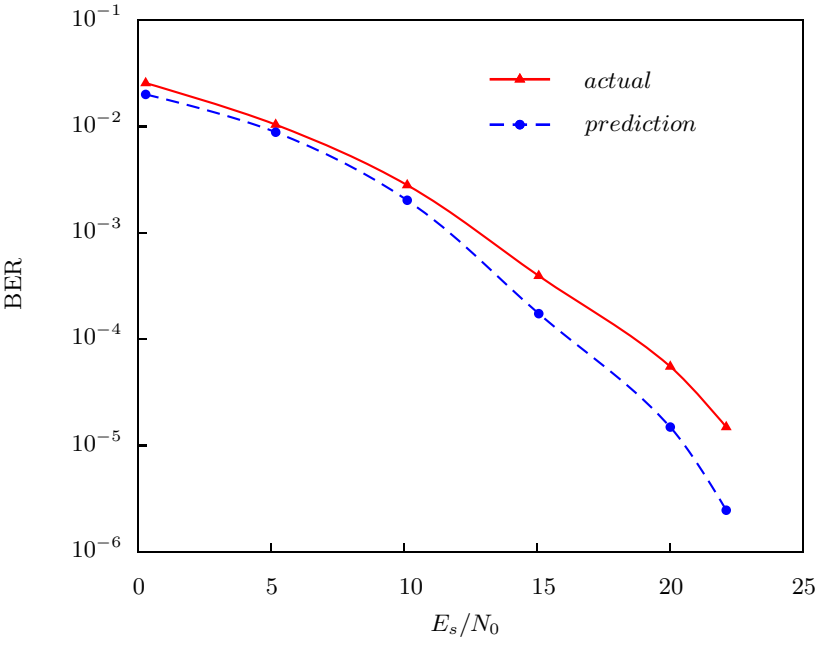


Fig. 1. Bit error rate (BER) as a function of E_s/E_0 for a un-coded 4×4 MIMO system

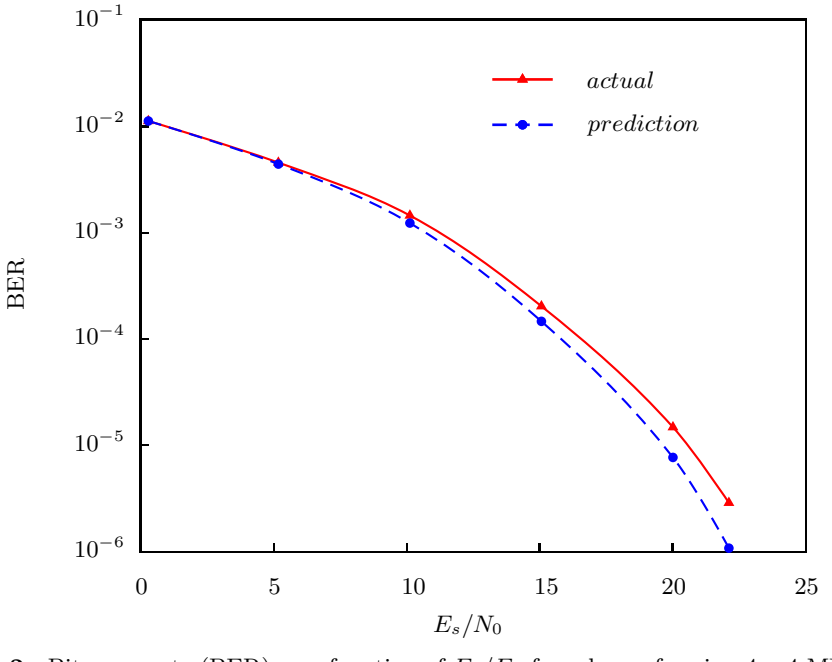


Fig. 2. Bit error rate (BER) as a function of E_s/E_0 for a beam-forming 4×4 MIMO system

or after some algebraic manipulations

$$\sigma_e^2 = E \mid (\hat{\mathbf{u}}_1^{\mathbf{H}} \mathbf{U}\mathbf{D}\mathbf{V}^{\mathbf{H}} \hat{\mathbf{v}}_1 - \hat{\sigma}_{max}) x \mid^2 \tag{19}$$

The RBF kernel with the width parameters: $\sigma = 0.5$ and the regularizotion parameter $\gamma = 500$ were selected, respectively.

The obtained values of the maximum singular value of the error matrix and the minimum singular value of the error matrix are given in Fig. 1 and Fig. 2, respectively.

5 Conclusion

A recurrent LS-SVM method was used in order to predict a MIMO channel. The received SNR for an un-coded and beam-forming MIMO system was derived. The proposed solution does not need to use the analytic mathematical model of performance measures. However, it can provide the parameters of the system in the varying time horizon. The experiment results show that the proposed method achieves the best performance for the RBF kernel function. This kernel function demands the use of scaling factors for all input parameters.

References

[1] Cortes, C., Vapnik, V.: Support Vector Networks. Machine Learning 20, 273–297 (1995)
[2] Fernández-Getino Garcia, M.J., Rojo-Álvarez, J.L., Alonso-Atienzo, F., Martinez-Ramón, M.: Support Vector Machines for Robust Channel Estimation in OFDM. IEEE Signal Processing Letters 13(7), 397–400 (2006)
[3] Jindal, N., Vishwanath, S., Goldsmith, A.: On the Duality of Gaussian Multiple-Access and Broadcast Channels. IEEE Trans. on Information Theory 50(5), 768–783 (2004)
[4] Rahman, S., Saito, M., Okada, M., Yamamoto, H.: An MC-CDMA Signal Equalization and Detection Scheme Based on Support Vector Machines. In: Proc. 1st Int. Symp. Wireless Communication Systems, pp. 11–15 (2004)
[5] 3GPP TR25.869 Tx Diversity Solutions for Multiple Antennas, v1.2.0 (August 2003)
[6] Sánchez-Fernández, M.P., de Prado-Cumlido, M., Arenas-Garcia, J., Perez- Cruz, F.: SVM Multiregression for Nonlinear Channel Estimation in Multiple-Input Multiple-Output Systems. IEEE Trans. on Signal Processing 52(8), 2298–2307 (2004)
[7] Suykens, J., Lukas, L., Vandewalle, J.: Least Squares Support Vector Machine Classifiers. Neural Processing Letters 8(3), 293–300 (1999)
[8] Suykens, J.A.K., Vandewalle, J.: Recurrent Least Squares Vector Machines. IEEE Trans. Circuits Systems and Systems I 47(7), 11109–11114 (2000)
[9] Suykens, J.A.K., Van Gestel, T., De Brabantter, J., De Moor, B., Vandewalle, J.: Least Squares Support Vector Machines. World Scientific Pub. Co., Singapore (2002)
[10] Vapnik, V.: Statistical Learning Theory. John Wiley and Sons, New York City (1998)
[11] Zheng, Y., Xiao, C.: Improved Models for the Generation of Multiple Uncorrelated Rayleigh Fading Waveforms. IEEE Trans. Commun. Letters 6(6) (2002)

A Neural Computation Model Based on nCRF and Reverse Control Mechanism

Hui Wei and Xiao-Mei Wang

Department of Computer Science, Laboratory of Cognitive Model and Algorithm, Brain Science Research Center, Fudan University, 200433 Shanghai, China
{weihui,071021059}@fudan.edu.cn

Abstract. Previous nCRF models are mainly based on fixed RF whose dynamic characteristics are not taken into account. In this paper, we establish a multilayer neural computation model with feedback for the basic structure of nCRF. In our model, GC's RF can dynamically and self-adaptively adjust its size according to stimulus properties. RF becomes smaller in local areas where the image details need distinguishing and larger where the image information have no obvious difference. The experimental results fully reflect the dynamic characteristics of GC's RF. Among adjacent areas in an image, similar ones are integrated together and represented by a larger RF, while dissimilar ones are separated and represented by several smaller RFs. Such a biology-inspired neural computation model is a reliable approach for image segmentation and clustering integration.

Keywords: Non-classical receptive field, Neural network, Image representation.

1 Introduction

Machine vision is of great significance to intelligent computing systems. However, the vast majority of machine vision researchers overlooked the principle of visual physiology. They proposed various algorithms from the engineering point of view. Therefore there is a lack of unitive and effective machine vision algorithms to solve the problem of complex scenes segmentation. In contrast, the human vision system has a strong image processing ability. Therefore, it is entirely possible that we find some new breakthroughs to solve the difficulties encountered in computer vision by applying the theories of human vision, neuropsychology and cognitive psychology. In the human vision system, the retina is also known as "peripheral brain". It processes visual information preliminarily. The ganglion cell (GC) is the final place of retinal information processing. The receptive field (RF) is the basic structural and functional units for information processing in visual system. GC's RF determines the processing result and response characteristic of the retinal complex network. Therefore, RF is instructive for visual algorithm to simulate the mechanism of information processing in the GC.

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 45–50.
springerlink.com

Each visual neuron only elicits response to the stimulation in a specific area of the retina (or visual filed). The area is called as the classical receptive field (CRF) of the neuron. The CRF of the GC has an antagonistic center-surround structure. With the development of in-depth studies on human vision system, many researchers [1, 2] found that there is still a large area outside the CRF. Stimulating such area alone fails to elicit neural responses but can modulate the neural response to CRF stimulation. The area is called as non-classical receptive field (nCRF). Sun C. et al. [3] found that stimulating CRF alone can also elicit neural response when the nCRF receive large area and appropriate stimuli. Disinhibitory nCRF helps to transmit the luminance and grads of image local area [4].

Some models were built to stimulate disinhibitory nCRF. Li Z. et al. [5] proposed a function with the shape of a volcano whose center is concave. Ghosh K. et al. [6] proposed a linear function of three-Gaussian and explained low-level brightness–contrast illusions. Qiu F. T. et al. [7] gave a model for the mechanism of mutual inhibition within disinhibitory nCRF. These models were established by simple reference to the mechanism of nCRF. They were just used in contrast enhancement, edge extraction, etc. and had no top-down feedbacks. However the modulations from high-level are very important in the neurobiology, which has been proved by electrophysiology, anatomy and morphology.

Research shows that ganglion RF can dynamically change its size [8]. The size of RF varies with the changes of brightness, background, length of time of stimulation, speed of moving objects and so on. Previous nCRF models are mainly based on fixed RF, whose dynamic characteristics are not taken into account.

Our computational model is not for segmentation. Simulating biological visual system needs to address three fundamental issues: first, which features to automatically choose for representation; second, what to use to achieve representation; third, can biological nervous systems achieve this task? To make a computational model solve these three problems simultaneously is very challenging and far exceed the scope of the classical segmentation. By establishing ganglion cell's dynamic receptive fields and its self-adaptive mechanism, we realized automatic analysis and extraction of block features. By simulating the array of ganglion cells we achieved representation. And the neural circuit we developed is supported by the neurobiological evidence. Therefore, this paper is aimed at how to self-adaptively extract features and form internal representations for images by a bio-inspired structure. Limited to the length of the paper, we just presented a limited number of experimental results.

2 Physiological Basis

2.1 Neurophysiological Mechanism of nCRF of Retinal GC

Neurophysiological studies show that [9] the horizontal cell (HC) with a wide range of electrical synaptic connections can receive inputs from large areas of retinal receptor cells. The HC affects the activities of receptor cells (RCs) by means of spatial summation and feedback and then elicits the GC responses to the stimuli within the disinhibitory nCRF through bipolar cells (BCs). Moreover, the amacrine

cell (AC) connects many of the nearby ganglion cells (GCs) in the horizontal direction through its extensive dendritic branches. Amacrine cells (ACs) also interconnect with each other. The breadth of AC's connection goes far beyond GC's CRF surround, So AC is properly related to the formation of nCRF with a wide range.

Thus, HC and AC play the role of information integration in the outer and inner plexiform layer respectively. Retinal GC receive inputs from many neurons in the outer and inner plexiform layer, hence HC and AC are connected with the formation of nCRF of retinal GC.

2.2 Reverse Control Mechanism of Retina

The IPC sends messages to HC and BC in the outer plexiform [10]. The works of Yang et al. [11] show that IPC ceasing activity in light enhances HC activity. After a long period of dark adaptation, IPC accelerates activity, inhibits HC and decreases the peripheral antagonist. In addition, some fibers from mesencephalic center (MC) are found into the centrifugal fibers of the retina through optic nerves in all vertebrates. The fibers from MC control AC and IC. Electrical stimulation to centrifugal fibers can increase response of GC to lights. It is possibly caused by the disinhibition of IPC and AC. This shows that IPC takes backward control of HC and BC to change the size of CRF surround, and MC takes backward control of IPC and AC to change the size of CRF center through centrifugal fibers.

3 Calculation Model

3.1 Neural Circuit for Dynamically Adjusting of RF

Based on the above neurophysiological mechanism of retinal ganglion cell nCRF, the reverse control mechanism and fixation eye movement, the neural circuit showed in Fig. 1 is designed to adjust RF dynamically. The details of retinal micronetwork are very complex and partly unclear. So we simplify the RF mode appropriately simplification by omitting the contribution of HC to the formation of nCRF.

In Fig. 1, the axons of several RCs form synaptic connections with the dendrites of a BC. The RCs compose the CRF center of the BC. HC connects many of the nearby RCs in the horizontal direction through its dendritic branches. Horizontal cells (HCs) also interconnect with each other. The HCs integrate the responses of the RFs, transfer them to the BC, and inhibit BC's CRF center, and form BC's CRF surround. BCs with antagonistic center-surround CRF transfer their responses into GC and form the GC's CRF with antagonistic center-surround structure. AC connects many of the nearby GCs in the horizontal direction through its extensive dendritic branches. ACs also interconnect with each other. The ACs integrate the responses of the BFs, transfer them to the GC, and inhibit BC's CRF surround, thus forming GC's nCRF. IPC takes feedback control of HC and BC and changes the size of CRF surround and MC takes feedback control of IPC and AC and changes the size of CRF center through centrifugal fibers.

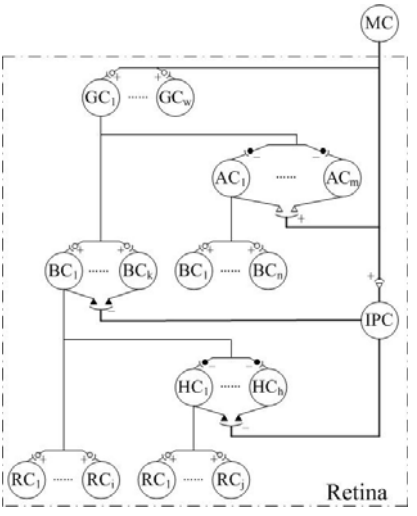


Fig. 1. Neural circuit for dynamically adjusting RF.

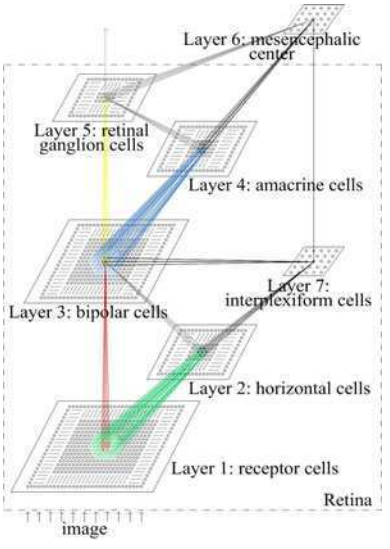


Fig. 2. The multi-layer network model for processing image.

3.2 Multi-layer Network Model for Image Processing

According to the above neural circuit for dynamically adjusting RF, we propose the multi-layer network model for processing image shown in Fig. 2.

3.3 Numerical Calculation Model of GC's RF

The output of the second layer can be expressed as follow:

$$GC(X,Y) = \sum_{y\in S_1}\sum_{x\in S_1} W_1 \cdot RC(x,y) - \sum_{y\in S_2}\sum_{x\in S_2} W_2 \cdot RC(x,y) + \sum_{y\in S_3}\sum_{x\in S_3} W_3 \cdot RC(x,y)$$
$$\text{where } W_1 = A_1 \cdot e^{-\frac{(x-x_0)^2+(y-y_0)^2}{\sigma_1^2}} \cdot \Delta s_1, W_2 = A_2 \cdot e^{-\frac{(x-x_0)^2+(y-y_0)^2}{\sigma_2^2}} \cdot \Delta s_2, W_3 = A_3 \cdot e^{-\frac{(x-x_0)^2+(y-y_0)^2}{\sigma_3^2}} \cdot \Delta s_3 \quad (1)$$

where, GC(X, Y) is the response of GC; RC(x, y) is image brightness projected onto RC within RF; S_1, S_2 and S_3 are CRF center, CRF surround and nCRF respectively; W_1, W_2 and W_3 are the weighting functions of the RCs within S_1, S_2 and S_3 respectively; A_1, A_2 and A_3 are the peak sensitivities of S_1, S_2 and S_3 respectively; σ_1, σ_2 and σ_3 are the standard deviations of the three weighting functions respectively, and $3\sigma_1$, $3\sigma_2$ and $3\sigma_3$ are the radii of S_1, S_2 and S_3 respectively; Δs_1, Δs_2 and Δs_3 are the bottom areas of the weights of the RCs in S_1, S_2 and S_3 respectively; x and y are position coordinates of RC; x_0 and y_0 are center coordinates of RF; X and Y are position coordinates of retinal GC.

In our experiment, pictures are imaged on the central retina with the range of 0-10 degrees eccentricity.

4 Results

We confirmed the change range of the size of RF of retina GC according to the physiological data. So, in the process of experiment, the change range was limited and cannot enlarge or reduce arbitrarily. We got the schematic diagram (as shown in Fig. 3) of RF dynamic adjustment process after tracking. In Fig. 3, red means that RF has reached the final size, and needn't to change; blue means that RF need to enlarge further; yellow means that RF need to reduce further; the real line denotes the actual size after this iteration; the broken line denotes the RF size that be replaced gradually in the process of iteration, which shows the change trail of RF. Observing the final size of RF (red RF) in Fig. 3, we found that the smallest RFs are always on the border of the image. In the background wall, the information is uniform within each color block, so the RFs representing the information within color block reach to the maximum. The RFs representing the information of the lamb, pillows, cup, flowerpot, the bottom of the sofa don't change when they just reach the edges. Therefore, from Fig. 4 we can see that RFs best represent the image information by dynamically adjusting their size in the limited change range.

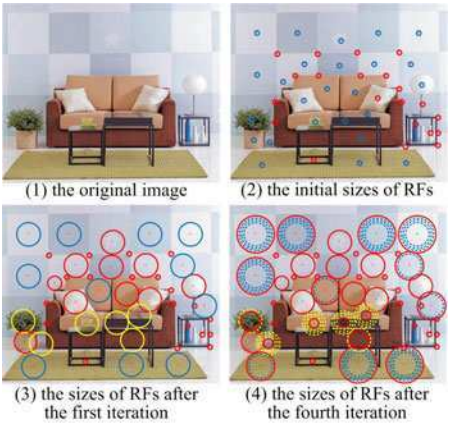


Fig. 3. The schematic diagram dynamic adjustment process.

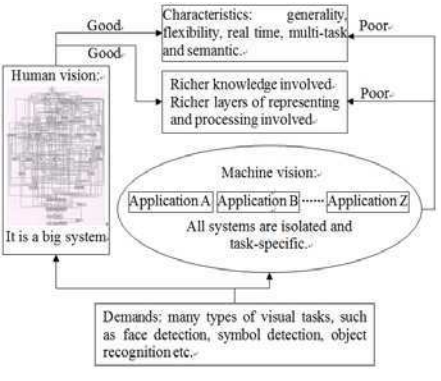


Fig. 4. The difference in image understanding between human vision and machine vision.

5 Discussion

The property of nCRF is not changeless. The nCRF can adjust its filter characteristics according to the spatial frequency components of images. And it is flexible toward changing contrasts and brightness of stimulations. Along with the changing of image spatial properties, nCRF sometimes turn into high spatial frequency filter and sometimes into low one. The nCRFs of most neurons are facilitatory under the condition of low contrast or low brightness, which shows that nCRF enhance the RF

response in order to detect contours of large-area image at this moment. After the increase of the contrast or the brightness, the nCRF will accordingly weaken the RF responses in order to detect the difference of the image inside and outside the RF. In large-area integration, this dynamic characteristic of nCRF provides necessary condition for visual system detecting Fig.-ground in a variety of stimulus conditions. Our current research shows the dynamic characteristic that RF varies with different stimuli.

Now there are two common senses in computer vision, saying that (a) an animal's vision system is much cleverer than a machine one, and (b) object recognition needs the help from higher level knowledge. To (a), the advantage of a biological vision system that is it has a systematic architecture with rich knowledge and rich layers of representing and processing. Fig. 4 shows the reason. To (b), what is knowledge needs to be clarified.

References

1. Ikeda, H., Wright, M.J.: The outer disinhibitory surround of the retinal ganglion cell receptive field. J. Physiol. 226, 511–544 (1972)
2. Krüger, J., Fischer, B.: Strong periphery effect in cat retinal ganglion cells. Excitatory responses in ON- and OFF-center neurons to single grid displacements. Exp. Brain Res. 18, 316–318 (1973)
3. Sun, C.: Spatial Property of Extraclassical Receptive Field of the Relay Cells in Cat's Dorsal Lateral Geniculate Nucleus and its Interaction with the Classical Receptive Field. PhD thesis, Fudan University (2004)
4. Li, C.Y., Pei, X., Zhou, Y.X.: Role of the extensive area outside the x-cell receptive field in brightness information transmission. Vision Res. 31, 1529–1540 (1991)
5. Li, Z., et al.: A New Computational Model of Retinal Ganglion Cell Receptive Fields- I. a Model of Ganglion Cell Receptive Fields with Extended Disinhibitory Area. Acta Biophysica Sinica 16, 288–295 (2000)
6. Ghosh, K., Sarkar, S.: Kamales Bhaumik: A possible explanation of the low-level brightness–contrast illusions in the light of an extended classical receptive field model of retinal ganglion cells. Biological Cybernetics 94, 89–96 (2006)
7. Qiu, F.T., Li, C.Y.: Mathematical simulation of disinhibitory properties of concentric receptive field. Acta Biophysica Sinica 11, 214–220 (1995)
8. Li, C.Y.: New Advances in Neuronal Mechanisms of Image Information Processing. Bulletin of National Natural Science Foundation of China 3, 201–204 (1997)
9. Yang, X.L., Gao, F., Wu, Samuel, M.: Modulation of horizontal cell function by GABA(A) and GABA(C) receptors in dark- and light-adapted tiger salamander retina. Vis. Neurosci. 16, 967–979 (1999)
10. Shou, T.D.: Brain mechanisms of Visual Information Processing. Shanghai Scientific & Technological Education Publishing House, Shanghai (1997)
11. Yang, X.L., Tornqvist, K., Dowling, J.E.: Modulation of cone horizontal cell activity in the teleost fish retina. II. Role of interplexiform cells and dopamine in regulating light responsiveness. J. Neurosci. 8, 2259–2268 (1988)

Particle Swarm Optimization of a Recurrent Neural Network Control for an Underactuated Rotary Crane with Particle Filter Based State Estimation

Sam Chau Duong[1], Hiroshi Kinjo[1], Eiho Uezato[1], and Tetsuhiko Yamamoto[2]

[1] Faculty of Engineering, University of the Ryukyus, Japan
[2] Tokushima College of Technology, Japan

Abstract. This paper addresses the control problem of an underactuated rotary crane system by using a recurrent neural network (RNN) and a particle filter (PF) based state estimation. The RNN is used as a state feedback controller which is designed by a constricted particle swarm optimization (PSO). As the study also considers the problem with assuming that the velocities of the system are not obtained, PF is utilized to estimate the latent states. Simulations show that the RNN could provide a superior evolutionary performance and less computational cost compared to a feedforward NN and that the PF is effective in estimating the unobserved states.

Keywords: recurrent neural network, particle swarm optimization, nonlinear control, underactuated system, particle filter, sequential Monte Carlo method.

1 Introduction

The recent immense growth of computing power has allowed several computation methods applying in broad areas. One of the most successful implementations is the use of stochastic and intelligent approaches in control engineering [1]. In particular, the applications of neural networks (NNs) and evolutionary algorithms (EAs) have brought much success and convenience in both offline and online designs [1] - [3].

Aside from several advantages of NNs, such as good nonlinear processing ability and robustness with inherently parallel architecture, recurrent NNs (RNNs) have interesting properties with great potential, such as attractor dynamics and internal memory, and they have shown superiority to feedforward NNs (FNNs). Since a recurrent neuron already has an internal feedback loop, it captures the dynamic response of a system without external feedback through tapped delays [4], RNNs are thus dynamic mappings and are more appropriate than FNNs when applied to dynamical systems. Nevertheless, despite important capabilities, RNNs are much less popular than FNNs because it is hard to develop a convenient learning algorithm as well as classic gradient-based algorithms are apparently insufficient [5]. In order to overcome the problems of RNNs training, the last two decades have shown an increasing amount of interest in applying EAs to construct RNNs (e.g., see [5], [6]). Among the EA family, particle swarm optimization (PSO) [7], [8] is known as one of the latest methods and it has been shown to have several advantages compared to conventional backpropagation in training FNNs

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 51–58.
springerlink.com

for nonlinear function approximation, such as less computational cost and faster convergence [9]. In this paper, attention is paid to the use of PSO as a training algorithm to construct an RNN-based controller for a nonlinear system.

On the other hand, due to the fact that in dynamic system control the system state is not always observable completely, state estimation is thus an important prerequisite to make the operation possible, safe and economical. For example, sensing the velocities in mechanical systems is hard or may require more sensors with high cost. We thus assume that the velocity states of the system are not observed and hence use a particle filter (PF) to estimate the velocities. Known as a stochastic estimation method, PF has emerged as a very potential technique during the last decade as it is applicable for nonlinear and non-Gaussian systems [10], [11].

In order to validate the method, this study considers a rotary crane to be the control object, which is a reduced two degree-of-freedom model with underactuated behavior. Underactuated systems, which are characterized by the fact that they process fewer actuators than degrees of freedom, have attracted much research during the last decades due to it pervasive applications. Such systems generate interesting but difficult control problems as they usually exhibit nonholonomic behavior and complex dynamics. As widely used in industry, crane systems have been the subject of several studies, presenting various methods for load swing control including classical and fuzzy approaches [12], [13]. The crane system being considered herein is hard to stabilize quickly and also known to be a nonholonomic system [13]. As is well-known, the class of nonholonomic systems cannot be asymptotically stabilized by continuous time-invariant static feedback [14]. Several approaches have been introduced for such systems, presenting time varying, discontinuous, dynamic state feedback control methods [14]. Generally, these methods aim to deal with asymptotic stabilization, considering the ideal system model while assuming that all system states are observed. This paper addresses the control of such a system with supposing that there exist unknown states (velocities).

