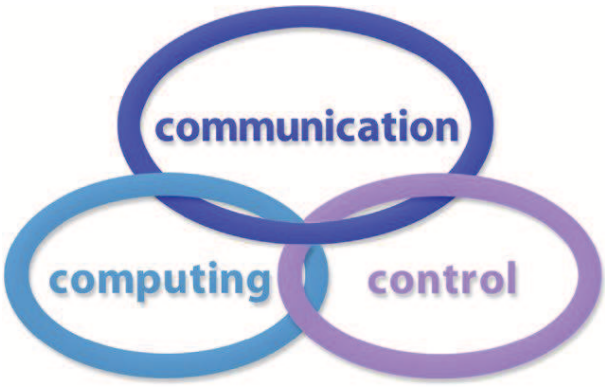
INTERNATIONAL JOURNAL

of

COMPUTERS COMMUNICATIONS & CONTROL

ISSN 1841-9836



A Bimonthly Journal
With Emphasis on the Integration of Three Technologies

Year: 2017 Volume: 12 Issue: 3 Month: June

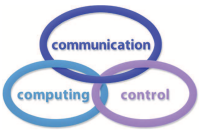
This journal is a member of, and subscribes to the principles of, the Committee on Publication Ethics (COPE).

http://univagora.ro/jour/index.php/ijccc/

CCC Publications

BRIEF DESCRIPTION OF JOURNAL

Publication Name: International Journal of Computers Communications & Control.
Acronym: IJCCC; **Starting year of IJCCC:** 2006.
ISO: Int. J. Comput. Commun. Control; **JCR Abbrev:** INT J COMPUT COMMUN.
International Standard Serial Number: ISSN 1841-9836.
Publisher: CCC Publications - Agora University of Oradea.
Publication frequency: Bimonthly: Issue 1 (February); Issue 2 (April); Issue 3 (June); Issue 4 (August); Issue 5 (October); Issue 6 (December).
Founders of IJCCC: Ioan DZITAC, Florin Gheorghe FILIP and Misu-Jan MANOLESCU.
Logo:



Indexing/Coverage:

- Since 2006, Vol. 1 (S), IJCCC is covered by Thomson Reuters and is indexed in ISI Web of Science/Knowledge: Science Citation Index Expanded.

  2016 Journal Citation Reports® Science Edition (Thomson Reuters, 2016):
  Subject Category: (1) Automation & Control Systems: Q4(2009,2011,2012,2013,2014,2015), Q3(2010); (2) Computer Science, Information Systems: Q4(2009,2010,2011,2012,2015), Q3(2013,2014).

  Impact Factor/3 years in JCR: 0.373(2009), 0.650 (2010), 0.438(2011); 0.441(2012), 0.694(2013), 0.746(2014), 0.627(2015).
  Impact Factor/5 years in JCR: 0.436(2012), 0.622(2013), 0.739(2014), 0.635(2015).

- Since 2008 IJCCC is indexed by Scopus (SNIP2014= 1.029):
  Subject Category: (1) Computational Theory and Mathematics: Q4(2009,2010,2012,2015), Q3(2011,2013,2014); (2) Computer Networks and Communications: Q4(2009), Q3(2010, 2012, 2013, 2015), Q2(2011, 2014); (3) Computer Science Applications: Q4(2009), Q3(2010, 2011, 2012, 2013, 2014, 2015).

  SJR: 0.178(2009), 0.339(2010), 0.369(2011), 0.292(2012), 0.378(2013), 0.420(2014), 0.319(2015).

- Since 2007, 2(1), IJCCC is indexed in EBSCO.

Focus & Scope: International Journal of Computers Communications & Control is directed to the international communities of scientific researchers in computer and control from the universities, research units and industry.

To differentiate from other similar journals, the editorial policy of IJCCC encourages the submission of original scientific papers that focus on the integration of the 3 "C" (Computing, Communication, Control).

In particular the following topics are expected to be addressed by authors: (1) Integrated solutions in computer-based control and communications; (2) Computational intelligence methods (with particular emphasis on fuzzy logic-based methods, ANN, evolutionary computing, collective/swarm intelligence); (3) Advanced decision support systems (with particular emphasis on the usage of combined solvers and/or web technologies).

Editorial Address:
Agora University/ R&D Agora Ltd. / S.C. Cercetare Dezvoltare Agora S.R.L.
Piata Tineretului 8, Oradea, jud. Bihor, Romania, Zip Code 410526
Tel./ Fax: +40 359101032
E-mail: ijccc@univagora.ro, rd.agora@univagora.ro, ccc.journal@gmail.com
Journal website: http://univagora.ro/jour/index.php/ijccc/

IJCCC EDITORIAL BOARD MEMBERS

Gheorghe Păun
Institute of Math. of Romanian Academy
Bucharest, PO Box 1-764, Romania
gpaun@us.es

Mario de J. Pérez Jiménez
Dept. of CS and Artificial Intelligence
University of Seville, Sevilla,
Avda. Reina Mercedes s/n, 41012, Spain
marper@us.es

Dana Petcu
Computer Science Department
Western University of Timisoara
V.Parvan 4, 300223 Timisoara, Romania
petcu@info.uvt.ro

Radu Popescu-Zeletin
Fraunhofer Institute for Open
Communication Systems
Technical University Berlin, Germany
rpz@cs.tu-berlin.de

Imre J. Rudas
Óbuda University
Budapest, Hungary
rudas@bmf.hu

Yong Shi
School of Management
Chinese Academy of Sciences
Beijing 100190, China &
University of Nebraska at Omaha
Omaha, NE 68182, U.S.A.
yshi@gucas.ac.cn, yshi@unomaha.edu

Athanasios D. Styliadis
University of Kavala
Institute of Technology
65404 Kavala, Greece
styliadis@teikav.edu.gr

Gheorghe Tecuci
Learning Agents Center
George Mason University
U.S.A.
University Drive 4440, Fairfax VA
tecuci@gmu.edu

Horia-Nicolai Teodorescu
Faculty of Electronics and
Telecommunications
Technical University "Gh. Asachi" Iasi
Iasi, Bd. Carol I 11, 700506, Romania
hteodor@etc.tuiasi.ro

Dan Tufiş
Research Institute for Artificial Intelligence
of the Romanian Academy
Bucharest, "13 Septembrie" 13, 050711, Romania
tufis@racai.ro

Lotfi A. Zadeh
Director,
Berkeley Initiative in Soft Computing (BISC)
Computer Science Division
University of California Berkeley,
Berkeley, CA 94720-1776
U.S.A.
zadeh@eecs.berkeley.edu

DATA FOR SUBSCRIBERS
Supplier: Cercetare Dezvoltare Agora Srl (Research & Development Agora Ltd.)
Fiscal code: 24747462
Headquarter: Oradea, Piata Tineretului Nr.8, Bihor, Romania, Zip code 410526
Bank: BANCA COMERCIALA FEROVIARA S.A. ORADEA
Bank address: P-ta Unirii Nr. 8, Oradea, Bihor, România
IBAN Account for EURO: RO50BFER248000014038EU01
SWIFT CODE (eq.BIC): BFER

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, Volume 12, Issue 3, June, 2017.

Contents

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 307-322, June 2017.

A Rating-based Integrated Recommendation Framework with Improved Collaborative Filtering Approaches

S. Cheng, B. Zhang, G. Zou

Shulin Cheng
1. School of Computer Engineering and Science, Shanghai University
99 Shangda Road, BaoShan District, Shanghai, 200444, PR, China
chengshulin@shu.edu.cn
2. School of Computer and Information, Anqing Normal University
1318 Jixian North Road, Anqing, Anhui Province, 246133, PR, China
chengshL@aqnu.edu.cn

Bofeng Zhang*, Guobing Zou
School of Computer Engineering and Science, Shanghai University
99 Shangda Road, BaoShan District, Shanghai, 200444, PR, China
*Corresponding author: bfzhang@shu.edu.cn
guobingzou@gmail.com

Abstract: Collaborative filtering (CF) approach is successfully applied in the rating prediction of personal recommendation. But individual information source is leveraged in many of them, i.e., the information derived from single perspective is used in the user-item matrix for recommendation, such as user-based CF method mainly utilizing the information of user view, item-based CF method mainly exploiting the information of item view. In this paper, in order to take full advantage of multiple information sources embedded in user-item rating matrix, we proposed a rating-based integrated recommendation framework of CF approaches to improve the rating prediction accuracy. Firstly, as for the sparsity of the conventional item-based CF method, we improved it by fusing the inner similarity and outer similarity based on the local sparsity factor. Meanwhile, we also proposed the improved user-based CF method in line with the user-item-interest model (UIIM) by preliminary rating. Second, we put forward a background method called user-item-based improved CF (UIBCF-I), which utilizes the information source of both similar items and similar users, to smooth item-based and user-based CF methods. Lastly, we leveraged the three information sources and fused their corresponding ratings into an Integrated CF model (INTE-CF). Experiments demonstrate that the proposed rating-based INTE-CF indeed improves the prediction accuracy and has strong robustness and low sensitivity to sparsity of dataset by comparisons to other mainstream CF approaches.
Keywords: personalized recommendation, collaborative filtering, rating integration.

1 Introduction

The recommender system [21] has been studied by many researchers in the past decade, which are widely applied in many fields like information retrieval [5], item recommendation [22], E-commerce [13]. Recommender system obtained relatively promising results and facilitated users, in which collaborative Filtering (CF) recommendation methods are classic and useful ones, and do well in recommending items with ratings such as products, movies, music. CF recommendation approaches [21] [19] [23] can be divided into memory-based approaches and model-based approaches. Memory-based approaches are heuristic and comprises item-based and user-based approaches, while model-based approaches are built based on machine learning theory. Item-based and user-based approaches both leverage the idea of neighbors to generate recommendation by measuring the similarities between the target item and other items, or, between

the target user and other users. And the similarities are viewed as weights between items or users in the process of rating prediction.

But many of item-based and user-based approaches predict unknown ratings from single perspective of either users or items, in which only partial information embedded in user-item matrix is utilized. Traditional approach [22] [19] with single view has relatively low performance due to the poor ability against the sparsity of user-item matrix except few ones like recommendation in Amazon [13]. Naturally, some researchers studied the imputation of missing data [6] [16] which produced relatively good performance. But they did not consider the information of multiple sources embedded in user-item rating matrix. Therefore, we study an integrated recommendation framework of CF approaches using the information of multiple sources from user-item matrix.

In this paper, we propose a rating-based integrated recommendation framework INTE-CF with improved CF methods. Our integrated framework is, to some extent, similar but different to hybrid recommendation approach which usually combined CF methods with content-based approaches applying the strategies of pre-fusion or/and post-fusion, or built linear combination of different CF methods. Our framework could directly obtain the values of optimal fusion parameters by one time of learning, whereas other methods like [16] [24] found out the suitable values of combination parameters by many times of learning and manual comparisons. In our framework, an objective optimization function for predicting unknown ratings is put forward by considering three varying information sources from different perspectives of improved traditional CF approaches. The framework can implement more accurate recommendations through learning the optimization parameters, whose advantages are fully leveraging the information embedded in user-item matrix from three different perspectives, reducing the dependence on missing data and balancing three CF methods by optimization parameters.

The remainder of the paper is organized as follows. We first summarized the related works in section 2. The rating-based integrated recommendation framework is presented in section 3. Section 4 presented the improved approaches of traditional item- and user-based methods. We designed a background rating prediction method based on both similar users and similar items in section 5. The details of integrated recommendation framework are demonstrated in section 6. The experimental results of the proposed scheme are discussed in section 7. Finally, we discussed the findings of our work along with the future work in the last section.

2 Related works

Since the recommender systems were generated, CF recommendation has been viewed as the most successful recommender method including memory-based heuristic approaches CF methods and model-based learning approaches [14]. There have been many CF recommender applications in academia and industry. To the best of our knowledge, Tapestry system [9] identifying like-minded user is the earliest real CF recommender system. And Amazon Web Site [13] is a famous application of CF approach.

In order to increase the recommendation accuracy, many scholars tried to improve CF approaches by varying similarity calculations between users or items. Breese [3] etc. compared the prediction accuracy of several similarity algorithms including correlation coefficient-based algorithm, vector-based algorithm and statistical Bayesian algorithm. Choi [4] etc. proposed a new similarity function for selecting neighbors for each target item. Others like Conditional Probability-Based Similarity algorithm [7] of item similarity and Genetic algorithm [2] of user similarity were also studied. Good similarity computation method as a kind of enhancement of CF methods indeed improves the recommendation accuracy to some extent. But it is sensitive to the data quality like the sparsity of dataset.

Despite the success of CF approaches, sparsity is still a major challenge and heavily affects

recommendation accuracy. The fact is that a large volume of entries' value in user-item matrix is missing. Therefore, some solutions were proposed to address the issues of sparsity. The simplest ways [6] are using either the value of zero or the average rating of users or items. Obviously, these two ways are too coarse and imprecise. Later on, some relatively better methods were adopted, such as dimensionality reduction based on matrix factorization [20], imputation based on preliminary rating prediction of missing data [16] [8]. In recent years, some other information related to users or items was adopted to alleviate sparsity, like the trust and distrust relationships between users, which were studied in open Dataset of Epinions [1]. The information in social networks such as friend relations, social influence, is also researched to alleviate sparsity by some scholars [27]. Indeed, these information is useful, but they are not always obtained like in the MovieLens dataset, unless in social networks. It is straightforward that the information that could be utilized in recommendation depends on the questions of a certain specific domain. Therefore in this paper, we consider only the information embedded in user-item rating matrix and the semantic information of items to reduce the sparsity according to the limited available information.

In addition to taking full advantage of improvements of similarity computation, dimensionality reduction and other related information sources, there is another important improvement way called hybrid filtering which combines CF with other recommendation approaches. Lu [15] etc. proposed the CCF approach for the news topic recommendation in Bing, which combined CF approach and content-based filtering method. In E-commerce, Song [24] etc. leveraged both demographic recommendation techniques and CF algorithms to put forward a hybrid algorithm in order to improve recommendation accuracy. Ma [16] etc. proposed a linear combination of user- and item-based methods based on the missing value prediction by finding suitable combination parameters and obtained better performance. Moin [17] etc. suggested feature hybrid weighting schemes for improving the precision of neighborhood based CF algorithms, while it increases the complexity of computation. From the perspective of optimization, Nilashi [18] etc. proposed hybrid recommendation for CF method based on multi-criteria to improve prediction accuracy. Hybrid recommendations absorb the advantages of each recommender algorithm and do improve the precision of recommendation. They effectively alleviated sparsity and solved the problem of Cold Start to some extent, especially the combination of content-based method and CF approach. In this paper, we also leverage a similar but different idea of content-based method to improve the item-based CF approach, which utilize inner similarity of items and is discussed in subsection 4.1.

To sum all, CF is applied successfully into all kinds of recommendation fields and obtained a lot of improvements, which focuses on the similarity improvements either in user-based or in item-based methods and combinations with other types of recommendation algorithms. In this paper, we emphasis on the improvement of accuracy by CF-self integration based on three types of ratings deriving from three perspectives of users, items and both users and items. The integrated model is proved to be more accurate, effective and interpretive than some other mainstream CF methods demonstrated in our experiments. What's more, the model is easy to be paralleled to improve the running efficiency.

3 Integrated recommendation framework of CFs

Our proposed integrated CF recommendation framework, namely, INTE-CF is shown in Fig.1, which comprises four core parts. The first part is to generate the first type of rating (Rating 1) from the perspective of item by improving conventional item-based CF approach based on the fusion of two kinds of similarities of item. The second part is in charge of generating the second type of rating (Rating 2) from the perspective of user by improving the traditional

user-based CF (UBCF) approach with UIIM extracted from user-item matrix. The third part demonstrates a combination model of generating the third type of prediction rating (Rating 3) based on both similar items and similar users from the two perspectives of item and user. The three types of ratings are integrated together to build an objective function f in the part 4, which is our proposed integrated optimal model. It is tuned by the optimization parameters which are learned by training sample data. The parts of 1, 2, and 3 serve the part 4. The details of generating each type of rating are demonstrated in the subsequent sections.

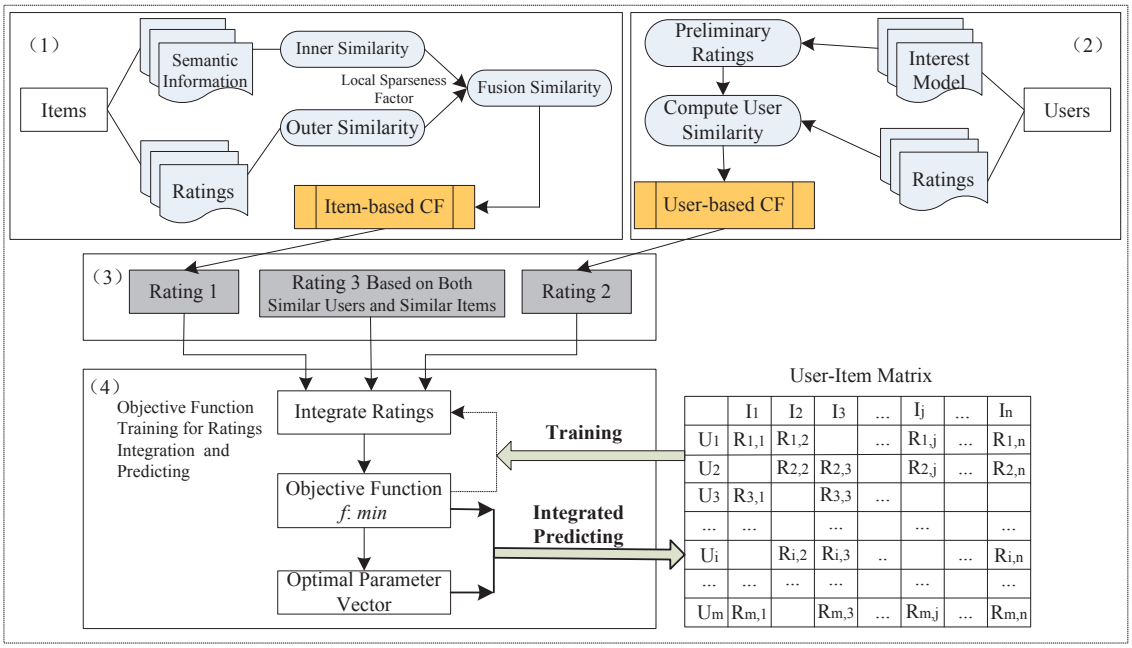


Figure 1: The rating-based integrated recommendation framework of CF approaches

4 Improvements of traditional item-based and user-based CF

Item-based and user-based CF approaches have similar rationale to predict the unknown ratings in user-item matrix. Firstly, the neighbors of target user or item are obtained by similarities in the two CF approaches. Then the unknown rating of each entry related to the target user or item in user-item matrix is predicted by these neighbors whose similarities to the target user or item are viewed as weights in calculation. Lastly, the top-K recommendation list is generated in accordance with the predicted ratings. The details of them can refer to [7, 13, 19, 23].

4.1 Improved item-based approach by similarity fusion

Similarities between items

The performance of recommender system partially depends on the computation of similarities between items. According to the principles related to dialectics, the relevance between things is determined by inner factors and outer factors. In conventional item-based CF approaches (IBCF), the similarities which are calculated in line with the ratings in user-item matrix are measured from outer factors, namely, the perspective of user evaluation. Actually, the similarities between items are also influenced, to a large extent, by the inner factors such as the properties of items, which embody item's inherent semantic information [26]. In other words, the similarities between

items depend both on inner factors and outer factors. In this paper, inner factors denote the properties of item, which are utilized to characterize items and depend on the specific objects. For instance, if the object is movie, the properties could be genres etc.; if the object is product or commodity, the properties could be appearance, genres, color, function, price, quality etc. Therefore, it is necessary to take the two kinds of factors into account to measure the similarities between items. For convenience, the similarity produced by outer factors is called outer similarity and the similarity produced by inner factors is called inner similarity. The outer similarity is calculated by the ratings of items in user-item matrix showed in Eq. (1). The inner similarity is computed by the properties of items showed in the Eq.(2).

$$sim^{\mathrm{I}}_{\mathrm{out}}(i,j) = \frac{\vec{I_i} \cdot \vec{I_j}}{||\vec{I_i}|| \times ||\vec{I_j}||} \tag{1}$$

$$sim^{\mathrm{I}}_{\mathrm{in}}(i,j) = \sum_{k=1} \varphi(k) sim(\Theta(k), i, j) \tag{2}$$

where Θ denotes the property set of item i and item j , $sim(\Theta(k), i, j)$ represents the similarity of item i and item j on property k in Θ, $\varphi(k)$ is the weight of property k like the genre property of a movie.

Item local sparsity factor

User-item matrix is commonly heavily sparse. The existing sparsity called global sparsity degree is measured by the ratio that is equal to the number of unknown ratings over the number of total entries in user-item matrix. When calculating the item similarity, we defined a local sparsity factor, i.e., *item local sparsity factor* which is used to describe the sparsity of the set of co-ratings from the local perspective of item.

Definition 1. (Item local sparsity) Let U^{I}_i be the set of users who rated on item i and U^{I}_j be the set of users who rated on item j, then the item local sparsity is defined as:

$$SP^{\mathrm{I}}_{i,j} = \frac{2 * |U^{\mathrm{I}}_i \cup U^{\mathrm{I}}_j| - (|U^{\mathrm{I}}_i + |U^{\mathrm{I}}_j|)}{2 * |U^{\mathrm{I}}_i \cup U^{\mathrm{I}}_j|} \tag{3}$$

Fusion of inner similarity and outer similarity of item

According to the aforementioned analysis, it is reasonable to fuse the inner similarity and outer similarity of items. The item local sparsity factor can be used to balance the outer similarity and inner similarity. Therefore, we define the weight function between inner and outer similarities incorporating the item local sparsity by sigmoid function as follows:

$$f(SP^{\mathrm{I}}_{i,j}) = \begin{cases} \frac{1}{1+e^{-SP^{\mathrm{I}}}} & 0 \le SP^{\mathrm{I}}_{i,j} < 1 \\ 1 & SP^{\mathrm{I}}_{i,j} = 1 \end{cases} \tag{4}$$

Clearly, $SP^{\mathrm{I}}_{i,j}$ is between 0 and 1, and $f(SP^{\mathrm{I}}_{i,j})$ belongs to 0.5 and 1, which guarantees inner similarity always be in the resulting item similarity, because inner similarity between two items is always useful and works. When $SP^{\mathrm{I}}_{i,j}$ equals 0, namely, two items have the complete common rating users and the set of co-rating users is full, the value of $f(SP^{\mathrm{I}}_{i,j})$ is 0.5, which means inner and outer similarities have same weights. When $SP^{\mathrm{I}}_{i,j}$ equals 1 that means item i and item j have

no common rating users, $f(SP^{\mathrm{I}}_{i,j})$ is set to 1, i.e., the similarity between item i and item j only depends on the inner similarity. The resulting similarity after being fused based on $f(SP^{\mathrm{I}}_{i,j})$ is as follows:

$$sim^{\mathrm{I}}(i,j) = f(SP^{\mathrm{I}}_{i,j}) * sim^{\mathrm{I}}_{\mathrm{in}}(i,j) + (1 - f(SP^{\mathrm{I}}_{i,j})) * sim^{\mathrm{I}}_{\mathrm{out}}(i,j) \tag{5}$$

The resulting similarity of items embodies the two aspects of inner factors and outer factors, effectively alleviates the dependence on the sparsity of user-item matrix and overcomes the item Cold Start problem. $f(SP^{\mathrm{I}}_{i,j})$ balances the inner factors and outer factors. Therefore, IBCF can be improved by the item fusion similarity and called IBCF-I.

4.2 Improved user-based CF for rating prediction based on user item interest model

Although conventional UBCF method can predict rating with some extent accuracy, it still has the space of improvement. The key of conventional UBCF approach is to find quality neighbors of a target user. So the similarity calculation between users is important. But due to the heavy sparsity of initial user-item matrix, sometimes the similarity calculation like cosine similarity in conventional UBCF method has relatively low accuracy, even not correct occasionally. In order alleviate the sparsity, Deng [6] etc. proposed an approach of preliminary rating of unknown rating entries, which, while, still suffered from the sparsity, since it only used the existing known ratings. Different from Deng [6], we propose a preliminary rating model (PRM) based on user-item-interest to conquer the sparsity, which is similar to imputation. And the rating prediction method UBCF-I is put forward based on PRM.

Applying CF method, user-item matrix is the information source leveraged to make study and analysis. The UIIM model is built based on KNN cluster approach using inner similarities between items. Generally, User rates similar items with similar ratings. Therefore, items that user has rated can be clustered into k clusters in line with their inner similarities for building UIIM. Then the nearest cluster to the target item with unknown rating entry is selected. Lastly, it is utilized to make preliminary rating for the unknown rating entry. So there are more co-ratings between users, which are used to calculate the similarities between users. Obviously it can produce better accuracy of user similarity than traditional UBCF method. The detailed process is discussed as follows.

Let I_p be the known rating item set of user u_p, I_q be the known rating item set of user u_q, $I^{\cup}_{p,q}$ be the union set of I_p and I_q, and $I^{\cap}_{p,q}$ be the intersect set of I_p and I_q with co-ratings namely, $I^{\cup}_{p,q} = I_p \cup I_q$, $I^{\cap}_{p,q} = I_p \cap I_q$. Then the unknown rating item sets N_p and N_q of user u_p and user u_q are $N_p = I^{\cup}_{p,q} - I_p$ and $N_q = I^{\cup}_{p,q} - I_q$, respectively.

The process of preliminary ratings of N_p and N_q are similar, here we take an example for N_p. Assume item $I_j \in N_p$, firstly, compute the semantic distances of item I_j to the k clusters in UIIM of the user u_p, and sort them by ascend. The cluster which is the nearest to the item I_j is selected as the neighbors called I_n. Then calculate the preliminary rating $R'_{p,j}$ of unknown rating entry of user u_p on item I_j according to the neighbors I_n, as follows:

$$R'_{p,j} = \frac{\sum_{l \in I_n} sim^{\mathrm{I}}(j,l) * R_{p,l}}{\sum_{l \in I_n} sim^{\mathrm{I}}(j,l)} \tag{6}$$

So far, each entry in the union set $I^{\cup}_{p,q}$ has co-ratings of user u_p and user u_q either known rating or preliminary rating. The resulting similarity between user u_p and user u_q is quality. Therefore, in user space the similarities between user u_p and other users can be calculated effectively. The nearest neighbor user set NU_p used to calculate the similarities to the target user u_p is formed in lines with the rule of top-K. Finally, UBCF-I is applied to predict the unknown ratings of user

u_p.