2 Nonlinear Control Using a Recurrent Neural Network and Particle Swarm Optimization with Particle Filter Estimation

Considering a discrete, nonlinear, non-Gaussian system represented by:

$$x_k = f_k(x_{k-1}, \eta_k) \tag{1}$$

$$y_k = g_k(x_k, \xi_k) \tag{2}$$

where y_k is the vector of observations, x_k is the state vector at time k ($k = 0, 1, 2...$); $f_k(.)$ is the (known) state transition function, $g_k(.)$ is the (known) observation function; η_k and ξ_k are noise vectors. The control task is to regulate the system from an arbitrary initial point x^{init} to a desired point x^{ref} with an assumption that there exist unobserved states. To do so, an RNN is used as a state-feedback controller which is optimized by PSO and the latent states are estimated by a PF. The schematic of the control system is shown in Fig. 1, which also includes a limiter for dealing with the constraints involving to the control input ($|u| \leq u_{\max}$ or $|\dot{u}| \leq \dot{u}_{\max}$).

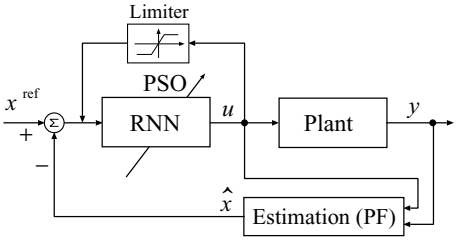


Fig. 1. The proposed control system.

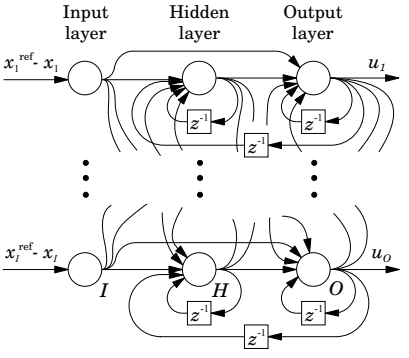


Fig. 2. Recurrent neurocontroller.

Fig. 2 shows the structure of the RNN used herein. The network consists of three layers: the input layer with I neurons, the hidden layer with H neurons, and the output layer with O neurons (i.e., it is a $I-H-O$ structured RNN). The input layer contains only feedforward connections and there is no recurrent/feedback loop in its own neurons or from/to other layers. In contrast, the hidden and output layers are fully recurrent.

The PSO and PF are employed in two following operators:

i) Design: Used to design a feedback controller in offline manner. That is, a PSO is utilized to evolve the RNN controller. A sum-squared error is used as the evaluation function for the PSO, which is:

$$E = \sum_{i=1}^{I} Q_i \left(x_i^{\text{ref}} - x_i^{\text{end}} \right)^2 + \sum_{m=1}^{O} Q_m \left(u_m^{\text{ref}} - u_m^{\text{end}} \right)^2 \tag{3}$$

where $Q_{(.)}$ are the weight coefficients, $(.)^{\text{end}}$ represents the values at the final state (at the end of control simulation), and $(.)^{\text{ref}}$ represents the desired values. In most cases, such as in this paper, $u^{\text{ref}} = 0$.

In this study we utilize the constricted version of PSO proposed in [7]. After finding the *local best* $PB_n = [\, pb_{n1},\ pb_{n2},\ \cdots,\ pb_{nD} \,]^{\mathrm{T}}$ and the *global best* $gb = [\, gb_1,\ gb_2, \cdots, gb_D \,]^{\mathrm{T}}$ in the j-th iteration, the n-th particle updates its velocity $V_n = [\, v_{n1},\ v_{n2},\ \cdots,\ v_{nD} \,]^{\mathrm{T}}$ and position $W_n = [\, w_{n1},\ w_{n2},\ \cdots,\ w_{nD} \,]^{\mathrm{T}}$ in the next iteration $(j+1)$ with the following equations:

$$v_{nd}^{j+1} = \chi \left(v_{nd}^{j} + c_1 r_1 \left(pb_{nd}^{j} - w_{nd}^{j} \right) + c_2 r_2 \left(gb_{d}^{j} - w_{nd}^{j} \right) \right) \tag{4}$$

$$w_{nd}^{j+1} = w_{nd}^{j} + v_{nd}^{j+1} \tag{5}$$

where $d = 1, 2, \cdots, D$ is the dimension number; $n = 1, 2, \cdots, N$ (N is the swarm population size); $j = 1, 2, \cdots, J$ is the iteration number; c_1 and c_2 are positive constants; r_1 and r_2 are random values in the range $[0,\ 1]$; and χ is the constriction coefficient defined as:

for $i = 1 : N_p$ **do**
  Sample: $x_k^i \sim q(x_k|x_{k-1}^i, y_k)$
  Weight update: $\omega_k^i \propto \omega_{k-1}^i \frac{p(y_k|x_k^i)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{k-1}^i, y_k)}$
end for
Calculate total weight: $\sum_{i=1}^{N_p} \omega_k^i$
for $i = 1 : N_p$ **do**
  Normalize: $\omega_k^i = \omega_k^i / \sum_{i=1}^{N_p} \omega_k^i$
end for
Effective no. of particles: $\hat{N}_{\text{eff}} = 1/\sum_{i=1}^{N_p} (\omega_k^i)^2$
if $\hat{N}_{\text{eff}} < N_{\text{threshold}}$
  Resample
end if

Fig. 3. Particle filtering pseudocode.

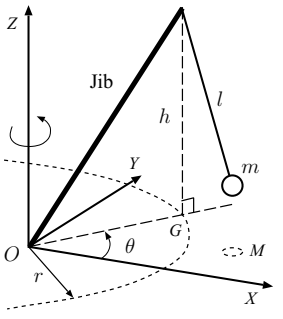


Fig. 4. Model of the rotary crane system.

$$\chi = \frac{2}{|\,2 - \varepsilon - \sqrt{\varepsilon^2 - 4\varepsilon}\,|}\,, \qquad \varepsilon = c_1 + c_2 \tag{6}$$

ii) Estimation: Used to estimate the latent states of the system. Namely, using the incomplete measurements y to obtain the estimate $\hat{x}$ by PF. After the *Design* finishes constructing a controller (using the system model with all sates completely observed), the control process of the real plant are then executed while the PF performs in every time step (online) to estimate the hidden states.

In principle, the objective of Bayesian estimation is to estimate the hidden x_k based on the set of all available (observed) measurements $y_{1:k} = \{y_i,\ i = 1, 2, ..., k\}$ up to time k. To do so, the posterior probability density function (pdf) $p(x_k|y_{1:k})$ is calculated recursively. Based on Monte Carlo simulation, PF approximates the posterior by representing the required posterior as a finite set of N_p random samples (particles) $X_k = \{x_k^i\}_{i=1}^{N_p}$ and associated weights $\{\omega_k^i\}_{i=1}^{N_p}$. Fig. 3 shows the sampling importance resampling (SIR) algorithm, a very commonly used PF. More details of PF approach can be found in [10], [11].

3 Rotary Crane System

The control object to be considered in this work is a rotary crane system as shown in Fig. 4, which is also known as a nonholonomic system [13]. Herein, θ is the rotation angle which serves as the control input, r is the radius of rotation, h is the height from the tip of the boom, m is the mass of the load, and l is the length of rope which is assumed to be not elastic or slack. Assume that the vibration of swinging load is sufficiently small, this study only considers the movement of the jib rotating around the Z axis.

Suppose that the projection of the load m on $X-Y$ plane is M, its position $(x_M,\ y_M)$ can be determined by the angle θ. The state of the system is defined as $x = [x_1,\ x_2,\ x_3,\ x_4]^{\mathrm{T}} = [x_M,\ \dot{x}_M,\ y_M,\ \dot{y}_M]^{\mathrm{T}}$. The dynamics of the system is: [13]

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\lambda^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\lambda^2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \lambda^2 r\cos\theta \\ 0 \\ \lambda^2 r\sin\theta \end{bmatrix} \tag{7}$$

where $\lambda = \sqrt{g/l}$ is the natural frequency, $g = 9.8[\mathrm{m/s}^2]$ is the acceleration of gravity.

It should be noted here that, in this rotary crane model, the control input is not the torque of the crane, but the angle θ. Angle-command system is realizable when using a pulse motor [15], the model (7) therefore can be used without the appearance of the load mass m. However, it is necessary to consider the limit of the angular velocity of the motor in order to avoid the loss of synchronism [15] (i.e., $|\dot{\theta}| \le \dot{\theta}_{\max}$).

The task in the crane system control is to regulate the load mass, by rotating the crane jib, from an arbitrary position x^{init} (determined by an initial angle θ_0) to a desired position, which can be defined as $x^{\mathrm{ref}} = [r,\ 0,\ 0,\ 0]^{\mathrm{T}}$ without loss of generality. This rotation movement usually accompanies with the vibration or swinging of the load, making the control problem more difficult as requiring to suppress the load sway.

4 Performance Test and Control Simulations

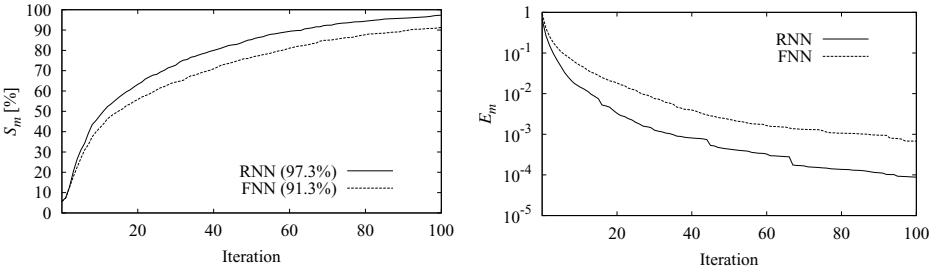
4.1 Comparative Performance Test

In order to show that the RNN is a suitable choice, the RNN is compared to a three-layer FNN. Under the same instances, the criteria for comparing the two networks are their evolutionary performance and computation time.

i) Evolutionary performance: The evolutionary performance of the two PSO-based control systems (using the RNN and FNN) are evaluated by the mean success rate S_{m}, i.e., the rate of successfully-evolved controllers over the total performed runs, and the mean (*global best*) error value E_{m} obtained at each iteration. A controller is considered to be successfully evolved if it can obtain an error smaller than a required accuracy after running the *Design* by PSO, that is, $E < E_{\mathrm{suc}} = 10^{-4}$. The above two factors are calculated over the replications of varying 20 different initial system positions, which are random values uniformly distributed in the range $\theta_0 \in [-\pi,\ \pi]$, and 50 times of randomly changing the initial population of the PSO for each of the initial positions.

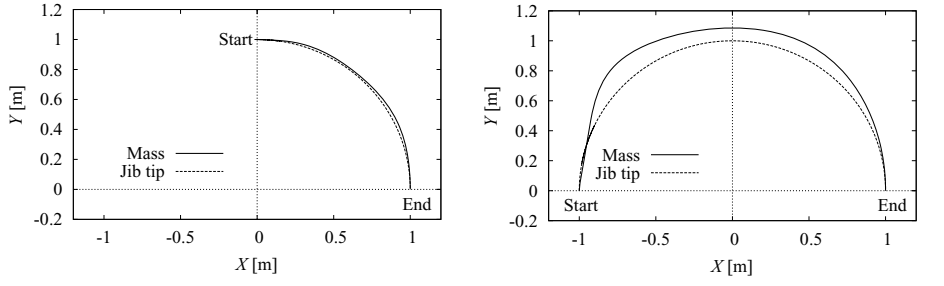
In the PSO, the population size is set to be 20 particles and the number of iterations is 100. These values are selected to be not too large with respect to computational cost. As suggested in [7], the constants in the constricted PSO should be set as $c_1 = c_2 = 2.05$, hence $\varepsilon = 4.1$ and $\chi \approx 0.7298$, for assuring the convergence. In our tests, the crane system parameters are $l = 2[\mathrm{m}]$, $r = 1[\mathrm{m}]$, the constraint is $|\dot{\theta}| \le \dot{\theta}_{\max} = 1.0\,[\mathrm{rad/s}]$, and the fourth-order Runge-Kutta method with time step of 0.01 seconds is utilized.

To demonstrate the advantages of the RNN over FNN, we first tune the number of hidden neurons and the range of PSO initial population (based on trial and error) such that the FNN achieves the best performance. That is, the number of hidden neurons



(a) Mean success rate (b) Mean best error

Fig. 5. Result of comparative evolutionary performance test.



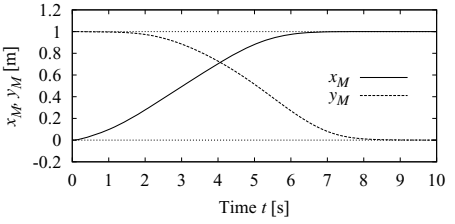
(a) $\theta_0 = \pi/2$ (b) $\theta_0 = \pi$

Fig. 6. Trajectories of the load mass on X-Y plane.

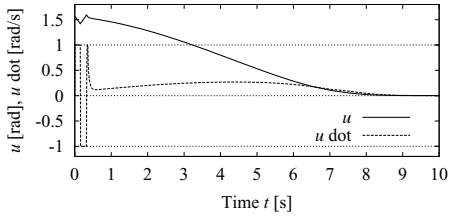
in the FNN is $H = 7$, and the initial population range is $[-1.5,\ 1.5]$; this range is then applied to the RNN. In the RNN, only $H = 2$ hidden neurons are used, which is set to be small with respect to computational cost. Namely, the structures of the FNN and RNN are $4-7-1$ and $4-2-1$, respectively. It should be noted that the above initial population range and the number of hidden neurons used for the RNN are not the optimal values for its performance. In both networks, we use a linear function $f_{\text{io.act}}(x) = x$ as the activation function of the input and output layers and a hyperbolic tangent function $f_{\text{h.act}}(x) = \tanh(x)$ for the hidden layer.

The mean success rates and mean *global best* errors of the PSO in the *Design* using the RNN and FNN are respectively shown in Figs. 5 (a) and (b). It turns out that the RNN could obtain a higher mean success rate and lower mean error compared to the FNN (the rates at the end of the evolution for the two networks are 97.3% and 91.3%).

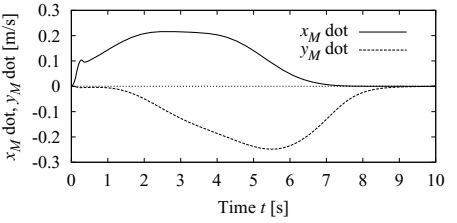
ii) Computational cost: The tests have been implemented in C programming language ran on a PC with a Duo-Core 2.2 GHz CPU and 2 GB of RAM running Ubuntu Linux. It was measured that a single run of the *Design* using the FNN took 1.89 seconds, while it was only 1.08 seconds in the case of using the RNN. Clearly, the use of the RNN noticeably reduced computation time.



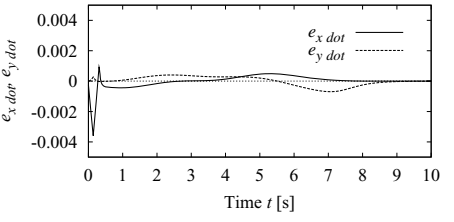
(a) Positions x_M and y_M vs. time



(b) Velocities $\hat{x}_M$ and $\hat{y}_M$ vs. time



(c) Input u and velocity $\dot{u}$ vs. time



(d) Estimate errors $e_{\dot{x}}$ and $e_{\dot{y}}$ vs. time

Fig. 7. Control results with $\theta = \pi/2$.

4.2 Control Simulations

Next, the control simulations are performed with $\theta_0 = \pi/2$ and $\theta_0 = \pi$, for example. Figs. 6 (a) and (b) show the trajectories of the crane using two of the successfully-evolved RNN controllers and the PF with 500 particles and $N_{\text{threshold}} = 0.3N_p$ for the *Estimation*, where, for simplicity, it is assumed that there is no disturbance in the system but the PF is used only to estimate the hidden velocity states $\hat{x}_M$ and $\hat{y}_M$ based on the observed position (x_M, y_M). More details of the control results are plotted in Fig. 7. Clearly, the controller is able to regulate the system to the desired position.

In order to see how well the PF performed in the example, we also plot the errors of estimations, i.e., the differences between the real states (velocities) and the estimates provided by the PF, which are:

$$e_{\dot{x}}(t) = \dot{x}_M(t) - \hat{\dot{x}}_M(t), \quad \text{and} \quad e_{\dot{y}}(t) = \dot{y}_M(t) - \hat{\dot{y}}_M(t) \tag{8}$$

It appears that, although there exist errors, the PF could obtain small errors (see Fig. 7 (d)), revealing that the estimated values are very similar to the real ones.

5 Discussion and Conclusion

Although underactuated systems with nonholonomic constraints seem to be a potential approach with several advantages bringing it to broad applications, it is a fact that control of such systems is a challenging task. The study has presented an intelligent control method for an underactuated crane with constraint by using an RNN optimized by PSO while considering the problem with hidden velocities which were estimated by PF.

Tests were carried out for comparing the RNN and FNN based controllers with small values of iteration number and population size. As it is our priority to achieve a good evolutionary performance under a given condition, the parameters of the FNN (i.e., the number of hidden neurons and the range of its initial connection weights) were thus tuned for the most suitable values. Although the parameters of the RNN were set unfairly in the comparison, the RNN has provided an attractive performance while significantly reduced the computation time. Aside from the self-connected neurons, the RNN used in this paper includes the hidden and output layers that have other feedbacks interacting each other. The RNN is thus considered to be suitable for such a sensitive nonlinear system like the crane.

References

1. Fleming, P.J., Purshouse, R.C.: Evolutionary algorithms in control systems engineering: a survey. Control Engineering Practice 10, 1223–1241 (2002)
2. Linkens, D.A., Nyongesa, H.O.: Learning systems in intelligent control: An appraisal of fuzzy, neural and genetic control applications. IEEE Proc. Control Theory and Applications 143(4), 367–386 (1996)
3. Kristinsson, K., Dumont, G.A.: System identification and control using genetic algorithms. IEEE Trans. Systems, Man, and Cybernetics 22(5), 1033–1046 (1992)
4. Zerkaoui, S., Druaux, F., Leclercq, E., Lefebvre, D.: Stable adaptive control with recurrent neural networks for square MIMO non-linear systems. Engineering Applications of Artificial Intelligence 22, 702–717 (2009)
5. Blanco, A., Delgado, M., Pegalajar, M.C.: A real-coded genetic algorithm for training recurrent neural networks. Neural Networks 14, 93–105 (2001)
6. Angeline, P.J., Saunders, G.M., Pollack, J.B.: An evolutionary algorithm that constructs recurrent neural networks. IEEE Trans. Neural Networks 5, 54–65 (1994)
7. Clerc, M., Kennedy, J.: The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. on Evolutionary Computation 6(1), 58–73 (2002)
8. Yasuda, K., Iwasaki, N., Ueno, G., Aiyoshi, E.: Particle Swarm Optimization: A numerical stability analysis and parameter adjustment based on swarm activity. IEEJ Trans. Electrical and Electronic Engineering 3, 642–659 (2008)
9. Gudise, V., Venayagamoorthy, G.: Comparison of Particle swarm optimization and backpropagation as training algorithms for neural networks. In: Proc. IEEE Swarm Intelligence Symp., pp. 110–117 (2003)
10. Kitagawa, G.: Monte-Carlo filter and smoother for non-Gaussian nonlinear state-space models. Journal of Computational and Graphical Statistics 5(1), 1–25 (1996)
11. Cany, J.V.: Bootstrap Particle Filtering. IEEE Signal Processing Magazine 24(4), 73–85 (2007)
12. Sakawa, Y., Shindo, Y.: Optimal control of container cranes. Automatica 18(3), 257–266 (1982)
13. Kondo, R., Shimahara, S.: Anti-sway control of a rotary crane via two-mode switching control. Trans. Society of Instrument and Control Engineers 41(4), 307–313 (2005) (in Japanese)
14. Kolmanovsky, N., McClamroch, N.H.: Developments in Nonholonomic Control Problems. IEEE Control Systems 15(6), 20–36 (1995)
15. Furuta, K., Kawaji, S., Mita, T., Hara, S.: Mechanical System Control, Ohm-sha, Japan, pp. 192–197 (1984) (in Japanese)

Extracting Connection Types in Process Models Discovered by Using From-to Chart Based Approach

Eren Esgin[1] and Pinar Senkul[2]

[1] Middle East Technical University, Informatics Institute,
065531 Ankara, Turkey
[2] Middle East Technical University, Computer Engineering Department,
065531 Ankara, Turkey
eesgin@ii.metu.edu.tr senkul@ceng.metu.edu.tr

Abstract. Although contemporary information systems are intensively utilized in enterprises, their actual impact in automating complex business process is still constrained by the difficulties coincided in design phase. In this study, a hybrid data analysis methodology to business process modeling that is based on using from-to chart is further enhanced to discover connection types and include in the process model. From-to chart is basically used as the front-end to figure out the observed transitions among the activities in event logs. The derived raw patterns are converted into activity sequence on from-to chart by using Genetic Algorithms. Finally a process model including AND/OR connection types is constructed on the basis of this activity sequence.

Keywords: From-to Chart, Business Process Modeling (BPM), Process Mining, Connection Types, Genetic Algorithms (GA), Event Logs.

1 Introduction

In conventional process modeling, a reference model for the target process is constructed. However, sometimes the actual enactment may deviate from the proposed reference model due to the reasons such as subjectivity of the domain experts or incomplete design phase. Process mining is proposed a remedy to these discrepancies by mining interesting patterns from the event logs and discovering the business process model automatically [5,6,7,8,9,10]. Unlike contemporary process modeling approaches, process mining is not biased or restrictive by normative perceptions [14].

A new approach for process mining based on using from-to chart is given in [2]. In this approach, from-to chart, which is fundamentally used in tracing material handling routes on the production floor [4], is adapted to the business process modeling (BPM) domain and is used for analyzing the event logs. The transitions among activities occurred in process instances are kept and analyzed in a from-to chart in order to check if there exists any specific order of the occurrences for representing in process model [2]. This approach is further improved in [3] in such a way that the runtime complexity of the approach is improved by adopting *Genetic Algorithms* (GA) [11].

In this work, we extend the work in [3] by extraction of AND/OR connection types. Through this extension, the type of connections are discovered for each parallel

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 59–65.
springerlink.com

predecessor or successor of underlying activities by interpreting *(i)* the structure of dependency/frequency graph and *(ii)* the final scores at from-to chart. Then dependency/frequency graph is converted into a block-oriented model named control flow graph. By this way, the process model is enhanced from a sequence form to a more extended form including connections.

This paper is organized as follows: Section 2 introduces the proposed approach. In Section 3, the application of the presented approach on a real case is described. Section 4 includes the related work. Finally, Section 5 presents the concluding remarks.

2 Extracting Connection Types in From-to Chart Based Process Discovery

The proposed approach consists of three basic steps: construction and rearrangement of from-to chart, relation extraction and constructing of process model that includes connection types.

2.1 Construction and Rearrangement of From-to Chart

The first operation of the proposed methodology is the creation of a FROMTOCHART database table by retrieving the activity types from the event logs and populating FROMTOCHART table. Predecessor and successor for each transition are parsed in every process cycle, which is ordered by timestamp in ascending order and the current score at (predecessor,successor)th element at FROMTOCHART table is incremented.

Then scores at FROMTOCHART table are evaluated in order to eliminate the effect of *weak* scores on finding the fittest activity sequence by pruning down them prior to rearrangement. Basically there are three evaluation metrics used in the proposed methodology: *confidence for from-to chart* (confidence FTC), *support for from-to chart* (support FTC) and *modified lift* [2].

Rearrangement operation is the *engine component* of proposed methodology, which aims to find out the activity sequence with the minimum total moment value at FROMTOCHART table in GA fashion. The coarse-grained GA stages are detailed in [3].

2.2 Relation Extraction

Basically there are three types of relation represented in dependency/frequency graph:

a. Immediate Succession. Let S be an optimum activity sequence over T, a range of activities, i.e. $S \in T^*$ (i.e. a permutation of T) and $A, B \in T$. Then B *immediately succeeds* A, if and only if position of A (p_A) and position of B (p_B) at S are two successive integers such that; $p_B = p_A + 1$, where p_A and $p_B \in [1, |S|]$.

b. Succession. Let S be an optimum activity sequence over T, a range of activities, i.e. $S \in T^*$ and $A, B \in T$. Then B *succeeds* A, if and only if position of A (p_A) and position of B (p_B) at S are two integers such that; $p_B > p_A$, where p_A and $p_B \in [1, |S|]$.

c. Back-tracking. Let S be an optimum activity sequence over T, a range of activities, i.e. $S \in T^*$ and $A, B \in T$. Then B *backtracks* A if and only if position of A (p_A) and position of B (p_B) at S are two integers such that; $p_B < p_A$, where p_A and $p_B \in [1, |S|]$.

2.3 Process Model Construction by Deriving AND/OR Connections

AND/OR connection type extraction handles relations of each activity type, which are held in successor and predecessor activity list, in a pair-wise fashion. There are two formulations for calculation connection scores with respect to the type of connections (i.e. join or split) as stated in [13]:

$$A \Rightarrow B \wedge C_{JOIN} = \left(\frac{|B > C| + |C > B|}{|B > A| + |C > A|} \right) \quad A \Rightarrow B \wedge C_{SPLIT} = \left(\frac{|B > C| + |C > B|}{|A > B| + |A > C|} \right) \tag{1}$$

While $A \Rightarrow B \wedge C_{SPLIT}$ term indicates the score of potential split-type connection between activities B and C, which are successors of activity A, $A \Rightarrow B \wedge C_{JOIN}$ term indicates the score of potential join-type connection between activities B and C, which are predecessors of activity A in dependency/frequency graph.

The denominators of these formulations, $|A > B| + |A > C|$ and $|B > A| + |C > A|$, indicate the total number of outgoing or incoming observations. The numerator, $|B > C| + |C > B|$, is the number of times that activity B and C appear directly after each other. Quantitatively connection scores, which are greater than the AND threshold, implies an AND-connection between the related activities. Otherwise it is an OR-connection.

3 Application of the Proposed Approach

The sample business process, which we use for illustrating the approach, consists of ten activity types: intern repair, make ticket, arrange survey, inform client survey, survey, repair ready, send ticket to financial admin, ready inform client, ticket ready and first contact. Table 1 shows the initial state of `FROMTOCHART` table for 333 process instances. Note that `frCon` activity is denoted as the initiator activity.

Table 1. Initial and Final States of `FROMTOCHART` Table (Element $a_{6,6}$ is set to zero due to evalution operation).

	frCon	makTic	arrSur	infCliSur	sur	intRep	repRdy	senTic	rdyInfCli	ticRdy	Row Total	CST	CCT
frCon	0	312	0	0	0	0	0	0	0	0	312	133.20	124.80
makTic	0	0	312	0	0	0	0	0	0	0	312	133.20	124.80
arrSur	0	0	0	312	0	0	0	0	0	0	312	133.20	124.80
infCliSur	0	0	0	0	312	0	0	0	0	0	312	133.20	124.80
sur	0	0	0	0	0	158	154	0	0	0	312	133.20	124.80
intRep	0	0	0	0	0	**17|0**	158	0	0	0	158	133.20	70.00
repRdy	0	0	0	0	0	0	0	137	175	0	312	133.20	124.80
senTic	0	0	0	0	0	0	0	0	137	175	312	133.20	124.80
rdyInfCli	0	0	0	0	0	0	0	175	0	137	312	133.20	124.80
ticRdy	0	0	0	0	0	0	0	0	0	0	0	133.20	0.00

Calculated Support Threshold (**CST**) = 333 * 0.4
Calculated Confidence Threshold (**CCT**) = row total of the underlying activity * 0.4

After FROMTOCHART table is created and parsed transitions in activity streams are populated to this table, scores are evaluated with respect to support and confidence thresholds (supportFTC=0.40 and confidenceFTC=0.40). Table 1 shows the initial and final states of FROMTOCHART table.

After scores are evaluated, GA engine reproduces activity sequence starting with firCon activity and then calculates fitness score for these activity sequences according to proposed population size (populationSize=80) and maximum number of iterations (maxIterationNum=200) settings.

As shown in Figure 1, initial population is quite diverse early on the process, so crossover frequently takes large steps early in the search space and fitness scores of the individuals tend to increase. Afterwards smaller steps are taken when most individuals are quite similar, i.e. the gap between average and maximum fitness score boundaries has shrunk. Consequently the fittest (and optimum) activity sequence is obtained as {firCon, makTic, arrSur, infCliSur, sur, intRep, repRdy, rdyInfCli, senTic, ticRdy} with a 0.779 fitness score.

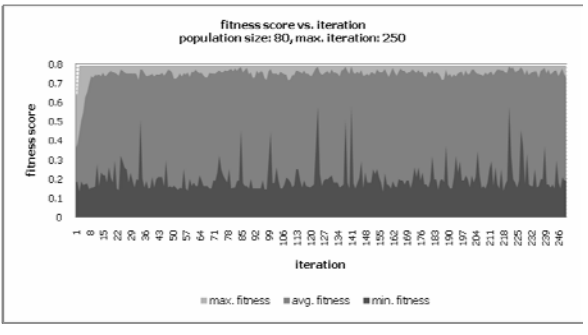


Fig. 1. Runtime Analysis for Rearrangement Operation.

After rearrangement of FROMTOCHART table, relations are extracted according to positive scores remarked at FROMTOCHART table. Consequently, dependency/frequency graph, which visually converts the fittest activity sequence (*genotype*) into the process model (*phenotype*), is constructed as shown in Figure 2.

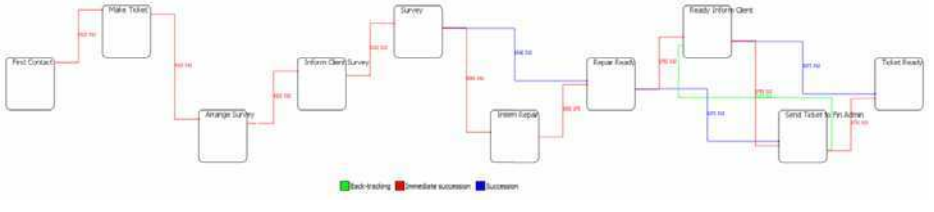


Fig. 2. Extracted Relations in Dependency/Frequency Graph.

As the final operation, type of connections at successor and predecessor activity list of each activity are derived with respect to the AND threshold (`ANDthreshold=0.10`). Hence dependency/frequency graph is converted into control flow graph as shown in Figure 3. Sample connection score calculations are as follows;

$$sur \Rightarrow intRep \wedge repRdy_{SPLIT} = \frac{158}{158+154} = 0.505, then\ AND-split\ type\ connection$$

$$repRdy \Rightarrow senTic \wedge rdyInfCli_{SPLIT} = \frac{137+175}{137+175} = 1, then\ AND-split\ type\ connection$$

$$rdyInfCli \Rightarrow repRdy \wedge senTic_{JOIN} = \frac{137}{137+175} = 0.438, then\ AND-\ join\ type\ connection$$

$$senTic \Rightarrow rdyInfCli \wedge ticRdy_{SPLIT} = \frac{137}{137+175} = 0.438, then\ AND-split\ type\ connection$$

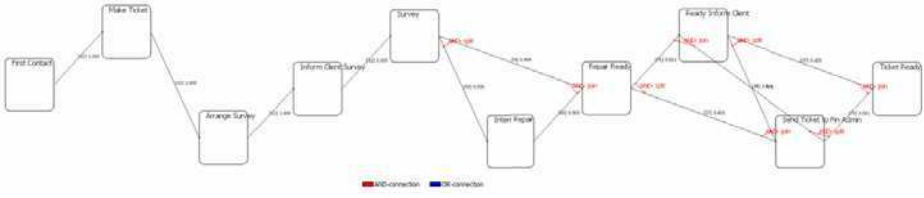


Fig. 3. Derived Connections in Control Flow Graph.

For this data set, the percentage of transitions in testing dataset that are represented in the dependency/frequency graph (i.e. recall) is 99.85%. Additionally, all extracted relations have corresponding complements in testing dataset (i.e. precision of the discovered process model is 100%) and a 58.87% improvement is achieved in processing time by the use of GA based rearrangement.

4 Related Work

In [5], Aalst et al. highlights the process mining usage in analysis and improvement of process flexibility. It proposes the enrichment of obtained event logs with semantic information. Likewise [5], [6] aims to handle both case and control-flow perspectives in an integrated fashion. While production workflow is strongly process-oriented, the case handling paradigm basically concentrates on the case itself, which is the fundamental object to be produced.

The idea of applying process mining in the context of workflow management was first introduced in [7]. In this study, two issues are handled. The first issue is to discover a workflow graph generating activities appearing in a given workflow log. The second issue is to find the definitions of the relation conditions.

Cook and Wolf have investigated similar issues in the context of software engineering process. In [8], they describe three methodologies for process discovery ranging from the purely algorithmic to purely statistical: RNET, KTAIL and MARKOV.

In [9], Weijters and Aalst propose a rediscovery technique to deal with noise and to validate workflow processes by uncovering and measuring the discrepancies between the prescriptive models and actual process executions. Detecting concurrency appears as one of the fundamental concerns. Therefore AND/OR connectors are aimed to be explicitly distinguished in the process model. Moreover, local and global metrics are proposed to find explicit representations for a broad range of process models. These metrics are revised in [1]. In [10], the proposed method, named alpha algorithm, seeks to rediscover sound Workflow nets. Given such an algorithm, it is aimed to find the class of workflow nets which can be rediscovered. As a way of representation, it generates concrete Petri nets for a broad range of process models rather than a set of dependency relation between events like in [9].

As a GA implementation in facility layout problem (FLP), Sarker et al. attempts to handle locating multiple identical machines in a linear layout, which is called tertiary assignment problem (TAP) in [12]. It is attempted to solve this problem type by identifying sets of identical machines which may be partitioned into individual, unique machines. In order to relax the underlying FLP and provide approximate solutions, amoebic matrix is used.

5 Conclusion

Process design is a complicated, time-consuming phase and typically, there may arise incompatibilities between process design and process enactment. Process mining techniques aim to handle such incompatibilities by extracting information on the process from the event logs. In this work, we present an enhancement on a hybrid process mining approach that uses from-to chart, in order to extract connection types and to form a process model. From-to chart, which is a fundamental analytical tool used in tracing material handling routes at production floor, is adapted to process modeling domain in order to analyze transitions among activities. The activity sequence, which is rearranged in a straight-line form by developed GA engine, constitutes the backbone of dependency/frequency graph without compromising accuracy. In order to construct the process model, the generated dependency/frequency graph is converted into control flow graph according to AND threshold. The applicability of the approach is demonstrated on a case study.

References

[1] Măruşter, L., Weijters, A.J.M.M.T., van der Aalst, W.M.P., van den Bosch, A.: Process mining: Discovering direct successors in process logs. In: Lange, S., Satoh, K., Smith, C.H. (eds.) DS 2002. LNCS, vol. 2534, pp. 364–373. Springer, Heidelberg (2002)

[2] Esgin, E., Senkul, P.: Hybrid Approach to Process Mining: Finding Immediate Successors of a Process by Using From-to Chart. In: Int. Conf. on Machine Learning and Applications, pp. 664–668. IEEE Computer Society, Los Alamitos (2009)

[3] Esgin, E., Senkul, P., Cimenbicer, C.: A hybrid approach for process mining: Using from-to chart arranged by genetic algorithms. In: Graña Romay, M., Corchado, E., Garcia Sebastian, M.T. (eds.) HAIS 2010. LNCS, vol. 6076, pp. 178–186. Springer, Heidelberg (2010)

[4] Apple, J.M.: Material Handling Systems Design. The Ronald Press Company, New York (1972)
[5] van der Aalst, W.M.P., Gunther, C., Recker, J., Reichert, M.: Using Process Mining to Analyze and Improve Process Flexibility. In: 7th Workshop on BPMDS 2006, CAiSE 2006 Workshop (2006)
[6] Gunther, C.W., van de Aalst, W.M.P.: Process Mining in Case Handling Systems. In: Proc. PRIMIUM Sub Conference at the Multikonferenz Wirtschaftsinformatik (2006)
[7] Agrawal, R., Gunopulos, D., Leymann, F.: Mining Process Models from Workflow Logs. In: Schek, H.-J., Saltor, F., Ramos, I., Alonso, G. (eds.) EDBT 1998. LNCS, vol. 1377, pp. 469–483. Springer, Heidelberg (1998)
[8] Cook, J.E., Wolf, A.L.: Discovering Models of Software Processes from Event-Based Data. ACM Transactions on Software Engineering and Methodology 7(3), 215–249 (1998)
[9] Weijters, A.J.M.M., van der Aalst, W.M.P.: Rediscovering Workflow Models from Event-Based Data Using Little Thumb. Integrated Computer-Aided Engineering 10(2), 151–162 (2003)
[10] van der Aalst, W.M.P., Weijters, A.J.M.M., Maruster, L.: Workflow Mining: Discovering Process Models from Event Logs. Transaction on Knowledge and Data Engineering 16(9), 1128–1142 (2004)
[11] Dianati, M., Song, I., Treiber, M.: An Introduction to Genetic Algorithms and Evaluation Stragies. Univ. of Waterloo, Canada (2004)
[12] Sarker, B.R., Wilbert, E.W., Hogg, G.R.: Locating Sets of Identical Machines in a Linear Layout. Annals of Operations Research 77, 183–207 (1998)
[13] Weijters, A.J.M.M., van der Aalst, W.M.P., Mederios, A.K.A.: Process Mining with the HeuristicMiner Algorithm. Paper presented at the BETA Working Paper Series, WP 166, Eindhoven University of Technology (2006)
[14] van der Aalst, W.M.P., Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., Weijters, T.A.J.M.M.: Workflow Mining: A Survey of Issues and Approaches. Data & Knowledge Engineering 47(2), 237–267 (2003)

Using Class Decomposition for Building GA with Fuzzy Rule-Based Classifiers

Passent El-Kafrawy and Amr Sauber

Dept of Math and CS, Faculty of Science
Menoufia University, Egypt
{passentmk,amrmausad}@gmail.com

Abstract. A classification problem is fully partitioned into several small problems each of which is responsible for solving a fraction of the original problem. In this paper, a new approach using class-based partitioning is proposed to improve the performance of genetic-based classifiers. Rules are defined with fuzzy genes to represent variable length rules. We experimentally evaluate our approach on four different data sets and demonstrate that our algorithm can improve classification rate compared to normal Rule-based classification GAs [1] with non-partitioned techniques.

Keywords: Genetic Algorithm, Rule-based Classification, Class Decomposition, Divide and Conquer, Fuzzy Rules.

1 Introduction

The interest in decision-making has been gaining momentum in recent years. Nowadays enormous amounts of information are collected continuously. The tremendously growing amount of data has made manual analysis by experts a tedious task and sometimes impossible. Many hidden and potentially useful relationships may not be recognized by the analyst. The explosive growth of data requires an automated way to extract useful knowledge. One of the possible approaches to this problem is by means of data mining or knowledge discovery from databases. Rule-based classification [2,3] is one of the most studied tasks in data mining community and is an active research area. Classification rules are typically useful in medical problems, specially in medical diagnosis. Such rules can be verified by medical experts and may provide better understanding of the problem by-hand.

Numerous techniques have been applied to classification in data mining over the past few decades, such as statistical methods, expert systems, artificial neural networks, database systems, and evolutionary algorithms ([4], [5], [6], [7], [8]). Among them, genetic algorithm (GA)-based approaches have attracted much attention and become one of the popular techniques for classification [9]. De Jong and Spears [1] considered the application of GA to a symbolic learning task—supervised concept learning from a set of examples. Corcoran and Sen [2] used GA to evolve a set of classification rules with real-valued attributes.

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 67–76.
springerlink.com

Ishibuchi et al. [3] examined the performance of a fuzzy genetic-based machine learning method for pattern classification problems with continuous attributes.

However, when GA is applied to larger-scale real-world classification problems, it still suffers from some drawbacks such as the inefficiency in searching large solution spaces, the difficulty in handling the internal interference of training data, and the possibility of getting trapped in local optima. A natural approach to overcome these drawbacks is to decompose the original task into several subtasks based on certain techniques. The purpose of decomposition methodology is to break down a complex problem into several manageable sub-problems while providing attention to each subproblem independently.

According to Michie [10], finding a good decomposition is a major tactic, both for ensuring the transparent solutions and for avoiding the combinatorial explosion. It is generally believed that problem decomposition can benefit from conceptual simplification of the problem, making the problem more feasible by reducing its dimensionality, achieving clearer results (more understandable), reducing run time by solving smaller problems and using parallel or distributed computation, and allowing different solution techniques for individual sub-problems. Various task decomposition methods have been proposed. These methods can be roughly classified into the following categories: functional modularity, domain modularity, class decomposition, and state decomposition, according to different partition strategies [11,13].

However, most of them are used in artifcial neural networks (ANN), very few find their applications in GA, especially GA-based classification [14]. As GAs have been widely used as basic soft computing techniques, the exploration of class decomposition with GAs becomes more important. Decomposition methods have been used in various fields, such as classification, data mining, clustering, etc. [15] presented a feature decomposition approach for improving supervised learning tasks. The original set of features is decomposed into several subsets. A classification model is built for each subset, and then all generated models are combined. A greedy procedure is developed to decompose the input features set into subsets and to build a classification model for each subset separately. Weile and Michielssen [16] explored the application of domain decomposition GAs to the design of frequency selective surfaces. Masulli and Valentini [17] presented a new machine learning model for classification problems. It decomposes multi-class classification problems into sets of two-class sub-problems which are assigned to non-linear dichotomizers. Chan and Zhsu [18] proposed a similar approach using GA and used Fisher's linear discriminant (FLD) Algorithm to re-assemble the rules. Apte et al. [19] presented a new measure to determine the degree of dissimilarity between two given problems and suggested a way to search for a strategic splitting of the feature space that identifies different characteristics. Watson and Pollack [20] used techniques from multi-objective optimization to devise an automatic problem decomposition algorithm that solves test problems effectively.

In this work we propose a new solution based on the class decomposition approach, in which a classification problem is fully partitioned into N class

modules; where each module is responsible for developing a classifier for a specific class. In other words, each class receives enough attention whatever the size of the input data that represents that class. Accordingly, rare occurring classes will not be ignored and still get their own attention. These modules are trained in parallel and independently and the results obtained are integrated to a final solution. Several benefits are gained using this class decomposition technique: all classes are fairly represented; parallelism can be obtained for faster solution; and scalability is possible as the problem is broken into several manageable sub-problems. After this detailed introduction and related work review given in this section, details about the methodology of class decomposition using a GA is given in section 2. In section 3 four experiments are conducted to evaluate the proposed technique. Finally, a brief conclusion is given in section 4.

2 Methodology

A GA is a search and optimization methodology from the field of evolutionary computation that was invented by Holland [21]. A GA is based on the Darwin's natural selection principle of the survival of the fittest, and is widely used for hard problems in engineering and computer science. A GA is a population-based method where each individual of the population represents a candidate solution for the target problem. This population of solutions is evolved throughout several generations, starting from a randomly generated one, in general. During each generation of the evolutionary process, each individual of the population is evaluated by a fitness function, which measures how good the solution represented by the individual is for the target problem. From a given generation to another, some parent individuals (usually those having the highest fitness) produce "offspring", i.e., new individuals that inherit some features from their parents, whereas others (with low fitness) are discarded, following Darwin's principle of natural selection. The selection of the parents is based on a probabilistic process, biased by their fitness value. We use rank-selection mechanism which means that a ranking process is performed on the fitness values then individuals with the higher fitness value are selected for the next generation. The generation of new offspring from the selected parents of the current generation is accomplished by means of genetic operators. This process is iteratively repeated until a stop criteria is reached.

2.1 Individuals Representation

An individual in our method is a classification rule where each gene represents the minimum and maximum values of intervals of each attribute that belongs to such rule. In rule-based classification, there are various representation methods in terms of rule properties (fuzzy or non-fuzzy) and attribute properties (nominal or continuous). In our approach, we use fuzzy IF-THEN rules with continuous attributes. A rule set (classifier) consisting of a - user determined - number of

rules as a solution candidate for a classification problem. An IF-THEN rule is represented as follows:

$R_i : IF\ (V1_{min} \leq x_1 \leq V_{1max}) \wedge (V_{2min} \leq x_2 \leq V_{2max}) \wedge ..... \wedge (V_{nmin} \leq x_n \leq V_{nmax})\ Then\ y = C$

where R_i is a rule and each rule represent a chromosome in which each gene represents an attribute and the consequence gene stands for a class; n is the number of attributes with $(x_1, x_2, ...x_n)$ are the input attribute set; and y is the output class category assigned with a value of C. V_{jmin} and V_{jmax} are the minimum and maximum bounds of the jth feature x_j respectively. We encode rule R_i according to Figure1.

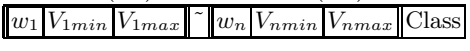


Fig. 1. An individual representation.

where the weight w_j is a real-valued variable taking values in the range [0..1]. This variable indicates whether or not the potential attribute is present in the corresponding classification rule. More precisely, when w_j is smaller than a user-defined threshold (called Limit) the attribute will not appear in the related rule. Therefore, the greater the value of the threshold Limit, the smaller the probability that the corresponding attribute will be included in the rule. Variable length rules is implemented by using weights, which simplifies the problem by ignoring insignificant attributes randomly. The weights are fuzzy indicators representing attribute significance. The less their significance than a user defined threshold, the attribute is then ignored. V_{jmin} and V_{jmax} are the limits of the intervals corresponding to the attribute A_i. Note that the above encoding is quite flexible with respect to the length of the rules. A traditional GA is very limited in this aspect, since it can only cope with fixed-length rule. In our approach, although each individual has a fixed length, the genes are interpreted (based on the value of the weight w_i) in such a way that the individual phenotype (the rule) has a variable length. The start of the first population consists of generating, arbitrarily, a fixed number of individuals during the evolution.

2.2 Genetic Operators

For the developed method, the usual one-point crossover operator is stochastically applied with a predefined probability, using two individuals of the selected pool. The crossover point is a percentage of the length of the individual that defines the starting point from where the crossover breaks the string. We use arithmetic crossover method in the experiments [22]. The employed method works as follows:

Consider two chromosomes $C_1 = (c_1^1, ..., c_n^1)$ and $c_2 = (c_1^2, ..., c_n^2)$. Applying the crossover operator on C_1 and C_2 generates two offspring $H_1 = (c_1^1, .., c_i^1, c_{i+1}^2, ..., c_n^2)$ and $H_2 = (c_1^2, .., c_i^2, c_{i+1}^1, ..., c_n^1)$.

The mutation operator is used to foster more exploration of the search space and to avoid unrecoverable loss of genetic material that leads to premature convergence to some local minima. In general, mutation is implemented by changing the value of a specific position of an individual with a given probability, denominated mutation probability. We developed new mutation operators tailored for our genome representation that Shift the starting location and/or the ending location towards the right or the left: the value in the starting/ending location of a randomly selected gene is increased or decreased by a fuzzy fraction of the upper and lower limits. This is implemented in such a way that the lower and upper bounds of the domain of the field are never exceeded.

2.3 Fitness Function

As each chromosome in our approach comprises one rule, the fitness function actually measures the collective behavior of the rule.

Definition 1. *Accuracy of a certain rule is the ratio of correctly classified instances by this rule to the whole number of instance to which this rule satisfied*

$$f_1 = \frac{C}{n} = \frac{Number\ of\ correctly\ classified\ instances\ by\ th \quad ule}{T \qquad umber\ of\ instances\ satisfy\ this\ rule}$$

2.4 Ending Criteria

As the GA searches the solution space, defined by the representation of the application at hand, some of the rules are selected in the second step as promising candidate solutions based on their evaluated fitness. They will undergo genetic operations to generate a new population. This process is repeated until the end criteria is met; either that the algorithm has reached a good enough solution or that the maximum number of generations have been iterated. We use a hybrid ending criteria that exits the execution whenever a good solution is met (human expert defined) otherwise the execution will continue after maximum number of generations.

2.5 Class Decomposition

Classification techniques usually assume homogeneity of class representation (the number of cases represent each class is close); therefore the absence of a class in the result classifer due to the trivial existence of that class in the training data is possible. Our technique addresses this problem by building a separate classifier for each class and then integrate them into one classifer. We used the training data of all classes to calculate the accuracy of each classifer (for each class) to

minimize the conflecting rules. Let us assume a classification problem that has c classes in the n-dimensional attribute space. The task of classification is to assign instances to one out of the pre-defined c classes, by discovering certain relation among the attributes. Then, the discovered rules can be evaluated by classification accuracy or error rate either on the training data or test data.

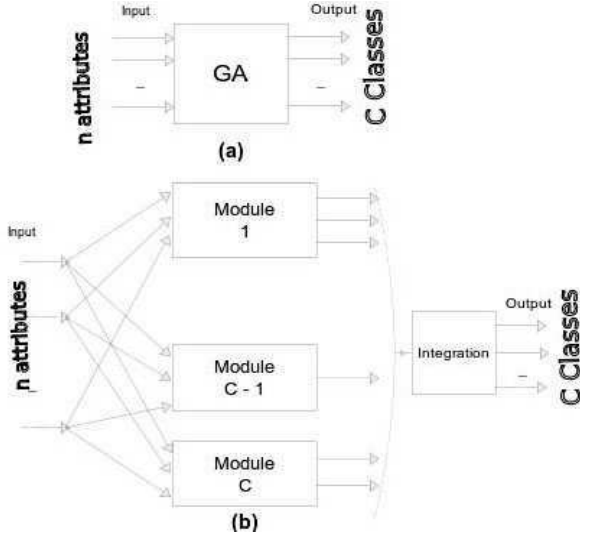


Fig. 2. Class Decomposition.

A traditional GA maps attributes to classes directly in a batch manner, which means all the attributes, classes, and training data are used together to train a group of GA chromosomes. In our approach, a GA is executed with class decomposition. As shown in Figure 2, the proposed solution consists of three steps. First, the original problem is divided into c sub-problems in terms of classes. Then, c GA modules are constructed for these sub-problems, where each module will be responsible for evolving a sub-solution. Finally, these sub-solutions are integrated to form the final solution of the original problem.

Definition 2. *Consider having c classes in the classification problem on hand, with n-dimensional pattern space, and p vectors where* $X_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $i = 1, 2, \ldots, p, p \ll c$ *are input instances. Accordingly, the given classification problem can be denoted as follows:*

$$f : X \rightarrow T$$

where $X \in R^n$ *is the set of instances with n attributes, and* $T \in R^c$ *is the set of output classes. Required a mapping f that maximizes* accuracy*. Assume that the*

c-class problem is divided into c subproblems, each contains the vectors X_i that present T^j. Hence the class set can be denoted as

$$T = T^1 \bigcup T^2 \bigcup \ldots \bigcup T^c$$

each subproblem can be formulated as required f_j with an optimal classification accuracy and score on T^j

$$f_j : X \rightarrow T^j.$$

Having c subproblems, c GA sub-modules are constructed and could be executed in parallel.

2.6 Integration

Although each GA module has evolved a portion of the solution, we cannot just simply aggregate their sub-solutions as the final one, because each GA module only classifies only one class. Therefore, when the sub-solutions are combined together, there may still exist conflicts among the sub-solutions. For example, rules from different modules may classify an instance into several classes. In order to resolve these conflicts and further improve the classification rate, the classifier employs some intelligent decision rules. The detailed integration process is explained as follows.

The classifier constructs an overall rule set by aggregating all rules from c modules. Some decision rules are added to help resolve the above-mentioned conflicts. We believe that the ending classification rate obtained from each module would be useful for this purpose. Currently, the following decision rules have been employed:

i) If an instance is classified into more than one class categories by the rule set, it will be classified into the one whose corresponding module achieves the highest classification rate in the training phase.

ii) If an instance is not classified into any class category by the rule set, it will be classified into the class whose corresponding module achieves the lowest classification rate in the training phase, if available.

2.7 Proposed Algorithm

To sum up, the process employed in this paper can be summarized by the following algorithm:

Input: Population size N; Maximum number of generations G; Crossover probability p_c; Mutation rate p_m.

Output: Classifier

1. GP = ϕ
2. for each Class C_i do
   (a) P_i:=Initialize(P,C_i)
   (b) while the termination criterion is not satisfied do

i. P_i^I:= Genetic Operators(P_i)
ii. P_i:= Rank and select fittest($P_i \cup P_i^I$)
(c) end while
(d) GP = $GP \cup P_i$
3. end for each
4. return (GP)

First, an initial Global Population GP is initialized and for each class C_i, an initial population P_i is generated in Step 2.a. Crossover and mutation operations are applied to each pair in P_i to generate the offspring population P_i^I in Step 2.b.i. The next population is constructed by choosing good solutions from the merged population $P_i \cup P_i^I$. A ranking based on fitness function is performed to select the fittest rules from the new generation to act as basis for the next generation, steps 2.b.i and 2.b.ii are repeated to obtain the set of rules (classifier) of the current class, in step 2.d the rule set P_i is added to the GP. Step 2 is to be repeated for each class, finally GP should contain all rules needed to identify all classes.

3 Experimental Results

We have implemented several classifiers running on four benchmark data sets to evaluate our approaches. The data sets chosen are the wine data, glass data, iris data and breast cancer data. They are all available in the UCI machine learning repository [23]. They all are real-world problems. We partition each data set into two parts with an equal number of instances. One half is for training, and the other half is for testing. We use the training data to train the rule set, and test the generalization power of resulting rule set with the test data. All experiments are completed on Pentium 1.8 GHz Dual Core, 1GB RAM PC. The results reported are averaged over five independent runs. The parameters, such as mutation rate used was 0.07, crossover rate was 0.7, generation limit used is 3000, and population size was 200.