UBCF-I method has some advantages (1) searching similar items for preliminary ratings is in a small item scope rather than in the whole item space, which derived from the most related items in UIIM; (2) avoiding the sparsity of computing the item similarity for making preliminary rating, especially for the case two users have many ratings but few common ones, since all the missing values between two users are imputed in our method when computing their similarity.

Although UBCF-I has some strong points, we don't intend to deeply research the serious user Cold Start problem in which the number of user ratings is close or equal to 0. To well solve the problem needs some other information like user social information [12, 27], trust relationships [1] and etc., which should be deeply studied but not always be achieved in traditional dataset such as MovieLens, unless in social network. Therefore, as for the very serious user Cold Start problem in which clustering is ineffective, considering averaging user's average rating and the item's average rating as the preliminary rating is a good selection [8].

5 Rating based on both similar users and similar items

IBCF-I and UBCF-I predict ratings from the perspectives of similar items and similar users, respectively. But only depending on one of them is undesirable [22, 23]. It is necessary to think about that taking both similar users and similar items into account, which correspond to the rows and columns in user-item matrix, respectively, can provide more effective information sources for predicting ratings. That means similar users making similar item ratings provides an extra and useful information source for prediction. But how to make full use of the information that derives from both similar users and similar items? Firstly, reorder user-item matrix according to the similarities of users and similarities of items. Second, generate the predictive ratings by fusing the two similarities towards the target user and the target item related to the entries with unknown ratings in user-item matrix. Therefore, we proposed one kind of CF method using the compound similarity based on the two-dimension coordinates to address the problem. For the convenience of expression, we call this method UIBCF-I. The rationale of UIBCF-I is shown in Fig.2 as follows.

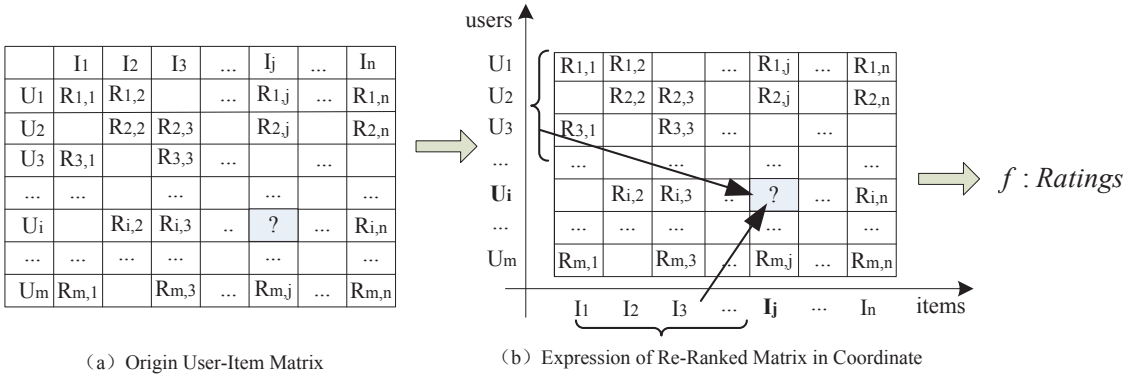


Figure 2: Principle of predicting rating based on UIBCF-I

The part (a) denotes the original user-item matrix and part (b) represents the rebuilding and mapping of user-item matrix in two-dimension coordinates. Horizontal axis denotes item and vertical axis represents user. The entries of user-item matrix correspond to the points in coordinates. All the users and items are ordered in descend by the magnitudes of similarities to the target user and the target item. The question mark denotes the entry of unknown rating related to the target user U_i and the target item I_j. The top-K most similar users U_{ss} and top-M

most similar items I_{ss} to the user U_i and the item I_j are selected, respectively. The predictive rating can be calculated by the compound similarity of similar users and similar items in Eq.(7).

$$R^p_{\text{ss}}(i,j) = \frac{\sum\limits_{k\in U_{\text{ss}}}\sum\limits_{m\in I_{\text{ss}}} sim^{\text{SS}}(i,j,k,m) * R(k,m)}{\sum\limits_{k\in U_{\text{ss}}}\sum\limits_{m\in I_{\text{ss}}} sim^{\text{SS}}(i,j,k,m)} \tag{7}$$

where sim^{SS} represents the compound similarity of similar users and similar items, which is computed in Eq. (8) as follows:

$$sim^{\text{SS}}(i,j,k,m) = \lambda_1 sim^{\text{U}}(i,k) + \lambda_2 sim^{\text{I}}(j,m) \tag{8}$$

λ_1 and λ_2 are tuning parameters, whose values are commonly denoted by 0.5 respectively.

6 Integrated CF recommendation model by ratings fusion

6.1 Overview

The core task of a recommendation algorithm is to predict which items a user relatively most likes based on his/her observed feedback which denotes ratings on items here. So far, we have obtained three types of ratings, namely, ratings of 1, 2 and 3 according to the aforementioned content. They are obtained by three different methods from three varying information sources. Each of them has its own strength and weakness. How to combine the three types of ratings represents a novel challenge. We proposed the optimal integration framework INTE-CF based on the three types of ratings by learning relevant parameters. In INTE-CF model, the ratings predicted by IBCF-I, UBCF-I and UIBCF-I from three varying perspectives and using different information sources which complement each other. And UIBCF-I could also be viewed as a background method of IBCF-I and UBCF-I and smooth the rating predictions generated by IBCF-I and UBCF-I. Therefore, the integration of the three types of ratings not only leverages the three varying information sources but also can reduce the dependence on data sparsity.

Let $U = \{u_1, u_2, ..., u_m\}$ be the set of m users and $I = \{i_1, i_2, ..., i_n\}$ be the set of n items. r_{ui},$\hat{r}_{ui}$ denote the rating and predicted rating of user u on item i, respectively. $\hat{r}^{(1)}_{ui}, \hat{r}^{(2)}_{ui}, \hat{r}^{(3)}_{ui}$ represent user $u\prime s$ predicted ratings by IBCF-I, UBCF-I and UIBCF-I on item i, respectively. $\vec{\hat{r}}_{ui}$ is a predicted rating vector composed of $\hat{r}^{(1)}_{ui}, \hat{r}^{(2)}_{ui}, \hat{r}^{(3)}_{ui}$. We also use $R \in \mathbb{R}^{m\times n}$ to represent the matrix of observed ratings and $\vec{w} = (w_1, w_2, w_2)$ to denote a parameter vector. For convenience, we use $S^*_u \subseteq U \times I$ to denote the set of user-item pairs of user u, for which the observed ratings are available.

6.2 Integration

We have obtained user $u\prime s$ three predicted ratings of $\hat{r}^{(1)}_{ui}, \hat{r}^{(2)}_{ui}, \hat{r}^{(3)}_{ui}$ on item i, which are leveraged to predict user $u\prime s$ preference on item i from three different perspectives. In order to achieve more accurate user $u\prime s$ predictive rating on item i, we proposed an algorithm of combining them with an integrated model to implement the aforementioned framework INTE-CF as follows:

$$\hat{r}_{ui} = \hat{r}^{(1)}_{ui} * w_1 + \hat{r}^{(2)}_{ui} * w_2 + \hat{r}^{(3)}_{ui} * w_3 = \vec{\hat{r}}_{ui}\vec{w}^{\text{T}} \tag{9}$$

Actually, it is an optimization problem of the following general form:

$$\min_{\vec{w}}(\ell(r_{ui}, \hat{r}_{ui}) + \mathbb{R}(\vec{w})) \tag{10}$$

Here $\ell(r_{ui}, \hat{r}_{ui})$ is a loss function measuring the discrepancy between the observed rating and the predicted rating of user $u\prime s$ on item i. The regularization function $\mathbb{R}(\vec{w})$ overly penalizes the model to suppress overfitting.

The goal of the model is to make the predicted rating $\hat{r}_{ui}$ as close to the observed rating r_{ui} as possible. The common and good selection for the loss function is to use squared error loss form:

$$\ell(r_{ui}, \hat{r}_{ui}) = \frac{1}{2} \sum_{(u,i) \in S_u^*} (r_{ui} - \hat{r}_{ui})^2 \tag{11}$$

Certainly, there are several other forms of loss function such as [14]. Here we used squared error form for loss function due to its simplicity and easiness of implementation.

We used the Frobenius norm of parameters to build the regularization function $\mathbb{R}(\vec{w})$, which was adopted by Koren [10] et al. due to its smooth differentiable property.

$$\mathbb{R}(\vec{w}) = \frac{1}{2} \lambda ||\vec{w}||_F^2 \tag{12}$$

where the parameter $\lambda \geq 0$ is used to control the strength of regularization and helps to balance between training error and model complexity. So the training model can be rebuilt as follows according to the equations of (10), (11) and (12):

$$f(\vec{\hat{r}}_{ui}, \vec{w}) = \min_{\vec{w}} (\ell(r_{ui}, \vec{r}_{ui}) + \mathbb{R}(\vec{w})) = \frac{1}{2} \min_{\vec{w}} (\sum_{(u,i) \in S_u^*} (r_{ui} - \vec{\hat{r}}_{ui} \vec{w}^T)^2 + \lambda ||\vec{w}||_F^2) \tag{13}$$

For convenience, we convert Eq.(13) into Eq. (14):

$$\begin{cases} f_{\min} = \frac{1}{2} (\sum\limits_{(u,i) \in S_u^*} (r_{ui} - \vec{\hat{r}}_{ui} \vec{w}^T)^2 + \lambda ||\vec{w}||_F^2) \\ s.t. \qquad ||\vec{w}||_F^1 = 1, \quad w_j \geq 0, \quad j \in \{1, 2, 3\} \end{cases} \tag{14}$$

Obviously, this is an optimization problem and could be solved by dynamic program with constraints. We adopted the Stochastic Gradient Descent (SGD) [11] method to learn the parameters in order to accelerate the optimization process. The optimization procedure is shown in Algorithm 1. The algorithm takes as input the matrix R of observed ratings, error ε and a group of vectors $\vec{\hat{r}}_{ui}$s which derive from the three types of predicted ratings.

7 Experiments

In order to verify our proposed integrated model INTE-CF, we experimented on the classic dataset of MovieLens[1] and EachMovie[2]. Due to the high similar results on the two datasets, we only report the experiment results of MovieLens (out of space consideration). The MovieLens dataset is comprised with 943 users, 1682 movies (items) and 100,000 ratings (1-5 scales) with the global sparsity of 0.93695, where each user has rated at least 20 items.

To better validate our proposed INTE-CF model, we conducted 4 groups of experiments corresponding to 200, 400, 600 and 800 users and relevant data extracted from the dataset at random. The purpose of dividing the dataset into 4 groups is to find out the difference of optimal parameters which, we thought, depend on the information of specific dataset, such as the size, sparsity. The experiments were finished in line with 10-fold cross-validation. We have two goals to conduct the experiments. One is to validate the higher prediction accuracy and more effectiveness of INTE-CF model. The other is to find out the component variation law of optimal parameter vector with varying scale dataset.

[1] http://www.grouplens.org/

[2] http://www.research.digital.com/SRC/EachMovie/

Algorithm 1 The optimization of INTE-CF model

Input: The rating matrix R, error ε and a group of vectors $\vec{\hat{r}}_{ui}$s of three types of predicted ratings.
Output: Model parameter vector $\vec{w} = (w_1, w_2, w_3)$.
Begin
Initialize the vector $\vec{w} = (0.333, 0.333, 0.334)$, $k \leftarrow 1$ and $f^{(0)} = 0$;
Calculate $f^{(1)}$;
while $|f^{(k)} - f^{(k-1)}| > \varepsilon$ **do**
    $k \rightarrow k + 1$;
    $s_1^{(k)} \leftarrow -\frac{\nabla f(w_1)}{||\nabla f(w_1)||}$, $s_2^{(k)} \leftarrow -\frac{\nabla f(w_2)}{||\nabla f(w_2)||}$;
    $w_1 \leftarrow w_1 + \alpha_k s_1^{(k)}$, $w_2 \leftarrow w_2 + \alpha_k s_2^{(k)}$;//$\alpha_k$ is learning parameter
    $w_3 \leftarrow 1 - w_1 - w_2$;
    Calculate $f^{(k)}$;
end while
Return vector $\vec{w} = (w_1, w_2, w_3)$
End

7.1 Preliminary

Here we conducted the experiments on the dataset of MovieLens. In order to calculate the inner similarities between items (movies) discussed in the sub-section of 4.1, we need to quantify the information of item's genre properties which characterize the movie's inherent features. The genre of each movie in the dataset is multi-valued and has 20 possible values such as drama, action and comedy. In general, more than one of these genres present with different degree in a movie. Some of them with high presence are called major or dominating genres for that movie. For example for the movie "*Copycat*" as presented in the MovieLens dataset [28] has "*Crime/Mystery/Thrill/Drama*", *Crime* is the most dominating genre value; Mystery is the second one, etc. The rest in 20 possible genres do not present. Therefore, in order to quantify the presence degree, we utilize a Gaussian-like function [28] to compute it.

$$\mu(g_i, I_j) = r_i / 2^{\sqrt{\alpha * N_j * (r_i - 1)}} \tag{15}$$

Where g_i denotes the genre i of item I_j, N_j represents the total number of the present genres, r_j denotes the rank position that indicates the magnitude of presence of g_i and $1 \leq r_i \leq N_j$. The rank positions of those genres with no presence equal 0. And $\alpha > 1$ is a constant threshold which controls the difference in presence degree of the g_i in the item I_j. Here we set $\alpha = 1.2$ which makes the calculation perfect [28]. Due to no information of rank positions of genres in the dataset, we complemented them by crawling the information from the online Movie Database(http://www.imdb.com/).

7.2 Metrics

As for assessing the accuracy of a recommender system with prediction ratings, one of the most popular evaluation metrics is Mean Absolute Error (MAE) [21, 27], which measures the average absolute deviation between the real rating assigned by the user and the predicted rating calculated by a certain recommendation algorithm. Therefore, we use MAE to measure the prediction quality of our proposed integrated framework INTE-CF with other mainstream CF methods, which is defined as follows.

$$\text{MAE} = \frac{\sum_{(u,i)\in R_{\text{test}}} |r_{u,i} - \hat{r}_{u,i}|}{|R_{\text{test}}|} \tag{16}$$

where R_{test} is the set of all user-item pairs (u, i) in the test set. The smaller MAE value means a better performance.

7.3 Preliminary experiments of verifying UBCF-I and IBCF-I methods

We first conducted a preliminary experiment to verify the effectiveness of our proposed improved CF methods IBCF-I and UBCF-I, which are compared to the conventional methods of UBCF and IBCF so that we could proceed to do the next further experiments of INTE-CF model. We randomly selected half of data of the MovieLens dataset to conduct the preliminary experiment. The data was split into two parts, namely, training set 80% and prediction set 20%. The related users and items are 444 and 1605, respectively. The global sparsity is 0.9298. The experimental results are showed in Fig.2.

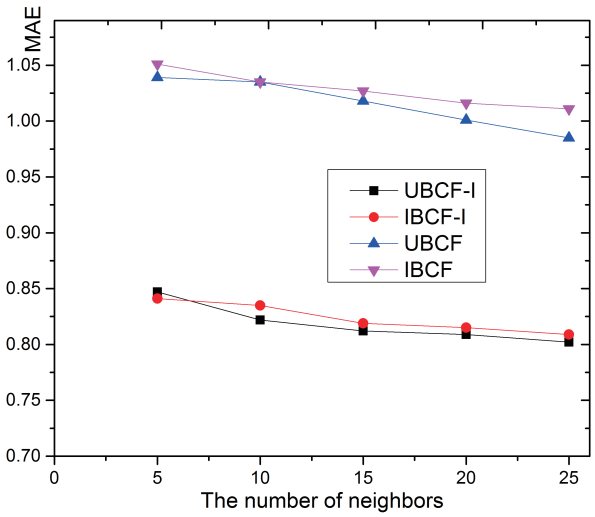


Figure 3: Preliminary experiment about comparisons of MAE between improved methods and conventional methods

Obviously, UBCF-I and IBCF-I methods are more accurate than the conventional methods of UBCF and IBCF from overall view. UBCF-I obtains maximum 19.57% and average 17.25% increases than UBCF, respectively. Similarly, IBCF-I obtains maximum 18.93% and average 18.12% increases than IBCF method, respectively. They all get lower MAE value with the increase of neighbors since UBCF-I approach benefits from UIIM and IBCF-I method benefits from fusion similarity composed of inner and outer similarities. We achieved the significant improvements of UBCF-I and IBCF-I approaches on MAE and not planned to further analysis and conduct the preliminary experiments. We emphasized on the subsequent experiments of our integrated framework model INTE-CF.

7.4 Experiments for predictive accuracy

We conducted 4 experiments in which users are divided into 4 groups of 200, 400, 600 and 800. For convenience, we call them G2, G4, G6 and G8, respectively. We compared our integrated model INTE-CF to two individual predictors, namely, UBCF-I and IBCF-I, and other two combination predictors, namely, our proposed UIBCF-I method and a linear combination method [16],

Table 1: Comparison to other CF methods: A smaller value means a better performance

Groups	G2	G4	G6	G8
INTE-CF	**0.792**	**0.744**	**0.731**	**0.711**
UIBCF-I	0.887	0.827	0.774	0.748
UBCF-I	0.845	0.775	0.763	0.759
IBCF-I	0.873	0.778	0.786	0.764
UI-Linear	0.861	0.819	0.762	0.728

Table 2: The value of average optimal vector in 4 groups

W	G2	G4	G6	G8
W3	0.625	0.697	0.711	0.725
W2	0.204	0.192	0.174	0.166
W1	0.171	0.111	0.115	0.109

which, for convenience, is called UI-Linear showed at the last row in Table 1. The optimal number of neighbors was 35 selected by many tests. Table 1 summarizes the results, showing the how INTE-CF approach outperforms the other methods in all 4 groups of experiments.

INTE-CF approach is the best recommendation method in Table 1. UBCF-I and IBCF-I have relatively low performance compared to INTE-CF, although they are improved based on standard CF method. UIBCF-I similar to UI-Linear is less accurate than UBCF-I and IBCF-I in G2 and G4 because of the less data when both considering similar users and similar items. But their accuracy increases fast with more users and items. If there are enough users and items, they will outperform UBCF-I and IBCF-I just like in G6 and G8, since they benefit from the strengths of combination. INTE-CF has the best performance which fuses three information sources deriving from UBCF, IBCF and UIBCF-I, and absorbs the advantages of them. We also found that the performances of all the methods have been improved with more users in dataset. It is evident that more users and items produce more ratings on whole, which provide more accurate prediction when applying enhanced CF methods.

7.5 Discussion about optimal parameter vector

Each group of experiments generates 10 optimal parameter vectors in 10-fold cross-validation experiments. In order to demonstrate the overall changes of vectors and proportions of UBCF-I, IBCF-I and UIBCF-I, we select the average optimal parameter vector of each group of experiments for the comparisons. Table 2 and Fig.3 show the changes of value derived from all the 4 groups of average optimal parameter vectors.

Each vector contains three components which are W1, W2 and W3 corresponding to the weights ω_1, ω_2 and ω_3 of IBCF-I, UIBCF-I and UIBCF-I in parameter vector, respectively. In each group of experiment, W1, W2 and W3 have the similar situations. W3 is dominant value and plays an important role in predicting ratings, especially in more rating data. W1 and W2 decreased with the increase of rating data, maybe since both more similar users to the target user and more similar items to the target item result in great influence on UIBCF-I. The optimal value of W3 is in vibration around 0.7 in most cases. Combining Table 1 and Fig.3, our proposed model of INTE-CF makes full use of the three kinds of information sources of IBCF-I, UBCF-I and UIBCF-I from varying views and obtains the best performance.

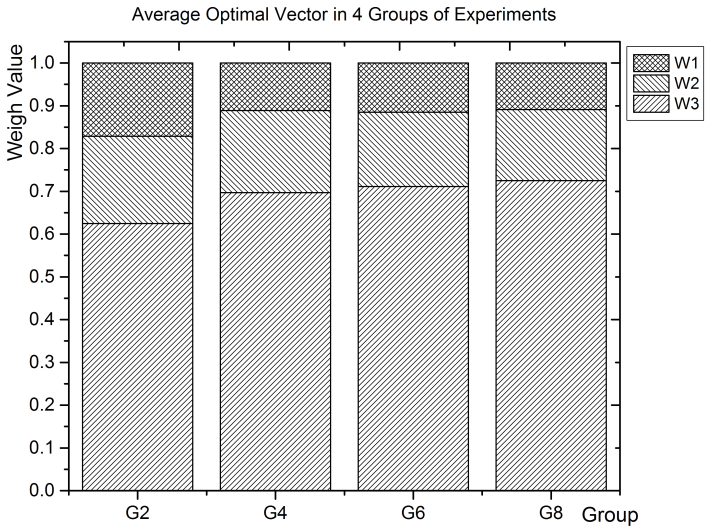


Figure 4: Three components of average optimal parameter vector in 4 groups

Table 3: Statistics information of 4 groups

Groups	G2	G4	G6	G8
Users	200	400	600	800
Items	1409	1484	1596	1655
Existing data count	22378	41826	64384	85697
Theoretical data count	281800	553600	957600	1324000
Global sparsity	0.9206	0.9296	0.9328	0.9353

Table 3 gives the statistics information of 4 groups of experiments. The global sparsity in each group is adjacent regardless of the increase of the rating data, and is close to the global sparsity 0.93695 of the whole dataset. The three components of the optimal parameter vector changes a little in 4 groups. The background method of UIBCF-I plays a great role in 4 groups, especially in G8 whose global sparsity is relatively large. The optimal parameter vector is low sensitive to the data size. The component weights of UIBCF-I in 4 groups are all high since compound similarity between items does work and UIBCF-I makes full use of both similar users and similar items.

8 Conclusions and future work

As for the shortage of individual predictors of conventional item-based and user-based CF recommendation approaches utilizing single information source, we proposed a rating-based integrated framework to combine three CF recommendation methods of IBCF-I, UBCF-I and UIBCF-I. UIBCF-I is considered as a background method to smooth the rating predictions of UBCF-I and IBCF-I. Meanwhile, we improved traditional item-based CF by inner similarity and outer similarity, and user-based CF by preliminary ratings based on UIIM. Furthermore, we built an optimal learning model INTE-CF of the framework by dynamic program with constraints to find out the optimal parameter vector in rating predictions. The experiments showed that our new integration framework of CFs is effective in improving the prediction accuracy of CF recommendation approaches. That is to say our integrated model INTE-CF leveraging the three kinds of information sources achieves the best performance. But INTE-CF pays the price of a little more running time which is not avoided but worthy. Fortunately, some calculations which accelerate the overall execution process are off line or incremental, including inner similarity of between items, clustering of UIIM, and the predictions of three ratings (rating 1, 2 and 3) can be parallel processing.

In the future work, we will compare INTE-CF model to more other methods such as SF [25] and evaluate it on more metrics. The parallelization of INTE-CF is also interesting when encountered big data. And we will continue to optimize INTE-CF by considering the rating biases of users and items and the influence of time.

Acknowledgment

We would like to thank all of the anonymous reviewers for their insightful comments and useful suggestions that must lead to a much higher quality of our manuscript.This work was partially supported by the National Science Natural Foundation of China (nos. 61303096).

Bibliography

[1] Anand D., Bharadwaj K.K. (2013); Pruning trust-distrust network via reliability and risk estimates for quality recommendations, *Social Network Analysis and Mining*, 3(1), 65-84, 2013.

[2] Bobadilla J., Ortega F., Hernando A.; Alcal J. (2011); Improving collaborative filtering recommender system results and performance using genetic algorithms, *Knowledge-based systems*, 24(8), 1310-1316, 2011.

[3] Breese J.S., Heckerman D., Kadie C. (1998); Empirical analysis of predictive algorithms for collaborative filtering, *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, 43-52, 1998.

[4] Choi K., Suh Y. (2013); A new similarity function for selecting neighbors for each target item in collaborative filtering, *Knowledge-Based Systems*, 37, 146-153, 2013.

[5] Das A. S., Datar M., Garg A., Rajaram S. (2007); Google news personalization: scalable online collaborative filtering, *Proceedings of the 16th international conference on World Wide Web*, 271-280, 2007.

[6] Deng A.L., Zhu Y.Y., Shi B. (2003); A collaborative filtering recommendation algorithm based on item rating prediction, *Journal of Software (Chinese)*, 14(9), 1621-1628, 2003.

[7] Deshpande M., Karypis G. (2004); Item-based top-n recommendation algorithms, *ACM Transactions on Information Systems (TOIS)*, 22(1), 143-177, 2004.

[8] Ghazanfar M.A., Pršgel-Bennett A. (2013); The Advantage of Careful Imputation Sources in Sparse Data-Environment of Recommender Systems: Generating Improved SVD-based Recommendations, *Informatica (Slovenia)*, 37(1), 61-92, 2013.

[9] Goldberg D., Nichols D., Oki B.M., Terry D. (1992); Using collaborative filtering to weave an information tapestry, *Communications of the ACM*, 35(12), 61-70, 1992.

[10] Koren Y. (2010); Collaborative filtering with temporal dynamics, *Communications of the ACM*, 53(4), 89-97, 2010.

[11] Li Q., Sato I., Murakami Y. (2007); Efficient stochastic gradient search for automatic image registration, *International Journal of Simulation Modelling (IJSIMM)*, 6(2), 114-123, 2007.

[12] Li W., Ye Z., Xin M., Jin Q. (2015); Social recommendation based on trust and influence in SNS environments, *Multimedia Tools and Applications*, 1-18, 2015.

[13] Linden G., Smith B., York J. (2003); Amazon.com recommendations: Item-to-item collaborative filtering, *IEEE Internet computing*, 7(1), 76-80, 2003.

[14] Liu N.N., Zhao M., Yang Q. (2009); Probabilistic latent preference analysis for collaborative filtering, *Proceedings of the 18th ACM conference on Information and knowledge management*, 759-766, 2009.

[15] Lu Z., Dou Z., Lian J., Xie X., Yang Q. (2015); Content-Based Collaborative Filtering for News Topic Recommendation, *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 217-223, 2015.

[16] Ma H., King I., Lyu M.R. (2007); Effective missing data prediction for collaborative filtering, *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 39-46, 2007.

[17] Moin A., Ignat C.L. (2014); *Hybrid weighting schemes for collaborative filtering (Doctoral dissertation, INRIA Nancy)*, France, 2014.

[18] Nilashi M., bin Ibrahim O., Ithnin N. (2014); Hybrid recommendation approaches for multicriteria collaborative filtering, *Expert Systems with Applications*, 41(8), 3879-3900, 2014.

[19] Park D. H., Kim H. K., Choi I.Y., Kim J.K. (2012); A literature review and classification of recommender systems research, *Expert Systems with Applications*, 39(11), 10059-10072, 2012.

[20] Paterek A. (2007); Improving regularized singular value decomposition for collaborative filtering, *Proceedings of KDD cup and workshop*, 5-8, 2007.

[21] Ricci F., Rokach L., Shapira B. (2011); *Introduction to recommender systems handbook*, Springer, 2011.

[22] Sarwar B., Karypis G., Konstan J., Riedl J. (2001); Item-based collaborative filtering recommendation algorithms, *Proceedings of the 10th international conference on World Wide Web*, 285-295, 2001.

[23] Shi Y., Larson M., Hanjalic A. (2014); Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges, *ACM Computing Surveys (CSUR)*, 47(1), 3-45, 2014.

[24] Song R.P., Wang B., Huang G.M., Liu Q.D., Hu R.J., Zhang R.S. (2014); A hybrid recommender algorithm based on an improved similarity method, *Applied Mechanics and Materials*, 475, 978-982, 2014.

[25] Wang J., De Vries A.P., Reinders M.J. (2006); Unifying user-based and item-based collaborative filtering approaches by similarity fusion, *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 501-508, 2006.

[26] Xu S. Y., Raahemi B. (2016); A Semantic-based service discovery framework for collaborative environments, *International Journal of Simulation Modelling (IJSIMM)*, 15(1), 83-96, 2016.

[27] Yang X., Guo Y., Liu Y., Steck H. (2014); A survey of collaborative filtering based social recommender systems, *Computer Communications*, 41, 1-10, 2014.

[28] Zenebe A., Zhou L., Norcio, A.F. (2010); User preferences discovery using fuzzy models, *Fuzzy Sets and Systems*, 161(23), 3044-3063, 2010.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 323-329, June 2017.

Learning Speed Enhancement of Iterative Learning Control with Advanced Output Data based on Parameter Estimation

G.-M. Jeong, S.-H. Ji

Gu-Min Jeong
School of Electrical Engineering
Kookmin University, Korea
gm1004@kookmin.ac.kr

Sang-Hoon Ji*
Robot R&BD Group
KITECH, Korea
*Corresponding author: robot91@kitech.re.kr

Abstract: Learning speed enhancement is one of the most important issues in learning control. If we can improve both learning speed and tracking performance, it will be helpful to the applicability of learning control. Considering these facts, in this paper, we propose a learning speed enhancement scheme for iterative learning control with advanced output data (ADILC) based on parameter estimation. We consider linear discrete-time non-minimum phase (NMP) systems, whose model is unknown, except for the relative degree and the number of NMP zeros. In each iteration, estimates of the impulse response are obtained from input-output relationship. Then, learning gain matrix is calculated from the estimates, and by using new learning gain matrix, learning speed can be enhanced. Simulation results show that the learning speed has been enhanced by applying the proposed method.
Keywords: iterative learning control, speed enhancement, parameter estimation, learning gain estimation.

1 Introduction

By using Iterative learning control (ILC), the tracking performance can be enhanced when the the same task is performed iteratively [7]– [9]. Among various ILC schemes, Iterative learning control with advanced output data (ADILC) [4] [5] has been proposed for the learning control in discrete time non-minimum phase (NMP) systems. ADILC stabilizes inverse mapping by using output-to-input mapping directly with time-advanced output data. Its learning structure is simple since it consists of an input update law that depends on the relative degree and number of NMP zeros.

On the other hand, due to the complexity of computation, learning speed enhancement is one of the most important issues in iterative learning control (ILC). Considering this, various approaches for direct learning control (DLC) [9] have been proposed.

In [5], an ADILC scheme based on the estimation of the impulse response is proposed for linear discrete-time NMP systems, whose model is unknown, except for the relative degree and the number of NMP zeros. Instead of using an approximate model of the system, the first part of impulse response is estimated and used for the ADILC. However, considering the computational cost of this method, we need a novel scheme to enhance learning speed.

In this paper, we propose a new speed enhancement scheme for discrete time NMP systems, extending the results in [5]. By using the estimates of the learning matrix, an estimate of the desired input is derived and the learning speed can be significantly enhanced. Further, an illustrative example is provided to demonstrate the applicability of the proposed method.

2 ADILC for discrete-time NMP systems

In this section, some preliminary results of the ADILC in [4] are briefly summarized. Let us consider a linear time invariant(LTI) system described by

$$\begin{aligned} x(i+1) &= Ax(i) + Bu(i) \\ y(i) &= Cx(i) \end{aligned} \tag{1}$$

where, $u \in \mathbb{R}^1$, $x = [x_1, \cdots, x_n]^T \in \mathbb{R}^n$, and $y \in \mathbb{R}^1$ are the input, the state, and the output of the system, respectively. A, B and C are matrices of appropriate dimensions. Let $x^d(i)$, $y^d(i)$ and $u^d(i)$ represent the state, the output and the input corresponding to the desired trajectory respectively. Further, let the desired output $y^d(i), i \in [\sigma, N+\sigma-1]$ be given and $\mathbf{u}_{[i,j]} := [u(i), \cdots, u(j)]^T, \mathbf{y}_{[i,j]} := [y(i), \cdots, y(j)]^T$.

The transfer function of the system is represented by $G(z) = \frac{\beta_1 z^{n-1} + \cdots + \beta_n}{z^n + \alpha_1 z^{n-1} + \cdots + \alpha_n}$. Here, it is assumed that the number of NMP zeros, d_0, and the relative degree, σ, are known *a priori* (i.e., $\beta_1 = \cdots = \beta_{\sigma-1} = 0$).

In the ADILC, the following input-output mapping is used to stabilize the inverse mapping.

$$\begin{aligned} \mathbf{y}_{[\sigma+d_0, N+\sigma+d_0-1]} &= \mathbf{H}x(0) + \mathbf{J}\mathbf{u}_{[0,N-1]}, \\ \mathbf{H} &= \left[\begin{array}{ccc} (H_{d_0+1})^T & \cdots & (H_{N+d_0})^T \end{array} \right]^T, \\ \mathbf{J} &= \left[\begin{array}{cccc} J_{d_0+1} & J_{d_0} & \cdots & 0 \\ J_{d_0+2} & J_{d_0+1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ J_{N+d_0} & J_{N+d_0-1} & \cdots & J_{d_0+1} \end{array} \right], \end{aligned} \tag{2}$$

where $H_l = CA^{\sigma+l-1}$, $J_l = CA^{\sigma+l-2}B$. The time interval for the output of interest is $[\sigma + d_0, N+\sigma+d_0-1]$ in (2), whereas it is $[\sigma, N+\sigma-1]$ for minimum phase systems (i.e., $d_0 = 0$).

For an ADILC, we set the input horizon to $[0, N+d_0-1]$ with $\mathbf{u}_{[N,N+d_0-1]} = 0$ and the output horizon to $[0, N+\sigma+d_0-1]$. The desired trajectory, y^d, is given in $[\sigma, N+\sigma-1]$. We set $\mathbf{y}^d_{[N+\sigma,N+\sigma+d_0-1]}$ to some appropriate constants. Further, at every iteration, we set $x^k(0) = x^d(0)$ and $u^k(i) = u^d(i) = 0, N \le i \le N-1+d_0$.

To analyze the stability of the inverse mapping, we need the following assumptions:

- (**A1**) The system is stable, controllable and observable.
- (**A2**) The matrix A is invertible.
- (**A3**) $\beta_n \neq 0$ in $G(z)$.
- (**A4**) The matrix $\mathbf{J}$ is nonsingular.

With these assumptions, Lemma 1 shows that the inverse mapping (2) is stable using the time advancing of the output data, even though it is an NMP system.

Lemma 2.1. *(Stable inversion using time advancing)*
The inverse mapping from $\mathbf{y}^d_{[\sigma+d_0,N+\sigma+d_0-1]}$ *to* $\mathbf{u}^d_{[0,N-1]}$ *is stable.*

The input update law is derived from Lemma 1 as follows:

$$\mathbf{u}^{k+1}_{[0,N-1]} = \mathbf{u}^k_{[0,N-1]} + \mathbf{S}^k \mathbf{e}^k_{[\sigma+d_0,N+\sigma+d_0-1]}, \tag{3}$$

where $\mathbf{e}^k_{[l,m]} = \mathbf{y}^d_{[l,m]} - \mathbf{y}^k_{[l,m]}$ and $\mathbf{S}^k \in \mathbb{R}^{N\times N}$ is the learning gain matrix.

The next lemma shows that the input $\mathbf{u}^k_{[0,N-1]}$ converges to $\mathbf{u}^d_{[0,N-1]}$ as $k \to \infty$ using the input update law (3). It should be noted that this inverse mapping is stable.

Lemma 2.2. *The uncertain system (1) satisfies (**A1**)–(**A4**). If the condition*

$$\|I - \mathbf{S}^k\mathbf{J}\| \le \rho < 1 \tag{4}$$

holds, the input $\mathbf{u}^k_{[0,N-1]}$ converges to $\mathbf{u}^d_{[0,N-1]}$ as $k \to \infty$.

3 ADILC with the estimation of the impulse response

In this section, the impulse response estimation scheme in [5] is slightly modified for the learning speed enhancement detailed in the next section. After estimating the first p impulse responses, we select the first $l \le p$ responses to obtain $\bar{\mathbf{J}}$, the estimate of $\mathbf{J}$ in (2). With $\bar{\mathbf{J}}$, which consists of the estimations of the first l impulse responses $(J_1,\cdots,J_l)$, the ADILC scheme can be applied to unknown NMP systems.

Since we set $x(0) = 0$ for the learning scheme, from (2), we can obtain

$$\mathbf{y}_{[\sigma+d_0,N+\sigma+d_0-1]} = \mathbf{J}\mathbf{u}_{[0,N-1]}. \tag{5}$$

By exchanging the location of $\mathbf{J}$ and $\mathbf{u}_{[0,N-1]}$, (5) can be changed into

$$\mathbf{y}_{[\sigma+d_0,N+\sigma+d_0-1]} = \mathbf{U}_{max}\mathbf{J}_{[1,N+d_0+1]}. \tag{6}$$

Here,

$$\mathbf{U}_{max} = \begin{bmatrix} u(d_0) & \cdots & u(0) & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ u(d_0+N-1) & \cdots & u(N-1) & \cdots & u(0) \end{bmatrix},$$
$$\mathbf{J}_{[1,N+d_0+1]} = [J_1,\cdots,J_{N+d_0+1}]^T. \tag{7}$$

To estimate the first p impulse responses, we make an approximation for $\mathbf{J}_{[1,p]}$. As i becomes larger, the impulse response J_i approaches 0. By selecting a sufficient large p and discarding the impulse responses from $p+1$, the approximation is made as

$$\mathbf{y}_{[\sigma+d_0,N+\sigma+d_0-1]} \approx \mathbf{U}_p\mathbf{J}_{[1,p]}. \tag{8}$$

Here,

$$\mathbf{U}_p = \begin{bmatrix} u(d_0) & u(d_0-1) & \cdots & 0 \\ u(d_0+1) & u(d_0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ u(d_0+N-1) & u(d_0+N-2) & \cdots & u(d_0+N-l-1) \end{bmatrix},$$
$$\mathbf{J}_{[1,p]} = [J_1,\cdots,J_p]^T. \tag{9}$$

Using the least square method, we can obtain $\bar{\mathbf{J}}_{[1,p]}$ consisting of the estimates of $\mathbf{J}_{[1,p]}$, as

$$\bar{\mathbf{J}}_{[1,p]} = (\mathbf{U}_p^T\cdot\mathbf{U}_p)^{-1}\mathbf{U}_p^T\mathbf{y}_{[\sigma+d_0,N+\sigma+d_0-1]}. \tag{10}$$

After estimating the impulse responses, we select the first $l \leq p$ impulse responses and obtain $\bar{\mathbf{J}}$ which is the estimates of $\mathbf{J}$ in (2).

In [5], a learning control scheme was presented based on impulse response estimation. At step 0, $\mathbf{u}^1_{[0,N-1]}$ can be determined by setting $\mathbf{S}^0$ as an appropriate matrix, e.g., αI and $\mathbf{u}^0_{[0,N-1]} = 0$. Then, we can estimate $\bar{\mathbf{J}}^k_{[1,N]}$ similarly to (10) and learning control can be performed using (3) with $\mathbf{S}^k = \alpha(\bar{\mathbf{J}}^k)^{-1}$ for some $\alpha, 0 < \alpha < 1$.

4 Learning speed enhancement using the estimation of the impulse response

In this section, a new learning speed enhancement algorithm is presented using (10) and the learning scheme for unknown NMP systems. Since the estimates of impulse responses can be used to estimate the desired input, the learning speed can be enhanced.

At $k = 0$, since $\mathbf{u}^0_{[0,N-1]} = 0$, $\mathbf{y}^0_{[\sigma+d_0,N+\sigma+d_0-1]}$ will be zero. Thus, $\mathbf{u}^1_{[0,N-1]} = \mathbf{S}^0\mathbf{y}^d_{[\sigma+d_0,N+\sigma+d_0-1]}$. Here, we set $\mathbf{S}^0$ to be an appropriate matrix, e.g., αI. For $k \geq 1$, we estimate the impulse response $\bar{\mathbf{J}}^k_{[1,p]}$ using (10) from $\mathbf{u}^k_{[0,N-1]}$ and $\mathbf{y}^k_{[\sigma+d_0,N+\sigma+d_0-1]}$, derive $\bar{\mathbf{J}}^k$ from the estimates of first l impulse responses.

Likewise, for a sufficient k, e.g., $k = 1$, we can obtain the estimate of the input $\bar{\mathbf{u}}^d_{[0,N-1]}$ as follows:

$$\bar{\mathbf{u}}^d_{[0,N-1]} = (\bar{\mathbf{J}}^k)^{-1}\mathbf{y}_{[\sigma+d_0,N+\sigma+d_0-1]}. \tag{11}$$

If the estimation is successfully made and l is sufficiently enough, $\bar{\mathbf{u}}^d_{[0,N-1]}$ will be considerably close to $\mathbf{u}^d_{[0,N-1]}$.

For $k+1$, e.g., $k = 2$, we can set $\mathbf{u}^k_{[0,N-1]} = \bar{\mathbf{u}}^d_{[0,N-1]}$ and $\mathbf{S} = \alpha(\bar{\mathbf{J}}^k)^{-1}$. If $\mathbf{S}$ satisfies the convergence condition, we can obtain the desired input with the proposed method. Throughout this approach, we can enhance the learning speed. We can summarize the learning rule as follows:

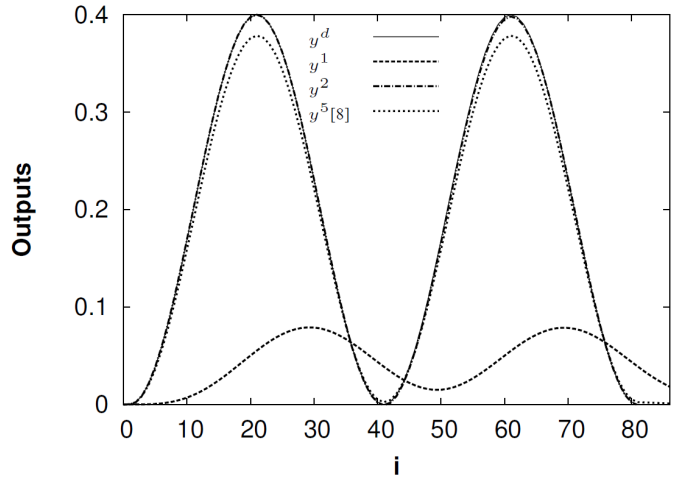
The proposed learning algorithm

- Step 0: When $k = 0$.
- Set $\mathbf{S}^0$ to be an appropriate matrix and obtain $\mathbf{u}^1_{[0,N-1]}$.
- Step 1: For the first iteration,
- Obtain $\mathbf{y}^1_{[\sigma+d_0,N+\sigma+d_0-1]}$.
- If $\|\mathbf{e}^k_{[\sigma+d_0,N+\sigma+d_0-1]}\| \leq \epsilon$, then stop.
- Else, derive $\bar{\mathbf{J}}^k$ using (10).
- Calculate the estimated value of the desired input $\bar{\mathbf{u}}^d_{[0,N-1]}$ from (11) and set $\mathbf{u}^2_{[0,N-1]} = \bar{\mathbf{u}}^d_{[0,N-1]}$.
- Step k: For the k-th iteration,
- If $\|\mathbf{e}^k_{[\sigma+d_0,N+\sigma+d_0-1]}\| \leq \epsilon$, then stop.
- Set $\mathbf{S} = \alpha(\bar{\mathbf{J}}^k)^{-1}$.
- Update the input using (3), increment k, and repeat Step k until termination.

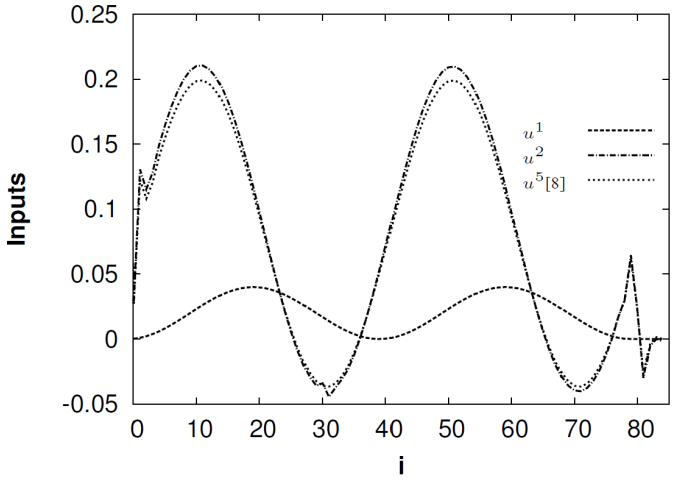
Theorem 1. The NMP system (1) satisfies (**A1**)-(**A4**), the relative degree and the number of NMP zeros are known, and the system dynamics may not be known completely. Let us assume that we update the input based on the proposed learning algorithm.

If the condition (4) holds for all $k \geq 1$, the input $\mathbf{u}^k_{[0,N-1]}$ converges to $\mathbf{u}^d_{[0,N-1]}$ as $k \to \infty$.

Proof: This can be easily shown using Lemma 2.1, and Theorem 1 from [5]. □



(a) Outputs using the proposed method and y^5 in [5]



(b) Inputs using the proposed method and u^5 in [5]

Figure 1: Outputs and inputs for different values of k

5 Simulation results

Let us consider an example of NMP system for a positioning table in [10] as follows:

$$G(z) = \frac{0.0082z^4 + 0.031z}{(z + 0.29)(z - 0.2)(z - 0.46)(z^2 - 1.7z + 0.73)}$$

This system has one NMP zero ($z = -3.7805$) and satisfies (**A1**)–(**A4**). The desired trajectory is given as

$$y^d(i) = \begin{cases} 0, & i = 0, 1, 83, 84, 85, 86 \\ -0.2\cos(0.05\pi(i-2)), & 2 \le i \le 82. \end{cases} \tag{12}$$

Here, we set $N = 85$, $\mathbf{S}^0 = 0.1I$ and $u(85) = u(86) = 0$. The input update law is given as $\mathbf{u}^{k+1}_{[0,84]} = \mathbf{u}^k_{[0,84]} + \mathbf{S}^k\mathbf{e}^k_{[2,86]}$. From (10), we set $p = 85$. The impulse response is estimated using

$$\bar{\mathbf{J}}^k_{[1,85]} = ((\mathbf{U}^k)^T \cdot \mathbf{U}^k)^{-1}(\mathbf{U}^k)^T\mathbf{y}^k_{[2,86]}. \tag{13}$$

In addition, $\bar{\mathbf{J}}^1$ is obtained using $l = 30$. Then, we set $\mathbf{u}^2 = \bar{\mathbf{u}}^d$ and enhance the learning speed. In this case, the convergence condition is satisfied as $\|I - \mathbf{S}^k\mathbf{J}^k\| < 0.568$ when $\mathbf{S}^k = 0.5(\bar{\mathbf{J}}^k)^{-1}$.

Fig. 1(a) and 1(b) show the outputs and inputs for different values of k, respectively. The root mean square (RMS) error for the output error is 0.0012 for $k = 2$, and is smaller than the RMS error of 0.0036 for $k = 10$ reported in [5]. From this example, we can see that the learning speed is significantly enhanced.

6 Conclusion

In this paper, we have proposed a new learning speed enhancement algorithm of ADILC for discrete-time NMP systems. First, we have presented an estimation algorithm of impulse responses based on the input-output mapping of ADILC. Next, learning speed enhancement algorithm has been derived from new learning gain, which is calculated with the estimates of impulse response. Simulation results for the NMP system have demonstrated the learning speed enhancement of the proposed method.

Robust algorithms over disturbances for learning control can be considered with the proposed method. It remains as a future work.

Acknowledgment

This work was supported by the National Research Foundation of Korea(NRF) Grant funded by the Korean Government(MSIP)(NRF-2016R1A5A1012966), and also supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2015R1D1A1A01060917)

Bibliography

[1] Arimoto S., Kawamura S., Miyazaki F. (1984); Bettering operation of robots by learning, *Journal of Robotic Systems*, 1(2), 123–140, 1984.

[2] Bien Z., Xu J.-X. (1998); *Iterative learning control analysis, design, integration and applications*, Kluwer Academic Publishers, 1998.

[3] Jang T.-J., Ahn H.-S., Choi C.-H. (1994); Iterative learning control for discrete-time nonlinear systems, *International Journal of Systems Science*, 25(7): 1179-1189..

[4] Jeong G.-M., Choi C.-H. (2002); Iterative learning control for linear discrete time nonminimum phase systems, *Automatica*, 38(2), 287–291, 2002.

[5] Jeong G.-M., Ji S.-H. (2013); Iterative learning control with advanced output data using an estimation of the impulse response, *IEICE Transactions on Fundamentals*, E96-A (6), 1488-1491, 2013..

[6] Ngo T., Wang Y., Mai T.L., Ge J., Nguyen M.H., Wei S. N. (2012); An adaptive iterative learning control for robot manipulator in task space, *International Journal of Computers Communications & Control*, 7(3), 518–529, 2012.

[7] Uchiyama M. (1978); Formulation of high-speed motion pattern of mechanical arm by trial, *Transactions of the Society of Instituteument and Control Engineers* (in Japanese), 14(6), 706–712, 1978.

[8] Xia C., Deong W., Shi T., Yan Y. (2016); Torque ripple minimization of PMSM using parameter optimization based iterative learning control, *Journal of Electrical Engineering and Technology*, 11(2), 709–718, 2016.

[9] Xu J.-X. (1997); Direct learning of control efforts for trajectories with different magnitude scales, *Automatica*, 33(12), 2191–2195, 1997.

[10] Yamada M., Riadh Z., Funahashi Y. (1999); Design of discrete-time repetitive control system for pole placement and application,*IEEE/ASME Transactions on Mechatronics*, 4(2), 110-118, 1999.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 330-346, June 2017.

Walking Motion Generation and Neuro-Fuzzy Control with Push Recovery for Humanoid Robot

P.E. Mendez-Monroy

Paul Erick Mendez-Monroy
IIMAS - Universidad Nacional Autonoma de Mexico
Parque CientĂfico y Tecnologico de Yucatan
Km. 5.5 Carretera Sierra Papacal - Chuburna
C.P. 97302 Sierra Papacal, Yucatan
erick.mendez@iimas.unam.mx

Abstract: Push recovery is an essential requirement for a humanoid robot with the objective of safely performing tasks within a real dynamic environment. In this environment, the robot is susceptible to external disturbance that in some cases is inevitable, requiring push recovery strategies to avoid possible falls, damage in humans and the environment. In this paper, a novel push recovery approach to counteract disturbance from any direction and any walking phase is developed. It presents a pattern generator with the ability to be modified according to the push recovery strategy. The result is a humanoid robot that can maintain its balance in the presence of strong disturbance taking into account its magnitude and determining the best push recovery strategy. Push recovery experiments with different disturbance directions have been performed using a 20 DOF Darwin-OP robot. The adaptability and low computational cost of the whole scheme allows is incorporation into an embedded system.
Keywords: push recovery, neuro-fuzzy systems, reinforcement learning, biped walking.

1 Introduction

With the recent growth of humanoid robots performing tasks within the human environment, research has focused on improving their movement and design for stable bipedal walking. Where dynamic walking is a general technique with predefined trajectories and feedback control to ensure that the robot is always rotating around a point in a base of support (BoS). To achieve dynamic walking the centre of gravity (COG) can be outside of the BoS of the robot, but the zero momentum point (ZMP) cannot. The ZMP is the point where the total angular momentum in the ankle is zero and the COG is the projection of the centre of mass (COM) on the ground, inside the BoS established by the feet. These concepts can be associated through an inverted pendulum model as the simplest model. But this technique is susceptible to external disturbance and irregularities in the surface, it requires strategies to maximise the balance and minimise the adverse effects of these disturbances.

The aim of push recovery is to perform an action that counteracts an external disturbance to prevent the robot from falling. The problem of push recovery has been researched using several approaches. Hyon *et al.* [5] use force-controlled actuators and a complete dynamic model to reject external disturbances. Park *et al.* [10] also use force-controlled actuators with optimisation algorithms for active balancing. Nevertheless, the requirements for multi-axis force sensors, an accurate model and intensive computing are their disadvantages.