We applied the four data sets to our proposed technique and also compared them by running them on Kaya's GA [24] and a normal GA where all classes are given as input without decomposition.

The proposed technique produced classification rules with higher accuracy as can be seen in table 1. Besides achieving higher accuracy, performance was increased through the parallel execution of the different modules of the problem.

Table 1. Comparison of the classifiers performance on the four test datasets.

	Normal GA	Kaya's	Proposed Tech.
Wine	0.29	0.81	0.89
Glass	0.15	0.51	0.78
Iris	0.89	0.93	0.90
Cancer	0.33	0.47	0.64

4 Conclusion

This paper proposed a new evolutionary class decomposition approach for classification tasks. A classification problem is decomposed into several modules and each module is responsible for solving a fraction of the original problem. These modules are trained in parallel and the sub-solutions obtained from them are integrated to further obtain a final solution. To evaluate our method, we have conducted some experiments. The results have shown that our algorithm is effective, efficient and promising.

References

1. DeJong, K., Spears, W.: Learning concept classification rules using genetic algorithms. In: International Joint Conference on Artificial Intelligence, pp. 651–656 (1991)
2. Corcoran, A., Sen, S.: Using real-valued genetic algorithm to evolve rule sets for classification. In: First IEEE Conference on Evolutionary Computation, Orlando, USA, pp. 120–124 (1994)
3. Ishibuchi, H., Nakashima, T., Murata, T.: Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems. IEEE Transactions on Systems, Man and Cybernetics, Part B 29, 601–618 (1999)
4. Weiss, S., Kulikowski, C.: Computer Systems that Learn Classification and Prediction Methods from Statistics. Neural Nets, Machine Learning, and Expert Systems., Morgan Kaufmann Publishers, San Mateo (1991)
5. Chang, Y.H., Zeng, B., Wang, X.H., Good, W.F.: Computer-aided diagnosis of breast cancer using artificial neural networks: Comparison of backpropagation and genetic algorithms. In: International Joint Conference on Neural Networks, Washington, DC, pp. 3674–3679 (1999)
6. Fidelis, M.V., Lopes, H.S., Freitas, A.A.: Discovering comprehensible classification rules with a genetic algorithm. In: Proceedings of Congress on Evolutionary Computation (2000)
7. Ghosh, A., Nath, B.: Multi-objective rule mining using genetic algorithms. Information Science 163, 123–133 (2004)
8. Tan, K.C., Yu, Q., Heng, C.M., Lee, T.H.: Evolutionary computing for knowledge discovery in medical diagnosis. Artificial Intelligence in Medicine 27, 129–154 (2003)
9. Merelo, J., Prieto, A., Moran, F.: Optimization of classifiers using genetic algorithms. In: Patel, M., Honavar, V., Balakrishnan, K. (eds.) Advances in the Evolutionary Synthesis of Intelligent Agents. MIT Press, Cambridge (2001)
10. Michie, D.: Problem decomposition and the learning of skills, pp. 17–31. Springer, Berlin (1995)
11. Guan, S., Li, S.: Parallel growing and training of neural networks using output parallelism. IEEE Transactions on Neural Networks 13, 1–9 (2002)
12. Jenkins, R., Yuhas, B.: A simplified neural network solution through problem decomposition: the case of the truck backerupper. IEEE Transactions on Neural Networks 4, 718–720 (1993)
13. Lu, B., Ito, M.: Task decomposition and module combination based on class relations: a modular neural network for pattern classification. IEEE Transactions on Neural Networks 10, 1244–1256 (1999)

14. Rodriguez, M., Escalante, D.M., Peregrin, A.: Efficient distributed genetic algorithm for rule extraction. Applied Soft Computing 11(1), 733–743 (2011)
15. Rokach, L., Maimon, O.: Improving supervised learning by feature decomposition. In: The Second International Symposium on Foundations of Information and Knowledge Systems, pp. 178–196 (2002)
16. Weile, D., Michielssen, E.: The use of domain decomposition genetic algorithms exploiting model reduction for the design of frequency selective surfaces. Computer Methods in Applied Mechanics and Engineering, 439–458 (2000)
17. Masulli, F., Valentini, G.: Parallel non-linear dichotomizers. In: IEEE-INNS-ENNS International Joint Conference on Neural Networks, vol. 2, pp. 29–33 (2000)
18. Guan, S.U., Zhu, F.: A class decomposition approach for ga-based classifiers. Engineering Applications of Artificial Intelligence 18, 271–278 (2005)
19. Apte, C., Hong, S., Hosking, J., Lepre, J., Pednault, E., Rosen, B.: Decomposition of heterogeneous classification problems. In: Liu, X., Cohen, P.R. (eds.) IDA 1997. LNCS, vol. 1280, pp. 17–28. Springer, Heidelberg (1997)
20. Watson, R., Pollack, J.: Symbolic combination in genetic algorithms. In: Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature, pp. 425–434 (2000)
21. Holland, J.H.: Adaptation in natural and artificial systems. University of Michigan Press, MI (1975)
22. Herrera, F., Lozano, M., Verdegay, J.L.: Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis. Artificial Intelligence Review 12, 265–319 (1998)
23. Blake, C.L., Merz, C.J.: Repository of machine learning databases (1998)
24. Kaya, M.: Autonomous classifiers with understandable rule using multi-objective genetic algorithms. Expert Systems with Applications 37, 3489–3494 (2010)

Improving the Performance of Multi-start Search on the Traveling Salesman Problem

Charles R. King, Mark McKenney, and Dan E. Tamir

Texas State University
Department of Computer Science
{charles.king,mckenney,dt19}@txstate.edu

Abstract. Constructive multi-start search algorithms are commonly used to address combinatorial optimization problems. Multi-start algorithms recover from local optima by restarting, which can lead to redundant configurations when search paths converge. In this paper, we investigate ways to minimize redundancy using record keeping and analyze several restart algorithms in the context of iterative hill climbing with applications to the traveling salesman problem. Experimental results identify the best performing restart algorithms.

Keywords: Combinatorial optimization, traveling salesman problem, iterative hill climbing, multi-start algorithms.

1 Introduction

Intuitively, combinatorial optimization problems (COPs) compute the "most favorable" outcome to a problem from a set of possible outcomes. A most favorable outcome may be the outcome with the highest profit, lowest cost, shortest distance, etc., depending on the problem being solved. For example, shipping companies have used COPs to find delivery routes that minimize left turns; by eliminating left turns, drivers spend less time in traffic waiting to turn, and use less fuel, resulting in monetary savings for the company. Clearly, COPs have great practical importance.

Many COPs are NP complete and characterized by exponential growth [1-3]; i.e., as the size of problem parameters grows, the number of potential solutions grows exponentially, causing an exhaustive search of outcomes to be impractical. A *heuristic search* technique is often used in order to assign weights to outcomes, or partial outcomes, such that partial solutions that appear to be leading to poor solutions are not considered, thus, drastically reducing the search space of outcomes. A drawback to heuristic techniques is that the optimal solution may not be found, but often a very good solution can be guaranteed. Therefore, heuristic techniques are applicable only when a sub-optimal solution is acceptable, or when the faster computation times of heuristic algorithms for COPs are required.

It is well known that many heuristic algorithms explore the same partial solution (i.e., a state in the state space) more than one time, thereby wasting computational resources. Several researchers have proposed record keeping mechanisms as a solution to this problem [4, 5], which we further investigate in this paper. In this sense, record keeping resembles Tabu search where the Tabu list includes a subset of

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 77–82.
springerlink.com

the states encountered so far [6, 7]. It differs from Tabu in that the objective of short term Tabu memory (e.g., a cache) is to explore the solution space more intelligently rather than to speed the search [7]. The goal of long-term Tabu is to enable revisiting explored search states in order to attempt to find a better local optimum [6, 7].

In [4], we analyzed the performance of a cache as a means for record keeping. In [8], we investigated several other record keeping mechanisms such as dedicated memory, and Bloom filters [9]. In this paper, we investigate the performance of record keeping multi-start COP algorithms using various restart algorithms where the record keeping mechanism is a cache. We investigate these in the context of the travelling salesman problem (TSP), using iterative hill climbing (IHC) heuristic search algorithm as the multi-start heuristic procedure [2, 3, 10]. The TSP is chosen since it is a classic, well-understood COP with many practical applications [10].

In this paper, we investigate the performance of six construction (restart) algorithms used in the context of IHC with randomly generated and benchmark TSP problems. The performance of the restart algorithms is empirically determined and the algorithms are compared for performance and solution quality.

The second contribution of this paper is to determine the amount of redundancy associated with various construction algorithms, and utilize record keeping techniques to reduce duplicate path exploration. We show empirically that a good choice of construction algorithm and record keeping mechanism provides improved convergence time. The evaluation of construction algorithms and the use of record keeping have been studied in previous work [4, 8]. Our work differs from that work in that we consider a larger group of construction algorithms, we provide a more in-depth study of the effects of different cache configurations, and we provide a comparison of cache performance to unbounded memory record keeping.

2 Construction Algorithms

In the case of the TSP, IHC can be implemented as a *constructive multi-start search*, in which multiple solutions are computed by constructing solutions from an initial solution. At each step, the concept of a *neighborhood* is used in which a neighborhood of a solution s is defined as the set of solutions S that are generated by making a minor modification, denoted a *move,* to s [11, 12]. For example, a TSP tour can be adjusted by an operation such as 2-opt which results in exchanging the visitation order of two cities [2]. The IHC algorithm proceeds by choosing the best $s' \in S$ as the next step. The move s is then assigned s' and the process is repeated until no improvements to s can be made. This solution is referred to as the locally optimal solution. In general, the algorithm repeatedly restarts with a new starting configuration until a sufficiently good solution is reached, or a set running time has expired. The goal of this paper is to determine the performance of various restart algorithms in generating solutions to the TSP in the context of iterative hill climbing and record keeping. The restart algorithms evaluated, described in details in ref. [8], are: 1) *Greedy Enumeration*, 2) *Greedy Jump*, 3) *GRASP* [12], 4) *Nearest Neighbor* [2], 5) *Clarke-Wright* [2], and 6) *Random*. Identical neighborhood generating algorithms are used for all implementations.

2.1 Record Keeping

A problem with the IHC using a constructive multi-start search method is that the same solution may be reached from multiple starting configurations. Therefore, many tours chosen are duplicate tours that have been explored in previous iterations of the algorithm. One method to combat this problem is to utilize record keeping mechanisms that record some or all tours that have been previously computed. Thus, if a tour is generated that has been considered previously, the algorithm restarts with a new starting configuration. We investigate two models of record keeping [4, 8]: 1) *Unbounded Memory,* 2) *Software Emulation of a Cache* [13]. We use set associative caches with small set sizes in our experiments.

3 Experiments

All experiments were performed using a set of TSP graphs from TSPLIB [14] and random graphs with a maximum of 100 vertices. The set of graphs consists of:

- 10 randomly-generated Euclidean graphs, 100 vertices each
- 10 random non-planar graphs, 100 vertices each
- 18 benchmark instances from TSPLIB [14], vertex counts of 16-100
- 400 random Euclidean graphs of 100 vertices each.

3.1 TSP Solution Quality

The experiment results are summarized in Table 1. Each row in the table shows the average quality of the solutions found by each of the algorithms in each of the experiments as a percentage of the best tour found by Concorde; lower numbers indicate better quality, with 100.0 being the best tour discovered. Note that the best tour may be found by multiple construction algorithms.

In the first set of experiments the IHC algorithm is executed using 10 randomly generated non-planar graphs. The average performance of the algorithms relative to the Concorde results, which are known to be Optimal with respect to this set of graphs, is shown in the first row of Table 1. Lower numbers are better with an ideal being 100 The second experiment runs the IHC algorithm using each of the construction algorithms over ten randomly generated planar Euclidean graphs are summarized in the second row of Table 1. The third experiment uses the TSPLIB benchmarks as the input for the construction algorithms [14].

The results of the experiments show that the nearest neighbor algorithm performs the worst of the six construction algorithms, and that the greedy based approaches perform well regardless of the structure of the input. Similarly, the Clarke-Wright

Table 1. Average performance of the construction algorithms.

	Greedy	Jump	GRASP	Random	CW	NN
Random graphs	**100.08**	**100.90**	**101.42**	**119.62**	**119.08**	**149.19**
Random Euclidean graphs	**100.83**	**100.83**	**100.50**	**100.29**	**102.80**	**110.43**
TSPLIB benchmarks	**100.66**	**100.37**	**100.73**	**100.18**	**101.75**	**109.72**

algorithm performs relatively poorly regardless the structure of the input graph. The random algorithm performed the best of the six algorithms on two of the data sets, but its poor performance on the non-planar graph data set brings its average performance to be less than the greedy approaches, and indicates that it is sensitive to the structure of the input graph.

3.2 Construction Algorithms and Tour Redundancy

The next set of experiments examines the quality and redundancy associated with some of the construction algorithms. Because the greedy algorithms perform among the best despite the structure of input graphs and due to the fact that the random restart does not exploit the record keeping efficiently, we perform this set of experiment only on them. Figure 1a shows the average redundancy and tour qualities produced when running IHC with the greedy construction algorithms. The average quality of all algorithms is nearly identical, but the greedy-jump algorithm has far less redundancy than the others. Figures 1b, 1c, and 1d show the distribution of solution quality for 400 random graphs solved using Concorde, GRASP, and GE. All the graphs show a distribution that is close to normal distribution [10], where Concorde has the best mean and lowest variance.

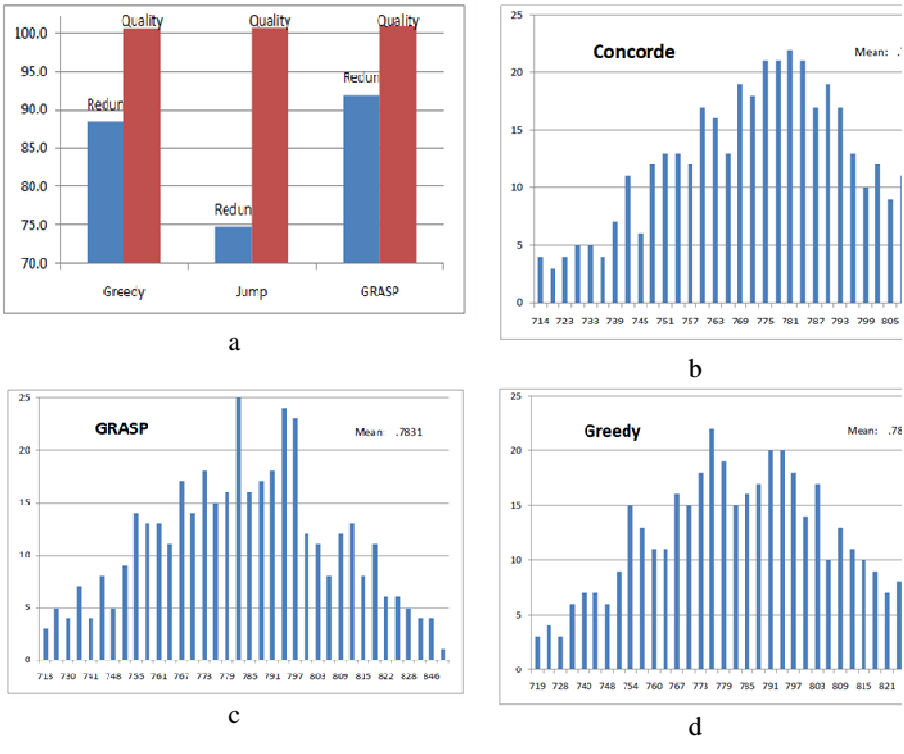


Fig. 1. Average tour Quality and Redundancy.

The mean and variance of GRASP and GE are almost identical and close to the mean and variance obtained by Concorde. Since GRASP has slightly better redundancy we used it as the subject for the next experiment.

3.3 Experiments with Record Keeping

Under the IHC algorithm, the generation of a tour that has been previously considered results in generating tours that have also been previously considered, and eventually leads to a local maximal tour that has been previously computed. Therefore, the obvious solution is to record all computed tours so that duplicates can be detected. In general, this is not feasible for large TSP instances, so we employ a cache mechanism to limit the size of memory used by the IHC algorithm. However, we first use unbounded memory record keeping on 50 random Euclidean graphs, of 100 vertices each, in order to determine an experimental upper bound on IHC speedup using record keeping. For IHC runs of 100,000 iterations, the average speedup using unbounded memory record keeping was 10.9. The same speedup is obtained with a 64,000 blocks cache (cache of 896KB). Figure 2 shows the speedup obtained with different cache configurations [2, 8, 13]. It is apparent that even with a small cache size 16KB, we still achieve a speedup of 6.7.

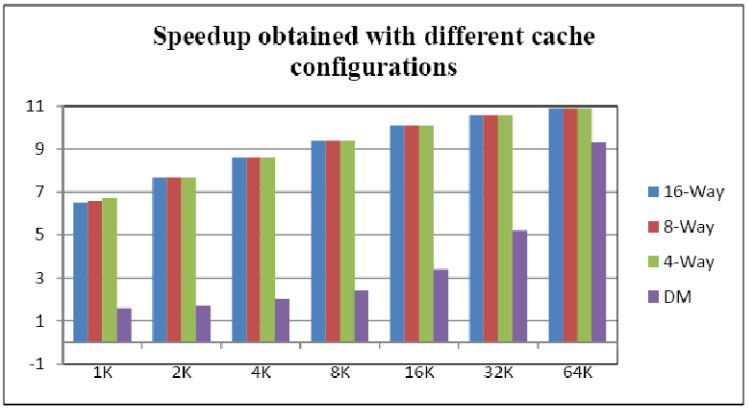


Fig. 2. Speedup obtained with different cache configurations.

4 Conclusion

In this paper, we evaluated the use of six different construction algorithms with the IHC method of solving the TSP. Our experimental results show that the greedy, greedy-jump, and GRASP construction algorithms all perform well. The greedy-jump construction algorithm also generates significantly fewer duplicate tours than the other algorithms, indicating that it may be more robust in situations where many local minima occur. Furthermore, we show that utilizing a record keeping mechanism modeled on cache memory is effective at improving IHC efficiency and can provide a speed up of up to 10.9x in the given configuration. Moreover, even when the memory is only large enough to hold a small percentage of duplicate tours generated, cache

still provides a significant performance improvement. Thus, regardless of the complexity of a TSP configuration and memory limits, even a small cache is useful. This work has applications in the study of the traveling salesman problem and in combinatorial optimization problems in general.

Future work involves further investigation of redundancy in the construction algorithms. We plan to investigate stochastic mechanisms to minimize redundancy and study its affects on algorithm performance and quality of the generated tours.

References

1. Cormen, T.H., et al.: Introduction to Algorithms. MIT Press, Cambridge (2001)
2. Johnson, D., McGeoch, L.: The Traveling Salesman Problem: A Case Study. In: Local Search in Combinatorial Optimization, pp. 215–310. John Wiley & Sons, Chichester (1997)
3. Garey, M.R., Johnson, D.S.: Computers and intractability: A guide to the theory of NP-completeness. W.H. Freeman, New York (1979)
4. Karhi, D., Tamir, D.E.: Caching in the TSP Search Space. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS, vol. 5579, pp. 221–230. Springer, Heidelberg (2009)
5. Allen, D., Darwiche, A.: Optimal Time-Space Tradeoff in Probabilistic Inference. In: Proceedings of the 21st International Joint Conference on Artificial Intelligence (2003)
6. Glover, F., Laguna, M.: Tabu Search. Kluwer Academic Publishers, Dordrecht (1997)
7. Glover, F., Taillard, E., de Werra, D.: A User's Guide to Tabu Search. Annals of Operations Research 41(1-4), 3–28 (1993)
8. King, C.R.: Improving the performance of constructive multi-start search using record keeping. In: Computer Science. Texas State University, San Marcos (2010)
9. Dawson, E., Wong, D.: Information Security Practice and Experience. Springer, Heidelberg (2007)
10. Applegate, D.L., et al.: The Traveling Salesman Problem, A Computational Study. Princeton University Press, Princeton (2006)
11. Ambite, J., Knoblock, C.: Planning by Rewriting. Journal of Artificial Intelligence Research, 207–261 (2001)
12. Pitsoulis, L.S., Resende, M.G.C.: Greedy Randomized Adaptive Search Procedures. In: Handbook of Applied Optimization, pp. 168–183. Oxford University Press, Oxford (2001)
13. Hennessy, J.L., Patterson, D.A.: Computer Architecture. In: A Quantative Approach. Morgan Kaufmann Publishers, Inc., San Francisco (2007)
14. Reinelt, G.: TSPLIB -A traveling salesman problem library. ORSA Journal on Computing 3(4), 376–384 (1991)

A Constraint Programming Application for Rotating Workforce Scheduling

Markus Triska and Nysret Musliu

Database and Artificial Intelligence Group
Vienna University of Technology
{triska,musliu}@dbai.tuwien.ac.at

Abstract. We describe *CP-Rota*, a new constraint programming application for rotating workforce scheduling that is currently being developed at our institute to solve real-life problems from industry. It is intended to complement *FCS*, a previously developed application. The advantages of *CP-Rota* over *FCS* are a significantly smaller and more maintainable code base, portability across a range of different language implementations and a more declarative approach that makes extensions easier and mistakes less likely. Our benchmarks show that *CP-Rota* is already competitive with *FCS* and even outperforms it on several hard real-life instances from the literature.

Keywords: Staff Scheduling, Cyclic Schedule, Manpower Scheduling, Timetabling.

1 Introduction

Computerized workforce scheduling has interested researchers for over 30 years. To solve rotating workforce scheduling problems, different approaches have been used in the literature, including exhaustive enumeration ([5], [2]), constraint (logic) programming, genetic algorithms ([8]) and local search methods.

In the present paper, we describe *CP-Rota*, a new constraint application for rotating workforce scheduling that is currently being developed at our institute to solve real-life problems from industry. It is intended to complement *FCS*, a previously developed application that is currently commercially used in many companies in Europe. *CP-Rota* builds upon, contributes to and improves previous constraint programming approaches to rotating workforce scheduling in the following ways:

- *CP-Rota* is written in portable Prolog and will eventually be released under a permissive licence to benefit both researchers and practitioners. Much of its code is already available on request at the time of publication.
- *CP-Rota* implements new allocation strategies (available as options for users to choose) that we discovered and discuss in this paper, which yield significantly improved performance on some real-life instances.
- Our benchmarks on real-life instances show the tremendous potential of constraint programming in rotating workforce scheduling, also and especially due to using different language implementations where they excel.

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 83–88.
springerlink.com

2 Related Work

Many different approaches for solving rotating workforce instances are documented in the literature. Balakrishnan and Wong [1] solved a problem of rotating workforce scheduling by modeling it as a network flow problem. Laporte [6] considered developing the rotating workforce schedules by hand and showed how the constraints can be relaxed to get acceptable schedules. Musliu et al. [9] proposed and implemented a method for the generation of rotating workforce schedules, which is based on pruning the search space by involving the decision maker during the generation of partial solutions. The algorithms have been included in a commercial product called First Class Scheduler (FCS) [4], which is used by many companies in Europe. In [10], Musliu applied a min-conflicts heuristic in combination with tabu search. Although this yields good performance on many instances, the resulting search method is incomplete and its results are therefore not directly comparable with FCS. This paper also introduced 20 real-life problems collected from different areas in industry and the literature.[1]

The use of constraint logic programming for rotating workforce scheduling was first shown by Chan in [3]. Recently, Laporte and Pesant [7] have also proposed a constraint programming algorithm for solving rotating workforce scheduling problems, implemented in ILOG and requiring custom extensions.

3 The Rotating Workforce Scheduling Problem

With *CP-Rota* and in the present paper, we focus on a specific variant of a general workforce-scheduling problem, which we formally define in this section. The following definition is from [9] and proved to be able to satisfactorily handle a broad range of real-life scheduling instances in commercial settings. A rotating workforce scheduling *instance* as discussed in the present paper consists of:

- n: Number of employees.
- A: Set of m shifts (activities) : a_1, a_2, ..., a_m.
- w: Length of the schedule. A typical value is $w = 7$, to assign one shift type for each day of the week to each employee. The total length of a planning period is $n \times w$ due to the schedule's cyclicity as discussed below.
- R: Temporal requirements matrix, an $m \times w$-matrix where each element $R_{i,j}$ shows the required number of employees that need to be assigned shift type i during day j. The number o_j of day-off "shifts" for a specific day j is implicit in the requirements and can be computed as $o_j = n - \sum_{i=1}^{n} R_{i,j}$.
- Sequences of shifts not permitted to be assigned to employees. For example, one such sequence might be ND (Night Day): after working in the night shift, it is not allowed to work the next day in the day shift. A typical rotating workforce instance forbids several shift sequences, often due to legal reasons and safety concerns.

[1] Examples available from *ht* /

- MIN_s and MAX_s: Each element of these vectors shows, respectively, the required minimal and permitted maximal length of periods of consecutive shifts of the same type.
- MIN_w and MAX_w: Minimal and maximal length of blocks of consecutive work shifts. This constraint limits the number of consecutive days on which the employees can work without having a day off.

The task in rotating workforce scheduling is to construct a *cyclic schedule*, which we represent as an $n \times w$ matrix $S_{n,w} \in A \cup \{\text{day-off}\}$. Each element $S_{i,j}$ denotes the shift that employee i is assigned during day j, or whether the employee has time off. In a cyclic schedule, the schedule for one employee consists of a sequence of all rows of the matrix S.

The task is called *rotating* or *cyclic* scheduling because the last element of each row is adjacent to the first element of the next row, and the last element of the matrix is adjacent to its first element. Intuitively, this means that employee i ($i < n$) assumes the place (and thus the schedule) of employee $i + 1$ after each week, and employee n assumes the place of employee 1. This cyclicity must be taken into account for the last three constraints above.

In the present paper, we consider the generation of a single schedule that satisfies all the hard constraints given in the problem definition. Fulfilling all these constraints is usually sufficient in practice. The same constraints that we use in this paper are used in the commercial software *FCS* for generating rotating workforce schedules. This system has been used since 2000 in practice for many companies in Europe and the scheduling variant we discuss in this paper proved to be sufficient for a broad range of uses. However, *FCS* has several shortcomings that led us to consider constraint programming as an alternative approach. We discuss these shortcomings and their remedies with *CP-Rota* in the next section.

4 Shortcomings of FCS and Their Remedies in CP-Rota

Although *FCS* has been in commercial use since 2000 in several companies and proved to be an acceptable solution for many applications in practice, it has several disadvantages that hinder its further development:

- The code base of *FCS* has gotten quite large and hard to maintain. This makes user modifications difficult and error-prone. New scheduling ideas cannot be rapidly prototyped but require substantial development effort.
- *FCS* was implemented in *Visual Basic* and thus depends on essentially a single supported language implementation, which is in addition also not freely available. This complicates the sharing of code with other researchers and practitioners for joint development and turns every mistake in the language implementation into a potentially show-stopping problem.
- From [10], it is known that local search approaches – although incomplete – can significantly outperform *FCS* on some instances. Clearly, it is highly desirable to improve the running times of *FCS* to more closely match such competing approaches while retaining the completeness of the search.

When we started to work on the successor of *FCS* for the above reasons, we initially looked into constraint programming in the hope to significantly reduce the size of its code base. The promise of constraint programming was to just state the necessary requirements with high-level constraints and to then use built-in enumeration methods to search for solutions.

We implemented the successor using the portable constraint programming model described in Section 5 and named it *CP-Rota*. Eventually (see Section 7), *CP-Rota* even outperformed its predecessor on many instances, making constraint technology a potential remedy for all of the above original shortcomings.

5 A Constraint Formulation for Rotating Workforce Scheduling

Our initial development environment was SWI-Prolog, which we chose due to its convenient libraries, tools and workflow, and also because it is freely available. We then ported the model to GNU Prolog due to its much better performance, and because it is also freely available. changes. When experimenting with custom allocation strategies (Section 6), GNU Prolog's lack of garbage collection hindered testing with larger instances, and we therefore ported the model also to B-Prolog, which is also a very efficient Prolog implementation and available free of charge for personal use. In all these systems, we model the rotating workforce problem as follows:

- The schedule is represented as a list of lists, and each element is a finite domain variable that denotes the shift type scheduled for this position.
- The temporal requirements are enforced via `global_cardinality/2` constraints on the columns of the schedule. In GNU Prolog, `fd_exactly/3` constraints are used instead.
- The minimal/maximal-length constraints on consecutive shifts of the same type are enforced via `automaton/3`.
- Reified constraints are used to map shifts of all types to either "work" or "day-off", and a second `automaton/3` constraint is used on these reified variables to limit the number of consecutive work and day-off shifts.
- Reified constraints are also used to express forbidden patterns. For example, if "0 4 3" is forbidden, the constraint is:

$$X_k \ \#= 0 \ \#/\backslash \ X_{k+1} \ \#= 4 \ \#==> X_{k+2} \ \#\backslash= 3$$

  for all variables X_k, also taking into account the schedule's cyclicity. In B-Prolog, `notin/2` (negated extensional) constraints are used for better performance.

It only took a few days to implement this basic model (700 LOC, including a 50 LOC parser for instance specification files and 50 LOC for visualisations) and to get it to run on all of the above Prolog implementations. Only built-in constraints are used in all systems. In contrast, the development of *FCS* took several months.

6 Labeling and Allocation Strategies

The default strategy in *CP-Rota* is to first label the (reified) work/"day-off" Boolean variables. Then, all original variables of the schedule are labeled with the "first-fail" option. We call this strategy S_1. When S_1 did not yield a solution within 1000 seconds, we used Strategy S_2, which is to label all schedule variables from left to right, trying their values from lowest to highest. If this does not yield a solution within 1000 seconds, S_3 is used: Reified constraints are used to compute, for each column, the number of still missing shifts of each type. Processing the columns in order, we then choose the shift type that misses the *least* number of elements in that column, and assign it to a feasible variable with *smallest* domain. When S_3 also fails to find a solution within 1000 seconds, S_4 is used: It is similar to S_3, except the columns are not processed from left to right, but in descending order of their number of still missing shifts of any type.

Table 1. Comparison between *FCS* and *CP-Rota*.

Ex.	n	*FCS* (time in sec)	*CP-Rota* (sec)	Strategy
1	9	0.9	**0.02**	S_1
2	9	0.4	**0.02**	S_1
3	17	1.9	**0.24**	S_1
4	13	1.7	**0.03**	S_1
5	11	3.5	**0.98**	S_1
6	7	2	**0.02**	S_1
7	29	16.1	**0.07**	S_2
8	16	**124**	964	S_1
9	47	>1000s	**19**	SWI, S_4
10	27	**9.5**	>1000s	–
11	30	**367**	>1000s	–
12	20	>1000s	>1000	–
13	24	>1000s	**114**	S_1
14	13	**0.54**	940	S_1
15	64	>1000s	>1000s	–
16	29	**2.44**	216	S_1
17	33	>1000s	**18**	SWI, S_3
18	53	**2.57**	>1000s	–
19	120	>1000s	>1000s	–
20	163	>1000s	>1000s	–

7 Comparison with the Commercial System FCS

Table 1 compares the performance of *CP-Rota* with that of *FCS* on 20 real-life instances from [10]. To the best of our knowledge, *FCS* is a state-of-the-art commercial system for generating rotating workforce schedules. We tested all instances on a Pentium 4, 1.8 GHZ, 512 MB RAM, using the latest versions of *FCS*, GNU Prolog (1.3.1) and SWI-Prolog (5.9.10). Except where stated otherwise, timing results are from GNU Prolog.

The table shows that *CP-Rota* nicely complements its predecessor so that 3 more instances than previously can now be solved. On 7 instances, *CP-Rota* outperforms *FCS* already with its default strategy (S_1), the converse holds for 6 instances.

8 Future Work

Future improvements to *CP-Rota* include the addition of a more convenient user interface, real-time interaction with decision makers and the implementation of additional allocation strategies.

References

1. Balakrishnan, N., Wong, R.T.: A network model for the rotating workforce scheduling problem. Networks 20, 25–42 (1990)
2. Butler, B.: Computerized manpower scheduling. Master's thesis, University of Alberta, Canada (1978)
3. Chan, P., Weil, G.: Cyclical staff scheduling using constraint logic programming. In: Burke, E., Erben, W. (eds.) PATAT 2000. LNCS, vol. 2079, pp. 159–175. Springer, Heidelberg (2001)
4. Gärtner, J., Musliu, N., Slany, W.: Rota: A research project on algorithms for workforce scheduling and shift design optimisation. Artificial Intelligence Communications 14(2), 83–92 (2001)
5. Heller, N., McEwen, J., Stenzel, W.: Computerized scheduling of police manpower. St. Louis Police Department, St. Louis (1973)
6. Laporte, G.: The art and science of designing rotating schedules. Journal of the Operational Research Society 50, 1011–1017 (1999)
7. Laporte, G., Pesant, G.: A general multi-shift scheduling system. Journal of the Operational Research Society 55(11), 1208–1217 (2004)
8. Mörz, M., Musliu, N.: Genetic algorithm for rotating workforce scheduling. In: Proceedings of second IEEE International Conference on Computational Cybernetics, Vienna, Austria, pp. 121–126 (2004)
9. Musliu, N., Gärtner, J., Slany, W.: Efficient generation of rotating workforce schedules. Discrete Applied Mathematics 118(1-2), 85–98 (2002)
10. Musliu, N.: Heuristic Methods for Automatic Rotating Workforce Scheduling. International Journal of Computational Intelligence Research 2(4), 309–326 (2006)
11. Pesant, G.: A regular language membership constraint for finite sequences of variables. In: Wallace, M. (ed.) CP 2004. LNCS, vol. 3258, pp. 482–495. Springer, Heidelberg (2004)

An Agent-Based Architecture for Multifaceted Online Dispute ResolutionTools

Davide Carneiro, Paulo Novais, and José Neves

Department of Informatics, University of Minho, Braga, Portugal
{dcarneiro,pjon,jneves}@di.uminho.pt

Abstract. Online Dispute Resolution (ODR) tools are being seen as a way to settle disputes out of courts, namely under virtual environments. However, the acceptance of these alternative methods is still relatively restricted, once existing tools are relatively undemanding anddomain-centered.Indeed, there is the need for conceptual tools whose parts may be arranged for particular use, i.e., to operate in different domains. Following this line of attack, in this paper it will be presented a new agent-based approach to ODR.It comes in an abstract and formal form, in order to beindependent of the legal domains, but specific enough to be applied to concrete ones. The main advantage is that functionality reuse is maximized, making architectures simpler to implement and to expand.

Keywords: Multi-agent Systems, Online Dispute Resolution, Intelligent Environments.

1 Introduction

The technological evolution led to unprecedented changes in virtually every aspect of the society. As an example, a significant percentage of contracts in the most different fields is now signed under an electronic or virtual environment, whereas in the past all contracts were signed with the physical presence of all the parties involved. Another difference is that in the past these contracts were paper-based, while now they are completely electronic, without a mandatory paper support. The most evident example of this new reality is visible on the recent e-commerce phenomenon [1], of which web-sites like the ones of Amazon and e-Bay are the best examples. However, similarly to traditional paper-based contracts, disputes can also arise in an electronic setting. In fact, according to Javelin Strategy and Research[1], only 45% of electronic costumers are satisfied with the quality of products sold online. It is nowadays clear that the traditional approach for solving disputes by means of litigation in court is not feasible anymore.

An alternative way of solving disputes arising out of electronic contracting is Online Dispute Resolution (ODR), allowing already traditional alternative dispute resolution methods such as negotiation, mediation or arbitration to be moved "from a physical to a virtual place" [2]. ODR makes availableto all the parties an undemanding way of litigation, and a simple and efficient way to deal with disputes, thus saving time and assets [3].

[1] https://www.javelinstrategy.com/research

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 89–94.
springerlink.com

In devising such systems, we are taking into consideration the Katsch/Rifkin vision of the four parties in an ODR process: the two opposing parties, the third neutral party and the technology that works with the mediator or arbitrator [4].We clearly assume a gradual tendency to foster the intervention of autonomous software agents, whichacteither as decision support systems or as real electronic mediators [5]. The consideration of this wider role for software agents is based in the use of Artificial Intelligence based methodologies and techniques eitherfor problem solving or knowledge representation and reasoning, such as Case-Based Reasoning [6].

In this paper we focus on the problematic of developing an architecture for multi-domain dispute resolution, whose objective is to enable a range of services targeted at assisting the disputant parties, independently of the domain of the dispute. This architecture is abstract in the sense that it encompasses perceptions that are common to the several domains addressed. It is also specific in the sense that it provides support for the specificities of each domain, in a transparent manner. Indeed, the objective of this work is to develop an agent-based architecture that can be used in several legal domains.It is being developed in the context of the Portuguese legal system, covering three major fields of The Law: Family Law [13], Consumer Law [12] and Labor Law [11].

2 Identifying Abstract Perceptions and Processes

In a dispute resolution process we can talk of abstract entities that are present regardless the domain of the dispute. As an example, to a certain point, a negotiation process will always be a sequence of rounds in which, in each round, each party states itsview about the current proposal on the table, i.e., the process goes on independently of the subject of the proposals. The same happens with certain perceptions. As an example, independently of the domain of the dispute, a party will always be interested in knowing its best and worst possible scenario. Therefore, in the development of an abstract architecture, firstly one has to identify which perceptions and processes are present in the quite a few domains that can be modeled.

2.1 Abstract Perceptions

On determining the abstract picture for a multi-domain ODR tool aimed at assisting the parties, one mustpursue by determining which information would actually be useful for the parties and then state the overall perception of the situation. Undeniably, it would be interesting for a party to determine to which extent is it reasonable to engage in a dispute resolution process, i.e.,would a better outcome be reached using an alternative dispute resolution process instead of litigation?

The thought that encompasses this is the BATNA - Best Alternative to a Negotiated Agreement, or the possible best outcome “along a particular path if I try to get my interests satisfied in a way that does not require negotiation with the other party” [7]. This conception is abstract as it is useful for any dispute, and it is constructive for parties once, at least, it contributes to the acknowledgement that an agreement may be disadvantageous [8]. A similar hypothesis is the Worst Alternative To a Negotiated Agreement (WATNA) [9]. A WATNA intends to estimate the worst possible

outcome along a litigation path. It can be quite relevant in the calculation of the real risks that parties will face in a judicially determined litigation, imagining the worst possible outcome for the party.

However, it couldalso be interesting to consider the roomalready settled between the BATNA and WATNA, as a useful element to be taken into account for making (or accepting) a proposal. Of course, excluding the space between BATNA and WATNA, less dangerous is for the party not to accept the agreement (unless, of course, its BATNA is really unfavorable). A wider room between BATNA and WATNA would usually mean that it becomes rather dangerous for the party not to accept the ODR agreement (except in situations when the WATNA is really not inconvenient at all for the party). This ideais evidently related to the Zone of Possible Agreement (ZOPA) proposed by Raiffa (1982) [10].

Moreover, it would be interesting for a party to understand the region of such a space in which aresult is more likely to come about, i.e., if the parties are going to solve the dispute through litigation, which is the most likely outcome? In fact, sticking only with the BATNA and WATNA may not be realistic. Thus, an informed party should also consider concept of MLATNA – Most Likely Alternative to a Negotiated Agreement. Using the same arguments, it may bring to a close that the existence of metrics that assess the probability of each possible outcome could also be extremely useful for a party. Thus, it will be possible to consider the view of plausible case: a possible outcome with an associated value of likeliness.

2.2 Abstract Processes

Likewise, there are processes that implement key functionalities, which are intangible enough to be reused in a number of legal domains. Concretely, two abstract processes were applied: case retrieval and negotiation. The former caters for a selection of a set of past known cases that can be relevant, considering the current dispute. Parties can then analyze these cases, and obtain valuable information about past similar disputes. The remaining one is concerning with negotiation, making available a common negotiation process in which two or more parties state their opinions in sequential rounds about the subject being negotiated. This process allows the parties to reach a consensus about the outcome of a dispute.

3 An Architecture for a Multifaceted ODR Platform

Let us now depict the architecture that implements the ideas described so far. There are two main issues: the actual abstract architecture and an ontology that allows for the agents to perform specific tasks according to the domain of discourse.

3.1 The Architecture

The architecture builds on a set of software agents, autonomous and proactive, that cooperate in order to achieve their goals. On the other hand, a development strategy was devised that organizes the agents in two categories, namely: high-level agents and low-level ones. High-level agents perform tasks that do not need explicit domain-dependant information. Low-level ones are closer to the legal domain, inneedof

methodologies for problem solving and/ormethods for knowledge representing and reasoning, in particular to think up the legal procedures. In general, high-level agents monitor and set the behaviour ofthe low level ones. Low-level agents act on the object-level data and knowledge, that make the extensions of the functions that describe the universe of discourse.

Let us take as example anegotiation process:high-level agents guide the process and determine when a new round should start, or finish; low-level ones have the autonomy to choose the actions to be performed, according to the state of the domain of discourse. That is, depending on the domain, low level agents will compute a valid output in each round.All the agents and their roles are depicted in [11]. These agents were defined according to the specific requirements of the project, following an iterative cut-down practiceof increasing specification.

3.2 The Ontology

Having specified the high level architecture, let us now show how it can be used to implement specific services. This will be done through the use of ontologies. The method consists in defining a specific ontology for each legal domain. Each ontology encodes a domain theory, actions, constraints and rules, revealing to the agent what action is to be executed and how to achieve it, i.e., according to the legal domain being attended to. The main advantage of this approach is that a single agent can be used to perform a similar task in a wide range of domains in opposition to a traditional one in which an agent would be used for each different domain. As an example, let us consider the action of searching for similar cases. Instead of having three different agents, one for each of the legal domains under equation (i.e., Commercial, Family and Labour), we have one single agent and three different ontologies.This line of attack significantly increases functionality reuse and allows for a single architecture supporting services in a wide range of domains. It also simplifies the task of extending the architecture for addressing new legal domains by developingan ontology foreveryone, with all the actions, rules, constraints and specific theories. This will tell the low-level agents how to act in a new domain.

Each ontology comprises four components: the vocabulary, the actions, the features and a theory. The vocabulary contains all the words that can be used to describe the entities that belong to the domain. Actions define how each action should be executed according to the domain of the ontology. The theory will define all the elements that make up the perceptions about the universe of discourse (e.g., type, number). Finally, features allow for add up of invariants to the perceptions and actions that make up the ontology.

As an example, let us consider the computation of the BATNA and WATNA in two different legal domains, namely: Commercial Law and Labour Law. These twoabstract conceptions denote, as it was described above, the best and worst possible scenario in a litigation process, i.e., it has a meaning that can be of use independently of the domain of the dispute. Nevertheless, depending on that domain, it is computed in different ways. In Labour Law we will have to consider views in terms of worker antiquity, monthly salary, seniority, a joust cause for dismissal, just to name a few. On the other hand, in Commercial Law key views would be the date in which the product

was bought, the type of the product, the type of the warranty, the state of the product, among others.Thus, two different ontologies (i.e., Labour and Commercial) define the act of computing the BATNA and WATNA, in different ways:

A simplification of the rules that allow the computation of the BATNA and WATNA for the Portuguese Labour Law, as it is given in Decree of Law (DL) 7/2009 (Portuguese laws)..

```
Def_Rule 396

if RULE_394 then
  WATNA := 3 * (M_SALARY + SENIORITY)
  if TEMPORARY_CONTRACT then
    if WATNA < M_REMAINING *(M_SALARY + SENIORITY) then
      WATNA := M_REMAINING *(M_SALARY + SENIORITY)
  if WATNA < 15 * (D_SALARY + SENIORITY) then
    WATNA := 15 * (D_SALARY + SENIORITY)
  BATNA := 45 * (D_SALARY + SENIORITY)
  if BATNA < DAMAGE then
    BATNA := +DAMAGE
```

A simplification of the rules that allow the computation of the BATNA for the Commercial Law, as it is given in DL 67/2003. I this example rule it will be considered only numbers 1 to 4 of Article 5th.

```
Def_Rule5
if IS_MOBILE then
  if DEFECT_COMPLAINT_DELAY < 60 then
    if WARRANTY_DELAY < 730 then
      BATNA := {"product repair in 30 days";
                "product replacement"}
     else BATNA := {"no indemnity due"}
  else BATNA := {"no indemnity due"}
else
  if DEFECT_COMPLAINT_DELAY <365 then
    if WARRANTY_DELAY < 1810 then
      BATNA := {"product repair in reasonable time";
```

4 Conclusion

One of the major challenges that Online Dispute Resolution faces is the lack of tools that can address more than one legal field. In fact, most of existing tools focus only on small and very limited domains. Following the approach presented in this paper, three prototypes focusing on different legal domains are being developed. These prototypes are based on a single agent-based architecture targeted at providing services for dispute resolution. The services implemented are abstract enough to be useful in several legal domains and contain the specific rules needed to be used in each specific domain. Using ontologies to specialize a single agent in several legal domains makes the architecture simple and makes it easy to expand it to other legal domains, by adding new ontologies. This, we believe, will foster the development of multi-domain Online Dispute Resolution Platforms.

Acknowledgments. The work described in this paper was developed under the TI-ARAC - Telematics and Artificial Intelligence in Alternative Conflict Resolution Project (PTDC/JUR/71354/2006), a research project supported by FCT (Science & Technology Foundation), Portugal.

References

1. Goel, R.: E-Commerce. New Age International (2007)
2. Bellucci, E., Lodder, A., Zeleznikow, J.: Integrating artificial intelligence, argumentation and game theory to develop an online dispute resolution environment. In: ICTAI 2004: 16thIEEE International Conference on Tools with Artificial Intelligence, pp. 749–754 (2004)
3. Klaming, L., Van Veenen, J., Leenes, R.: I want the opposite of what you want: summary of a study on the reduction of fixed-pie perceptions in online negotiations. In: Expanding thehorizons of ODR, Proceedings of the 5th International Workshop on Online Dispute Resolution (ODR Workshop 2008), Firenze, Italy, pp. 84–94 (2008)
4. Katsch, E., Rifkin, J.: Online Dispute Resolution – resolving conflicts in cyberspace. Jossey-Bass Wiley Company, San Francisco (2001)
5. Peruginelli, G., Chiti, G.: Artificial Intelligence in Online Dispute Resolution. In: Proceedings of the Workshop on the law of electronic agents – LEA (2002)
6. Kolodner, J.L.: Case-based Reasoning. Morgan Kaufmann, San Francisco (1993)
7. Notini, J.: Effective Alternatives Analysis In Mediation: BATNA/WATNA Analysis Demystified (2005), http://www.mediate.com/articles/notini1.cfm
8. Klaming, L., Van Veenen, J., Leenes, R.: I want the opposite of what you want: summary of a study on the reduction of fixed-pie perceptions in online negotiations. In: Expanding the horizons of ODR, Proceedings of the 5th International Workshop on Online Dispute Resolution (ODR Workshop 2008), Firenze, Italy, pp. 84–94 (2004)
9. Steenbergen, W.: Rationalizing Dispute Resolution: From best alternative to the most likely one. In: Proceedings 3rd ODR Workshop, Brussels (2005)
10. Raiffa, H.: The art and science of negotiation: how to resolve conflicts and get the best out of bargaining. The Belknap Press of Harvard University Press, Cambridge (1982)
11. Carneiro, D., Novais, P., Andrade, F., Zeleznikow, J., Neves, J.: The Legal Precedent in Online Dispute Resolution. In: Governatori, G. (ed.) Proceedings of the Jurix 2009 - the 22nd International Conference on Legal Knowledge and Information Systems, Legal Knowledge and Information Systems, pp. 47–52. IOS press, Rotterdam (2009)
12. Costa, N., Carneiro, D., Novais, P., Andrade, F.: An Agent-based Approach to Consumefs Law Dispute Resolution. In: Proceedings of the 12th International Conference on Enterprise Information Systems - ICEIS (2010)
13. Café, A., Carneiro, D., Novais, P., Andrade, F.: Online Dispute Resolution tool for Property Division - Conjugal and Hereditary. In: INFORUM (2010) (in portuguese)

Towards an Agent-Based Knowledge Management System

Marwan Altawil[1], Asma Moubaiddin[2], and Nadim Obeid[1]

[1] Department of Computer Information Systems,
King Abdullah II School for Information Technology,
The University of Jordan
obein@ju.edu.jo
[2] Department of Linguistics,
Faculty of Foreign Languages,
The University of Jordan

Abstract. In this paper, we propose a knowledge management system that employs agents and case-based reasoning to extract tacit knowledge and support organizational learning and decision making at the level of individual workers, group workers and the whole organization. Agents will perform the tasks of inquiry, investigation, sharing and updating of knowledge bases, including previous similar cases, in order to help in finding the required knowledge for a specific problem. We have applied the system to develop a banking training assistant system and have run several experiments to measure a number of functionalities such as reliability, efficiency, the impact of weight parameters on extracted results, and the degree of match between the results produced by our system and the opinions of randomly selected experts in the chosen application.

Keywords: Knowledge Management, Case-Based Reasoning, Multi-Agent.

1 Introduction

Knowledge is considered as a critical component of organization competency and an essential asset that requires careful management so that organizations can enhance and facilitate the decision making process [6]. We distinguish between explicit and tacit knowledge. Explicit knowledge could be explored and easily shared with others using documents, databases, and so on [1]. Tacit (embodied) knowledge is embedded inside a person's mind, therefore it is difficult to extract and share with others [4]. Organizational Knowledge/Memory (OM) refers to more than just databases and information systems involved in an organization. It encompasses processes and people. Decision makers (DM) in an organization often need to employ both explicit and tacit knowledge that are related to the specific problem which needs to be solved. The study of OM seeks to understand how knowledge is created, stored, accessed, transformed, handled and used effectively. Knowledge Management (KM) should provide relevant knowledge to assist processes in performing knowledge intensive tasks [12].

A Knowledge Management System (KMS) should involve providing knowledge, experiences and insights at the right time, to the right place, in the right form to the right knowledge worker [9, 10, 12]. These features characterize intelligent agents.

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 95–100.
springerlink.com

With agents, it is possible to capture the representation, coordination, and cooperation between heterogeneous processes and other agents. An appropriate KMS should have the ability to extract proper tacit knowledge when requested to do so [9, 10, 11] and has to support the process of DM and Organizational Learning (OL).

In this paper, we propose a KMS that employs agents and Case-Based Reasoning (CBR) to extract tacit knowledge and support OL and decision making at the level of individual worker, group workers and the whole organization. Agents will perform the tasks of inquiry, investigation, sharing and updating of knowledge bases, including previous similar cases, in order to help in finding the required knowledge for a specific problem. We have applied the system to develop a banking training assistant system and have run several experiments to measure a number of functionalities such as reliability, efficiency, the impact of weight parameters on extracted results, and the degree of match between the results produced by our system and the opinions of randomly selected experts in the chosen application. Section 2 will be concerned with Multi-Agent Systems (MAS), OL and CBR. In section 3 we present an agent-based learning KMS. Section 4 will be concerned with data collection and implementation of a training assistant **system** for the central bank of Jordan.

2 MAS, OL and CBR

Agents are specialized problem solving entities with well-defined boundaries with the ability to communicate with other agents. They are designed to fulfill a specific purpose and exhibit flexible and pro-active behavior. They are autonomous in the sense that they can operate on their own without the need for guidance. They have control both over their internal state and over their actions.

A MAS can be defined as a collection of agents with their own problem solving capabilities and which are able to interact among them in order to reach an overall goal [5]. KM is mainly concerned with using, spreading, sharing, representing and storing the available knowledge of an organization. OM is about how to collect, store, and provide access to experiences and skills of both stored records and tacit knowledge. OL is concerned with enhancing the organization problem solving process [13]. In [3], a model of learning is proposed. It includes three learning levels: individual, group and organizational and two routes of flow: upward (individual to organizational) and downward. This suggests that knowledge be structured to capture the three learning levels.

CBR is a mechanism for solving new problems based on the solutions of similar past problems [7]. CBR supports many of the activities of KM and has been frequently proposed as a methodology for KM applications. CBR has been employed in the design of many KM systems [2].

3 Towards an Agent-Based Learning KMS

An important feature of a KMS is Knowledge Communication (KC). KC allows for making knowledge accessible to and reusable by its different components [10, 11, 12]. We employ **four** types of intelligent agents: (1) Knowledge Cooperative Agent (KCA) that will be responsible for analyzing and searching the Knowledge-Bases

(KBS) of the OM; (2) Reasoning Agent (RA) that will be involved with CBR tasks; (3) Learning Agent (LA) that will be responsible for modifying the OM and Task Agent (TA) that will be responsible for identifying when and how to apply KC. TA will be responsible for identifying to what level a specific problem belongs, whether in the CBR-Base or the KB. RA will investigate the CBR-base to identify similar cases to the given problem, and then RA will find percentages of similarities to cases and give the results back to the TA. If a desired case is founded then no need to call KCA. Otherwise, TA will call KCA in order to find the appropriate experts who may give a consultation about the problem.

The different components of the system are illustrated in Figure 1 below.

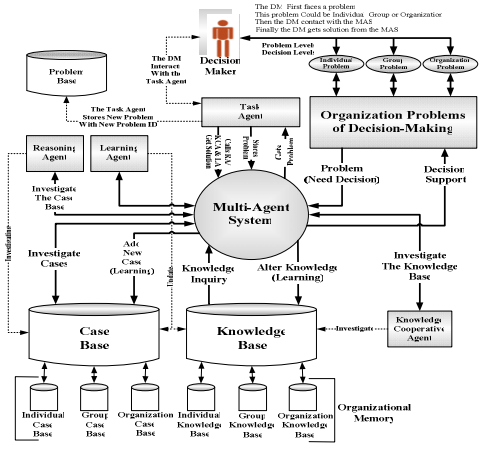


Fig. 1. A Learning KMS.