Other research considers biomechanical movements using the knowledge of human behaviour in responses of unexpected disturbances with simple movements. [11], [17] [7]. Generally, these strategies are classified into three groups: Ankle, hip and stepping.

Some researchers have presented their work in push recovery when a robot walking in place. Pratt *et al.* [11] introduced the concept of "capture point" to determine where the foot should step after being pushed to return to its original pose. Pratt *et al.* [12] extended the concept to multiple steps.

The use of learning strategies to adapt the push recovery has increased in the last years. Rebula *et al.* [14] used capture point learning to improve simple step push recovery. While Missura *et al.* [8] proposed online learning to define the amplitude of the leg depending on the robot posture and the error in the desired step.

Some researchers have integrated several methods to complement the simple strategies for this level of complexity. Stephens [17] combines hip and ankle strategies with the purpose of balance control by defining a larger area return of a push and he established theoretical analytic bounds for both strategies. Hyon *et al.* [5] presented a multi-level posture balance method for humanoid robots and demonstrated push recovery with a biped SARCOS pushed from behind. Semwal *et al.* [15] used a hierarchical fuzzy control to decide by several push recovery strategies and used a simulated robot with good performance and low-cost computation but without online learning obtaining the same boundaries for each push recovery strategy as an analytic method.

However, few researchers have presented push recovery strategies in the walking phase, in order to continue the walking before a push. Komura *et al.* [6] proposed a feedback controller for the biped walking and applied a hip strategy as push recovery in a simulated robot. Wieber *et al.* [19] used a predictive control model to minimise the jerk and ZMP error, this improved the robustness to disturbances in walking.

The objective of this work is to design a practical strategy according to external disturbances using a neuro-fuzzy system to define the best push recovery behaviour to counteract the disturbance. The novel push recovery approach can maintain the robot walking, counteracting pushes from any direction and in any phase with a certain magnitude of disturbance. This scheme has the advantages of using a simple model and a controller of low computational cost. Furthermore, the design is applicable to generic robots with position controlled actuators, inertial measurement unit and joint position sensors.

The remainder of the paper proceeds as follows. Section 2.1 reviews three biomechanical push recovery controllers and their implementation on position-controlled actuators. Section 2.1 explains the details of the neuro-fuzzy system, structure, design and estimation. Section 2.1 describes the reinforcement learning process with a modified Levengerg-Marquardt algorithm. Section 2.1 explains the walking pattern. Section 6 shows the experimental results using the neuro-fuzzy system with reinforcement learning. The paper is finished with some concluding remarks and future work.

2 Biomechanical push recovery

Biomechanical studies of human walking have shown that humans perform three different movement patterns as recovery strategies (ankle, hip, step) to reject a sudden external disturbance. The ankle strategy modulates torque at the ankle joint to keep humans upright, the hip strategy converts unpredicted linear momentum into angular motion at the torso, and the stepping strategy moves the swing foot in the direction of the disturbance. Recent work has focused on using such biomechanical strategies [12] [17]. The next review of these biomechanical strategies assumes simplified physical models of the robot and explains how they can be implemented on real humanoid robots with position controlled actuators.

The ankle strategy maintains the COG in the BoS by applying torque to the ankle joints. For position controlled actuators, the torque at the ankle can be modified either by controlling the ZMP or modulating the target angle of the ankle servo. In this approach, the latter is used

as the strategy of push recovery in the ankle.

$$\delta_{zmp} = -K_a x_{a,swing} \tag{1}$$

where K_a is the ankle feedback gain and the $x_{a,swing}$ is the current position of the swinging ankle.

The hip strategy applies angular acceleration in the torso to generate a reaction force that returns the COG to the BoS from a elapse time $\{T_0, T_1\}$, decelerate in time $\{T_1, T_2\}$ and revert to the initial position slowly $\{T_2, T_3\}$, this control called "bang-bang" can be represented for position controlled actuators as

$$\delta\theta_t = \begin{cases} K_h & 0 \leq t < T_2 \\ K_h \frac{T_3 - t}{T_3 - T_2} & T_2 \leq t < T_3 \end{cases} \tag{2}$$

where $\delta\theta_t$ is the torso position bias, T_3 the time to complete the control, and K_h is the hip feedback constant.

For even larger disturbances, no amount of body torquing will result in push recovery. In this case, it is necessary to take a step. The stepping strategy moves the BoS by taking a step. This strategy looks for the point on the ground where the robot can step to bring it to its original speed or even full stop [17]. The new step (walking phase) or modified step (stance phase) is calculated using the relative target foot position x_{step} [1].

$$x_{step} = \dot{\theta}_x \frac{\sqrt{z_c}}{g} = K_s \tag{3}$$

The definition of a real analytic boundary to apply each strategy is difficult without the exact knowledge of the current system state as well as the kinematic and mechanic restrictions. In addition, its design becomes impractical if some of the parameters, such as the weight or COG change. Instead, the problem is formulated through learning techniques and the parameters are obtained online using past experiences. Some learning strategies have been performed in other robots [3], [18], where the aim is to learn a policy that maps robot states and establish appropriate actions.

3 Neuro-fuzzy system for push recovery

The push recovery is one of the most complex tasks for humanoid robots and it requires an adaptive intelligent controller, The push recovery method is achieved using simulation, the manipulation of the real system and with expert knowledge on intuitive biped push recovery. Thus, a neuro-fuzzy system is proposed to select the appropriate control parameters for the three strategies mentioned above. The selection uses the current walking state and the torso position. The fuzzy controller is designed using system constraints (eg. maximum torque for the actuators, maximum angles for the joints, maximum step size, etc.), expert knowledge and is updated using new reinforcement training with a low computational cost.

Figure 1 shows the architecture of the neuro-fuzzy system with an "actor-critic" reinforcement learning. The strategy is divided in the sagittal and the coronal planes but, only the sagittal plane will be described as the design for the coronal plane is similar. The controller determines a proper action of push recovery in the presence of an external disturbance using the current states $\{\theta_t, \dot{\theta}_t\}$ and the internal reinforcement signal $\hat{\hat{r}}$ from the predictor.

The scheme separates the push recovery system from the motion generator to simplify the design. Thus, the motion generator performs its normal movements when there is no external

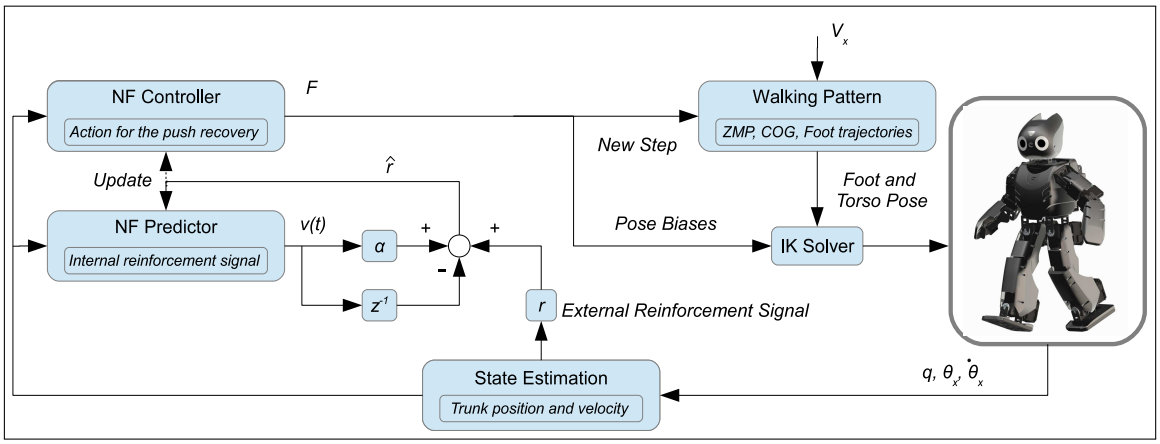


Figure 1: The neuro-fuzzy reinforcement structure

disturbances or unbalance. As soon as an external disturbance or unbalance occurs the neuro-fuzzy controller will establish the best learning strategy to counteract its effect.

The neuro-fuzzy controller uses the trunk position θ_t and its angular velocity $\dot{\theta}_t$ to define the feedback gains for the ankle, hip and stepping strategies. It is initialised offline using simulation, expert knowledge and the robot constraints (limit of the actuator torques, maximum step size, etc.) and it learns using reinforcement learning with low computational cost.

The neuro-fuzzy predictor has the same structure as the controller but with the internal reinforcement signal as the output. Its objective is to evaluate the performance of the whole neuro-fuzzy system using the external reinforcement signal (Reward) to update the parameters of the controller and predictor. Learning is performed by a modified Levenberg-Marquardt algorithm.

In the state estimation, the trunk position and angular velocity in the roll and pitch are estimated based on the accelerometer and gyroscope measurements and mixing with the measured joint angles (q) to use in the forward kinematic equations.

A walk pattern generator is presented to modify the walking trajectory according to the adopted push recovery strategy.

3.1 Neuro-fuzzy controller

The neuro-fuzzy controller (*"actor"*) can determine a better action for the next time instant based on the current system state and the internal reinforcement signal. The internal reinforcement signal from the predictor enables both the controller and the predictor to learn without waiting for external reinforcement, which may only be available at a time long after a sequence of actions has occurred. This scheme can speed up the learning of the complete system.

The fuzzy structure comprises five principal components: inputs, antecedent labels, rules, normalisation and consequent labels. Moreover, at the computational level, a fuzzy system can be seen as a feedforward neural network with five layers. Thus, The fuzzy structure is defined in five layers with node's basic functions for an online training.

Each layer has a input vector $\{u^k_{1\ldots p}\}$, a parameter vector $\{w^k_{1\ldots p}\}$ and a output vector $\{o^k_{1\ldots q}\}$ defining a activation function.

$$u^k_i = f\left(u^k_1, u^k_2, \cdots, u^k_p, w^k_1, w^k_2, \cdots, w^k_p\right) \tag{4}$$

$$o^k_j = a_j(u^k_i) \quad where\ a(\cdot)\ is\ the\ activation\ function \tag{5}$$

where $k = \{1, ..., 5\}$ is the kth layer, with p inputs and parameters for the jth output of the layer.

The next paragraph describes the functions of the nodes in each of the five layers of the proposed neuro-fuzzy controller with $X = \{x_1, ..., x_{in}\}$ inputs, $Y = \{y_1, ..., y_{out}\}$ outputs with r defined fuzzy rules like:

$$R_j : \textbf{if } x_1 \textit{ is } \mu_{1,j} \textit{ and ... and } x_{in} \textit{ is } \mu_{in,j} \textbf{ then } y_1 \textit{ is } \nu_{1,j} \textit{ and ... and } y_{out} \textit{ is } \nu_{out,j} \tag{6}$$

where μ_{ij} and ν_{lj} are fuzzy sets, the indexes $i = \{1, ..., in\}$ inputs, $l = \{1, ..., out\}$ outputs and $j = \{1, ..., r\}$ fuzzy rules.

Layer 1 (Inputs): The nodes in this layer transmit the inputs directly to the next layer (the vector parameters has unity values $w_i^1 = 1 \ \forall i$) eq.(7).

Layer 2 (Antecedents): Each single node to perform a membership function. The Gaussian membership function is used in this work eq.(8). where m_{ij}^2 and σ_{ij}^2 correspond to the centre and the width of the Gaussian function of the jth rule with the ith input variable.

Layer 3 (Rules): Implements the conjunction of all or some antecedent conditions in the jth rule. Hence, the rule nodes should perform the fuzzy "*and*" operation, in this case, the product function is used, in fuzzy theory, it is called *fire strength* eq.(9).

Layer 4 (Normalised): The number of nodes in this layer is equal to that of the rules layer. The effect of the layer is to perform a normalisation on each rule. It is called *normalised fire strength* for the jth rule eq.(10).

Layer 5 (Consequent): This layer will have as many nodes as there are output variables. Each output node combines the normalised fire strength with the fuzzy action and the defuzzification is computed with the centre of area method eq.(11) [4].

$$f_i^1 = u_i^1 \quad and \quad a_i^1 = f_i^1 \quad i = 1, 2, ..., in \tag{7}$$

$$f_{ij}^2 = \mu_{ij}\left(a_i^1;\ m_{ij}^2,\ c_{ij}^2\right) = -\frac{\left(a_i^1 - m_{ij}^2\right)^2}{\left(\sigma_{ij}^2\right)^2} \quad and \quad a_{ij}^2 = exp(f_{ij}^2) \tag{8}$$

$$f_j^3 = \prod_{i=1}^{2} a_{ij}^2 \quad and \quad a_j^3 = f_j^3 \quad w_{ij}^3 = 1 \ \forall i \tag{9}$$

$$f_j^4 = \frac{a_j^3}{\sum_{j=1}^{r} a_j^3} \quad and \quad a_j^4 = f_j^4 \quad where \quad \sum_{l=1}^{r} a_j^4 = 1 \tag{10}$$

$$f_l^5 = \frac{\sum_{j=1}^{r}(m_{lj}^5 \sigma_{lj}^5) a_j^4}{\sum_{j=1}^{r} \sigma_{lj}^5 a_j^4} \quad and \quad a_l^5 = f_l^5 \tag{11}$$

The complete fuzzy system F with fuzzy rules (7) is:

$$F = \{F_l,\ l = 1, ..., out\}, \quad F_l = \frac{\sum_{j=1}^{r} \nu_{lj}^{-} \prod_{i=1}^{in} \mu_{ij}(x_i)}{\sum_{j=1}^{r} \prod_{i=1}^{in} \mu_{ij}(x_i)}, \quad \nu_{lj}^{-} = \frac{\sum_{j=1}^{r}(m_{lj}^5 \sigma_{lj}^5) a_j^4}{\sum_{j=1}^{r} \sigma_{lj}^5 a_j^4} \tag{12}$$

where ν_{lj}^{-} is the centre of the lth output fuzzy set and the jth fuzzy rule. Thus, F is the fuzzy control action with the parameter vector $p_c = \{m_{ij}^2,\ \sigma_{ij}^2,\ m_{lj}^5,\ \sigma_{lj}^5\}$.

The parameter vector $p_c = \{m_{ij}^2,\ \sigma_{ij}^2,\ m_{lj}^5,\ \sigma_{lj}^5\}$ of the all weights in the controller is updated by minimising the internal reinforcement signal $\hat{r}$, such that, the system ends up with a good recovery strategy and avoids failure. The learning rule used to adjust the parameter vector is

a gradient-based adaption δp_c and Levenberg Marquardt with the objective function E_c to be minimised in the controller.

$$\delta p_c = \eta_c \frac{\partial E_c}{\partial p_c} = \eta_c \frac{\partial E_c}{\partial v} \frac{\partial v}{\partial F} \frac{\partial F}{\partial p_c}, \qquad E_c(t) = \frac{1}{2} v^2(t) \tag{13}$$

where η_c is the learning rate factor and v is the prediction of the external reinforcement signal.

3.2 Neuro-fuzzy predictor

To update a neuro-fuzzy system supervised learning [4] is the most common strategy used. These schemes need accurate training data to indicate the correct desired output to compute the training error. But, in the case of biped robots is difficult to get the correct output for each push recovery strategy because is a dynamic system with simplifications and uncertainties in the model and controller.

However, most of the real systems can provide a right-wrong binary decision (reinforcement signal) based on the current control, making a reinforcement training a better option to learn. Thus, a neuro-fuzzy predictor (*"critic"*) is used to determinate the parameter update using the external reinforcement signal r and to predict an internal reinforcement signal $\hat{r}$. The predictor structure is the same as the controller but only with the internal reinforcement signal like output.

The learning algorithm will be applied to the fuzzy controller and the fuzzy predictor simultaneously and only conducted by an external reinforcement feedback (R) using an orbital energy.

$$R = \frac{\dot{\theta}_x^2 + w^2 \theta_x^2}{2} \tag{14}$$

This is the "critic" information available for learning and indicates whether the output is right or wrong. In this paper, R is mapped to a range $r = \{-1 \rightarrow failure, 1 \rightarrow success\}$ to get a continuous reinforcement signal. Thus, the predictor output v approximates the discounted total reward-to-go $\tilde{R}(t) = r(t+1) + \alpha r(t+2)...$ [16]. It is the future accumulative reward-to-go, α is a discount factor for the infinite-horizon problem and $r(t+1)$ is the external reinforcement.

Defining the prediction error E_p and the objective function to be minimised as

$$e_p(t) = \alpha v(t) - v(t-1) - r(t) \qquad E_p(t) = \frac{1}{2} e_p^2(t) \tag{15}$$

Therefore, the parameter vector $p_p = \{m_{ij}^{2p},\ \sigma_{ij}^{2p},\ m_j^{5p},\ \sigma_j^{5p}\}$ is updated using a a Levenberg Marquardt algorithm and gradient-based update rule given by

$$\delta p_p = \eta_p \frac{\partial E_p}{\partial p_p} = \eta_p \frac{\partial E_p}{\partial e_c} \frac{\partial e_c}{\partial v} \frac{\partial v}{\partial p_p} \tag{16}$$

3.3 State estimation

State estimation uses the joint angles measured by the encoders, the inertial acceleration and angular velocity of the trunk measured by the IMU and a forward kinematic model to reconstruct the whole-body pose of the robot. With the reconstruction of the whore-body pose is obtained the current COM and which leg is the standing in.

By fusing the gyroscope and accelerometer data is estimated the trunk position. The trunk position $\hat{\theta}_t = (\theta_{tx}, \theta_{ty})$ is expressed using the inertial acceleration vector $a = (a_x, a_y) \in [-1, 1]$

measured by the accelerometer, first, a raw angle estimate is obtained. Then, it is mixed with the angular velocity in the n iteration of the control loop (eq. 19).

$$rl\dot{\theta}_{tx} = arcsin(a_x) \tag{17}$$

$$\dot{\theta}_{ty} = arcsin(a_y) \tag{18}$$

$$\theta_n = (1-\kappa)(\theta_{n-1} + \rho(\hat{\dot{\theta}}_n - b_n)) + \kappa\hat{\theta}_n \tag{19}$$

where $\hat{\dot{\theta}}_n$ is the raw angular velocity, κ is a blending factor, ρ is the control time, and b_n is the gyro drift. There are other more advanced options for mixing the gyroscope and accelerometer measurements [2].

On the other hand, the measurements of the joints (q) obtained through the encoders of each motor are applied to a direct kinetic algorithm to get the current pose. This kinematic model is rotated around the centre of the supporting foot such that the trunk position equals the roll and pitch angles $\theta_q = (\theta_{tx}, \theta_{ty})$ obtained in equation (19).

Thus, the difference between the trunk angles achieved by the IMU and the joint angles is used by the neuro-fuzzy controller.

$$\begin{aligned} \Delta\theta &= \theta_t - \theta_q \\ \dot{\theta} &= \hat{\dot{\theta}} \end{aligned} \tag{20}$$

4 Reinforcement Levenberg-Marquardt learning

As mentioned earlier, a neuro-fuzzy system is used in both the controller and the predictor where their parameters are updated using reinforcement learning. According to eq.(13) and eq.(15), it is possible use a modification of the Levenberg-Marquardt algorithm [13] to speed up the update to the neuro-fuzzy system. The parameters $p = \{p_c, p_p\}$ are updated using normalisation in both networks to confine the parameter values into some appropriate range (21).

$$p(t+1) = \frac{p(t) + \delta p(t)}{\|p(t) + \delta p(t)\|_1} \quad with \quad \delta p(t) = -(H(p))^{-1} J^T(p) e(p) \tag{21}$$

where approximation of the Hessian matrix H and the Jacobian matrix is calculated using the backpropagation algorithm.

$$H(p) = J^T(p)J(p) + \mu I \quad with \quad J = [\frac{\partial E}{\partial p_1}, \frac{\partial E}{\partial p_2}, ..., \frac{\partial E}{\partial p_N}] \tag{22}$$

where p_0 are the initial parameter values for the controller and predictor, ε_1, ε_2 and ε_3 are specific stop criteria in the LM algorithm, k_{max} is the maximum iteration number and ε_4 is the controller or predictor error prediction. E_{new} is the value of the objective function evaluated with the new parameters.

The Jacobian matrix in eq.(22) for the controller is formed by partial derivatives in eq.(13), they are calculated as

$$\frac{\partial E_c}{\partial v} = v \qquad \frac{\partial v}{\partial F} \approx \frac{dv}{dF} \approx \frac{v(t) - v(t-1)}{F(t) - F(t-1)} \tag{23}$$

the term $\partial v/\partial F$ in eq.(13) is indirectly dependent on F and complex computation, so that an approximation can be computed by the instantaneous difference ratio. The other term $\partial F/\partial P_c$ is more tractable. So that, with F known and differentiable, a few applications of the chain rule

Algorithm 1 The modified Levenberg-Marquardt algorithm

Data: p_c or p_p, E_c or E_p, e_c or e_p
Result: p_{new}
$k := 0$; $v := 2$; $p = p_o$
$H := J^T J$; $g := J^T e\,(k)$
stop:=$(\| g \|_\infty \leq \varepsilon_1)$; $\lambda = \tau \cdot max\,(diag\,(H))$
while (not stop) and $(k < k_{max})$ and $(E\,(k) > \varepsilon_4)$ **do**
  $h = -\,(H + \lambda I)^{-1}\, g$
  if $(\| h \| \leq \varepsilon_2 \| p \|)$ **then**
    stop=true
  else
    $p_{new} = p + h$
    $\rho := \frac{\|E(k)\|^2 - \|E_{new}\|^2}{h^T(\mu h + g)}$
    if $\rho > 0$ **then**
      $p = p_{new}$
      $H := J^T J$; $g := J^T e\,(k)$
      stop=$(\| g \|_\infty \leq \varepsilon_1)$ or $\left(\| e\,(k) \|^2 \leq \varepsilon_3\right)$
      $\mu = \mu \cdot max\left(1/3, 1 - (2\rho - 1)^3\right)$; $v = 2$
    else
      $\mu = \mu \cdot v$; $v = 2v$
    end if
  end if
end while

through the five layers of the controller give the following set of learning rules.

$$\begin{aligned}
\frac{\partial F}{\partial m_{lj}^5} &= \frac{\partial F}{\partial a_j^5}\frac{\partial a_j^5}{\partial f_j^5}\frac{\partial f_j^5}{\partial m_{lj}^5} = \left(f_l^5\right)\left(1\right)\left(\frac{\sigma_{lj}^5 a_j^4}{\sum_{j=1}^r \sigma_{lj}^5 a_j^4}\right) \\
\frac{\partial F}{\partial \sigma_{lj}^5} &= \frac{\partial F}{\partial a_j^5}\frac{\partial a_j^5}{\partial f_j^5}\frac{\partial f_j^5}{\partial \sigma_{lj}^5} = \left(f_l^5\right)\left(1\right)\left(\frac{\left(\sum_{j=1}^r \sigma_{lj}^5 a_j^4\right) m_{lj}^5 a_j^4 - \left(\sum_{j=1}^r m_{lj}^5 \sigma_{lj}^5 a_j^4\right) a_j^4}{\left(\sum_{k=1}^r \sigma_{lj}^5 a_j^4\right)}\right)
\end{aligned} \tag{24}$$

The equations in (24) are the partial derivatives to update the consequent parameter values $p_{lj}^5 = \{m_{lj}^5,\ \sigma_{lj}^5\}$.

The antecedent parameter values $p_{ij}^2 = \{m_{ij}^2,\ \sigma_{ij}^2\}$ are updated when the error is propagated to the preceding layers as

$$\begin{aligned}
\frac{\partial F}{\partial p_{ij}^2} &= \frac{\partial F}{\partial a_j^5}\frac{\partial a_j^5}{\partial f_j^5}\frac{\partial f_j^5}{\partial p_{ij}^2} = \frac{\left(\sum_{j=1}^r \sigma_{lj}^5 a_j^4\right)\left(m_{lj}^5 a_j^4\right)\frac{\partial a_j^4}{\partial p_{ij}^2} - \left(\sum_{j=1}^r m_{lj}^5 \sigma_{lj}^5 a_j^4\right)\left(\sigma_{lj}^5\right)\frac{\partial a_j^4}{\partial p_{ij}^2}}{\left(\sum_{k=1}^r \sigma_{lj}^5 a_j^4\right)^2} \\
\frac{\partial a_j^4}{\partial p_{ij}^2} &= \frac{\partial a_j^4}{\partial f_j^4}\frac{\partial f_j^4}{\partial a_j^3} = \frac{\sum_{j=1}^r a_j^3 \frac{\partial a_j^3}{\partial p_{ij}^2} - a_j^3 \frac{\partial a_j^3}{\partial p_{ij}^2}}{\left(\sum_{k=1}^r a_j^3\right)^2} \\
\frac{\partial a_j^3}{\partial p_{ij}^2} &= \frac{\partial a_j^3}{\partial f_j^3}\frac{\partial f_j^3}{\partial a_{ij}^2}\frac{\partial a_{ij}^2}{\partial f_{ij}^2}\frac{\partial f_{ij}^2}{\partial p_{ij}^2} = \left(\prod_{k \neq i} a_{kj}^2\right)\left(exp(f_{ij}^2)\frac{\partial f_{ij}^2}{\partial p_{ij}^2}\right)\left(\frac{2\left(a_i^1 - m_{ij}\right)}{\left(\sigma_{ij}^2\right)^2}\right)\left(\frac{2\left(a_i^1 - m_{ij}\right)^2}{\left(\sigma_{ij}^2\right)^3}\right)
\end{aligned} \tag{25}$$

In the case of the predictor, the partial derivatives are very similar to those of the controller. Where the differential is:

$$\frac{\partial E_p}{\partial e_p} = e_p \qquad \frac{\partial e_p}{\partial v} = \alpha v \tag{26}$$

The partial derivative $\partial v/\partial p_p$ is calculated as being equal to $\partial f_l^5/\partial p_c$.

5 Walking pattern

The walking pattern generates trajectories from the design parameters (mode, sway amplitude, step time, direction) in real time. The parameters are modified at the start of each step.

5.1 ZMP and COG trajectories

The objective of defining the trajectories of the zero moment point (ZMP) and the centre of gravity (COG) is to guarantee the balance of the robot in open loop, keeping these trajectories within the base of support (BoS). The trajectory is defined in the sagittal plane, the trajectory in the coronal plane is designed in a similar way.