OM is divided into two parts: one for KBs of related entities and the other for CBR-Bases. Figure 2 illustrates the structure of an OM. There are three knowledge levels. An ontology is employed to structure the case bases into hierarchies (cf. Figure 3). It stores an abstraction and the mapping of all domain-specific concepts and meanings and can be used to verify the existence of all acquired concepts in KB. A similarity measure is defined in order to retrieve similar cases. For each attribute of a given case and a given query, a similarity can be calculated. The attribute similarities can be combined to an overall similarity value. The employment of similarity measure makes it possible to rank the solutions.ïïA case represents a previous experience of a situation. It consists of two parts: a problem description together with the context/environment in which it has occurred and a solution to this case. KB can be accessed during the retrieve phase using a similarity based measure. When presented with a particular case, DM calls TA in order to identify the type/level of problem based on its parameters. Weights can be associated with parameters to indicate their importance. Weights help in fine-tuning and reaching more accurate decisions during reasoning by enabling RA to retrieve the most similar and relevant cases. TA will generate a unique problem-ID and store the problem in the problem-base. RA can use the problem-ID and ontology in order to investigate the level to which the problem

belongs in the CBR-Base. The results are then passed on to TA to be studied in order to determine whether or not there is a need to involve KCA whose task is to finds the appropriate Knowledge Expert/s (KE) that can deal with the task. The process involves determining the appropriate knowledge level that contains similar solutions for the current problem. TA then presents the solution case to the KE for approval. If approved, LA will add the new case at the appropriate level.

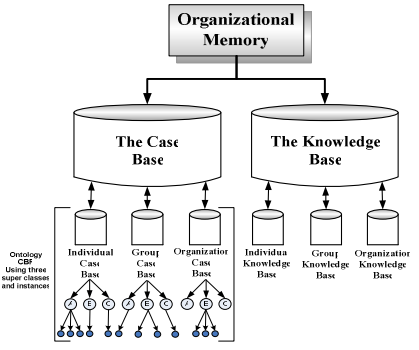


Fig. 2. Structure of OM.

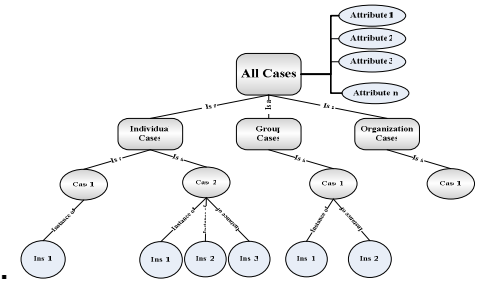


Fig. 3. Case Base Structure using Ontology.

4 Data Collection and System Implementation

The Computer Department (CD) in The Central Bank of Jordan (CBJ) has to provide the technical support for the functional operations carried out at CBJ. The Training Division (TD) in the **CBJ** has to regularly prepare the Training Need Matrix (TNM) **for CD**. Some training bodies may make offers regarding their current and future training courses including their specifications.

We have collected: (1) 1000 precious individual-cases, (2) 150 previous group-cases, (3) 50 organization-level previous course cases, (4) information/knowledge about 45 individual employees's job such as names, descriptions and responsibilities and (5) information about Groups's jobs such as names, descriptions and responsibilities.

We have used PL-SQL developer Ver. 6.0.3.893. **a**s our Multi-Agent development tool for developing TA, RA**, LA** and **KCA**. Reader may refer to http://www.ju.edu.jo/sites/Academic/obein/Lists/Academic%20Conferences/AllItems.aspx for the pseudo-code of the agents.**.** We have run several experiments where each is designed to measure a specific functionality as follows:

1. **Test-Retest Reliability:** The reliability of the model depends on reliability of the agents. We have measured the reliability of TA, RA, **KCA** and LA**.**
2. **Measuring Internal Consistency Reliability for RA**: We measure internal consistency reliability for RA with regard to a single parameter. Figure 4 shows the values of internal consistency reliability for 5 trials. The minimum value is 80%.
3. **Measuring System Efficiency:** We have run experiments to measure the efficiency of RA and KCA. We increase the number of cases and experts in the following order (1000, 2000, …, 6000) for the case-base and (45, 90, 135, 180, 225, 270) for KB. Figure 5 shows the additional execution times RA needed to investigate the new cases. Figure 6 shows that the performance RA varies in a stable manner when the number of cases of additional organizations increases. (cf. Figure 7).

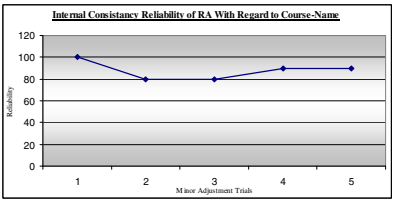


Fig. 4. Measuring the internal consistency reliability for RA.

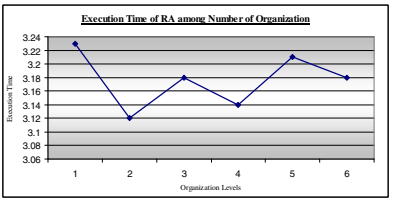


Fig. 5. Execution times that RA needs to investigate new cases.

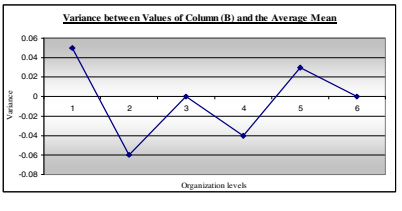


Fig. 6. Variance values of execution vs Org. levels & average mean.

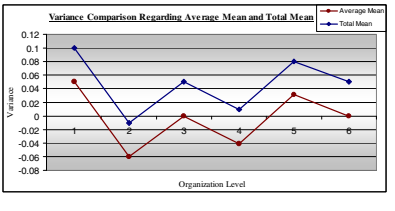


Fig. 7. Variance to average mean vs variance to total mean.

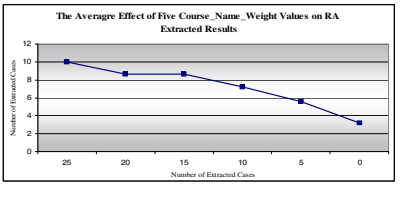


Fig. 8. Extracts by RA as course_name increases.

4. **Measuring the Impact of Parameters' Weight on Results:** We have run 10 different trials, on each parameter while decreasing its value such as 25, 20, …, 0 for course_name of a random problem. The other parameters are kept fixed. Figure 8 above shows the results that RA extracts are affected by the weight of course_name. When the weight increased the number of extracted cases increased.

6 Concluding Remark

We have, in this paper, proposed a KMS that employs agents and CBR to extract tacit knowledge and support OL and decision making. The experimental results, using various measurement methods, are reliable and efficient. We compared the results of the system with that of a committee of 7 experts from CBJ. The results were very positive. Out of 37 cases, there was a match on the 1st choice of 23 cases, 12 on the 2nd choice and 2 on the 3rd choice made by the system. Furthermore, the experts indicate that the system has indeed the ability to learn.

References

1. Abdullah, M., Benest, I., Evans, A., Kimble, C.: Knowledge Modeling Techniques for Developing Knowledge Management Systems. In: 3rd European Conf. on Knowledge Management, pp. 15–25 (2002)
2. Althoff, K.-D., Weber, R.: Knowledge management in case-based Reasoning. The Knowledge Engineering Review 20(3), 305–310 (2006)
3. Castaneda, D., Rios, M.: From Individual Learning to Organizational Learning. The Electronic Journal of Knowledge Management 5(4), 363–372 (2007)
4. Crossan, M., Lane, H.W., White, R.E.: An Organizational Learning Framework: From Intuition to Institution. Academy of Management Review 24(3), 522–537 (1999)
5. Ferber, J.: Multi-Agent Systems. Addison-Wesley, Reading (1999)
6. Frank, L.: Knowledge Management or the Management of Knowledge? In: King, W. (ed.) Knowledge Management and Organizational Learning, Ann. of Info. Systems, vol. 4 (2009)
7. Garrido, J., Hurtado, M., Noguera, M., Zurita, J.: Using a CBR Approach Based on Ontologies for Recommendation and Reuse of Knowledge Sharing in Decision Making. In: 8th Int. Conf. on Hybrid Intelligent Systems (2008)
8. Hans, K., Henk, S., Sylvia, S.: How to Improve Your Knowledge Intensive Organization: Implementing a Knowledge Management Scan Within Public and Private Sector Organizations. Elec. Journal of Knowledge Management 7(1), 77–86 (2009)
9. Moubaiddin, A., Obeid, N.: Towards A Formal Model of Knowledge Acquisition Via Cooperative Dialogue. In: 9th Int. Conf. on Enterprise Information Systems (2007)
10. Moubaiddin, A., Obeid, N.: Dialogue and Argumentation in Interdisciplinary Communication. In: 12th World Multi-Conf. on Systemics, Cybernetics and Informatics, 3rd Int. Symp. on Knowledge Communication and Conferences, pp. 318–323 (2008)
11. Moubaiddin, A., Obeid, N.: Partial Information Basis for Agent-Based Collaborative Dialogue. Journal of Applied Intelligence 30(2), 142–167 (2009)
12. Obeid, N., Moubaiddin, A.: Towards a Formal Model of Knowledge Sharing in Complex Systems. In: Nguyen, N., Szczerbicki, E. (eds.) Intelligent Systems for Knowledge Management. SCI, Springer, Heidelberg (2009)
13. Zhang, C., Tang, D., Liu, Y., You, J.: A Multi-Agent Architecture for Knowledge Management System. In: 5th Int. Conf. on Fuzzy Systems and Knowledge Discovery, pp. 433–437 (2008)

An Approach to Auction-Based Web Server Admission Control

Jolanta Wrzuszczak-Noga and Leszek Borzemski

Institute of Informatics, Wroclaw University of Technology,
50-370 Wrocław, Poland
{Jolana.Wrzuszczak,Leszek.Borzemski}@pwr.wroc.pl

Abstract. In this paper innovative web admission auction-based algorithms are presented combined with load-balancing and QoS characteristics. Performance indexes as income of the web service or client satisfaction is discussed. Simulation experiments have been performed to validate the proposed mechanisms. They showed benefits of our developments.

Keywords: auction mechanisms, admission control, load-balancing, scheduling algorithms in web service, quality of service.

1 Introduction and Background

Web services are used to provide the increasing volume of data in the computer network systems. Web clients are waiting for delivering data in a proper time, i.e. within the bounded values of the time delay. In many web services it is required to pay charges for utilizing their resources. Such services should be equipped with efficient admission control algorithms providing right and guaranteed service level (Quality of Service, QoS) to their users.

In the literature there are various approaches to the problem of controlling the distribution of service resources to the clients e.g. [3], [4], [6], [8]. The most common idea is based on the FIFO scheduling, so that resources are assigned to users according to the earliest time stamp of reported requests (offers). Other proposed scheduling policies taken into consideration are: the mean time of serving requests known from the history or distinguishing some requests classes according to requested resource size. The aspects of service level guarantee in web services were discussed in [3], [9] as a resources delivery in a proper time or as transfer parameter (bandwidth) assuring.

The auction-based algorithms were also presented in many knowledge domains, i.e. for energy resources management, for radio spectrum selling, for bandwidths or telecommunication goods management [2], [6], [7]. Proposed systems do not concentrate on serving the single users, thus they are oriented on providers and often use agent systems.

Modern web services are designed for business profit. They try to apply some market mechanisms to solve scheduling problems and to serve user requests according to market prices in such a way to increase the benefit of web service. This may be done by application of auction algorithms which are aiming for buying and selling of goods

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 101–106.
springerlink.com

through a dynamic pricing strategy. Various multi-access and multi-functional auction systems were designed and implemented, including e-Bay, Allegro or Merkato.

Several researches had tackled with the problem of decision making based on a specific auction policy named *Vickrey auction*. The basic idea of this kind of auction algorithms is that the price, which should be paid by the winners, is lower than declared [1], [6], [7], [10], [11].

In this paper two new auction-based scheduling algorithms were introduced and Vickrey mechanism for pricing offers was applied. The paper considers a combined approach of auction algorithm, QoS and admission control of requests, related to individual clients. The main contribution of this paper is an application of new mechanism of auction based on admission control of user requests arriving to the web service. Our proposal is featured with a charging schema in respect with acceptable QoS levels and closely connected with a client satisfaction. Proposed algorithms guarantee the time delivering of service resources (i.e. multimedia files, audio books files, books as text file) [10].

The process of offers collection from web clients will be discussed in the second section. The third section formulates the problem. In the fourth section the auction based scheduling mechanisms will be proposed. The fifth section presents simulation results. In the last section the conclusions are given.

2 User Offers

The role of web service will be to schedule web client requests and to guarantee them defined service level (defined bandwidth) for downloading files. Every user (web client) will be obtained to reveal his offer. Every offer ($o_n = \{r_j, p_{u^i k}, b_{u^i k}\}_n$) will be characterized by triple of following variables: resource name, price, and bandwidth.

Let's analyze an example: offer o_1=[r_1, 25$, 128 kb/s] includes resource r_1 for 25$ for bandwidth 128 kb/s, offer o_2=[r_1, 40$, 192 kb/s] includes resource r_1 for 40$ for bandwidth 192 kb/s. Allowable is to report many offers for one resource by one client, but the declared price for increasing number of bandwidth pieces should reflect non-decreasing valuation function.

3 Problem Formulation

The aim of the web service will be to manage of the bandwidth pieces assuring assigned transmission speed. The service will support delivering of data files, for which web client will be obligated to pay a charge related to requested bandwidth.

The problem will be formulated as follows:
For given:

- o_1, …., o_N – the set of offers, each offer contains the resource name, price and auctioned bandwidth,
- R – the set of data files (resources), $r_j \in R$,
- T – scheduling time interval,
- B – total available bandwidth,

- the performance index defining an income of the web service (sum of submitted offers),

a scheduling admission strategy should be found to select the set of offers S_d for processing, which increases the performance index and assures the client satisfaction simultaneously in every time instant (T). In this paper only the situation, in which server resources are not sufficient to satisfy all users, because of the fact that the total sum of expected bandwidth exceeds the total volume of bandwidth offered by the web service, is considered.

4 Algorithms

The managing process should increase the income of the web service (the sum of serviced offers) and should satisfy the web clients. Two scheduling algorithms will be discussed and examined. The first one – class-based auction algorithm is more justified and the second one – profitability based auction algorithm based only on proposed offers.

The first algorithm guarantees, that some offers from every request for every class will be selected in respect of the just treatment and customer satisfaction . The later policy does not consider class based selection and can lead to postponement of large files because of the same price both for small and large files. In both cases also a modified criterion could be defined, taking in account the client satisfaction. It could be e.g. the price paid by users (charge). When the client will pay finally the lower price than declared, he will be satisfied [1], [8], [9], [10]. The scheduling algorithm will run in following periods, which are defined by the periodic instances of the scheduling interval T (i.e. every 30 min).

4.1 Class-Based Auction Algorithm

In this approach, it is assumed, that every resource belongs to one class [3]. The classes are determined by the data size, i.e. three classes are there distinguished:

- the *bronze* for data size less than 10MB,
- the *silver* for data size between10MB and 50 MB,
- the *gold* for data greater than 50MB.

The next assumption is that the defined piece of total bandwidth is assigned justify for all classes, it means that from every class some offers will be served.

The rest will be assigned to offers containing the maximal value of profitability index. The approach algorithm consists of three phases.

Phase 1. The first phase of the algorithm based on the clear rules – to every class some bandwidth will be assigned. Let's assume that 60% of the total bandwidth is assigned for all the requests classes.

We consider three classes, so that, to every class 20% of total bandwidth will be assigned. For every class the offers with the highest price will be scheduled until the bandwidth for class is available.

The third assumption is that every class has a specific bandwidth, i.e.:

- *bronze* class (128kb/s),
- *silver* class (256 kb/s),
- *gold* class (512 kb/s).

Phase 2. The second stage of this algorithm takes a profitability metrics into consideration. For every offer an index will be calculated and then the offers with the highest value will be scheduled until the bandwidth is given. The profitability metrics φ_n can be defined as follows (1):

$$\varphi_n = \frac{p_{u_i k}}{t_i} \tag{1}$$

where

$$t_n = \frac{sizeof(r_j)}{b_{u_i k}} \tag{2}$$

t_n - time of transfer j-*th* resource for bandwidth $b_{u_i k}$ (2).

The classification to some classes in the second stage doesn't matter.

During this stage following set of offers will be chosen (3):

$$S_d = \{o_n\} => \max \sum_{i=1}^{L} \varphi_n \tag{3}$$

and by assumption (4)

$$\sum_{i=1}^{L} b_{u_i k} \leq 40\% B \tag{4}$$

Phase 3. The third phase is used to serve the dynamic requests. In case, when during the defined scheduling interval *T* some requests are rejected, the free bandwidth will be assigned to dynamic offers. The price will be calculated basing on the individual offer and requested bandwidth. The assigned bandwidth will base also on separately negotiation process. The third phase takes place only during the scheduling interval (when 0 < dynamic scheduling time < *T*).

4.2 Profitability-Based Auction Algorithm

This approach is based only on the value of profitability index defined in the second stage of the class-based auction algorithm. The following set of offers will be chosen (5):

$$S_d = \{o_n\} => \max \sum_{i=1}^{L} \varphi_n \tag{5}$$

and by assumption (6)

$$\sum_{i=1}^{L} b_{u_i k} \leq B \,. \tag{6}$$

The class division is not taken into consideration. The second phase is similar to the third phase presented in the class-based auction approach.

Some algorithms were developed to increase the income of web service and for the client satisfaction, and were based on the second-highest price (Vickrey) approach.

5 Results of Simulation

Three algorithms: Class-based Auction Algorithm, Profitability based Auction Algorithm and FIFO Scheduling Algorithm were implemented in the Matlab environment. The stream of offers was simulated as a Poisson Process. For different number of offers, and different available bandwidths (10Mbps and 62.5 Mbps) the profit of web service was measured and compared.

Fig. 1 and Fig. 2 present the performance index defined as the sum of the income of web service vs. number of serviced offers. The biggest profit (service income) was observed for Profitability based Auction Algorithm (depicted by squares) and a bit worse for Class-based Auction Algorithm (depicted by rhombus). The results for FIFO Algorithm (depicted by triangles) were always the worst. For the increasing number of offers the difference between Profitability based Auction Algorithm and Class-based Auction Algorithm was practically fixed whereas the FIFO algorithm stands out and the difference to others is increasing for growing number of offers.

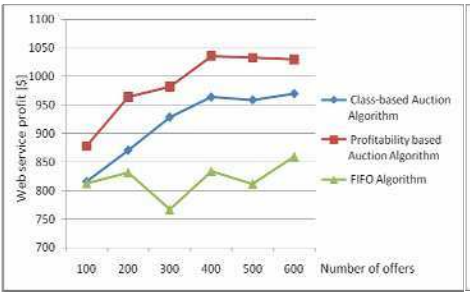


Fig. 1. Web service profit versus number of offers for 10 Mbps available bandwidth.

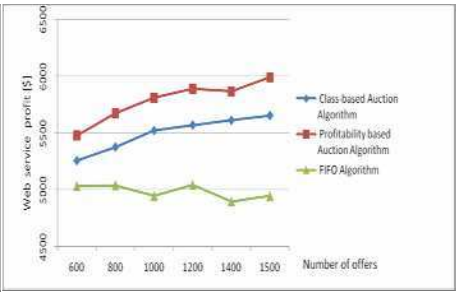


Fig. 2. Web service profit versus number of offers for 62.5 Mbps available bandwidth.

6 Conclusions

In this paper two new scheduling algorithms based on auction approach were presented. The classification of offers was depicted in an example. The profitability index was shown. The idea of scheduling of offers was shown and discussed. Some features of presented mechanisms were distinguished.

The results of simulations were also presented. The experiments showed that the auction based scheduling process could be judged as very effective in the context of decision making in a web service with QoS.

New approaches will be discussed, as offering more resources or the possibility of delivering data in later time.

The later research will be focused on experiments in the environment of the IBM blade cluster system.

References

1. Borzemski, L., Wrzuszczak, J.: Management of Web service delivering multimedia files based on the bid strategy. In: Information Systems Architecture and Technology, ISAT 2008, pp. 13–23 (2008)
2. Brazier, F., Cornelissen, F., Gustavsson, R., Jonker, C., Lindeberg, O., Polak, B., Treur, B.: A multi-agent system performing one-to-many negotiation for load balancing of electricity use. Electronic Commerce Research and Applications 1, 208–222 (2002)
3. Cherkasova, L., Phaal, P.: Peak Load Management for commercial Web servers using adaptive session-based admission control. In: Proceedings of the 34th Hawaii International Conference on System Sciences (2001)
4. Cramton, P., Shoham, Y., Steinberg, R.: Combinatorial auctions. The MIT Press, Cambridge (2006)
5. Lee, M., Lui, J., Yau, D.: Admission Control and Dynamic Adaptation for a Proportional-Delay DiffServ-Enabled Web Server. In: Proceedings of the 2002 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, pp. 172–182 (2002)
6. Maille, P., Tuffin, B.: Pricing the Internet with multibid auctions. IEEE/ACM Trans. on Networking 14(5), 992–1004 (2006)
7. Perez-Bellido, A., Salcedo-Sanz, S., Portilla-Figueras, J.A., Ortiz-Garcia, E.G., Garcia-Diaz, P.: An agent system for bandwidth allocation in reservation-based networks using evolutionary computing and vickrey auctions. In: Nguyen, N.T., Grzech, A., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2007. LNCS (LNAI), vol. 4496, pp. 476–485. Springer, Heidelberg (2007)
8. Vickrey, W.: Counterspeculation, Auctions, and Competitive Sealed Tenders. The Journal of Finance 16(1), 8–37 (1961)
9. Walsch, M., Wellman, P., Wurman, J., MacKie-Mason, K.: Some Economics of Market-Based Distributed Scheduling, Distributed Computing Systems. In: Proceedings of 18th International Conference Amsterdam, pp. 612–621 (1998)
10. Wrzuszczak, J.: Auction mechanism in management of processing nodes in a computer cluster. Contemporary Aspects of Computer Networks 2, 259–265 (2008)
11. Wrzuszczak, J., Borzemski, L.: Management of Web services based on the bid strategy using the user valuation function, Computer Networks. Communications in Computer and Information Science 39, 19–25 (2009)

Adaptive Fuzzy Rules Emulated Networks Controller for 7-DOF Robotic Arm with Time Varying Learning Rate

C. Treesatayapun

Department of Robotic and Manufacturing, CINVESTAV-Saltillo,
Ramos Arizpe, 25900, Mexico

Abstract. This article presents an adaptive controller based on fuzzy rules emulated network (FREN) with the proposed on-line learning algorithm. The human knowledge about the unknown plant, 7- DOF robotic arm in this case, is transferred to be if-then rules for setting the network structure. All adjustable parameters are tuned by the on-line learning mechanism with time varying step size or learning rate. The main theorem is introduced to improve the system performance and stabilization through the variation of learning rate. Experimental system based on Mitsubishi PA-10 is demonstrated the control algorithm validation.

Keywords: Discrete-time; adaptive control; neuro-fuzzy; 7-DOF robotic arm.

1 Introduction

To design the controller for robotic manipulators, dynamic and kinematic models usually are the normal requirements even those controllers has been constructed by artificial intelligence tools such as neural networks (NNs) and fuzzy logic systems [1,2]. The system performance is commonly related on the model accuracy or approximated model in the case of using NNs to estimate robotic models. In this work, on the other hand, a direct adaptive controller based on fuzzy rules emulated networks [3] (FRENs) is introduced with out any requirement of robotic system modeling. According to an on-line learning, the close-loop performance can be guaranteed by using time varying learning rate.

Recently, to ensure the system performance small learning rates or step sizes are often used in gradient search methods because of the system stability but the small step size can slow down the reaching solution [5]. In [6], the suitable learning rate which can guarantee the system stability has been widely discussed but the using limit or operation range has been considered because of fixed variable learning rate.

2 Robotic System with Non-affine Discrete-Time Systems

According to design the controller with out any requirement of robotic system modeling, this section, we consider these robotic systems as a class of non-affine formulation as follows:

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 107–112.
springerlink.com

$$x(k+1) = f_N(\boldsymbol{\mu}(k), u(k)), \tag{1}$$

when $f_N(\cdot)$ is an unknown nonlinear function and $\boldsymbol{\mu}(k) = [x(k)\, x(k-1) \cdots x(k-n)\, u(k-1)\, u(k-2) \cdots u(k-m)]^T$. According to some assumptions and the approximation mentioned in [4], the system (1) can be rearranged as

$$x(k+1) = F_N(\boldsymbol{\mu}(k)) + G_N(\boldsymbol{\mu}(k))u(k). \tag{2}$$

In this work, let us assume that these nonlinear functions $F_N(k)$ and $G_N(k)$ are all unknown. The control effort $u(k)$ is directly determined by FREN as

$$u(k) = \boldsymbol{\beta}^T(k)\boldsymbol{\phi}(k), \tag{3}$$

when $\boldsymbol{\beta}(k) \in R^l$ is an adjustable parameter vector and $\boldsymbol{\phi}(k) \in R^l$ is FREN's basis function vector where l denotes as the number of fuzzy rules.

3 Parameters Tuning Algorithm

In this work, only on-line leaning mechanism is applied with the associate of some designed parameters. The gradient descent method with the proposed time varying step size is introduced to adjust these parameter β_i for $i = 1, 2, \cdots, l$.

The cost function $E(k)$, which is needed to be minimized, can be defined as

$$E(k) = \frac{1}{2}e^2(k), \tag{4}$$

where $e(k) = x_d(k) - x(k)$. At time index $k+1$, all adjustable parameters β_i can be determined by

$$\beta_i(k+1) = \beta_i(k) - \eta(k)\frac{\partial E(k+1)}{\partial \beta_i(k)}, \tag{5}$$

when $\eta(k)$ is a time varying learning rate. In this work, we introduce the determination method to obtain the possible biggest learning rate when the system stability can be guaranteed.

Apply the chain rule through (4) and (2), we obtain

$$\begin{aligned}\frac{\partial E(k+1)}{\partial \beta_i(k)} &= \frac{\partial E(k+1)}{\partial x(k+1)}\frac{\partial x(k+1)}{\partial u(k)}\frac{\partial u(k)}{\partial \beta_i(k)}, \\ &= -[x_d(k+1) - x(k+1)]y_p(k)\phi_i(k).\end{aligned} \tag{6}$$

Thus, the tuning law can be rewritten as

$$\beta_i(k+1) = \beta_i(k) + \eta_i(k)e(k+1)y_p(k)\phi_i(k), \tag{7}$$

where $y_p(k)$ denotes $\frac{\partial x(k+1)}{\partial u(k)}$. Let us consider the system formulation in (2) again, clearly, we have

$$y_p(k) = G_N(\boldsymbol{\mu}(k)). \tag{8}$$

4 Closed Loop Performance

The key of this work is to determine the learning rate $\eta(k)$ for every time index k. Substitute the control effort $u(k)$ given by (3) into the system formulation (2), we have

$$x(k+1) = F_N(k) + G_N(k)\boldsymbol{\beta}^T(k)\boldsymbol{\phi}(k). \tag{9}$$

Thus, the next time step error can be rewritten as

$$e(k+1) = x_d(k+1) - F_N(k) - G_N(k)\boldsymbol{\beta}^T(k)\boldsymbol{\phi}(k), \tag{10}$$

Substitute (10) into (7), the adaptation law can be obtained as

$$\begin{aligned}\boldsymbol{\beta}(k+1) = [I - \eta(k)G_N^2(k)\boldsymbol{\phi}^T(k)\boldsymbol{\phi}(k)]\boldsymbol{\beta}(k) + \eta(k)[x_d(k+1),\\ -F_N(k)]y_p(k)\boldsymbol{\phi}(k).\end{aligned} \tag{11}$$

For the convenient presentation, let us select learning be

$$\eta(k) = \frac{\gamma}{G_u^2\boldsymbol{\phi}^T(k)\boldsymbol{\phi}(k)}, \tag{12}$$

when G_u is the positive upper bound of $\frac{\partial f_N(\boldsymbol{\mu}_k, u_k)}{\partial u_k}$ as given by assumption 2b and $0 < \gamma < 2$ is the designed parameter.

Next, we introduce the proof of system performance based on the proposed controller with the following theorem.

Theorem 2. *System stability(Closed loop system convergence)*

Let the desired trajectory $x_d(k)$ be bounded and the upper bound of $G_N(k)$ be known as G_u. Determine the control effort $u(k)$ by (3) and tune parameters by (7) with the varying learning rate given by (12) when $0 < \gamma < 2$. Then the tracking error $e(k)$ is bounded for the nonlinear system given by (1).

Proof. Let define the Lyapunov function candidate as

$$V(k) = \frac{1}{2}e^2(k), \tag{13}$$

thus the change of Lyapunov function can be given as

$$\begin{aligned}\Delta V(k) &= V(k+1) - V(k),\\ &= \Delta e(k)[e(k) + \frac{\Delta e(k)}{2}].\end{aligned} \tag{14}$$

Let consider the next time index error can be written by $e(k+1) = e(k) + \Delta e(k)$, where $\Delta e(k)$ can be estimated by $\Delta e(k) \approx \Delta_{e,\beta}^T(k)\Delta\boldsymbol{\beta}(k)$, when $\Delta_{e,\beta}(k) = [\frac{\partial e(k+1)}{\partial \beta_1(k)}\ \frac{\partial e(k+1)}{\partial \beta_2(k)}\ \cdots\ \frac{\partial e(k+1)}{\partial \beta_l(k)}]^T$. By using the chain rule, we have

$$\begin{aligned}\frac{\partial e(k+1)}{\partial \beta_i(k)} &= \frac{\partial e(k+1)}{\partial x(k+1)}\frac{\partial x(k+1)}{\partial u(k)}\frac{\partial u(k)}{\partial \beta_i(k)},\\ &= -G_N(k)\phi_i(k).\end{aligned} \tag{15}$$

Thus, we obtain $\Delta e(k) \approx -G_N(k)\boldsymbol{\phi}^T(k)\Delta\boldsymbol{\beta}(k)$. From the tuning law obtained by (11), the change of adjustable parameters can be rewritten as

$$\Delta\boldsymbol{\beta}(k) = -\eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2\boldsymbol{\beta}(k) + \eta(k)[x_d(k+1) - F_N(k)]G_N(k)\boldsymbol{\phi}(k). \quad (16)$$

By using (16), we have

$$\begin{aligned}\Delta e(k) &= \eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2[-x_d(k+1) + F_N(k) + G_N(k)\boldsymbol{\phi}^T(k)\boldsymbol{\beta}(k)],\\ &= -\eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2[e(k+1)]. \end{aligned} \quad (17)$$

We can rearrange (17) as

$$\Delta e(k) = \frac{-\eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2 e(k)}{1+\eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2}. \quad (18)$$

Substitute (18) into (14), we obtain

$$\Delta V(k) = \frac{-\eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2 e^2(k)}{1+\eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2}\Big[1 - \frac{\eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2}{2+2\eta(k)G_N^2(k)||\boldsymbol{\phi}(k)||^2}\Big]. \quad (19)$$

With the learning rate give by (12), the change of Lyapunov function candidate can be rearranged by

$$\Delta V(k) = \frac{-\gamma\big[\frac{G_N(k)}{G_u}\big]^2}{1+\gamma\big[\frac{G_N(k)}{G_u}\big]^2}\Big[1 - \frac{\gamma\big[\frac{G_N(k)}{G_u}\big]^2}{2(1+\gamma\big[\frac{G_N(k)}{G_u}\big]^2)}\Big]e^2(k) \le 0. \quad (20)$$

□

Remark: According to (20), sign or direction of $G_N(k)$ is not needed.

The importance thing that we need to estimate is G_u. With the following examples, the parameter G_u is clearly designed and discussed.

5 Experimental Setup and Results

In this experimental setup, the proposed control algorithm is implemented to control a 7-DOF Mitsubishi PA-10 robotic arm system. The overall system configuration is illustrated in Fig. 1. Seven FRENs are designed to control each of joints independently. This robotic system is operated in the velocity mode control which means FRENs must generate the velocity commands to move very joints to follow the desired trajectory.

To begin the controller design, the suitable IF-THEN rules are all needed to be specified. Based on the knowledge of PA-10, those IF-THEN rules can be given as the followings:

If $e(k)$ is PL	Then $\omega_1(k) = \beta_{PL}(k)\phi_1(k)$,	Large e in positive;	Fast ω in positive,
If $e(k)$ is PS	Then $\omega_2(k) = \beta_{PS}(k)\phi_2(k)$,	Small e in positive;	Slow ω in positive,
If $e(k)$ is Z	Then $\omega_3(k) = \beta_{Z}(k)\phi_3(k)$,	Reach desired position;	Stop moving !!!,
If $e(k)$ is NS	Then $\omega_4(k) = \beta_{NS}(k)\phi_4(k)$,	Small e in negative;	Slow ω in negative,
If $e(k)$ is NL	Then $\omega_5(k) = \beta_{NL}(k)\phi_5(k)$,	Large e in negative;	Fast ω in negative,

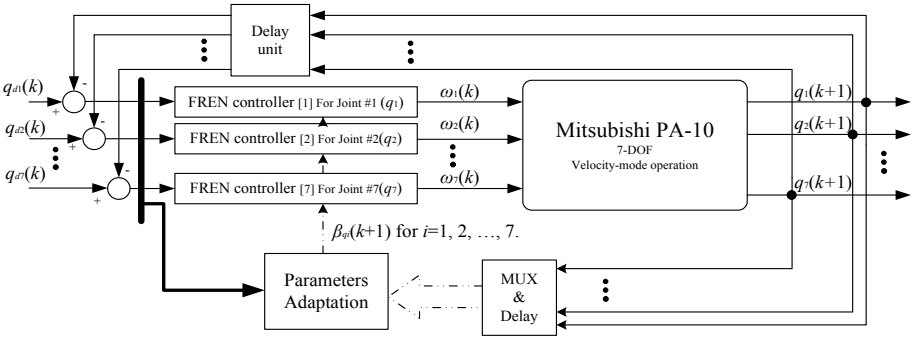


Fig. 1. PA-10 FREN controllers configuration block diagram.

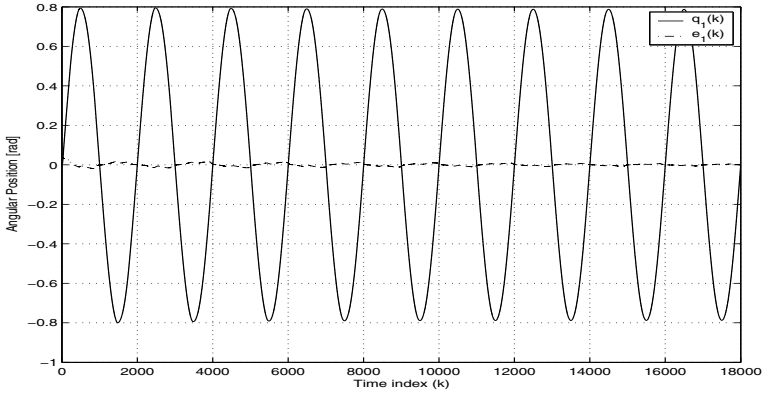


Fig. 2. PA-10 Tracking performance $q_1(k)$ and $e_1(k)$.

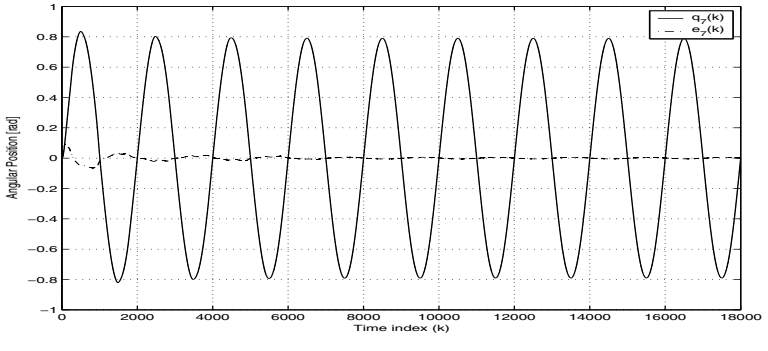


Fig. 3. PA-10 Tracking performance $q_7(k)$ and $e_7(k)$.

when ω denotes the velocity command determined by FREN and $e(k)$ is the position error defined by $q_d(k) - q(k)$ where $q_d(k)$ is the desired position at time index k.

The tracking performance for every joints q_1 and q_7 can be illustrated in Fig. 2 and 3, respectively, with theirs error signals. This experimental has been carried out for all 7 joints but we display only the first and seventh joints because of the page number limitation.

6 Conclusion

An adaptive control based on fuzzy rules emulated networks for 7- DOF robotic arm has been introduced in this article. The designer knowledge about the robotic arm including the physical limit has been directly integrated to FRENs through its IF-THEN rules and initial parameters. To control all DOF, the parallel configuration of FRENs has been demonstrated with the closed-loop performance analysis. The variation of learning rate has guaranteed the system stability along the on-line learning phase. The system validation has been confirmed by the experimental setup with Mitsubishi PA-10 robotic arm.

Acknowledgment

The authors would like to thank CONACyT (Project # 84791) for the financial support through this work.

References

1. Wai, R., Huang, Y., Yang, Z., Sliih, C.: Adaptive fuzzy-neural-network velocity sensorless control for robot manipulator position tracking. IET. Control Theory & Applications 4(6), 1079–1093 (2010)
2. Ferreira, J., Crisostomo, M., Coimbra, A.: SVR Versus Neural-Fuzzy Network Controllers for the Sagittal Balance of a Biped Robot. IEEE Transactions on Neural Networks 20(12), 1885–1897 (2009)
3. Treesatayapun, C.: Nonlinear discrete-time controller with unknown systems identification based on fuzzy rules emulated network. Applied Soft Computing 10, 390–397 (2010)
4. Treesatayapun, C.: A discrete-time stable controller for an omni-directional mobile robot based on an approximated model. Control Engineering Practice 19, 194–230 (2011)
5. Lin, C.T.: Neural fuzzy systems. Prentice-Hall, Englewood Cliffs (1996)
6. Jagannathan, S.: Neural Network Control of Nonlinear Discrete-Time Systems. Taylor & Francis, Boca Raton (2006)
7. Spooner, J.T., Maggiore, M., Ordonez, R., Passino, K.M.: Stable Adaptive Control and Estimation for Nonlinear Systems: Neural and Fuzzy Approximator Techniques. John Wiley & Sons, New York (2002)

Belief Management for Autonomous Robots Using History-Based Diagnosis

Stephan Gspandl, Ingo Pill, Michael Reip, and Gerald Steinbauer*

Institute for Software Technology
Graz University of Technology
Inffeldgasse 16b/2, A-8010 Graz, Austria
{sgspandl,ipill,mreip,steinbauer}@ist.tugraz.at

Abstract. Non-deterministic reality is a severe challenge for autonomous robots. Malfunctioning actions, inaccurate sensor perception and exogenous events easily lead to inconsistencies between an actual situation and its internal representation. For a successful mission in dynamic environments, a robot is thus required to efficiently cope with such inconsistencies.

In this paper, we present a belief management system based on the well-known agent programming language IndiGolog and history-based diagnosis. Extending the language's default mechanisms, we add a belief management system that is capable of handling several fault types that lead to belief inconsistencies. First experiments in the domain of service robots show the effectiveness of our approach.

Keywords: belief management, diagnosis, history-based diagnosis, autonomous robot.

1 Introduction

There is an increasing interest to target autonomous mobile robots for complex tasks in dynamic (non-deterministic) environments. Related target applications range from simple transportation services via visitor guidance in a museum to autonomous car driving[1,2]. The complexity of such application domains raises the demands regarding autonomous reasoning capabilities of deployable systems. Appropriate robots have to consider, for instance, a high and dynamic number of entities and objects, including their complex and spatial relations. In this context, inherent challenges a robot has to live up to are those of facing novel scenarios and having to cope with situations where a robot's assumptions about a situation are in conflict with reality.

Imagine for instance a delivery robot whose task is to move objects between rooms and assume that it perceives some object that is known to be at some other place. Obviously, there is more than one explanation for this crucial inconsistency. For instance, (1) that the robot drove to the wrong room (execution fault), (2) a sensing fault, and (3) that someone moved the object (unobserved exogenous event). A robot that is unable to handle such inconsistencies would neither be able to successfully finish the given

* The authors are listed in alphabetical order. The work has been partly funded by the Austrian Science Fund (FWF) by grants P22690 and P22959.

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 113–118.
springerlink.com

task, nor to continue operation from a sane situation. Thus, in order to ensure quality of service, a reasoning mechanism and belief management is needed that is able to deal with this ambiguity by considering multiple hypotheses.

For our implementation of such a belief management system, based on the agent programming language IndiGolog, we use the situation calculus and adapt history-based diagnosis [3] developed by Iwan and colleagues. Our system allows us to detect inconsistencies, describe belief ambiguities, and generate multiple hypotheses as well as rank them. While we adopt the favorite hypothesis for immediate operation, we keep track of the alternatives in case the chosen favorite is proven to be wrong by future data.

The remainder of this paper is organized as follows. Related work is covered in Section 2. In Section 3 we depict our belief management system, where section 3.1 contains some preliminaries and Section 3.2 covers the details of our system. Experiments can be found in Section 4, followed by conclusions drawn in Section 5.

2 Related Research

In the following we discuss previous work in three research areas that are relevant to our paper: (1) fault diagnosis, (2) hypothesis discrimination and (3) dealing with uncertainties in acting and sensing.

Diagnosis, i.e. the detection and localization of faults, is a major topic in research. Many approaches like [4,5] specifically address the diagnosis of sensing and/or actuator faults. For complex and dynamic environments, classic diagnosis like consistency-based diagnosis [6] is unfortunately too static. While it blames faulty components, for dynamic environments it is more appropriate to focus on correct action (event) sequences [7]. In [3], Iwan describes a diagnosis approach for robots in dynamic environments that is based on the situation calculus, which we will discuss in Section 3.1.

Once a set of diagnoses has been found, the correct one has to be isolated, or at least we have to identify the most probable one(s). In [8] the authors proposed a planning method to derive action sequences that improve the diagnosis quality by ruling out inappropriate ones. Handling uncertainty in sensing and acting is another issue relevant to our work. There are several approaches using the situation calculus to deal with this problem. For example, formalizing a planning problem, one can decide the relevance of unexpected changes for an actual plan [9]. The authors of [10] employ situation calculus as well as decision-theoretic planning to control an autonomous robot. Whenever passive sensing differs from expectations, re-planning is initiated. In [11] Thielscher et al. presented a formalization using the fluent calculus to derive explanations for unexpected situations. Their approach employs active sensing to minimize the impact of such events. Weld et al. [12] extended GraphPlan to handle sensing actions and uncertainty, but they do not maintain the consistency of a system's belief.

3 History-Based Diagnosis and Belief Management for IndiGolog

In this section we present a belief management system that ensures the consistency of a robot's belief with reality (as it is perceivable), preceded by necessary preliminaries.

3.1 Situation Calculus, IndiGolog and History-Based Diagnosis

IndiGolog (Incremental Deterministic Golog) [13] is a logic-based programming and planning language used for agents and robots. It lets the user choose whether computational resources should favor planning or imperative programs, and is based on Reiter's variant of the situation calculus [14]. With the situation calculus, all activities and changes in the world are encoded as actions. A situation is therefore completely determined by some initial situation and an action sequence (that is also called history). This entails that the terms situation and history are interchangeable. Properties are described by situation-dependent predicates and functions, named fluents. The user specifies dependencies between a fluent and an action by *successor state axioms* (SSAs). The so-called *basic action theory* (BAT) [14] models the environment and the robot's capabilities. IndiGolog programs are interpreted by a transition semantics in a step-by-step fashion. The semantics of program statements are defined by *Trans* and *Final* predicates. $Final(\sigma, s)$.

Program interpretation in IndiGolog is based on a five step main cycle [13]:

1. All exogenous events reported in the last cycle are integrated into the history.
2. The currently considered history fragment is moved forward if necessary. Taking only a fragment into consideration keeps the history at a processable length.
3. The program σ is checked if it may terminate in the current situation s using *Final*.
4. The program σ evolves a single step using the predicate *Trans*, a primitive action is executed if applicable and the history changes accordingly.
5. Sensing results obtained in the current cycle are integrated into the history.

This cycle is repeated until either the program terminates legally or there is no further executable legal transition.

In [3], Iwan describes a diagnosis approach, named *history-based diagnosis* (HBD), for robots in dynamic environments. HBD generates action sequences (histories) that are consistent with some observation Φ that contradcits the original history. Based on the situation calculus, alternative histories are generated by two operations: (1) variation of individual actions in the original sequence (action and sensing faults), and (2) insertion of exogenous actions. In order to isolate the most attractive history out of a possibly large set of hypotheses, probabilities and the amount of variation from the original one are used to rank the alternatives.

3.2 A Belief Management System for Dynamic Environments

We designed and implemented our belief management system around history-based diagnosis and IndiGolog. For our purpose, we had to adapt the standard features and the interpretation cycle of IndiGolog by adding diagnostic reasoning and related decision procedures as well as features to handle multiple hypotheses.

Our concept for handling multiple hypotheses mainly bases on an adapted IndiGolog temporal projector that is able to maintain a pool of hypotheses S (of configurable size) as well as a reference to the current favorite hypothesis $s_f \in S$. While the current steps in the IndiGolog cycle were only adapted for the changes to the temporal projector,

we added another step 1a (executed after step 1 of the main cycle) responsible for the control of our belief maintenance system.

The main stages of our new step 1a are

1. update all $s_i \in S : s_i \neq s_f$
2. check consistency of s_f
3. perform diagnosis if necessary
4. decide on a new s_f if necessary and proceed with stage 2 of the main cycle

The first stage is necessary as steps one and five of the standard cycle deal only with s_f and we have to adopt related changes for the alternative hypotheses in S as well.

In Stage 2 we verify s_f's consistency with reality. For this purpose background knowledge (common sense, domain knowledge) is used to derive new facts and conflicts from perception results while executing a plan π. Although results from perception can be handled by SSAs the same way as exogenous or endogenous actions, they do not change the environment. Thus every fluent update resulting from a sensing action has to be checked whether it masks an inconsistency. This can be done by assuring that a fluent is not restricted in such a fashion that would contradict the set of values proceeding the sensing action.

Stage 3 triggers a diagnosis process in the case that s_f is found to be inconsistent. In this case, we derive diagnoses satisfying the definition in Section 3.1. As already pointed out, a history is made up of endogenous, exogenous and sensing actions. Endogenous and sensing actions are performed actively, so it is a good guess for it or some variation to be part of the correct hypothesis, whereas exogenous actions can occur anytime and thus have to be considered with specific conditions.

We adapted history-based diagnosis according to Iwan [3] with these ideas in mind and created a predicate *Varia*(*OrigAct*, *Var*, *Cond*, *PV*) that is used to generate a valid variation *Var* of some action *OrigAct* with a preference value *PV* if condition *Cond* is met by situation s. The preference value is one of $\{1, 2\}$, where the higher the value the less likely this variation occurs. We use a value of 2 for incomplete initial situations (which we discuss later) and a value of 1 otherwise. The preference value of a derived sequence is defined as the sum of the individual PVs.

Derived action sequences have to be feasible, that is, it has to be possible to follow them step-by-step from a given initial state. As this implies that the initial assignment for all relevant fluents has to be known, our system has to be able to deal with incomplete initial belief. We cope with this issue by calculating also possibly executable hypotheses under the assumption of an incomplete S_0 and assign them higher preference values. As in general the number of derivable situations is very large, we apply pruning and abandon (partly) generated situation whenever they reach a predefined threshold.

In stage 4, we decide on a new s_f in case the actual one is found to be inconsistent. The preference value derived in stage 3 for each $s_i \in S$ is used for a ranking, where in a straight forward approach we choose that hypothesis with the highest rank. Unless there is no valid candidate for s_f, execution is continued with stage 2.

With our basic concept we provide an effective belief management system that is able to handle multiple hypotheses and several type of faults, and that can be easily integrated into IndiGolog. First experiments are discussed in the next section.

4 Experimental Analysis

We simulated and analyzed the performance of our belief management system for the scenario of an office delivery robot, a promising application area for robotics. The robot has to move objects between 59 rooms that are connected via 12 hallway segments. We simulate a robot that is able to pick up an object, move between adjacent rooms, and put the object currently carried down again. The robot is equipped with a vision sensor registering objects within the same room, and a pressure sensor indicating whether it currently carries an object. In order to resemble real robot systems more closely, continuous sensing was implemented as a new feature in IndiGolog.

The SSAs and action pre-conditions were implemented in a straightforward fashion. Our background model states basic rules, for example that the robot has to currently carry some object if the pressure sensor signals it. In order to prevent influences from planning, we use a simple IndiGolog program, that chooses the next action to execute out of a given sequence. Typical execution traces consist of about 40 steps.

We defined three fault scenarios: In **F1** the robot might fail to pick up an object with a probability of 40% and pick up the wrong object with 20%. In **F2** the robot additionally fails to release an object with a probability of 30%. In **F3** we add a probability of 5% that a sensing action fails. These faults resemble issues that frequently occur in reality. We define three sensing rates: In **S1** there is sensor data after every, in **S2** after every second, and in **S3** after every third action. For a single mission, the robot has to deliver three particular objects to three particular offices. A mission is accomplished if at the end all objects are at the desired locations. It is failed if the interpreter aborts the program, the execution exceeds a 2 minute time-out, or some object ends up misplaced.

A simple base line agent and another one equipped with our belief management system have been tested. They had to solve 50 different missions 10 times each with varying seeds for any combination of fault and sensing rates. The percentage of successful missions serves as performance measure.

The results of our experiments are reported in Figure 1. The robot using our belief management system significantly outperforms the robot without this extension. From the figure, it is also evident that while with rising difficulty (more fault types) the performance of both robots declines, belief management helps to keep it at a decent level. A declining sensing rate has a negative impact on performance too, based on the fact that the robot has less information to detect and repair inconsistent situations.

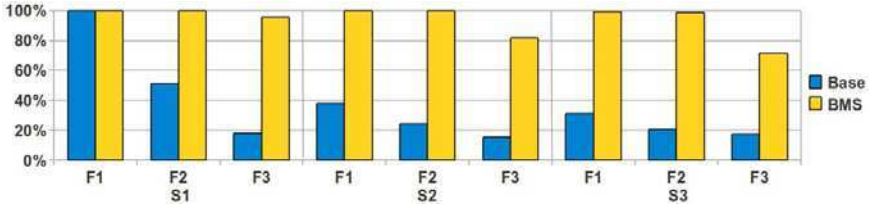


Fig. 1. Successfully completed missions with (BMS) and without belief management (Base).

5 Conclusions and Future Work

In this paper we presented a belief management system that seamlessly integrates into the agent programming language IndiGolog. Our basic concept was to integrate history-based diagnosis into the interpreter in order to allow it to repair inconsistent situations and maintain several situation hypotheses. First experiments with an office delivery robot scenario show that a robot equipped with our system is able to handle significantly more erroneous situations successfully than a simple robot. In contrast to classical diagnosis approaches, the situation calculus enables us to deal also with faults that occurred in the past, rather than just faults in the current situation.

Future work will include formally proving the correctness of our approach, as well as a setup for our real robots. Moreover, we will investigate the run-time properties of our approach that are important for a broad applicability of our idea. Finally, we will try to move the focus from the described passive behavior to a more active management of diagnoses such as hypothesis discrimination via actively performed actions.

References

1. Thrun, S., Bennewitz, M., Burgard, W., Cremers, A.B., Dellaert, F., Fox, D., Hähnel, D., Rosenberg, C., Roy, N., Schulte, J., Schulz, D.: Minerva: A second-generation museum tour-guide robot. In: IEEE International Conference on Robotics and Automation (1999)
2. Iagnemma, K., Buehler, M.: Special Issue on the DARPA Grand Challenge. Journal of Field Robotics 23(8-9) (2006)
3. Iwan, G.: History-based diagnosis templates in the framework of the situation calculus. AI Communications 15(1), 31–45 (2002)
4. Goel, P., Dedeoglu, G., Roumeliotis, S.I., Sukhatme, G.S.: Fault detection and identification in a mobile robot using multiple model estimation and neural network. In: IEEE International Conference on Robotics and Automation (2000)
5. Verma, V., Gordon, G., Simmons, R., Thrun, S.: Real-time fault diagnosis. IEEE Robotics & Automation Magazine 11(2), 56–66 (2004)
6. Reiter, R.: A theory of diagnosis from first principles. Art. Intelligence 32(1), 57–95 (1987)
7. McIlraith, S.: Explanatory diagnosis: Conjecturing actions to explain observations. In: International Workshop on Principles of Diagnosis (1997)
8. Kuhn, L., Price, B., de Kleer, J., Do, M., Zhou, R.: Pervasive diagnosis: the integration of active diagnosis into production plans. In: Int. Workshop on Principles of Diagnosis (2008)
9. Fritz, C.: Monitoring the Generation and Execution of Optimal Plans. PhD thesis, University of Toronto (April 2009)
10. Ferrein, A., Fritz, C., Lakemeyer, G.: On-line Decision-Theoretic Golog for Unpredictable Domains. In: International Cognitive Robotics Workshop (2004)
11. Fichtner, M., Großmann, A., Thielscher, M.: Intelligent execution monitoring in dynamic environments. Fundamenta Informaticae 57(2-4), 371–392 (2003)
12. Weld, D.S., Anderson, C.R., Smith, D.E.: Extending graphplan to handle uncertainty and sensing actions. In: Fifteenth National/Tenth Conference on Artificial iItelligence/Innovative Applications of Artificial Intelligence, pp. 897–904 (1998)
13. De Giacomo, G., Lespérance, Y., Levesque, H.J., Sardina, S.: IndiGolog: A High-Level Programming Language for Embedded Reasoning Agents. In: Multi-Agent Programming: Languages, Tools and Applications, pp. 31–72. Springer, Heidelberg (2009)
14. Reiter, R.: Knowledge in Action. Logical Foundations for Specifying and Implementing Dynamical Systems. MIT Press, Cambridge (2001)

Enhancing Believability of Virtual Soccer Players: Application of a BDI-Model with Emotions and Trust

Tibor Bosse and Daniel Höhle

Vrije Universiteit Amsterdam, Department of Artificial Intelligence
de Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
tbosse@few.vu.nl, daniel.hohle@gmail.com

Abstract. Despite significant progress in the development of virtual soccer games, the affective behavior of virtual soccer players, both in software and hardware applications, is still limited. To improve this situation, this paper presents a generic model for decision making of virtual agents in relation to emotions and trust. The model takes the BDI framework as point of departure, and extends this with mechanisms to represent the dynamics of emotions and trust. After testing the model by means of simulation experiments, it has been incorporated into the virtual agents within the RoboCup 2D soccer environment. A preliminary evaluation pointed out that the model successfully enhances believability of virtual soccer players.