In the case of the ZMP, a dynamic ZMP can be represented with two equations in axes X-Y, using the simple inverted pendulum model (figure 2) and assuming a constant movement z_c of the pelvis in the transverse plane, the equation (27) is simplified. In the case of the COG, it is assumed to be identical to the centre of the pelvis.

Thus, the ZMP equation in the sagittal plane is represented as

$$x_{zmp} = \frac{\sum_{i=1}^{n} m_i(\ddot{z}_i + g)x_i - \sum_{i=1}^{n} m_i\ddot{x}_i z_i - \sum_{i=1}^{n} I_{iy}\ddot{\omega}_{iy}}{\sum_{i=1}^{n} m_i(\ddot{z}_i + g)} \approx x_{cog} - \ddot{x}_{cog}\frac{l}{g} \tag{27}$$

where m_i is the mass, x_i, z_i is the mass position, I_i is the inertia moment and ω_i is the angular velocity. The first term (x_{cog}) is the static component and the second term ($\ddot{x}_{cog}$) is the dynamic component of the ZMP.

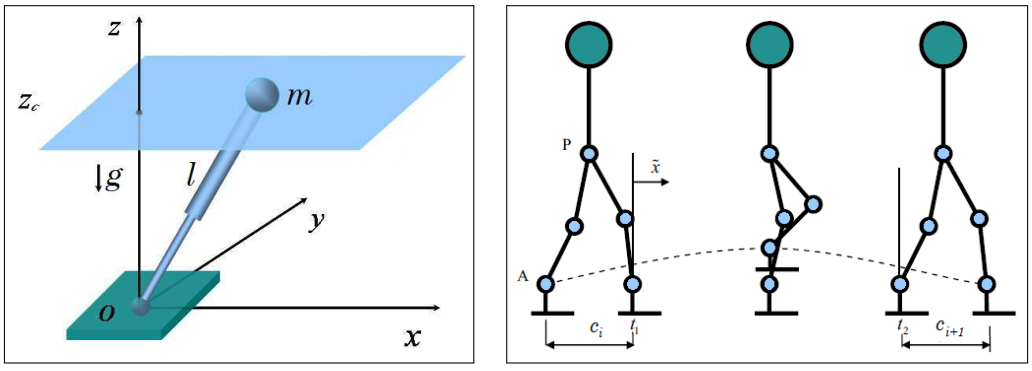


Figure 2: (Inverted pendulum model (*left*); Step parameters in the x direction (*right*)

In this paper a third-order polynomial interpolation is used to obtain the desired ZMP trajectory. A polynomial can be used to generate a trajectory in a straightforward and direct form. Moreover, the COG trajectory is designed with this trajectory.

The polynomials in eq.(28) represent the trajectories of position and velocity of the ZMP in the sagittal plane setting a normalised time to start $t_n = 0$ and end $t_n = 1$ of the step [10] without loss of generality.

$$x_{zmp}(t_n) = \sum_{t=0}^{3} b_i t_n^i \qquad \dot{x}_{zmp}(t_n) = \sum_{t=1}^{3} i b_i t_n^{i-1} \qquad b = \Omega^{-1} x_{zmp}^{boundary} \tag{28}$$

The boundary conditions $\{x_{zmp}(0),\ \dot{x}_{zmp}(0)\}$ and $\{x_{zmp}(1),\ \dot{x}_{zmp}(1)\}$ of the lifting and landing of the ZMP in each step define the polynomial coefficients in eq.(28) to get a smooth ZMP trajectory:

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} x_{zmp}(0) \\ \dot{x}_{zmp}(0) \\ x_{zmp}(1) \\ \dot{x}_{zmp}(1) \end{bmatrix} \qquad \Omega b = x_{zmp}^{boundary} \tag{29}$$

By assuming that the COG x_{cog} is located to the centre of the pelvis, it is defined with the same form as (29) but with different coefficients. The boundary conditions for the COG $\{x_{cog}(0),\ \dot{x}_{cog}(0),\ x_{cog}(1),\ \dot{x}_{cog}(1)\}$ are used to design the polynomial coefficients a_i.

$$x_{cog}(t_n) = \sum_{t=0}^{3} a_i t_n^i \quad \dot{x}_{cog}(t_n) = \sum_{t=1}^{3} i a_i t_n^{i-1} \quad a = \Omega^{-1} x_{cog}^{boundary} \tag{30}$$

By substituting the x_{cog} and $\dot{x}_{cog}$ in eq.(30) and x_{zmp} in eq.(27), a direct match between ZMP and COG is obtained:

$$\sum_{t=0}^{3} b_i t_n^i = \sum_{t=0}^{3} a_i t_n^i - \left(\frac{l}{g}\right) \sum_{t=1}^{3} i a_i t_n^{i-1} \tag{31}$$

The relationship between the boundary conditions allows defining all the coefficients of the polynomials of the ZMP and the COG using (31) in a matrix form $b = \Gamma a$, where Γ is the coefficient matrix.

$$x_{zmp}^{boundary} = \Phi \Gamma \Omega^{-1} x_{cog}^{boundary} \qquad with \quad \Gamma = \begin{bmatrix} 1 & 0 & -2\frac{l}{g} & 0 \\ 0 & 1 & 0 & -6\frac{l}{g} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{32}$$

In the sagittal plane, each step can be defined by the amplitude of the feet during the double support phase before and after the single support phase (figure 2). The boundary conditions of COG are defined by with these parameters.

$$\begin{bmatrix} x_{cog}(0) & \dot{x}_{cog}(0) & x_{cog}(1) & \dot{x}_{cog}(1) \end{bmatrix}^T = \begin{bmatrix} -0.5c_i & \alpha c_i & 0.5c_{i+1} & \alpha c_{i+1} \end{bmatrix}^T \tag{33}$$

where $S_x = c_i + c_{i+1}$ is the length stride in X.

To define this trajectory, it is assumed that the position of the pelvis during a step is in the centre of the feet. The boundary conditions are parametrized by a factor α to modify the pelvis velocity. This allows acceleration at the start and end of the step and deceleration when the leg is swinging with a high value of α.

By establishing trajectories with a direct relationship, and defining boundary conditions based on the step parameters, it is simpler to define trajectories and modify them in the presence of a disturbance.

5.2 Ankle trajectory

Once the COG and ZMP trajectories are defined, it is necessary to identify the trajectories of the ankles (*support* and *swing*) according to several movement conditions, which complete the pattern.

The swinging foot trajectory ($\hat{x}_{a,swing}$) is defined by a cycloid function. It has zero velocity at the start and the end, and a maximum velocity in the swinging phase. The time (t_d) for the

double support phase (DSP) is considered. The trajectory $\hat{x}_{a,swing}$ is defined from the DSP to the other DSP, between the start time (t_1) and the final time (t_2) (figure 3 (*left*)).

$$\hat{x}_{a,swing}(t_a) = (S_x)\left(t_a - \frac{sin(2\pi t_a)}{2\pi}\right) - c_i \quad with \qquad t_a = \frac{t-(t_1+t_d/2)}{t_2-t_1-t_d} \tag{34}$$

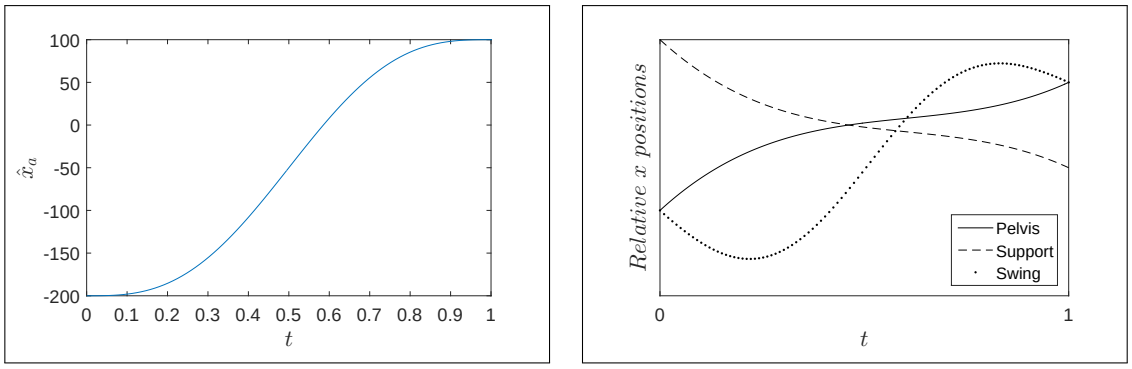


Figure 3: Position of the swinging ankle (*left*); Relative trajectories of the feet and the centre of the pelvis (*right*)

The supporting foot trajectory ($\hat{x}_{a,support}(t)$) must be the opposite action of the centre of the pelvis (x_p). These trajectories are generated in the local coordinate frame located in the front of supporting foot, and for its implementation must be transferred to the global coordinates of the robot (figure 3). In fig.(3 (*left*)) an example of the trajectories of the ankles is shown, it is assumed that the right ankle is the one supporting and the left ankle is swinging. The relative position in the global coordinate frame is:

$$x_{a,support} = -x_p(t_n) \qquad x_{a.swing} = \hat{x}_{a,swing}(t_a) - x_p(t_n) \tag{35}$$

The walking pattern is completely defined for the sagittal plane. Where the input parameters (S_x, t_1, t_2) can generate the trajectories of the ankles, knees and hip, the ZMP and the COG directly. The tuning parameters (α, t_d) are determined for the forward, turning and sideways modes.

The trajectories for the coronal plane have the same structure but with different condition boundaries.

6 Experimental results

A DARwIn-OP humanoid robot developed by the RoMeLa lab was used to implement the neuro-fuzzy system experimentally. It is 45 cm tall, weighs 2.8kg, and has 20 degrees of freedom. It has a web camera for visual feedback, and 3-axis accelerometer and 3-axis gyroscope for inertial sensing. Position-controlled Dynamixel servos are used for actuators, which are controlled by a CM730 microcontroller connected by an Intel Atom-based embedded PC at a control frequency of 100hz.

The neuro-fuzzy design begins when the fuzzy rules eq.(6) are defined incorporating information from the Darwin robot and expert knowledge. Once the structure is defined, it is necessary to reinforce learning. First, learning is applied to a realistic physical simulation to avoid damages in the real system. Finally, the neuro-fuzzy system is implemented in the Darwin robot and tested with random pushes, its performance and strategies show the advantages of the proposed scheme.

6.1 Design

The neuro-fuzzy system for the controller and the predictor are started with the same rules ($r = 9$) and with two inputs $\{\Delta\theta_x, \dot{\theta}_x\}$. But, the controller has three outputs $\{k_a, k_h, k_s\}$ and the predictor has one output $\{v\}$. So, with r defined fuzzy rules:

$$R_j : \textbf{if } \theta_x \textit{ is } \mu_{1j} \textit{ and } \dot{\theta}_x \textit{ is } \mu_{2j} \textbf{ then } k_a \textit{ is } \nu_{1j} \textit{ and } k_h \textit{ is } \nu_{2j} \textit{ and } k_s \textit{ is } \nu_{3j} \tag{36}$$

where μ_{ij} and ν_{lj} are fuzzy sets, the indexes $i = \{1, 2\}$ inputs, $l = \{1, 2, 3\}$ outputs and $j = \{1, ..., r\}$ fuzzy rules. $\{k_a, k_h, k_s\}$ are the recovery variables for the ankle eq. (4), hip eq.(5) and stepping eq.(3) strategies respectively.

With the fuzzy system defined, expert knowledge and limitations of the humanoid robot will be employed to set the initial rule base before its learning. Three membership functions are used for each input variable, and five membership functions for each output variable. The rule base is defined as follows:

Table 1: Fuzzy rule base for the feedback gains

k_a		$\Delta\theta_x$		
		NT	ZT	PT
$\dot{\theta}_x$	NV	PA	SPA	SNA
	ZV	SPA	ZA	SNA
	PV	SPA	SNA	NA

k_h		$\Delta\theta_x$		
		NT	ZT	PT
$\dot{\theta}_x$	NV	PH	SPH	SNH
	ZV	SPH	ZH	SNH
	PV	SPH	SNH	NH

k_s		$\Delta\theta_x$		
		NT	ZT	PT
$\dot{\theta}_x$	NV	NS	ZS	SPS
	ZV	SNS	ZS	SPS
	PV	SNS	ZS	PS

The membership functions in table 1 are defined using expert knowledge and the minmax ranges for each push recovery strategy. The term sets of the inputs and output variables and their limits are selected as follows:

$$\begin{aligned}
T(\theta_x) &= \{\mu_{11}, \mu_{12}, \mu_{13}\} = \{NT, ZT, PT\} && \{-45, 45\} \\
T(\dot{\theta}_x) &= \{\mu_{21}, \mu_{22}, \mu_{23}\} = \{NV, Z, PV\} && \{-120, 120\} \\
T(k_{xa}) &= \{\nu_{11}, \nu_{12}, \nu_{13}, \nu_{14}, \nu_{15}\} = \{NA, SNA, ZA, SPA, PA\} && \{-0.35, 0.35\} \\
T(k_{xh}) &= \{\nu_{21}, \nu_{22}, \nu_{23}, \nu_{24}, \nu_{25}\} = \{NH, SNH, ZH, SPH, PH\} && \{-1, 1\} \\
T(k_{xs}) &= \{\nu_{31}, \nu_{32}, \nu_{33}, \nu_{34}, \nu_{35}\} = \{NS, SNS, ZS, SPS, PS\} && \{-0.85, 0.85\}
\end{aligned} \tag{37}$$

where the letters for each fuzzy set are: Negative (N), Small Negative (SN), Zero (Z), Small Positive (SP), Positive (P), Trunk (T), Velocity (V), Ankle (A), Hip (H) and Step (S). For example, SPA meaning Small Positive Ankle.

6.2 Learning

The neuro-fuzzy system learns from random pushes and during walking and is able to successfully absorb multiple pushes. Simulation trials were performed for 3 push types (frontal, behind and lateral) with 6 repetitions in different strength ranges (small, medium, large). Reinforcement learning in the simulation was applied to 54 trials resulting in the following fuzzy surface fig.(4), for each output gain K.

The ankle gain surface shows that the strategy is applied in all ranges, with a large or medium push. Its gain is high, but with a small push it is regulated. The hip strategy is applied when

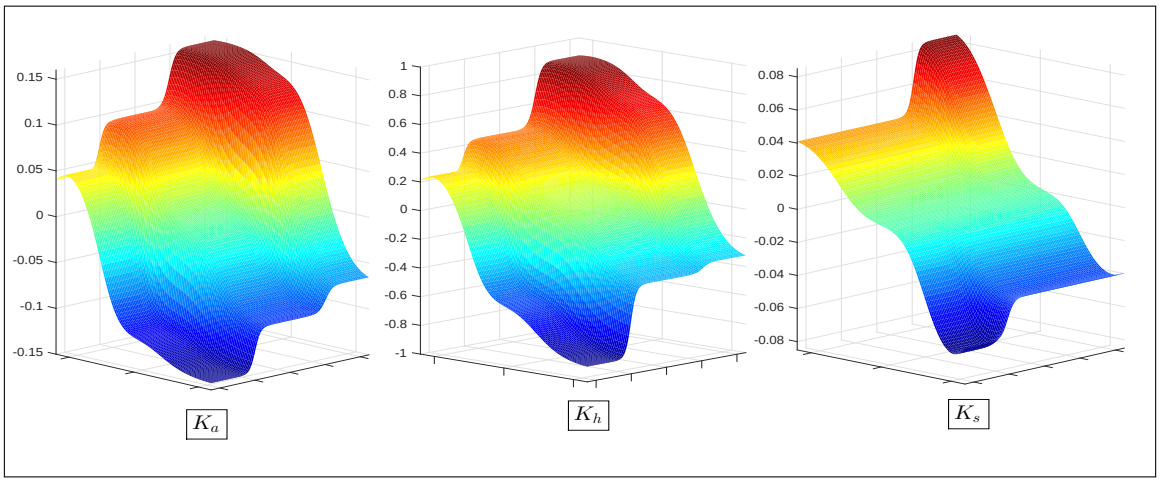


Figure 4: Ankle feedback surface after simulation learning (*left*); Hip feedback surface after simulation learning (*middle*); Step feedback surface after simulation learning (*left*)

the disturbance is greater, allowing a recovery without having to execute a step. Finally, the step strategy surface shows boundaries where it is inevitable to apply the step, but inside the robot kinematic.

6.3 Results

After designing and training the neuro-fuzzy system in a physically realistic simulation, the system was implemented in the Darwin robot. A trial was defined with multiple pushes to evaluate the ability of the neuro-fuzzy system to represent expert knowledge and respecting the limits, as well as assess the reinforcement learning algorithm to update the system.

Figure (5) shows an experimental test with 6 random pushes in walking forward mode, The first and second plots show θ_x and $\dot{\theta}_x$, respectively. The unknown strength push is frontal ($3s$,$29.9s$ and $40s$) and behind ($9s$, $15.5s$ and $21.6s$) are marked with vertical dash lines in the plots. The first push is the strongest requiring application of the three strategies as shown in the fig.(6), where the robot steps forward and corrects the trunk position angle to an upright position. The feedback gains are weighted using the effect of the inferior strategy learnt in the simulation trails respecting the limits defined in the design.

Note the sudden velocity change at the moment of pushes. Then, the robot is able to recover its velocity to the normal level and maintain walking. This is because the external reinforcement signal is applied with minimum delay to the neuro-fuzzy system, and this has a low computational cost with 18 fuzzy rules and 64 parameters to update. In a comparison between the Levenberg Marquardt algorithm and the backpropagation algorithm. The proposed algorithm has an average time of 0.5 ms with 8 average iterations, while, the backpropagation has an average time of 4 ms with 80 average iterations. This allows efficient implementation in embedded systems.

Finally, a phase plot between θ_x and $\dot{\theta}_x$ is shown in fig. (7), it shows the theoretical boundaries for each push recovery strategy using a lineal inverted pendulum model. The boundaries are calculated using the following parameters: $z_c = 0.295$, $d = 0.05$, $T_H = 0.3$, $m = 2$, $g = 9.81$,

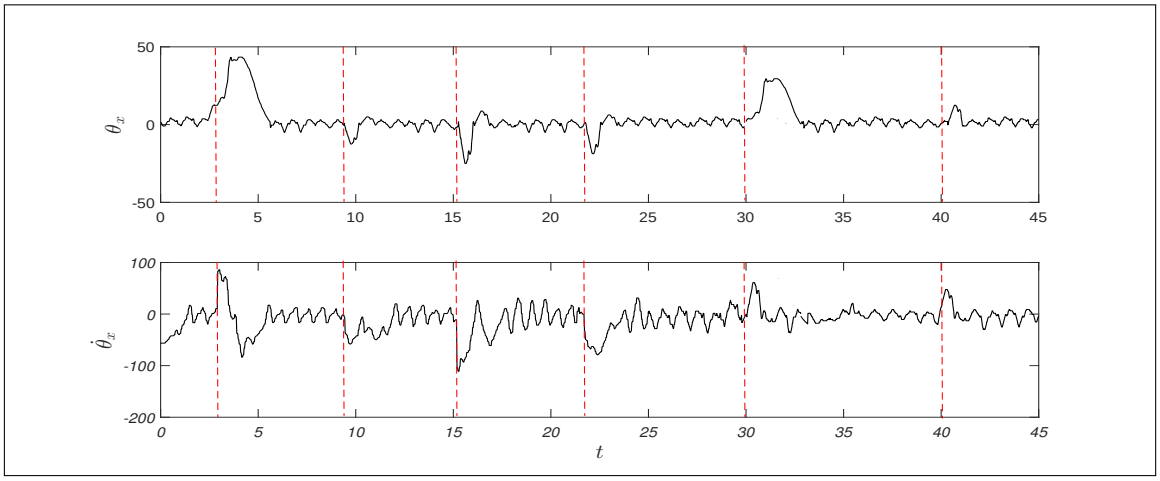


Figure 5: Experimental test with 6 random pushes without failure

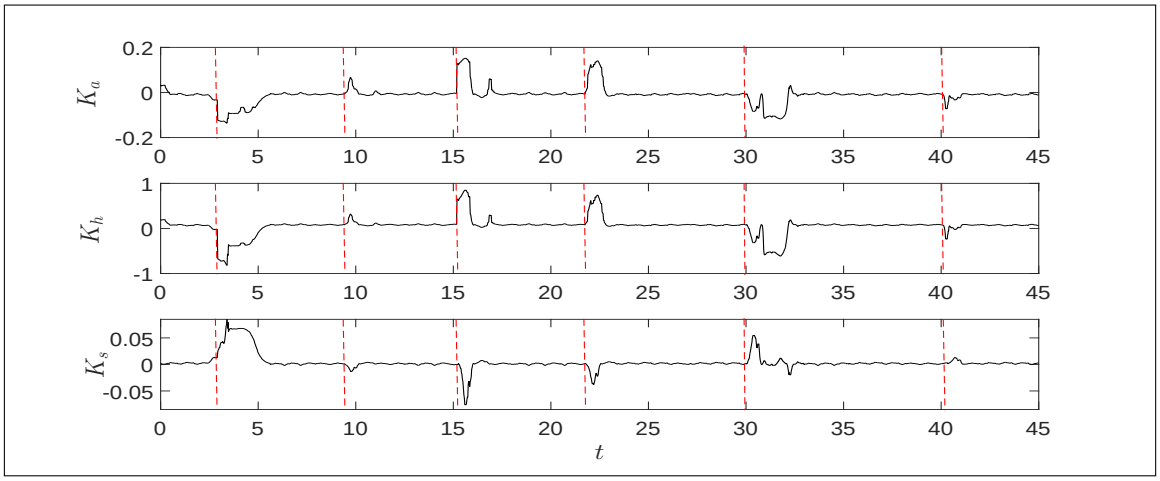


Figure 6: Gains of the 3 push recovery strategies with 6 random push after learning

$w = \sqrt{g/z_c}$, $\tau_{hip}^{max} = 1$ and $x_{capture}^{max} = 0.08$ [17].

$$\begin{aligned} Ackle: &\quad \left|\frac{\dot{\theta}_x}{w} + \theta_x\right| < \frac{d}{z_c} \\ Ankle + Hip: &\quad \left|\frac{\dot{\theta}_x}{w} + \theta_x\right| < \frac{d}{z_c} + \frac{\tau_{hip}^{max}\left(e^{-wT_H}-1\right)^2}{mgz_c} \\ Ankle + Hip + stepping: &\quad \left|\frac{\dot{\theta}_x}{w} + \theta_x\right| < \frac{d}{z_c} + \frac{\tau_{hip}^{max}\left(e^{-wT_H}-1\right)^2}{mgz_c} + \frac{x_{capture}^{max}}{z_c} \end{aligned} \tag{38}$$

The phase plot shows that the neuro-fuzzy system improved the robustness to strong pushes with

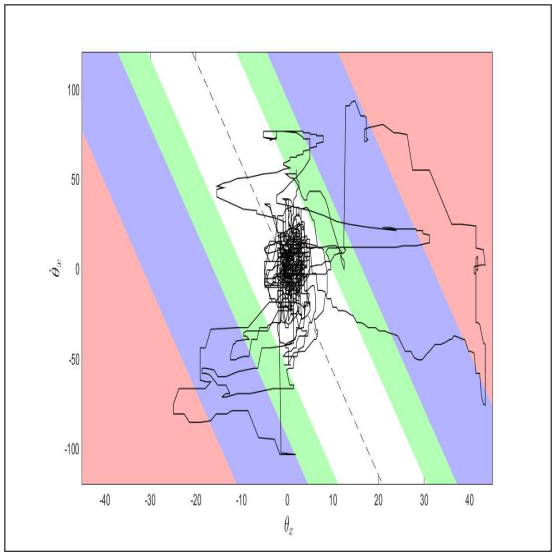


Figure 7: The phase plot shows how the neuro-fuzzy system responds to 6 unknown pushes. The coloured areas are the theoretical boundaries for the ankle(white), hip (light green), step (light blue) and unstable (light red).

a larger stable region. So, The robot already learnt the right ankle, hip and stepping strategies to cope with pushes from any direction.

7 Conclusions

In this paper, a practical method to implement a full body push recovery system on a general humanoid robot without specialised sensor and actuators is proposed. The method consists of a neuro-fuzzy system designed by combining expert knowledge, system constraints and online reinforcement learning.

This approach has a number of advantages over previous approaches. It is able to recover from unknown disturbances from any direction performing a combination of push recovery strategies based on the current system state without an accurate dynamical model of the robot or environment.

A straightforward and robust online reinforcement learning algorithm updates the parameters for the neuro-fuzzy controller and predictor to improve the push recovery performance using both simulation environment as well as on a small-size humanoid. Implementation using the modified LM algorithm guarantees low-cost computation.

It is integrated with a non-periodic, step-omnidirectional walking generator where step parameters produces soft curves and can be flexibly changed as a response to a disturbance. Moreover, it does not attempt to follow a future ZMP reference, only the joint trajectories are prescribed using the step parameters and a simple inverted pendulum model.

Experimental results show that the trained system can successfully develop a full body push recovery under external perturbations while walking in an arbitrary direction. There is a significant improvement of the stable region in the sagittal plane from a low number of experiences, but enough for the neuro-fuzzy controller to produce convincing push recovery capabilities.

Possible future work includes incorporating strategies to unreliable terrains and robustness.

Acknowledgment

Thanks to the support of the Laboratory of Humanoid Robots at PCC-UNAM for the facilities provided. To Dr. Fernando Arambula Cosio and Dr. Mario Pena Cabrera for the facilities and knowledge provided during the development of the project.

Bibliography

[1] Adiwahono A.H., Chew C.M., Liu B. (2013); Push recovery through walking phase modification for bipedal locomotion, *International Journal of Humanoid Robotics*, DOI: http://dx.doi.org/10.1142/S0219843613500229, 10(3), 1350022, 2013.

[2] Allgeuer P., Behnke S.(2014); Fused angles for body orientation representation, *Proceedings of 9th Workshop on Humanoid Soccer Robots of the IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, DOI: 10.1109/IROS.2015.7353399, 366-373, 2014.

[3] Faber F., Behnke S. (2007); Stochastic optimization of bipedal walking using gyro feedback and phase resetting, *7th IEEE-RAS International Conference on Humanoid Robots*, 203-209, doi: 10.1109/ICHR.2007.4813869, 2007.

[4] Fuller R. (2013); *Introduction to neuro-fuzzy systems*, Vol. 2, Springer Science and Business Media, 2013.

[5] Hyon S.H., Osu R., Otaka Y. (2009); Integration of multi-level postural balancing on humanoid robots, *IEEE Int. Conf. Robotics and Automation (ICRA)*, doi: 10.1109/ROBOT.2009.5152434, 1549-1556, 2009.

[6] Komura T., Leung H., Kudoh S., Kuffner J. (2005); A feedback controller for biped humanoids that can counteract large perturbations during gait, *IEEE Int. Conf. Robotics and Automation (ICRA), Barcelona, Spain*, 1989-1995, 2005.

[7] Koolen T., de Boer T., Rebula J.R., Goswami A., Pratt J.E. (2012): Capturability-based analysis and control of legged locomotion: Theory and application to three simple gait models - Part 1, *Int. J. of Robotics Research*, 31(9), 1094-1113, 2012.

[8] Missura M., Behnke S. (2015); Gradient-Driven Online Learning of Bipedal Push Recovery, IEEE/RSJ, *Int. Conf. on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, doi: 10.1109/IROS.2015.7353402, 387-392, 2015.

[9] Missura M., Behnke S. (2013); Omnidirectional Capture Steps for Bipedal Walking, *The IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 14-20, doi: 10.1109/HUMANOIDS.2013.7029949, 2013.

[10] Park J.-W., Kim J.-Y., Oh J.-H. (2008); Online Walking Pattern Generation and Its Application to a Biped Humanoid Robot - KHR-3 (HUBO), *Advanced Robotics*, 22(2), 159-190, 2008.

[11] Pratt J., Carff J., Drakunov S., Goswami A. (2006); Capture point: A step toward humanoid push recovery, *IEEE-RAS Int. Conf. Humanoid Robots*, Genova, Italy, 200-207, 2006.

[12] Pratt J.E., Koolen T., de Boer T., Rebula J.R., Cotton S., Carffer J., Johnson M., Neuhaus P.D. (2012); Capturability-based analysis and control of legged locomotion: Application to M2V2, a lower-body humanoid - Part 2, *Int. J. of Robotics Research*, 31(10), 1117-1133, 2012.

[13] Ranganathan A. (2004), The Levenberg-Marquardt Algorithm, *Tutorial on LM algorithm*, 2004.

[14] Rebula J., Pratt J., Canas F., Goswami A. (2007); Learning capture point for improved humanoid push recovery, *IEEE-RAS Int. Conf. Humanoid Robots, Pittsburgh, PA*, doi: 10.1109/ICHR.2007.4813850, 65-72, 2007.

[15] Semwal V.B., Chakraborty P., Nandi G.C. (2015); Less computationally intensive fuzzy logic (type-1)-based controller for humanoid push recovery. *Robotics and Autonomous Systems*, 63, 122-135, 2015.

[16] Si J., Wang Y.T. (2001); On-line Learning Control by Association and Reinforcement, *IEEE Trans. on Neural Networks*, doi: 10.1109/72.914523, 12(2), 264-276, 2001.

[17] Stephens B. (2007); Humanoid push recovery, *IEEE-RAS Int. Conf. Humanoid Robots*, IEEE Press, Pittsburgh, PA, 589-595, 2007.

[18] Tedrake R. (2004); Stochastic policy gradient reinforcement learning on a simple 3d biped, *Proc. of the 10th Int. Conf. on Intelligent Robots and Systems*, 2849-2854, 2004.

[19] Wieber P.B. (2006); Trajectory free linear model predictive control for stable walking in the presence of strong perturbations, *IEEE-RAS Int. Conf. Humanoid Robots*, IEEE Press, Nashville, doi: 10.1109/ICHR.2006.321375, 137-142, 2006.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 347-364, June 2017.

System Selection and Performance Evaluation for Manufacturing Company's ERP Adoption

B.Z. Niu, K.L. Chen, H.Z. Huang, Y. Li, L. Chen

Baozhuang Niu, Kanglin Chen, Lei Chen*
School of Business Administration, South China University of Technology
Guangzhou, 510640, P.R.China
bmniubz@scut.edu.cn, kanglin_chen@qq.com
*Corresponding author: jayden_business@foxmail.com

Huizhong Huang
Groupe Danone, Guangzhou, 510620, P.R.China
290092790@qq.com

Yuan Li
Lingnan College, Sun Yat-sen University
Guangzhou, 510275, P.R.China
jerry_yuanli@163.com

Abstract: Enterprise Resource Planning (ERP) system is an important investment for manufacturing companies that can affect their competitive advantages and operational performance. However, the implementation of ERP can be a complicated process, where many strategic decisions have to be made. We focus on two critical decisions in ERP implementation: (1) ERP system selection, and (2) ERP operational performance evaluation. For the former, we use Analytic Hierarchy Process (AHP) to design the key performance indicator (KPI) system. For the later, we combine AHP and Fuzzy Integrated Evaluation (FIE) methods to effectively evaluate the implementation of ERP. We use a typical industrial example and data analysis to illustrate our framework.
Keywords: ERP system selection, ERP performance evaluation, analytic hierarchy process, fuzzy integrated evaluation, manufacturing companies.

1 Introduction

Nowadays, severe market competition has dramatically transformed the business environment. For manufacturing companies, whose competitive advantages are mainly low cost operations and quick-response management, the implementation of information systems becomes critical. It is widely accepted that Enterprise Resource Planning (ERP) has the ability to integrate the flow of material, finance, and information and to support organizational strategies [10]. However, the implementation of ERP system can be a highly complicated process, especially for those contract manufacturing companies who have multiple businesses such as self-branded business, manufacturing business, and design business [4]. According to an independent research report [3], in 2014, 42% of the surveyed companies consider their ERP projects as a "neutral", or "not clear" , or "failed" project.

An important reason for the failure of ERP system implementation is that the ERP systems on the market do not fit the company's operations properties. For the successful implementation of ERP system, the adjustment of business process, the selection of suitable ERP system and IT tools, and the effective performance evaluation are the most critical decisions [15], [8], although all of them are hard to make. Being aware of these, managers in the manufacturing industry turn to consulting companies (e.g., IBM and Accenture) to find ERP solutions.

Meanwhile, many IT service companies identify this demand and build online service platform to help manufacturing companies to implement ERP systems. For example, TECTEC (http://www.technologyevaluation.com) proposes a ERP selection and assessment approach for its customers, and this approach is proven to be effective in their application cases for manufacturing companies that produce pharmaceutical and botanical products, industrial machinery products, and electronics and high-tech products [9], [10].

Recently, we have consulting interactions with a multinational manufacturing company which is Austria-headquartered. They turned to us for suggestions to implement ERP system to manage their supply chain. We conducted surveys and found that the standard approach proposed by TECTEC need be detailed. Thus we develop a framework to help the company to select the ERP system based on analytic hierarchy process (AHP) and use fuzzy integrated evaluation (FIE) method to measure the performance of their ERP implementation. Our work is summarized in this paper. All the data that is used to illustrate our framework comes from this consulting project. We combine the objectives of choosing the most appropriate ERP system and vendor with different criteria. We also compare the alternatives based on evaluators' opinions and identify the most appropriate ERP system.

2 Literature

Selecting the suitable ERP system for enterprise can help avoid the failure of ERP system implementation, so it is important to select the appropriate ERP system. There are several common methods to choose appropriate ERP system or the other management information system ([10], [15], [8]). The scoring method is one of the most popular methods, which is simple and intuitive, but does not guarantee the feasibility of resources. For example, [12] uses 10 criteria to evaluate the ERP system and develop a framework based on nominal group technique (NGT) and analytic hierarchy process (AHP) to select the ERP system. Some other methods are developed to improve the efficiency of ERP system implementation procedures, for example, [13], [14], and [4]. In practice, many companies use some financial indicators to select ERP systems. Since financial indicators are reported by professional institutions, they can be viewed as a trustable data resource, and can be used to index the implementers of ERP system [7]. Useful information includes the market size, the vendors and the overall system performance, etc.

Industry and academia also pay attention to the evaluation of ERP system. Companies want to use timely, accurate and objective performance evaluation to continuously adjust and improve the ERP project. Academia also try to identify the factors affecting the performance of ERP system through empirical study, and then construct the evaluation system to evaluate the performance of ERP system implementation ([5], [15]). [13] points out that it takes companies a long time to see the effect of ERP system on the performance. As a result, in the study on the comparison of performance between companies adopting ERP system and companies without ERP system, researchers couldn't find a significant difference [5]. When the time window is large enough that can eliminate this effect, there is a significant difference on the performance between companies using ERP system and companies without it [15].

3 Selection of ERP system

3.1 The criteria for selection

Before we discuss the criteria for selection of ERP system, determining the strategic objectives of ERP project is very necessary. Strategic objectives guide the team and indirectly coordinate

the interests of different departments inside the company.

The implementation of the ERP system including software and vendors. The quality of the system itself decides the influence of ERP system to the company. ERP vendors are responsible for ERP system development, implementation and maintenance services. Without vendors, the companies is unable to successfully implement ERP project. These two aspects are essential to the success of an ERP project. Therefore, we defined two objectives: Selecting the most appropriate ERP system and selecting the most appropriate ERP system vendor.

After determining the strategic objectives, we need to find the specific attributes of criteria according to two objectives. The following will discuss the attributes of criteria for ERP system selection and ERP system vendor selection.

ERP system

Most of the enterprisers have gradually understood the benefits from ERP system. According to the report of Panorama in 2014 [11], the most popular reason enterprisers implement ERP project is to improve the business (15%). The reason followed by is to better integrate the cross-regional and cross-department system (14%), and to get better service to customers (12%).

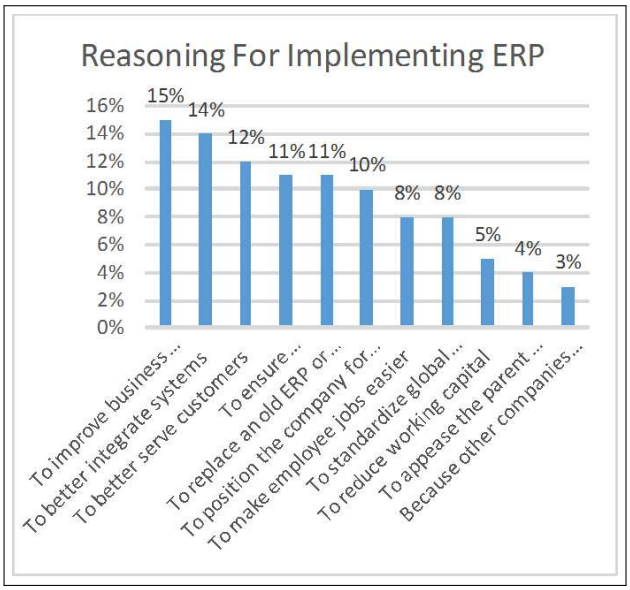


Figure 1: Reasoning for implementing ERP

We can analyze the attributes of criteria to select ERP system from the reasons for implementing ERP system:

Corresponding to the reason to improve business performance, ERP system should have complete functionality and help improve the company's performance by integrate business process through the complete module and fit function. Meanwhile, user-friendly interface and operations can help the internal and external personnel operate and understand the system, which can also help improve the operation efficiency and improve business performance.

Corresponding to the reason to better integrate the cross-regional and cross-department system, ERP system should have excellent system flexibility, providing the ease of in-house development and the ease of integration. The compatibility is particularly important to integrating the cross-regional and cross-department system.

Corresponding to better customer service, ERP system should have high system reliability, high system stability. Recovery ability can help avoid the loss of customers in the face of mistakes.

At the same time, long-term maintenance can also improve customers' satisfaction.

In addition, the total cost of the ERP system implementation is a factor that company must take into consideration. The total cost including system purchase price, consultant cost expenses, system maintenance cost and infrastructure cost. According to the report of Panorama in 2014, it shows that more than 54% of project will exceed the budget for unexpected technical or organizational issues. Considering a long-time implementation of ERP system, the implementation time is also an important attributes. According to the report of Panorama, 63% of the ERP system implementation will take more time than expected.

ERP vendor

According to the report of Panorama in 2014 [11], among the global famous vendors of ERP system, Oracle (34%) is the most popular, followed by Microsoft Dynamics (20%) and SAP (16%). Companies will pay great attention on the reputation of ERP system vendors. The financial condition, scale of vendor and market share will be taken into consideration.

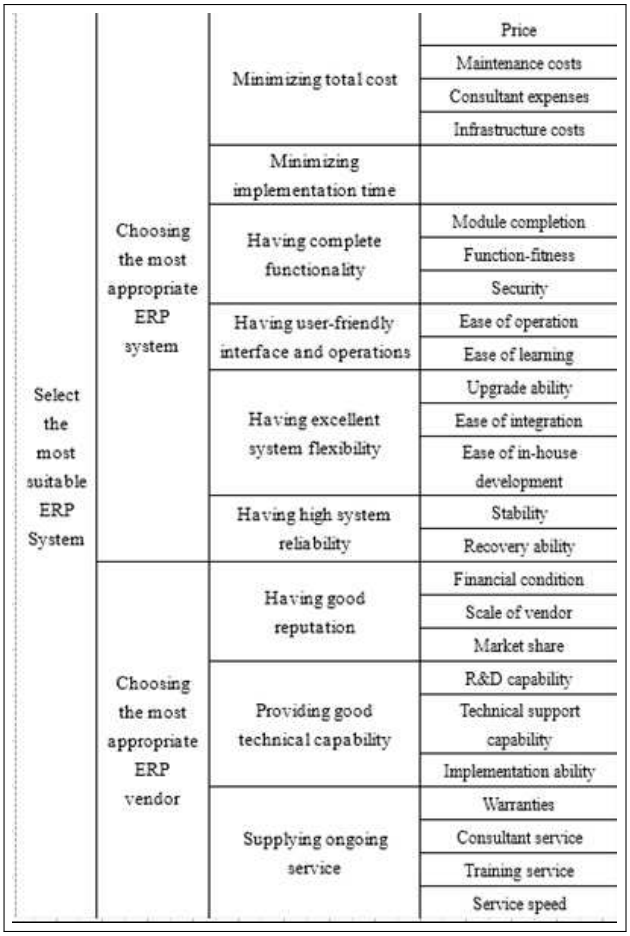


Figure 2: Factors to help select ERP systems

In addition, the service that vendors provide matters a lot to the companies. In the service that vendors provide, the implementation of ERP is the most common (21%), followed by related training (19%), organizational change management (14%), software selection (11%).The service is very important because the vendors have professional knowledge and the development and maintenance of system largely depends on the vendors. With the service of vendors, the

companies can integrate internal resources and external knowledge and play an important role on the development, implementation and maintenance of ERP system project. Therefore, in order to choose suitable system vendors, companies need to consider the vendors' technical capability, including R&D capability, technical support capability and implementation ability; Also, ongoing service needs considering, which includes warranties, consultant service, training service and service speed.

Based on the attributes of criteria above for the objective of ERP system and ERP system vendor, we sums up the attributes affecting the selection of ERP system:

Choosing the most appropriate ERP system: minimizing total cost (price, maintenance costs, consultant expenses, infrastructure costs), minimizing implementation time, having complete functionality (module completion, function-fitness, security), having user-friendly interface and operations (ease of operation, ease of learning), having excellent system flexibility (upgrade ability, ease of integration, ease of in-house development), having high system reliability(stability, recovery ability).

Choosing the most appropriate ERP vendor: having good reputation (financial condition, scale of vendor, market share), providing good technical capability (R&D capability, technical support capability, implementation ability), and supplying ongoing service (warranties, consultant service, and training service, service speed).

3.2 AHP-based approach to select ERP system

Introduction of AHP

Analytic Hierarchy Process (AHP) was developed by Thomas Saaty in 1971, mainly used in decision-making problems with uncertain circumstances and many criteria [16]. The main property of AHP is that it can turn qualitative problem quantitative. It gives a quantitative importance of each level and uses mathematical method to determine the weights of all elements [5]. Basic steps are as follows:

(a) *Determine the objectives and criteria* P attributes $u = \{u_1, u_2, ...u_p\}$,

(b) *Pairwise comparison and judgment matrix* The pairwise comparison show the importance of one attributes to another. This subjective judgment can be convert to a numerical value using a scale of 1-9. We can draw the judgment matrix from pairwise comparison.

(c) *Weights calculation and aggregation* Calculate the greatest characteristic root and characteristic vector of the judgment matrix S. The characteristic vector is the importance of each evaluation attributes or alternatives and also is the distribution of weight coefficient.

(d) *Check the consistency.* We need to check the consistency of the judgment matrix with $CI = \frac{\lambda_{max}-n}{n-1}$. If the consistency index CI of judgment matrix is less than 0.10, we believe the results of the analytic hierarchy sort have satisfactory consistency and the weights is reasonable; Otherwise, the pairwise comparison matrix need to adjust and redistribute the weights.

An example of company A

Company A is a large multinational manufacturing corporation. Company A has its own production workshop and assembly workshop in mainland China. Its product is involved in seven industries and there are thousands of different types of products. After ten years of development, Company A has rapidly expanded business and grown fast. The staff team has

grown from dozens to more than 1500 and the annual sales expand to millions of dollars from only six hundred million.

With the growing of business, the difficulty of company's management is also appearing, which makes the implement of ERP system become necessary.

Within the company, the sales department is only responsible for the order fulfillment. The lack of a standard process makes the sales department low-efficient. Besides, there is contradiction between purchasing department and project department. Project department, as the service department of the purchasing groups, makes the final decision in the procurement process. The purchasing department can only give suggestions and fulfill the order. This mismatching between right and duty during the procurement process in two departments induces many conflicts. Besides, the financial department uses an independent financial system which only manages the cash flow in that department but not the whole company.

Outside the company, purchasing department does not collaborate well with the suppliers. Most of the suppliers are small and medium-sized companies and the information construction remains to be improved.

The company cannot timely access to the useful information and the low standardization level of business operation process brings lots of troubles to company A. ERP system may help company A to integrate the departments and manage the information within and outside the company.

The leaders of company A are thinking whether they should purchase the professional ERP system and form a project team. They have already selected three ERP systems from different vendors, denoted as system 1, system 2, system 3.

The required function for ERP system of different industries has a huge difference. Therefore, enterprisers need to know its industry characteristics and function requirements, when choosing the appropriate ERP system. In addition, the enterpriser need consider the scale of company.

We ask three leaders as evaluators for a questionnaire survey and propose ERP system selection framework as follows.

a) Identify the ERP system characteristics For the ERP system selection, we collect the opinions through the purchasing department, project department, finance department, human resources department and marketing department. It is decided that the system selection is considered from two aspects: One is the ERP system itself; the other is the ERP system vendor.

b) Organize the hierarchy structure of Objectives, Criteria, and Alternatives. Objectives are the target of the problem. Criteria is to extract the attributes for evaluating ERP systems. Alternatives are the feasible solutions of the problem. In the case of company A, there are two Objectives with different Criteria. The first objective is screening out the most appropriate ERP system. There are six attributes for evaluating the ERP system, including minimizing total cost (C1), minimizing implementation time (C2), having complete functionality (C3), having user-friendly interface and operations (C4), having excellent system flexibility (C5), having high system reliability(C6). There are three alternatives, called as system 1,system 2 and system 3.

The second objective is choosing the most appropriate ERP vendor. There are three attributes for evaluating the ERP vendor, including having good reputation (D1), providing good technical capability (D2), supplying ongoing service (D3). The alternatives are same, called as system 1, system 2 and system3.

c) The comparison of attributes among Criteria (for example)

C1:Minimizing total cost; C2:Minimizing implementation time.

If the ratio is 3:1, the evaluator think that minimizing total cost is more important than minimizing the implementation time. The importance degree of former is 3 compared to the later. If the ratio is 1:5, the evaluator think that minimizing the implementation time is more important. Its importance degree is 5 compared to minimizing total cost.

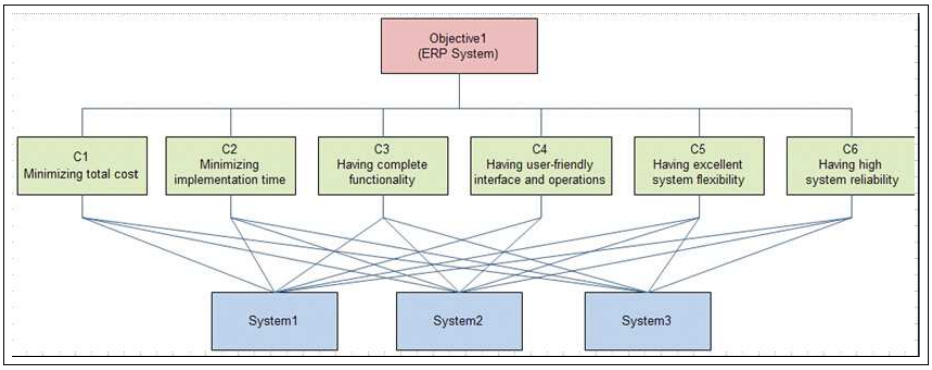


Figure 3: AHP-based ERP system selection framework

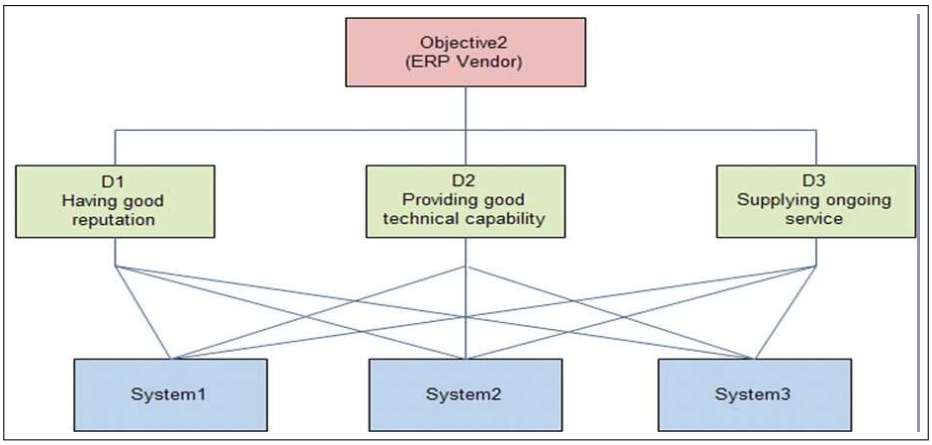


Figure 4: AHP-based ERP vendor selection framework

Table 1: Criteria questionnaire

	9:1	8:1	7:1	6:1	5:1	4:1	3:1	2:1	1:1	1:2	1:3	1:4	1:5	1:6	1:7	1:8	1:9	
C1																		C2

Table 2: Alternatives questionnaire

	9:1	8:1	7:1	6:1	5:1	4:1	3:1	2:1	1:1	1:2	1:3	1:4	1:5	1:6	1:7	1:8	1:9	
S1																		S2

Table 3: Judgment matrix of attributes for ERP system

Evaluator 1	C1	C2	C3	C4	C5	C6	Wi(weight)
C1 Minimizing total cost	1	1/3	1/3	1	1/3	3	0.0766
C2 Minimizing implementation time	3	1	1/4	3	1/3	5	0.1500
C3 Having complete functionality	3	4	1	5	3	7	0.3954
C4 Having user-friendly interface and operations	1	1/3	1/5	1	1/5	5	0.0766
C5 Having excellent system flexibility	3	3	1/3	5	1	5	0.2690
C6 Having high system reliability	1/3	1/3	1/7	1/5	1/5	1	0.3240
λ_{max}: 65.222; Consistency: 0.0829							

The comparison of alternatives: Known from the AHP selection framework, each attributes correspond three alternatives which need to take account of the project, so decision-makers need to compare the alternatives for each attributes.

d) Select ERP system We can obtain the corresponding judgment matrix through the pairwise comparison after we collect the questionnaire of the evaluators. We need to check the consistency of the judgment matrix with $CI = \frac{\lambda_{max}-n}{n-1}$. We find that the consistency index CI of judgment matrix is all less than 0.10.The following only show the judgment matrix for ERP system and each alternative of evaluator 1.

Comparing the importance of attributes to each evaluator, we can find that three evaluators tend to share the same opinion:

For the attributes of ERP system, “having complete functionality " is considered as a very important attributes for three evaluators, of which the relative weight comes to be the 1st for evaluator 1 and evaluator 2, 2nd for evaluator 3. “having excellent system flexibility " is also of great importance, respectively to be the 2nd,2nd and 1st for evaluator1, 2 and 3. While, “having high system reliability "is considered as the least important attribute for all three evaluators.