Keywords: soccer simulation, cognitive modeling, affective computing, trust.

1 Introduction

Association football, commonly known as soccer, has been the world's most popular sport for many decades. Thanks to the rapid developments in the video game industry as well as scientific areas such as Computer Science and Artificial Intelligence, nowadays, soccer games are not only played in the real world, but also in the virtual world. Overall, the purpose of virtual soccer games (or soccer simulations) is three-fold. First and foremost, virtual soccer games are played for entertainment purposes; in these cases, one or multiple human users control the soccer players in a virtual environment, with the goal to defeat each other or a computer opponent [12]. Second, soccer simulations are increasingly being used for the purpose of analysis, for instance by managers and coaches of real soccer teams; here, the idea is that the virtual environment provides an easy framework to test various strategies that cannot easily be tested in the real world. For example, in case a coach intends to investigate what might happen if his team uses a particular tactical strategy in a particular game, he could run a simulation of this game, where each of the simulated players behaves according to this strategy. Although this area is yet to be explored in much more detail, some anecdotical cases have been reported in which soccer simulations were able to make accurate predictions (e.g., [15, 17]). Third, virtual soccer games have proved to be an excellent test bed for the application of a variety of AI techniques, including (real-time) multi-agent planning, multi-agent communication, behavior modeling, and learning. These challenges have been the main motivation for the foundation of the RoboCup competition in 1997 [10].

K.G. Mehrotra et al. (Eds.): Developing Concepts in Applied Intelligence, SCI 363, pp. 119–128.
springerlink.com

As a result of these developments, realism of virtual soccer games has increased very rapidly over the past years. Not only has the physical appearance of players and environment become more and more appealing, also the technical movements as well as the tactical decisions of the players have become much closer to reality. In contrast, one particular aspect that has stayed a bit behind is the players' mental behavior, and in particular their affective behavior. For instance, in state-of-the-art video games, the only moments in which players' emotions are really apparent are right after a goal has been scored. In such cases, players usually show some built-in emotional expressions (e.g., cheering, or looking angry). However, complex emotional states that change dynamically during the game and influence the players' behavior are usually very limited. This is in conflict with the behavior of human soccer players, which has shown extremely sensitive to the dynamics of their affective states [7]. Consequently, existing soccer simulations are not as realistic (and appealing) as they could be.

To deal with this problem, the current paper presents a generic model for decision making in relation to emotions and trust, and applies this to the domain of virtual soccer. As also put forward in [16], endowing virtual soccer players with emotions will enhance their believability, thereby making the game more engaging for spectators. Both emotions and trust are mental states which have been found to have a serious impact on people's decisions, in general as well as in sports context [7, 9]. The presented model takes the BDI framework [13] as point of departure, and extends this with mechanisms to represent the dynamics of emotions and trust.

The remainder of this paper is structured as follows. The generic simulation model is presented at a conceptual level in Section 2. In Section 3, this model is illustrated for the context of soccer games, by showing a number of simulations generated in the LEADSTO simulation environment. Next, Section 4 describes how the model was incorporated into the virtual agents within the RoboCup 2D soccer environment, and presents some preliminary results. Section 5 concludes the paper with a discussion.

2 Conceptual Model

In this section, the model for decision making with emotions and trust will be described at an intuitive, conceptual level, using the agent-based modeling language LEADSTO [2]. This language allows the modeler to integrate both logical (qualitative) and numerical (quantitative) aspects. In LEADSTO, direct temporal dependencies between two state properties in successive states are modeled by executable dynamic properties. The format is defined as follows: let α and β be state properties of the form 'conjunction of ground atoms or negations of ground atoms'. In LEADSTO the notation $\alpha \rightarrow\!\!\!\!\rightarrow_{e, f, g, h} \beta$ means:

If state property α holds for a certain time interval with duration g,
then after some delay (between e and f), state property β will hold for a certain time interval of length h.

Here atomic state properties can have a qualitative, logical format, such as an expression desire(d), expressing that desire d occurs, or a quantitative, numerical format such as an expression has_value(x, v) which expresses that variable x has value v. For more details, see [2].

Below, the LEADSTO model for decision making with emotions and trust is introduced (at a conceptual level) in three steps. In Section 2.1, the basic Belief-Desire-Intention (BDI) model is briefly introduced. Next, in Section 2.2 and 2.3, this model is extended with mechanisms to represent the dynamics of emotions and trust, respectively. For the complete LEADSTO specification, see the M.Sc. Thesis in [6].

2.1 BDI-Model

The BDI-model is a standard paradigm within the Agent Technology domain, which describes how agents decide to perform actions based on their beliefs, desires and intentions (e.g., [13]). This model forms the foundation of the model presented in this paper. See the middle part of the global overview in Figure 1. Here the box indicates the borders of the agent, the ovals denote components of the agent model representing different mental states, and the arrows indicate influences between mental states.

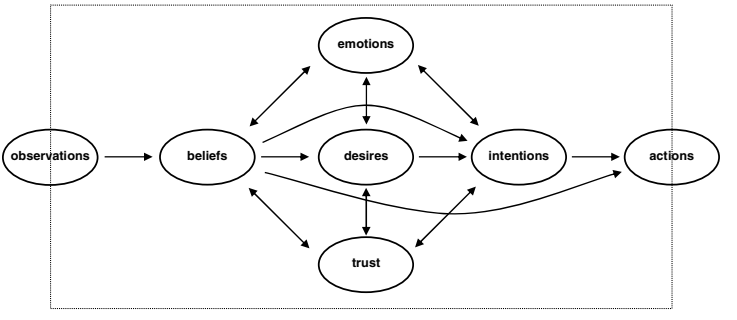


Fig. 1. Global overview of the BDI-model with Emotions and Trust.

For instance, an *action* A is executed when the agent has the *intention* to perform this action and it has the *belief* that certain circumstances in the world are fulfilled such that there is an opportunity to perform the action. Beliefs are created on the basis of observations. Moreover, the intention to do a specific type of action is created if there is some *desire* to reach state S, and the agent believes that performing this action will fulfill this desire. Such relations within the general BDI-model can be specified in formal LEADSTO format as follows (where the timing parameters e, f, g, h, as well as a parameter Ag indicating which agent has the mental states, have been omitted for simplicity):

$\forall$S:STATE $\forall$A:ACTION
desire(S) $\wedge$ belief(satisfies(A, S)) $\twoheadrightarrow$ intention(A)

$\forall$A:ACTION
intention(A) $\wedge$ belief(opportunity_for(A)) $\twoheadrightarrow$ performed(A)

Note that the beliefs used here depend on observations (see Figure 1), or on common knowledge. In the remainder of this paper, desires and intentions are parameterized with a real number I (between 0 and 1) to represent the strength of the mental state.

2.2 Emotions

In order to represent emotional states, a main assumption made is that the intensity of different emotions can be represented via a number between 0 and 1, as often done within emotion models within Artificial Intelligence (e.g., [5]). In the current paper, only one type of emotion is modeled, namely *happiness* (with 0 = very sad and 1 = very happy), but the model can also be used to represent emotions like anger and fear.

As shown in the upper part of Figure 1, the component to represent emotions interacts with the basic BDI-model in various manners. First, for generation of emotions, it is assumed that they are the result of the evaluation of particular events against the individual's own goals (or desires), as described by *appraisal theory* [5]. This process is formalized, among others, by the following LEADSTO rules (stating that reaching a goal state S leads to happiness with an intensity that is proportional to the strength of the desire, and inversely for not reaching a goal state):

$\forall$S:STATE $\forall$I:REAL
desire(S, I) $\wedge$ belief(S) $\rightarrow\!\!\!\!\rightarrow$ happiness(S, I)
desire(S, I) $\wedge$ belief(not(S)) $\rightarrow\!\!\!\!\rightarrow$ happiness(S, 1-I)

For example, in case an agent desires to score a goal, but misses, then the agent becomes sad about this. In addition to these event-specific emotional states, the agents' long term *mood* is modeled. Moods are usually distinguished from emotions in that they last longer and are less triggered by particular stimuli. Thus, in the model a global mood is assumed with a value between 0 and 1 (representing its positive valence), which depends on the values of all specific emotions in the following way:

$\forall$S1, ..., Sn:STATE $\forall$I1, ..., In, J:REAL
happiness(S1, I1) $\wedge$ ... $\wedge$ happiness(Sn, In) $\wedge$ mood(J) $\rightarrow\!\!\!\!\rightarrow$
mood(β * J + (1-β) * (w1*I1 + ... + wn*In))

Thus, the new mood state is calculated based on the valence J of the old mood state, combined with the weighted sum of the emotional values Ik for the different aspects Sk in the domain of application. Here, the wk's (which sum up to 1) are weight factors representing the relative importance of the different aspects, and the β (between 0 and 1) is a persistence factor representing the rate at which mood changes over time.

Finally, various rules have been formulated to represent the impact of emotions and mood on beliefs, desires, and intentions. These rules are mostly domain-specific; for instance, a positive mood increases the desire to cooperate with teammates, and a negative mood increases the desire to behave aggressively. These rules are described in detail in [6].

2.3 Trust

According to [4], trust is a (dynamic) mental state representing some kind of expectation that an agent may have with respect to the behavior of another agent or entity. The authors propose that a *belief in competence* of the other entity is an important contributor to the development of trust (in addition to a second aspect, the belief in willingness; however this aspect is currently ignored). This mechanism is formalized in the current paper by assuming that trust in another agent is based on a weighted sum of the beliefs in the agent's capabilities (using a similar formula as above for mood update):

∀A:AGENT ∀X1, ..., Xn:ACTION ∀I1, ..., In, J:REAL
belief(has_capability(A, X1, I1)) ∧ ... ∧ belief(has_capability(A, Xn, In)) ∧ trust(A, J) →→
trust(A, β * J + (1-β) * (w1*I1 + w2*I2 + ... + wn*In))

Thus, the new trust state in agent A is calculated based on the level J of the old trust state, combined with the weighted sum of the beliefs in capabilities Ik for the different actions Xk in the domain of application. For example, a soccer player trusts his teammate more if he believes that he is good at attacking as well as defending. Again, the wk's are weight factors, and the β is a persistence factor.

A next step is to define how beliefs in capabilities are determined. For this, the mechanism put forward in [8] is reused, which states that a new trust state in some entity is partly based on the old trust state, and partly on an experience. This is modeled via the following LEADSTO rule (where the experiences are observed actions performed by teammates):

∀A:AGENT ∀X:ACTION ∀I:REAL
belief(has_capability(A, X, I)) ∧ observed(performed(A, X, succeeded)) →→
belief(has_capability(A, X, 1 - γ + γ*I))
belief(has_capability(A, X, I)) ∧ observed(performed(A, X, failed)) →→
belief(has_capability(A, X, γ*I))

For instance, in case agent X believes that agent Y's capability with respect to tackling is 0.6, and Y performs a successful tackle, then this belief is strengthened (where γ is an update speed factor). Note that the mechanism to update trust is also applied to the self, to model some kind of self-confidence.

Finally, as with emotions, trust states also have an impact on the other states in the BDI-model. For example, a high trust in a teammate increases the strength of the intention to pass the ball to this player. Again, these mechanisms are represented using (mostly domain-specific) rules, which can be found in [6].

3 Simulation Results

To test the basic mechanisms of the model, it has been used to generate a number of simulation runs within the LEADSTO simulation environment. To this end, various scenarios have been established in the context of a (simplified) soccer game. The game has been simplified in the sense that we did not simulate a complete soccer match (as is the case in the RoboCup environment), including computer-generated teammates and opponents, and environmental processes (e.g., movements of the ball). Instead, the tests focused on the behavior of one particular agent (player X). To test this behavior, the scenarios consisted of a series of predefined events (e.g., 'teammate Y passes the ball to teammate Z', 'teammate Z shoots on target'), which were provided as input to player X. Based on these inputs, all actions and emotions derived by the agent were observed and evaluated, and in case of inappropriate behavior, the model was improved (manually). Since the main goal of the simulations was to test the model for decision making in relation to emotions and trust, this simplified setup was considered a necessary first step. As a next step (see Section 4), the model was tested in a more complete setting (the RoboCup environment).

To represent the domain-specific aspects of virtual soccer, the different logical *sorts* introduced in Section 2 were filled with a number of elements. For example,

some instances of the sort STATE were {win_game, ball_in_possession, ball_nearby_goal, score_other_goal}, some instances of the sort BELIEF were {is_at_location(AGENT, REAL), has_capability(AGENT, ACTION, REAL), satisfies(ACTION, STATE), opportunity_for(ACTION)}, and some instances of the sort ACTION were {shoot_at_goal, pass_from_to(AGENT, AGENT), dribble, run_free}.

To illustrate the behavior of the model, Figures 2 and 3 show some fragments of an example LEADSTO simulation trace, addressing a simple scenario in which a number of subsequent (positive and negative) events occur. In both figures, time is on the horizontal axis. Figure 2 shows a number of actions that are performed during the scenario by player Z, a teammate of player X. A dark box on top of a line indicates that an action is being performed at that time point. Figure 3 shows the emotional state (in this case the amount of happiness, on a [0,1] scale) that player X has with respect to player Z.

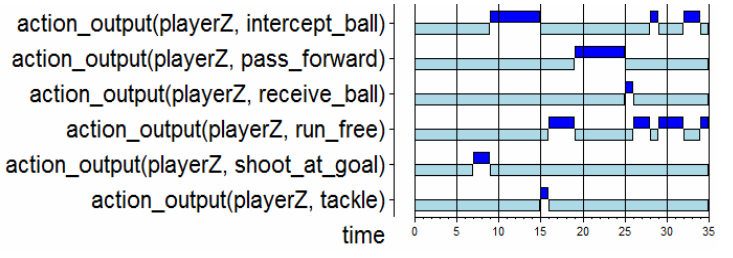


Fig. 2. Example simulation results - Actions performed by player Z.

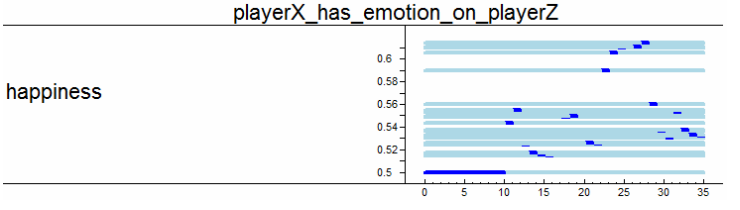


Fig. 3. Example simulation results - Emotional state of player X with respect to player Z.

As can be seen from the figures, player X's emotional state regarding player Z changes as a consequence of player Z's actions. Note that it takes about 3 time points in the simulations to change an emotional state, which explains the short delays. For instance, at time point 7-9, player Z shoots at the goal. Apparently, this attempt is appreciated by player X (i.e., it contributes to fulfillment of this agent's desires), since at time point 10-12, the agent's happiness regarding agent Z increases from 0.5 to 0.55. Similarly, player Z's attempts to pass the ball forward (time point 19-25) also lead to an increase of agent X's happiness. Note that not all of player Z's actions lead to an increase of happiness. There are two potential causes for this: either the action fails (not shown in the figure), or the action does not contribute to player X's desires.

In addition to the scenario shown above, various other scenarios have been simulated in a systematic manner (varying from scenarios in which only positive or negative events occur to scenarios in which both positive and negative events occur). Due to space limitations, the details of these simulations are not shown here. To improve the model's accuracy, each simulation run has been evaluated in the following manner. Based on the literature in decision making, emotion, and trust (see, e.g., [4, 5] and references in those papers), a number of requirements for the behavior of the agents have been formulated; some examples are the following:

- 'events that contribute to an agent's goals lead to an increase of happiness'
- 'agents that perform more successful actions are trusted more'
- 'happy agents perform different actions than sad agents'

These requirements have been tested against all simulated traces. In case a requirement was not fulfilled, small modifications in the model were made, either by correcting bugs in the specification or (more often) by adapting the values of the parameters involved. After a number of iterations, the model was considered ready to be implemented in a more realistic setting.

4 RoboCup Application

After the initial tests mentioned above pointed out that the model showed acceptable behavior, it has been implemented within the RoboCup 2D virtual soccer environment. RoboCup is an international competition founded in 1997 with the aim to develop autonomous (hardware and software) agents for various domains, with the intention to promote research and education in Artificial Intelligence [10]. RoboCup has four competition domains (Soccer, Rescue, Home, and Junior), each with a number of leagues. For the current project, the RoboCup Soccer 2D Simulation League was used, since this league provides good possibilities for rapid prototyping for games with many inter-player interactions.

In order to implement the model for decision making with emotions and trust in an efficient manner, part of the existing code of the 2005 Brainstormers team [14] was reused. This code contains a number of explicit classes for different actions (such as 'shoot_at_goal', 'pass_from_to', 'dribble', 'tackle', and 'run_free'). To create optimal strategies at the level of the team, the Brainstormers used Machine Learning techniques to learn which actions were most appropriate in which situations. Instead, for the current paper, a different approach was taken: for each individual agent, the original action classes were directly connected to the output (the generated actions) of the implemented decision making model. Thus, instead of having a team of agents that act according to a learned collective strategy, each agent in our implementation acts according to its personal role, (partly emotion-driven) desires and intentions[1].

This approach turned out to be successful in generating (intuitively) realistic soccer matches, in which the behavior of the players is influenced by their emotions and trust states. A screenshot of the application is shown in Figure 4. The top of the figure

[1] Recall that the aim of the current paper is *not* to develop optimal strategies, but rather to enhance believability of the virtual players.

shows the soccer field, including two teams of 11 agents (the colored circles) and the ball (the white circle), which is held by the goalkeeper on the right. On top of each agent, a small number is displayed, which indicates the player's personal number. Below, some information is displayed in plain text, to give some insight in the mental states of the agents. Obviously, there is no space to display all aspects of the agents' mental states (emotions, moods, trust states, beliefs, desires, and intentions). Therefore the user can select which information is shown. In the example shown in Figure 4, five columns of information are shown, indicating, respectively, the current time, and each agent's number, mood, emotion regarding itself, and current action.

Fig. 4. Screenshot of the RoboCup application.

An example animation illustrating the behavior of the application can be found at http://www.youtube.com/watch?v=I-ZamC-louo. This movie shows a scenario where player 6 passes the ball to player 10, who then dribbles with it towards the goal, and scores. As a result of these successful actions, various aspects of player 10's mental state change (see the textual information below the playing field): its emotion with respect to itself increases (from 0.366 to 0.686), as well as its overall mood (from 0.398 to 0.557), its trust in itself (from 0.522 to 0.552), and its belief about its own capability to shoot at the goal (from 0.509 to 0.602). As a result, in the rest of the scenario the player will behave more confidently, and will try to score more often.

The application has been tested extensively, using a large amount of different parameter settings (e.g. for initial trust, emotion and mood states, trust and emotion flexibility, and action preferences, see also [6]). In all tests, the behavior of all players was analyzed in a step-wise manner, and compared with expectations of the modelers (based on common sense). Various interesting types of behavior were observed,

which were found to be consistent with behavior as observed in real soccer games. For instance, players that had scored a number of goals were trusted more by their teammates, and received more passes from them. Also, players with negative emotions committed more fouls than other agents.

To further evaluate the application, 10 participants (experienced video gamers between 25 and 32 years old) were asked to judge the behavior of the virtual players, by answering questions like “do you think the players behave in a realistic manner?”, “do you think the players show appropriate emotions?”. The initial results of this evaluation were promising: the participants very much appreciated the players’ abilities to show emotions and trust. Overall, they had the idea that the presented model made the agents more believable, and that it enhanced the experienced fun when watching the soccer games. Nevertheless, in a later stage, a more elaborated evaluation experiment will be performed, in cooperation with colleagues from Social Sciences. In this experiment, we plan to compare virtual players that use different variants of the presented model with players that do not have emotions and trust.

5 Discussion

To enhance believability of virtual soccer players’ affective behavior, this paper presented a generic model for decision making of virtual agents in relation to emotions and trust. The backbone of the presented model is the BDI framework, which was extended with mechanisms to represent the dynamics of emotions and trust. After testing the model by means of simulation experiments, it has been incorporated into the virtual agents within the RoboCup 2D soccer environment. Although preliminary, an initial evaluation pointed out that the model has the potential to enhance believability of virtual soccer players.

Related work in the area of believable virtual soccer players is scarce. Since the initiation of the RoboCup competition in 1997, a number of participants focused on the cognitive aspects of the players (e.g., [3, 11]), but emotional aspects were mostly ignored. A welcome exception is presented by Willis [16], who proposes an architecture to endow the robots in the physical league with emotional intelligence. Unlike the model put forward in the current paper, this architecture has not been implemented yet, nor does it address the concept of trust. Finally, some papers (e.g., [1]) address implementation of emotions within the spectators and commentators of virtual soccer games; however, to the best of our knowledge, these approaches have never been applied to the player agents.

As mentioned in the introduction, endowing virtual soccer players with more human-like behavior can be useful for several reasons (see [16] for an extensive overview): for instance, the games become 1) more fun to watch and 2) more faithful to reality, which makes it possible to use them as an analytical tool for coaches. A third reason put forward in [16] is that soccer teams with emotional intelligence may have a competitive advantage over other teams. Although this was not the main focus of the current paper, future research may investigate how well our agents perform against similar agents without emotions and trust.

Other future work will include a further elaboration of the model, among others, by focusing on other emotions such as anger and fear, and the interaction between

multiple emotions. Also, possibilities will be explored to visualize players' emotional states in such a way that they are easily observed by spectators. For instance, a straightforward way to implement this would be using different colors to represent different emotions. It is expected that these extensions will enhance spectators' engagement in virtual soccer games even further.

References

1. Binsted, K., Luke, S.: Character design for soccer community. In: Asada, M., Kitano, H. (eds.) RoboCup 1998. LNCS (LNAI), vol. 1604, pp. 22–33. Springer, Heidelberg (1999)
2. Bosse, T., Jonker, C.M., van der Meij, L., Treur, J.: A Language and Environment for Analysis of Dynamics by SimulaTiOn. International Journal of AI Tools 16(3), 435–464 (2007)
3. da Costa, A.C.P.L., Bittencourt, G.: UFSC-team: A cognitive multi-agent approach to the roboCup 1998 simulator league. In: Asada, M., Kitano, H. (eds.) RoboCup 1998. LNCS (LNAI), vol. 1604, pp. 371–376. Springer, Heidelberg (1999)
4. Falcone, R., Castelfranchi, C.: Trust dynamics: How Trust is Influenced by Direct Experiences and by Trust Itself. In: Proc. of AAMAS 2004, pp. 740–747 (2004)
5. Gratch, J., Marsella, S.: A domain independent framework for modeling emotion. Journal of Cognitive Systems Research 5, 269–306 (2004)
6. Höhle, D.: A General Agent Model of Emotion and Trust using the BDI Structure. M.Sc. Thesis, Vrije Universiteit Amsterdam (2010), http://hdl.handle.net/1871/16276
7. Jones, M.V.: Controlling Emotions in Sport. In: The Sport Psychologist, vol. 17, pp. 471–486. Human Kinetics Publishers (2003)
8. Jonker, C.M., Treur, J.: Formal Analysis of Models for the Dynamics of Trust based on Experiences. In: Garijo, F.J., Boman, M. (eds.) MAAMAW 1999. LNCS, vol. 1647, pp. 221–232. Springer, Heidelberg (1999)
9. Jowett, S., Lavallee, D. (eds.): Social Psychology in Sport. Human Kinetics, Champaign (2007)
10. Kitano, H., Asada, M., Noda, I., Matsubara, H.: RoboCup: robot world cup. IEEE Robotics and Automation Magazine 5(3), 30–36 (1998)
11. Muñoz-Hernández, S., Sari Wiguna, W.: Fuzzy cognitive layer in roboCupSoccer. In: Melin, P., Castillo, O., Aguilar, L.T., Kacprzyk, J., Pedrycz, W. (eds.) IFSA 2007. LNCS (LNAI), vol. 4529, pp. 635–645. Springer, Heidelberg (2007)
12. Poole, S.: Trigger Happy: Videogames and the entertainment revolution. Arcade Publishing, New York (2000)
13. Rao, A.S., Georgeff, M.P.: Modelling Rational Agents within a BDIarchitecture. In: Allen, J., et al. (eds.) 2nd International Conference on Principles of Knowledge Representation and Reasoning, KR 1991, pp. 473–484. Morgan Kaufmann, San Francisco (1991)
14. Riedmiller, M., Gabel, T., Knabe, J., Strasdat, H.: Brainstormers 2D – Team Description 2005. In: Bredenfeld, A., et al. (eds.) RoboCup 2005. LNCS (LNAI), vol. 4020, pp. 219–229. Springer, Heidelberg (2006)
15. Silva, C.F., Garcia, E.S., Saliby, E.: Soccer championship analysis using Monte Carlo simulation. In: Yücesan, E., Chen, C.-H., Snowdon, J.L., Charnes, J.M. (eds.) Proceedings of the 2002 Winter Simulation Conference, vol. 2, pp. 2011–2016 (2002)
16. Willis, M.: RoboCup as a Spectator Sport: Simulating Emotional Response in the Four-Legged League. In: Proceedings of the 5th Australasian Conference on Interactive Entertainment, vol. 391. ACM Press, New York (2008)
17. http://www.ea.com/uk/news/ea-sports-predicts-spain-win-2010-world-cup

Author Index