It is necessary to analyze the result with the situation of company. Company A, a large manufacturing enterpriser, is an integrated supplier providing complete sets of production lines, equipment and services. It has its own production workshop and assembly workshop with complete functional departments. More specifically, the number of project team is large and the organizational structure is loose. There are too many types of equipment and spare parts to procure. There is too much communication between project teams and different functional departments, so are the purchasing department and suppliers of company A. Moreover, the difficulty of purchasing has worsen the information distortion and then deepen the contradictions between project teams and functional departments. Thus, an information system covered multi-department is particularly important, which can reduce the conflicts between different de-

Table 4: Judgment matrix of alternatives for C_1 and importance of attributes

The pairwise comparison of alternatives for C1				
	System 1	System 2	System 3	Wi(weight)
System 1	1	3	5	0.6370
System 2	1/3	1	3	0.2583
System 3	1/5	1/3	1	0.1047
λ_{max}: 3.0385; Consistency: 0.0370				

Table 5: Judgment matrix of alternatives for C_2 and importance of attributes

The pairwise comparison of alternatives for C2				
	System 1	System 2	System 3	Wi(weight)
System 1	1	5	1/3	0.2790
System 2	1/5	1	1/7	0.0719
System 3	3	7	1	0.6491
λ_{max}: 3.0649; Consistency: 0.0624				

Table 6: Judgment matrix of alternatives for C_3 and importance of attributes

The pairwise comparison of alternatives for C3				
	System 1	System 2	System 3	Wi(weight)
System 1	1	7	3	0.6694
System 2	1/7	1	1/3	0.0879
System 3	1/3	3	1	0.2426
λ_{max}: 3.0070; Consistency: 0.0068				

Table 7: Judgment matrix of alternatives for C_4 and importance of attributes

The pairwise comparison of alternatives for C4				
	System 1	System 2	System 3	Wi(weight)
System 1	1	1/3	1/7	0.0879
System 2	3	1	1/3	0.2426
System 3	7	3	1	0.6694
λ_{max}: 3.0070; Consistency: 0.0068				

Table 8: Judgment matrix of alternatives for C_5 and importance of attributes

The pairwise comparison of alternatives for C5				
	System 1	System 2	System 3	Wi(weight)
System 1	1	5	1	0.4806
System 2	1/5	1	1/3	0.1400
System 3	1	3	1	0.4054
λ_{max}: 3.0291; Consistency: 0.0279				

Table 9: Judgment matrix of alternatives for C_6 and importance of attributes

The pairwise comparison of alternatives for C6				
	System 1	System 2	System 3	Wi(weight)
System 1	1	1	1/5	0.1336
System 2	1	1	1/7	0.1194
System 3	5	7	1	0.7471
λ_{max}: 3.0126; Consistency: 0.0121				

Table 10: Judgment of importance of attributes

	Attributes	Evaluator 1	Evaluator 2	Evaluator 3
ERP System	minimizing total cost	0.0766(4)	0.1500(3)	0.0378(5)
	minimizing implementation time	0.1500(3)	0.1500(3)	0.1790(3)
	having complete functionality	0.3954(1)	0.3910(1)	0.3356(2)
	having user-friendly interface and operations	0.0766(4)	0.0565(5)	0.0566(4)
	having excellent system flexibility	0.2690(2)	0.2085(2)	0.3710(1)
	having high system reliability	0.0324(6)	0.0420(6)	0.0200(6)
ERP Vendor	having good reputation	0.0719(3)	0.0554(3)	0.0995(3)
	providing good technical capability	0.2790(2)	0.5990(1)	0.3355(2)
	supplying ongoing service	0.6491(1)	0.3456(2)	0.5650(1)

partments and project teams by sharing the information effectively. To sum up, we can find the great importance of “having complete functionality".

At the same time, company A is a foreign multinational enterprisers. It has different branches in 44 countries around the world and five business areas. There are many different brands and products in each business area. While in the implementation of procurement, the boundaries between different departments is clear and they independently do different works in the business. There is no communication and no collaboration which generates a lot of repetitive work and additional costs such as assessment, repeated negotiation, travel cost, quality control cost and so on. It is necessary for company A share information and resources cross areas and departments, so the system flexibility is very valued.

For the attributes of ERP vendor, “providing good technical capability" and “supplying on-going service" are of great importance while “having good reputation" is not considered as an important attribute.

“Providing good technical capability" includes R&D capability, technical support capability and implementation ability. Besides the initial system development, maintenance and upgrade stage of ERP project also need excellent technical support. In the adjustment stage, ERP system need continuous operation maintenance, and even need a new version or new functions. The system’s maintenance and upgrade require a long-term technical support.

“Supplying on-going service" includes the most basic warranty service, consultant service and training services. The users’ feedback is an important criterion to see whether ERP system is running smoothly. It is necessary that users approve and understand the system. The implementation of ERP project requires users to master the complicated operation skills. If the employee do not understand how the system works, it will ultimately affect the entire ERP system. The success of ERP system must be based on reasonable operation.

In the ERP system implementation of company A, training objects include the suppliers of A company besides the employees. There are more than 500 suppliers and quite of them are small and medium-sized companies, which adopt the traditional manufacturing management mode and are lack advanced management philosophy. The information cannot be inputted and processed timely and the information management system is incomplete and imperfect. To help the suppliers to adapt the ERP system is necessary and challenging. Therefore, ongoing training service is valued.

After three evaluators give a weight to all attributes, we can get the evaluation score of three ERP systems through the two judgment matrix (including criteria judgment matrix and alternative judgment matrix). Considering that we develop two objectives (ERP system and ERP vendor), we give equal weights to them. Finally we get the evaluation score of three ERP systems. Following is the result of evaluator 1:

Table 11: Evaluation score of ERP systems (1)

Evaluator 1	ERP system	ERP vendor	final score
System 1	0.4957	0.3129	0.4043
System 2	0.1185	0.323	0.22075
System 3	0.3858	0.3641	0.37495
(1) Equal weights to ERP system and ERP vendor			

Table 12: Evaluation score of ERP systems (2)

	Evaluator 1	Evaluator 2	Evaluator 3	final score
System 1	0.40430(1)	0.37155(2)	0.27362(2)	0.34982(2)
System 2	0.22075(3)	0.18145(3)	0.23123(3)	0.21114(3)
System 3	0.37495(2)	0.44700(1)	0.49515(1)	0.43904(1)
(2) Equal weights to each evaluator				

e) Get the final result. Evaluator 2 and evaluator 3 prefer to choose system 3 while evaluator 1 prefers to choose system 1. We can find that the score of evaluator 1 to system 1 and system 3 is close. Moreover, system 3 gets the highest final score. The result show that system 3 is the most appropriate ERP system for company A.

4 Evaluation of ERP system

Company A started to promote the ERP project after selecting the appropriate ERP system. In order to implement ERP project successfully, company A set up a team to take charge of the entire implement. In the preparatory stage, company A focused on the training and helped employee understand the ERP system. Then, company A started to research and analyze, even the specific operation of each departments, in order to adapt the ERP system to match the company.

After this, company A formally set up ERP system. They built a complete system framework taking full consideration of opinions from each departments and vendors. Company A fully combined the original function with the business process. In addition, ERP project team also optimized the mismatch between ERP system and company's business. Next is to import massive data. Company A successfully imported the internal and external data before changing the system and checked the accuracy of data. In November 2013, company A officially started using ERP system. After cautious consideration and selection, the new system still bring impact to the company on business. With time goes by, employees have been familiar with ERP system and it has run methodically.

Reviewing the ERP project of company A, it went well during the implementation, but we need to see whether it brings significant benefits to company A. The evaluation of implementation of ERP system is of great importance, which involved the influence on company's strategy, the impact on performance of management and the business process. The following will show the evaluation for the performance of the ERP system implementation.

4.1 The framework of evaluation

The success of ERP project is far more than that system goes live. How to judge or define the success of ERP project is also different for different companies or different industries. According

to company A's business and structure, we listened the opinion of managers and sort out the following performance evaluation structure.

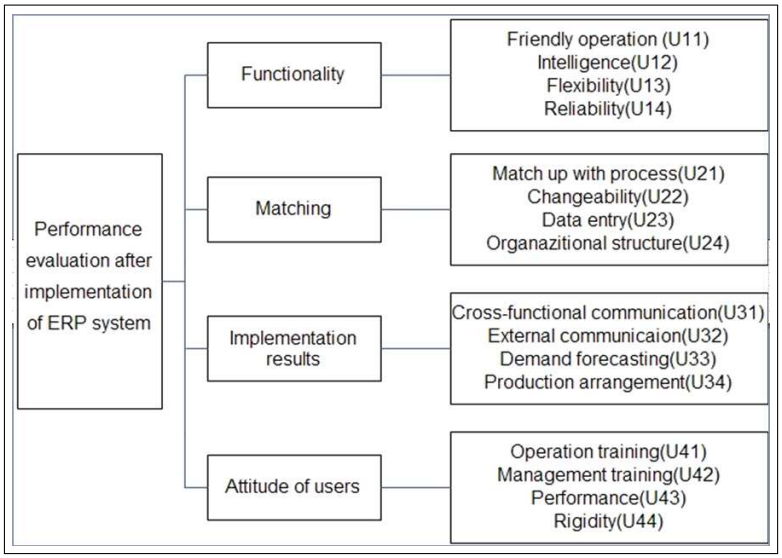


Figure 5: Structure of performance evaluation

The description of each factors in the structure of evaluation above is as follows. It's worth mentioning that "functionality" and "implementation result" are positive statements, "matching" and "attitude of users" is reverse. This is to reduce the respondents' deflection and is also helpful to remove those regardless of content. As a result, we need to adjust correspondingly when scoring. In the following results, "average score" is not adjusted and the "real score" is adjusted.

Functionality: ERP system has user-friendly interface and operations; ERP system has high intelligence; ERP system has excellent flexibility and compatibility; ERP system has high reliability.

Matching: ERP system does not match the company's operation process; ERP system does not adapt to the mismatch; The data entry and processing of ERP system does not match with the original model; ERP System does not match company's organizational structure and strategy.

Implementation results: ERP system helps improve the efficiency and communication cross-department; ERP system promotes the collaboration with suppliers; ERP system help the company with demand forecasting and capacity management; ERP system help the company improve the quality of the products and arrange the production reasonably.

Attitude of users: The users of ERP system do not get the corresponding training and do not understand ERP system; The management does not know the implementation of ERP system implementation and give no support to it; The users of ERP system think that it does not improve the performance. The performance get even worse than before; ERP system lacks flexibility and makes the company lose advantages.

4.2 The analysis based on FIE

30 questionnaires were distributed within the company, we recycled 30 questionnaires and the 23 of them were valid. The results are as follows:

From the data we can see that the mean of real average score is 3.3894 and the total real score is 54.2308, higher than the total real average score 48, under normal distribution assumption. If

Table 13: Evaluation score

		Strongly disagree	Disagree	Neutral	Agree	Strongly agree	Average score	Real score	Standard deviation
Functionality	1	0.0000	0.0769	0.2308	0.6923	0.0000	3.6154	3.6154	0.62
	2	0.0000	0.3846	0.3077	0.3077	0.0000	2.9231	2.9231	0.83
	3	0.0000	0.3077	0.4615	0.2308	0.0000	2.9231	2.9231	0.73
	4	0.0000	0.0769	0.3846	0.4615	0.0769	3.5385	3.5385	0.75
Matching	5	0.0000	0.3077	0.5385	0.1538	0.0000	2.8462	3.1538	0.73
	6	0.0000	0.5385	0.3846	0.0769	0.0000	2.5385	3.4615	1.12
	7	0.0000	0.3846	0.3077	0.3077	0.0000	2.9231	3.0769	0.84
	8	0.1538	0.3846	0.3077	0.1538	0.0000	2.4615	3.5385	1.42
Implementation result	9	0.0000	0.2308	0.3077	0.4615	0.0000	3.2308	3.2308	0.80
	10	0.0000	0.0000	0.3077	0.6154	0.0769	3.7692	3.7692	0.58
	11	0.0000	0.1538	0.3846	0.4615	0.0000	3.3077	3.3077	0.72
	12	0.0000	0.0769	0.0769	0.7692	0.0769	3.8462	3.8462	0.66
Attitude of users	13	0.0769	0.4615	0.4615	0.0000	0.0000	2.3846	3.6154	1.38
	14	0.2308	0.3846	0.3846	0.1538	0.0000	2.7692	3.2308	1.26
	15	0.2308	0.5385	0.1538	0.0769	0.0000	2.0769	3.9231	2.02
	16	0.0000	0.3077	0.4615	0.2308	0.0000	2.9231	3.0769	0.75
							Total	54.2308	

only judging from this data, we can say that company A thinks the ERP project help improve the performance.

There are some shortcomings in the classical statistical analysis. It cannot show the overall attitude intuitively and cannot directly show the proportion of different order of evaluation. Given the order of evaluation in questionnaires is fuzzy, we analyze the data using the fuzzy integrated evaluation (FIE) method. When determining the weights of each sub-factor, we use the AHP method.

FIE

FIE is a method to evaluate after fuzzy transform according to the criteria and measured values. The process of FIE: Assume the evaluation target as a fuzzy set composed of a number of factors; Then set order of evaluation to these factors and make up a fuzzy set; Next calculate the membership degree of each factors to the order of evaluation; And then according to the weights of factors in the evaluation, calculate the quantitative value [18].

The evaluation of ERP implementation of company A

a) The factors set U_t:

$U = \{U_1, U_2, U_3, U_4\}$= {functionality, matching, implementation results, attitude of users}

$U_1 = \{U_{11}, U_{12}, U_{13}, U_{14}\}$= {friendly operation, intelligence, flexibility, reliability}

$U_2 = \{U_{21}, U_{22}, U_{23}, U_{24}\}$= {match up, changeability, data entry, organizational structure}

$U_3 = \{U_{31}, U_{32}, U_{33}, U_{34}\}$= {cross-department communication, external communication, demand forecasting, production arrangement}

$U_4 = \{U_{41}, U_{42}, U_{43}, U_{44}\}$= {operational training, management training ,performance, rigidity}

b) The evaluation set for factors: Evaluation set is a collection of all results of the evaluation by evaluators. V= strongly disagree, disagree, neutral, agree, strongly agree

c) The fuzzy relationship matrix R: According to the questionnaire statistics, we can get the proportion of different order of evaluation .The statistical records are as follows:

Table 14: Factor set U_1

Factor set U_1					
	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
Friendly operation	0.00%	7.69%	23.08%	69.23%	0.00%
Intelligence	0.00%	38.46%	30.77%	30.77%	0.00%
Flexibility	0.00%	30.77%	46.15%	23.08%	0.00%
Reliability	0.00%	7.69%	38.46%	46.15%	7.69%

Table 15: Fuzzy relationship matrix

$$\mathbf{R_1} = \begin{pmatrix} 0.00\% & 7.69\% & 23.08\% & 69.23\% & 0.00\% \\ 0.00\% & 38.46\% & 30.77\% & 30.77\% & 0.00\% \\ 0.00\% & 30.77\% & 46.15\% & 23.08\% & 0.00\% \\ 0.00\% & 7.69\% & 38.46\% & 46.15\% & 7.69\% \end{pmatrix}$$

$$\mathbf{R_2} = \begin{pmatrix} 0.00\% & 15.38\% & 53.85\% & 30.77\% & 0.00\% \\ 0.00\% & 7.69\% & 38.46\% & 53.85\% & 0.00\% \\ 0.00\% & 30.77\% & 30.77\% & 38.46\% & 0.00\% \\ 0.00\% & 15.38\% & 30.77\% & 38.46\% & 15.38\% \end{pmatrix}$$

$$\mathbf{R_3} = \begin{pmatrix} 0.00\% & 23.08\% & 30.77\% & 46.15\% & 0.00\% \\ 0.00\% & 0.00\% & 30.77\% & 61.54\% & 7.69\% \\ 0.00\% & 15.38\% & 38.46\% & 46.15\% & 0.00\% \\ 0.00\% & 7.69\% & 7.69\% & 76.92\% & 7.69\% \end{pmatrix}$$

$$\mathbf{R_4} = \begin{pmatrix} 0.00\% & 0.00\% & 46.15\% & 46.15\% & 7.69\% \\ 0.00\% & 15.38\% & 38.46\% & 38.46\% & 23.08\% \\ 0.00\% & 7.69\% & 15.38\% & 53.85\% & 23.08\% \\ 0.00\% & 23.08\% & 46.15\% & 30.77\% & 0.00\% \end{pmatrix}$$

We can get the fuzzy relationship matrix R_1 from U_1. Similarly, we can get fuzzy relationship matrix R_2 from U_2. While this subset is disjunctive, we should reverse the arrangement.

d) The weight of each factor: In the evaluation system, the importance of each factor to realize the goal of system is different. The weight of each factor show the different importance. Set the weights reasonably and appropriately is important for evaluation. Here we use AHP to get the weights. The Supervisors of company score the four factors: functionality, matching, implementation results, and attitude of users. We get the following results:

Calculate the greatest characteristic root and characteristic vector of the judgment matrix. The characteristic vector is the importance of each evaluation factors and also is the distribution of weight coefficient. $U = [0.2477, 0.1259, 0.5538, 0.0727]$. Similarly, calculate the weight of each factor under the four dimensions according to the experts' scoring:

Each weight of factors passes the consistency check.

e) Get the evaluation results. $B = U * R$: $B_1 = U_1 * R_1$= (0.0000 0.2299 0.3975 0.3571 0.0154); $B_2 = U_2 * R_2$= (0.0000 0.1679 0.4360 0.3590 0.0370); $B_3 = U_3 * R_3$= (0.0000 0.0927 0.2966 0.5657 0.0451); $B_4 = U_4 * R_4$= (0.0000 0.0799 0.3147 0.4684 0.1618). $D = U * R$= (0

Table 16: Comparison matrix of factors

Comparison matrix of factors					
	U_1	U_2	U_3	U_4	Wi(weight)
U_1 functionality	1	3	1/3	3	0.2477
U_2 matching	1/3	1	1/5	1/3	0.1259
U_3 implementation result	3	3	1	5	0.5538
U_4 attitude of users	1/3	5	1/5	1	0.0727
λ_{max}: 4.1975; Consistency: 0.0740					

Table 17: Comparison matrix of sub-factors in U_1

Comparison matrix of sub-factors in U1					
	U_{11}	U_{12}	U_{13}	U_{14}	Wi(weight)
U_{11} functionality	1	3	1/3	1/2	0.1612
U_{12} matching	1/3	1	1/5	1/3	0.0740
U_{13} implementation result	3	5	1	5	0.5641
U_{14} attitude of users	2	3	1/5	1	0.2006
λ_{max}: 4.2219; Consistency: 0.0831					

Table 18: Comparison matrix of sub-factors in U_2

Comparison matrix of sub-factors in U2					
	U_{21}	U_{22}	U_{23}	U_{24}	Wi(weight)
U_{21} match up	1	5	3	3	0.5244
U_{22} change ability	1/5	1	1/2	1/2	0.0957
U_{23} data entry	1/3	2	1	1/3	0.1390
U_{24} organizational structure	1/3	2	3	1	0.2408
λ_{max}: 4.1575; Consistency: 0.0590					

Table 19: Comparison matrix of sub-factors in U_3

Comparison matrix of sub-factors in U3					
	U_{31}	U_{32}	U_{33}	U_{34}	Wi(weight)
U_{31} cross-department communication	1	1/3	4	3	0.2854
U_{32} external communication	3	1	3	4	0.4944
U_{33} demand forecasting	1/4	1/3	1	2	0.1290
U_{34} production arrangement	1/3	1/4	1/2	1	0.0912
λ_{max}: 4.2367; Consistency: 0.0886					

Table 20: Comparison matrix of sub-factors in U_4

Comparison matrix of sub-factors in U4					
	U_{41}	U_{42}	U_{43}	U_{44}	Wi(weight)
U_{41} operational training	1	3	1/2	3	0.3089
U_{42} management training	1/3	1	1/3	3	0.1612
U_{43} performance	2	3	1	3	0.4369
U_{44} rigidity	1/3	1/3	1/3	1	0.0930
λ_{max}: 4.2148; Consistency: 0.0805					

0.1352 0.3405 0.4810 0.0452).

f) Analyze the results. The results show 1.64% of evaluators strongly disagree that ERP project bring positive effect; 18.75% of them disagree; 34.05% remain neutral; 42.87% agree the positive effects of ERP project and 2.88% strongly agreed with it. According to the maximum membership degree principle, the conclusion is "agree". Multiply the raw score (1-5) in Likert scale by the number of sub-factors, 16. Then we get the level parameters in evaluation set and the column vector is: $p = DE = 016 + 0.135232 + 0.340548 + 0.481064 + 0.045280 = 55.0677$. The result is close to the statistical analysis result, 54.2308. It shows that the result based on FIE is consistent with the result based on classical statistical analysis. Company A recognizes ERP project as a beneficial project.

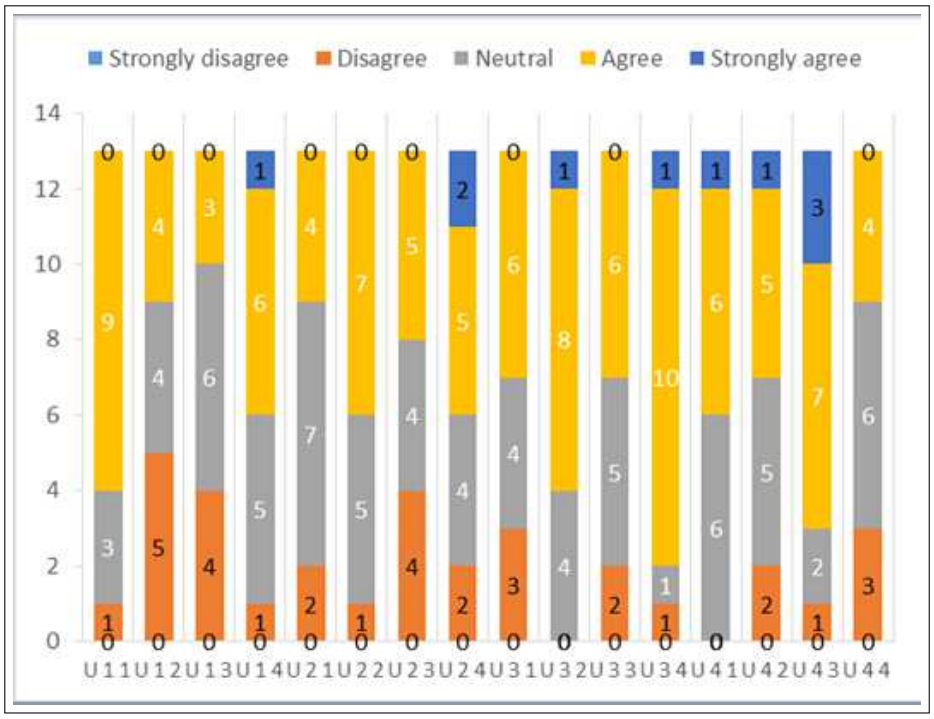


Figure 6: 6 S Assessment scale

We can see the factor “attitude of users" get the highest score (60.5846). Factor “implementation results" follows (57.0085). The other two factors “matching degree "(52.2384) and “functionality" (50.5253) is not ideal, which are lower than the average score (55.0677). This result is meaningful for company A’s management. They should focus on improving the matching degree and the system’s functionality in the future.

Regarding the factor “matching degree", we can see that in most cases, company’s organizational structure and process mismatch the ERP system’s functionality. When there exist mismatches between ERP system and company’s business process, what to do depends on different situations. On the one hand, if the operation of ERP system is inefficient, company can ask the vendors to adjust the system to adopt company. On the other hand, if the ERP system can improve more efficient performance, then company can make appropriate changes on the operation process to adapt to ERP system.

Regarding the factor “functionality", although company A pays much attention to system’s functionality during the selection, we still find that the respondents is not very satisfactory with the functionality. It reflects that there exists difference between the effects after t implementation and expectations. Company A should fully understand the ERP is a long-term project and it is

ongoing to look for problems and put forward the solution. The project team needs to stay close to the ERP vendors and solve the problems together.

5 Conclusions

It is necessary to use the proper process control method during the implementation of a successful ERP system project. This research focuses on the selection and performance evaluation of ERP system.

We study the criteria of the ERP system selection and develop a framework to select ERP system based on AHP method. We combine the objective and criteria, then compare the importance of attributes among criteria and alternatives, which represents the opinion of different evaluators from different departments. Finally, we select the most appropriate ERP system.

The selection based on AHP helps the ERP system match with the strategies of the company. With the help of AHP, we can divide the goal of company into simple ones. This help the goal be put into practice. The selection framework based on AHP could be adjusted according to the development of company and has a high degree of flexibility.

After the selection of ERP system, we study the evaluation of ERP implementation in company A. We combine the Likert scale, AHP and FIE methods, from four dimensions (the system's functionality, the matching degree, implementation results and attitude of users), to evaluate the implementation of ERP project. We find that objective and accurate evaluation of the ERP implementation can help company allocate resources.

We develop an evaluation framework based on FIE and also use AHP to determine the weights, which reducing the subjectivity of evaluation. Regarding the future work, creating a practical decision support software package could serve. This can be valuable for companies facing similar decision-making problems as company A, which we have extensively studied in this paper.

Acknowledgement

This paper is supported in part by National Natural Science Foundation of China (No. 71571194, 71301032, 71201175), and Excellent Young Teachers Program of Guangdong Universities and Colleges (YQ2015014).

Bibliography

[1] Borne P., Popescu D., Filip F. G., Stefanoiu D., Dubuisson B.(2013); *Optimization in Engineering Sciences: Exact Methods*, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2013.

[2] Elisabeth J. U., Ronald R. H., Umble M. M. (2003); Enterprise resource planning: Implementation procedures and critical success factors, *European Journal of Operation Research*, 146, 241-257, 2003.

[3] Filip, F. G. (2011); Designing and building modern information systems; A series of decisions to be made, *Computer Science Journal of Moldova*, 19(2), 119-129, 2011.

[4] Filip F. G. (2014); A decision-making perspective for designing and building information systems, *International Journal of Computers Communications & Control*, 7(2), 264-272, 2014.

[5] Lai V.S., Wong B.K., Cheung W. (2002); Group decision making in a multiple criteria environment: A case using the AHP in software selection. *European Journal of Operational Research*, 137(1), 134-144, 2002.

[6] Moriso M., Tsoukias A. (1997); JusWare: A methology for evaluation and selection of software roducts, *IEE Proc-Softw. Eng.*, 144(2), 162-174, 1997.

[7] Nicolaou A. I. (2014); Firm performance effects in relation to the implementation and use of enterprise resource planning systems. *J Information System*, 18, 79-105, 2014.

[8] O'Leary D. (2000); *Enterprise resource planning system: system, life cycle, electronic commerce and risk*. New York: Cambridge University Press, 2000.

[9] Panorama Consulting (2014); The 2014 Manufacturing ERP Report. http://panorama-consulting.com/resource-center/erp-industry-reports/

[10] Panorama Consulting (2014); *A Panorama Consulting Solution 204 ERP Research Report.* http://Panorama-Consulting.com/resource-center/2014-erp-report/.

[11] Poston R., Grabski S. (2001); Financial impacts of enterprise resource planning implementation. *Int J Account Inf Syst*, 2(4), 271-94, 2001.

[12] Saaty T. L. (1979); Application of analytical hierarchies, *Mathematics and Computers in Simulation*, 21(1), 1-20, 1979.

[13] Shang, S., Seddon, P. (2002); Assessing and managing the benefits of enterprise system: the business manager's perspective. *Information System Journal*, 20(12), 271-299, 2002.

[14] Siriginidi S.R. (2000); Enterprise resource planning in re-engineering business. *Business Process Management Journal*, 6(5), 376-391, 2000.

[15] Teltumbde A. (2000); A framework of evaluating ERP projects. *International Journal of Prodcution Research*, 27(8), 12-16, 2000.

[16] Wang Y., Niu B., Guo P. (2013); On the advantage of quantity leadership when outsourcing production to a competitive contract manufacturer. *Production and Operations Management*, 22 (1), 104-119, 2013.

[17] http://www.technologyevaluation.com/products-and-services/our-proven-approach/

[18] Zadeh L.A. (1999); Fuzzy logic and the calculi of fuzzy rules, fuzzy graphs, and fuzzy probabilities. *Computers & Mathematics with Applications*, 37(11-12), 35, 1999.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 365-380, June 2017.

Computational Intelligence-based $PM_{2.5}$ Air Pollution Forecasting

M. Oprea, S.F. Mihalache, M. Popescu

Mihaela Oprea*, Sanda Florentina Mihalache, Marian Popescu
Automatic Control, Computers and Electronics Department
Petroleum-Gas University of Ploieşti
Romania, 100680 Ploieşti, Bd. Bucureşti, 39
{mihaela, sfrancu, mpopescu}@upg-ploiesti.ro
*Corresponding author: mihaela@upg-ploiesti.ro

Abstract: Computational intelligence based forecasting approaches proved to be more efficient in real time air pollution forecasting systems than the deterministic ones that are currently applied. Our research main goal is to identify the computational intelligence model that is more proper to real time $PM_{2.5}$ air pollutant forecasting in urban areas. Starting from the study presented in [27][a], in this paper we first perform a comparative study between the most accurate computational intelligence models that were used for particulate matter (fraction $PM_{2.5}$) air pollution forecasting: artificial neural networks (ANNs) and adaptive neuro-fuzzy inference system (ANFIS). Based on the obtained experimental results, we make a comprehensive analysis of best ANN architecture identification. The experiments were realized on datasets from the AirBase databases with $PM_{2.5}$ concentration hourly measurements. The statistical parameters that were computed are mean absolute error, root mean square error, index of agreement and correlation coefficient.
Keywords: computational intelligence, $PM_{2.5}$ air pollution forecasting, ANFIS, ANN, ANN architecture identification.

[a]Reprinted (partial) and extended, with permission based on License Number 3957050363449 [2016] ©IEEE, from "Computers Communications and Control (ICCCC), 2016 6th International Conference on".

1 Introduction

This paper is an extension of [27] (doi: 10.1109/ICCCC.2016.7496746). A comprehensive comparative study is here presented. In addition, an extended analysis for the identification of best neural network architecture is included. We report our new experimental results.

Air pollution forecasting is an important research topic especially for the improvement of life quality in cities. Among currently used forecasting methods, computational intelligence methods proved to be more efficient in real time forecasting systems. The deterministic methods which take into account many variables related to the forecasted parameter and use a precise mathematical model with embedded physical and chemical factors (e.g. those based on climate models) give better solutions, but in a longer period of time. In contrast, computational intelligence based methods are approximate methods, that give solutions with a good forecasting accuracy, in short periods of time. Thus, the real time forecasting systems used for urban population early warning of air pollution episodes occurrence can be based on computational intelligence techniques.

Artificial intelligence (AI) provides several techniques for building forecasting systems, mainly from its computational intelligence part and less from its symbolic part. Such applications in different domains were reported in the literature, most of them in the economic, energy and environmental fields [1], [18], [25]. Symbolic AI is used as a knowledge based approach in selecting some important parameters that influence the forecasting systems performance, usually for the prediction model features selection. On the other hand, computational intelligence techniques

can be used as effective predictors' builders (see e.g. [5], [12], [16], [29], [30]). An important environmental problem that needs better solutions nowadays is urban air quality improvement and reducing human health effects due to air pollution in cities. For this, good real time air pollution short-term forecasters have to be developed and included in the environmental management systems or in the early warning system of intelligent environmental decision support systems. Particulate matter with diameter less than 2.5 μm ($PM_{2.5}$) is an air pollutant that has potential negative effects on human health, when its concentration exceeds the admissible standard upper level. Our research work focuses on the development of a good real time $PM_{2.5}$ air pollution forecasting model that will be integrated in the ROKIDAIR Decision Support System (ROKIDAIR DSS) to be used in two pilot Romanian cities, Ploieşti and Târgovişte, by the ROKIDAIR Early Warning System. Starting from a literature review and a comparative study between most used computational intelligence based forecasting methods, we have selected the best model which is a neural network model and we have performed an analysis of best neural network architecture identification via trial and error method.

Computational intelligence is a paradigm introduced in [2] which combines mainly three computing technologies: fuzzy computing, neural computing and evolutionary computing. Fuzzy computing and neural computing are used as forecasting models, while evolutionary computing can be applied mainly for the optimization of a forecasting model. In the last decade, other nature-inspired computing methods were added to computational intelligence, such as swarm intelligence with various techniques: ant colony optimization (ACO), particle swarm intelligence (PSO), artificial bee colony algorithm (ABC) etc. These last techniques are commonly used for optimizing the forecasting model and not as a forecasting model.

The remainder of the paper is organized as follows. In Section 2 it is described computational intelligence based forecasting focusing on most used techniques: artificial neural networks and adaptive neuro-fuzzy inference system (ANFIS). A comparative study of the two techniques (ANN and ANFIS) applied to $PM_{2.5}$ air pollution forecasting is presented in Section 3, concluding that the best experimental results were obtained by the ANN forecasting model. The identification of the best $PM_{2.5}$ ANN forecasting architecture is discussed in Section 4. The final section concludes the paper and highlights some future work.

2 Computational intelligence based forecasting

Computational intelligence provides data-driven methods. The neural methods applicable to solve forecasting problems are: artificial neural network (ANN) and adaptive neuro-fuzzy inference system (ANFIS). They can perform air pollution forecasting more efficiently than the deterministic methods by capturing the knowledge accumulated in the historical data sets (time series) which is learned via a training algorithm and is used to accurately predict specific air pollution parameters (e.g. air pollutants concentrations).

2.1 Artificial Neural Networks (ANN)

Artificial neural networks are universal approximators of non-linear functions [14]. They are composed by a number of non-linear processing units named artificial neurons which are structured in layers. Forecasting problems are solved mainly by feed-forward ANNs (multi-layer perceptron - MLP and radial basis function - RBF) and recurrent ANNs. Figure 1 shows the general architecture of a feed-forward ANN, which has an input layer, some hidden layers and an output layer.

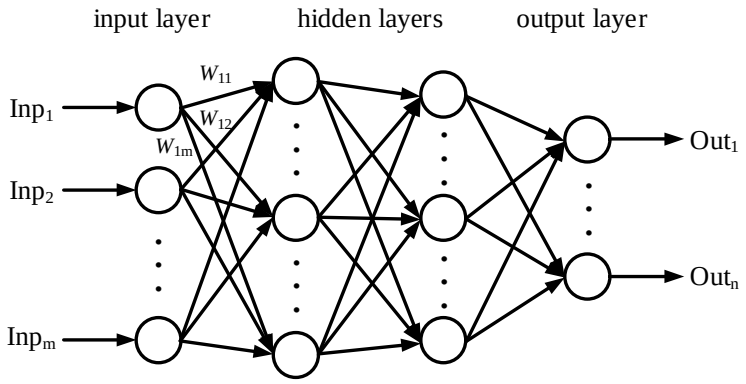


Figure 1: The general architecture of a feed-forward ANN

A recurrent neural network is a type of artificial neural network that has a directed cycle made by the connections between the artificial neurons, allowing it to have an internal state. Thus, it exhibits a dynamic behaviour which provides the ability to process and predict chaotic time series for long-terms [19]. A recurrent ANN is an ANN that has feedback, allowing arbitrary connections between neurons, both forward and backward (i.e. recurrent). Thus, it propagates data bi-directional, from input to output and from output to input. Recurrent ANNs are universal approximators [34]. They provide a very good performance in temporal structures modeling, as well as in real world problem solving ([4], [6]). Recurrent ANNS exhibit a dynamic temporal behavior and can process arbitrary sequences of inputs (e.g. chaotic time series).

The best structure of an ANN is experimentally determined. Usually, a single hidden layer is enough to capture the nonlinearity of any function. Deep networks are ANNs with several hidden layers. The number of hidden nodes is chosen by experiments, while the number of input and output nodes is set according to the forecasting problem that has to be solved. The number of input nodes represents the input window (in the case of time series, the number of past hours measurements) and the number of output nodes represents the forecast horizon (number of future time steps, hours, days etc., for which the prediction is determined). The ANN is trained with a training set (which is extracted from a data set) by using a specific training algorithm (the most used being backpropagation and the Levenberg-Marquardt algorithm), after that following the validation and testing steps which are executed on the validation set and testing set, respectively. Details on the ANN computational algorithms are given in the literature (see e.g. [13]). The Levenberg Marquardt algorithm [20] is an iterative algorithm that estimates the weights vector of the ANN model by minimizing the sum of the squares of the deviation between predicted and target values. If ANN training is too long then overfitting can occur. To avoid this, the ANN training is stopped earlier, as soon as the performance on testing data is not improved any more.

2.2 ANFIS

The ANFIS method applied to prediction uses a hybrid architecture composed by a fuzzy inference system FIS enhanced with ANN features proposed by Jang [15]. The advantages of FIS are mainly its design that emulates human thinking and the simple interpretation of the results. Integrating the ANN part into a fuzzy inference system enhanced the FIS part with learning/adapting capabilities. The prediction model does not use a mathematical model as well as the case of ANN. The ANFIS architecture has the structure given in Figure 2.

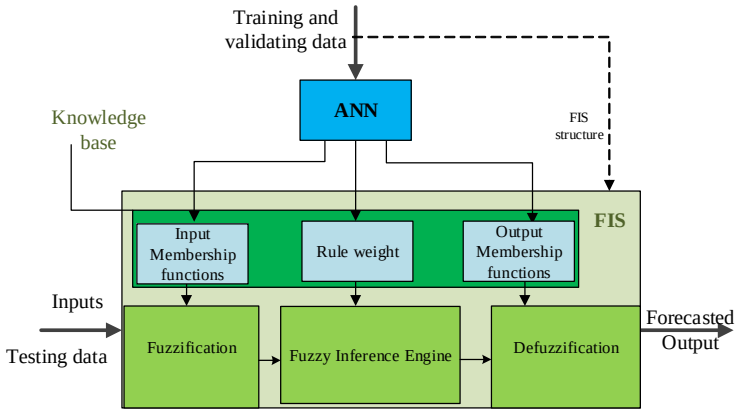


Figure 2: The general architecture of ANFIS

The FIS part is formed by five functional units: a fuzzification unit (from crisp value to fuzzy set), a defuzzification unit (from fuzzy set to a crisp value), the database unit (containing the description of membership functions for input/output variables), a rule base unit (all the rules defined for FIS), and the decision unit (performing the inference operations on the fuzzy rules) [26]. The neuro-fuzzy architecture is capable to learn new rules or membership functions, to optimize the existing ones (Figure 2). The training data determine restrictions on the design methods for the rule base and membership functions. Usually the particular type of datasets for $PM_{2.5}$ eliminates the subclustering method in generating the FIS structure, a good choice being the grid partition method.

The ANFIS architecture (Figure 2) has five layers, with Takagi-Sugeno rules. The first layer (adaptive) forms the premise parameters (the IF part with inputs and their membership functions). The second layer computes a product of the involved membership functions. The third layer normalizes the sum of inputs. In layer 4, the adaptive i-node computes the contribution of i-th rule to ANFIS output, forming the consequence parameters (the THEN part with output and its membership function). The fifth layer makes the summation of all inputs. The ANN part can improve the membership functions associated with FIS structure. Usually these membership functions are the tuning parameters of the FIS. Their initial values are chosen from experience or trial and error methods. In the training mode the ANN finds the most suited membership functions for the input-output relation described by FIS, according to training and checking dataset.

ANFIS applies a hybrid learning algorithm (H) or backpropagation (BP) algorithm. The hybrid learning algorithm identifies premise parameters with gradient method and consequence parameters with least square method. At feedforward propagation step from H, the system output reaches layer 4, and the consequence parameters are formed with least square method. With backpropagation (BP) optimization method, the error signal is fed back and the new premise parameters are computed through gradient method. The prediction method is tested with datasets that respect the main features of training dataset. The prediction precision decreases with enlarging the prediction window from one hour in advance to six hours in advance.

2.3 An overview on $PM_{2.5}$ computational intelligence based forecasting

The main computational intelligence techniques that were used for $PM_{2.5}$ forecasting are: feed forward ANN, radial basis function ANN, recurrent ANN, and ANFIS. Other AI techniques such as genetic algorithms and swarm intelligence were applied to optimize the forecasting model. We have selected some $PM_{2.5}$ forecasting systems based on computational intelligence that were

reported in the literature.

In the first years of the current century, the $PM_{2.5}$ forecasting ANN models (usually, of MLP type) were compared mostly with statistical models such as linear regression, ARMA, ARIMA, revealing a very good performance of the neural models. One of the earlier $PM_{2.5}$ forecasting neural models were proposed in [21] and [31]. The first ANN model was applied in Canada, while the latter was applied in Santiago de Chile. The experimental results described in both papers showed a very good performance of the ANN model in comparison with the traditional statistical models (e.g. linear regression).

In the next years, various comparisons between different types of ANNs models used to $PM_{2.5}$ forecasting were performed. For example, an analysis of three ANN models (MLP, RBF-ANN and square MLP) applied to PM2.5 short term prediction in an area on the US-Mexico border is presented in [28]. The experimental results revealed that for the analyzed area the RBF-ANN model outperformed the other two models. Another work that analyzes the performance of different neural network models and regression model applied to forecasting expressway fine PM (i.e. $PM_{2.5}$) in Indiana, USA is described in [35].

A feed forward ANN with backpropagation training algorithm is described in [**?**], for 3 days in advance forecasting of PM_{10}, SO_2 and CO air pollutants (AP) levels in the Besiktas district in Istanbul, Turkey. The ANN is integrated in the AirPol system (http://airpol.fatih.edu.tr). The ANN inputs are daily meteorological forecasts and the AP indicator values. The authors applied some geographical models, the most complex one being based on the distance between two sites in the case of using three selected neighborhood districts.

In [10] it is demonstrated the efficacy of using EnviNNet, a prototype stochastic ANN model for air quality forecasting in cities from Italy (Rome, Milan and Napoli) to predict PM_{10} in Phoenix, Arizona in comparison with the use of CMAQ system. The ANN is a MLP that uses the conjugate-gradient method for training.

In the last years, the research work was focused on combining $PM_{2.5}$ forecasting ANN models with other techniques, such as statistical techniques, data mining, genetic algorithms, wavelet transformation, deep learning, that can improve the forecasting accuracy. Some examples are briefly described as follows. A research work that reports the successful use of ANNs and principal component analysis for PM_{10} and $PM_{2.5}$ forecasting in two cities, Thessaloniki (Greece) and Helsinki (Finland) is described in [36]. The authors used a MLP and the meteorological and AQ pollutants to predict the next day mean concentration of PM_{10} and $PM_{2.5}$. A genetically optimized ANN and k-means clustering was applied in [8] to predict PM_{10} and $PM_{2.5}$ in a coastal location of New Zealand. A hybrid $PM_{2.5}$ forecasting model that uses feed forward ANN combined with rolling mechanism and accumulated generating operation of gray model, that was experimented in three cities from China is introduced in [11]. Another recent work on $PM_{2.5}$ neural forecasting is described in [9]. The authors propose a novel hybrid model that combines air mass trajectory analysis with wavelet transformation in order to improve the accuracy of the average $PM_{2.5}$ concentration two days in advance neural forecasting. The ANN is a MLP trained with a backpropagation algorithm and the Levenberg-Marquardt (LM) algorithm. Also, early stopping was used to avoid overfitting. Some meteorological parameters were used. Finally, one of the newest achievements for $PM_{2.5}$ prediction in 52 cities from Japan is reported in [24]. Each city has several thousands of $PM_{2.5}$ sensors. A deep recurrent neural network (DRNN) is proposed for $PM_{2.5}$ prediction based on real data sensors, which applies a novel pre-training method (DynPT) using time series prediction especially designed auto-encoder. The experimental results showed that the DRNN model outperformed the current climate models used in Japan, which are based on Eulerian and Lagrangian grids or Trajectory models.

The use of pure FIS for specific air pollutants is reported in literature. In [3] it is presented a study for Mexico City air pollution. The paper present a solution of classifying different air

parameters via fuzzy reasoning and include this solution in the air quality index calculation. In [7] it is formulated a fuzzy based forecasting model used in Polish Environmental Agency, a model that forecast specific air pollutants based on decades of measurements. Examples of ANFIS based systems for the prediction of air pollutants concentrations are presented in [22], [23], [32] and [33]]. In [23] it is proposed a fuzzy inference system to forecast $PM_{2.5}$ concentrations at specific hours using as additional input the medium temperature. In [33] it is developed a fuzzy inference system to forecast ozone concentrations levels based on other pollutants and meteorological parameters. The provided FIS model accuracy has the best values for coefficient of determination. For the city of Konya, Turkey the literature present the solution of ANFIS forecasting model for PM_{10} trained with large datasets [32]. In [22] there are presented three case studies from three different cities from Romania. Each time the proposed ANFIS forecasting model for PM_{10} is tested and there are made recommendations on how to adjust the model parameters to improve forecasting accuracy.

The main conclusion of the literature overview is that an efficient $PM_{2.5}$ forecasting ANN model can be derived only by experiment, depending on the $PM_{2.5}$ measurements data sets and several characteristics (climatic, geographic, industrial, economic, social etc) specific to the analyzed area. Thus, there is no pre-set ANN model type for $PM_{2.5}$ that is proper for any area. Another important conclusion is that both ANN and ANFIS can model non-linear time series.

3 A Comparative study of $PM_{2.5}$ forecasting with ANFIS and ANN

3.1 Data sets

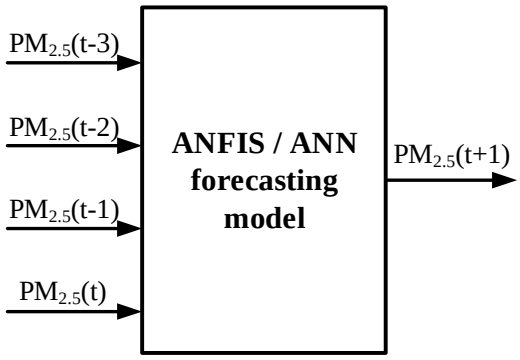


Figure 3: Structure of the proposed architecture

For this study were chosen two urban traffic $PM_{2.5}$ monitoring stations. Taking into consideration the lack of data for the Romanian stations, the data used for training and validation were from a München (Germany) station where sufficient data are available (4 years) in the Airbase database. In addition, the testing was done using also data (from 2015) from Ploieşti (Romania), a city which presents interest as part of a larger project (ROKIDAIR project- www.rokidair.ro). The data set from München contains 34000 samples and the $PM_{2.5}$ hourly concentration has a range of 0.52-168.5 $\mu g/m^3$. The additional testing data from Ploieşti contains 4000 samples and the domain of $PM_{2.5}$ concentration is 3.24-36.45 $\mu g/m^3$.

The data were normalized and then divided (randomly) as follows: 70% for training, 15% for validation and 15% for testing the model.

If the amount of data is large enough, it is supposed that the data contains the effect of other pollutants and meteorological data, so the proposed architecture (Figure 3) for the forecasting

model (ANFIS or ANN) has as inputs the values of $PM_{2.5}$ concentrations from current hour to three hours ago and as output the prediction for the next hour.

In order to determine the suitability of the models the following statistical parameters were calculated: MAE - mean absolute error; RMSE - root mean square error; IA - index of agreement; R - correlation coefficient.

The two errors measure how close the predicted data are to the true values and have to be as small as possible, and the last two indices are numbers that indicate how well the data fit the prediction model and they should be close to 1.

The experiments presented in the following sections were performed using MATLAB© environment.

3.2 Experimental results for ANFIS model

In this study, for the generation of the fuzzy inference system was used the grid partition method, imposed by the specific of data. The membership functions were triangular and Gaussian and the optimization algorithms of the ANN were backpropagation and hybrid.

The ANFIS model was tested for all combinations between the modifiable parameters, namely the membership functions types and the optimization methods.

Table 1 presents the experimental results of the ANFIS model tested with data from München.

Table 1: Statistical indices for ANFIS model (München)

ANFIS Structure	**MAE [$\mu g/m^3$]**	**RMSE [$\mu g/m^3$]**	**IA**	**R**
Trimf/Hybrid	1.9614	3.2564	0.9796	0.9604
Gauss/Hybrid	**1.9419**	**3.2089**	**0.9801**	**0.9616**
Trimf/Backprop.	2.0494	3.3933	0.9778	0.9573
Gauss/Backprop.	2.2177	3.4471	0.9770	0.9556

Analyzing Table 1 it can be seen that the best configuration is when a Gaussian function is used for the membership functions associated to inputs, and for the training of the neural network a hybrid algorithm is used. In this case, the RMSE is the smallest, and the IA and R have the biggest values. The smallest training and validation errors are around 0.02.

The best configuration of the ANFIS structure was also tested with data from the Ploieşti station. The values of the statistical parameters are presented as follows: MAE [$\mu g/m^3$]: 1.0166; RMSE [$\mu g/m^3$]: 1.9160; IA: 0.9711; R: 0.9447.

3.3 Experimental results for ANN model

The structure of the neural network contains four neurons in the input layer, one hidden layer and one neuron in the output layer.

There were used two types of neural networks, namely *feed forward backpropagation (FF)* and *layer recurrent (LR)*, with *Levenberg-Marquardt* as training algorithm, and the adaptive learning functions were gradient descent with momentum weight and bias (*learngdm - LGDM*) and gradient descent weight and bias (*learngd - LGD*). The simulations were performed modifying also the number of neurons in the hidden layer (from three to twelve).

The training and validation errors are with two orders of magnitude smaller than the ones obtained in the case of ANFIS model, with values around 0.0004.

In Table 2 is presented a selection with the values of the statistical parameters for the ANN models. The best configurations of the ANN models are highlighted, being associated to the

layer recurrent structure with 4 neurons in the hidden layer and the *learngd* adaptation learning function, and the feedforward structure with 5 neurons in the hidden layer and the *learngdm* adaptation learning function, respectively. In this case the mean absolute error and the root mean squared error have the smallest values, and IA and R indices have the biggest values.

Table 2: Statistical indices for ANN model with one hidden layer (München)

ANN Structure		**MAE [μg/m^3]**	**RMSE [μg/m^3]**	**IA**	**R**
4x4x1/LGDM	FF	1.9421	3.2086	0.9802	0.9616
	LR	1.9683	3.2359	0.9798	0.9609
4x4x1/LGD	FF	1.9421	3.2086	0.9802	0.9616
	LR	**1.9278**	**3.1931**	**0.9804**	**0.9619**
4x5x1/LGDM	**FF**	**1.9340**	**3.1966**	**0.9804**	**0.9619**
	LR	1.9836	3.2360	0.9797	0.9609
4x5x1/LGD	FF	1.9609	3.2152	0.9800	0.9614
	LR	1.9486	3.2138	0.9801	0.9614
4x6x1/LGDM	FF	1.9415	3.2149	0.9801	0.9614
	LR	1.9676	3.2207	0.9800	0.9613
4x6x1/LGD	FF	1.9471	3.2292	0.9799	0.9611
	LR	1.9519	3.2182	0.9801	0.9613
4x7x1/LGDM	FF	1.9402	3.2185	0.9801	0.9613
	LR	1.9528	3.2252	0.9800	0.9612
4x7x1/LGD	FF	1.9489	3.2128	0.9801	0.9615
	LR	1.9567	3.2224	0.9800	0.9612
4x8x1/LGDM	FF	1.9516	3.2218	0.9800	0.9612
	LR	1.9498	3.2130	0.9801	0.9615
4x8x1/LGD	FF	1.9433	3.2148	0.9801	0.9614
	LR	1.9613	3.2290	0.9799	0.9611
4x10x1/LGDM	FF	1.9657	3.2272	0.9799	0.9611
	LR	1.9527	3.2183	0.9800	0.9613
4x10x1/LGD	FF	1.9568	3.2167	0.9800	0.9614
	LR	1.9791	3.2367	0.9797	0.9609

In addition, as in the ANFIS case, the best configurations of the RNN and FFNN structure were tested with data from the Ploieşti station and the values of the statistical parameters from Table 3 were obtained.

Table 3: Statistical indices for ANN model (Ploieşti)

	MAE [μg/m^3]	**RMSE [μg/m^3]**	**IA**	**R**
RNN	**0.9672**	**1.3713**	**0.9852**	**0.9714**
FFNN	0.9852	1.3970	0.9846	0.9702

The experimental results confirms that the best ANN configuration (RNN) previously obtained provides very good results for the testing data set from Ploieşti.

3.4 Discussion

A comparison between the best values of the statistical parameters obtained for the ANFIS model and the ANN models, respectively, using München datasets is synthesized in Table 4.

Table 4: Comparison between best results

Forecasting Method	**MAE [$\mu g/m^3$]**	**RMSE [$\mu g/m^3$]**	**IA**	**R**
Best ANFIS	1.9419	3.2089	0.9801	0.9616
Best RNN	**1.9278**	**3.1931**	**0.9804**	**0.9619**
Best FFNN	1.9340	3.1966	0.9804	0.9619

It can be observed that the results obtained with ANN are slightly better than the ANFIS results, and corroborated with the fact that for Ploieşti data the statistical indices are much better in the case of ANN (see sections 3.2 and 3.3) and the time for training is much smaller (with at least 10 times) for ANN, it can be concluded that the ANN forecasting model is most suitable for the prediction of $PM_{2.5}$ concentrations. Therefore, in the next section the focus will be on the ANN forecasting model.

4 Best ANN architecture identification for $PM_{2.5}$ forecasting

Usually, an ANN with one hidden layer is enough for any complex nonlinear function (see e.g. [37]). However, in order to make a complete analysis, several ANN architectures with one hidden layer and two hidden layers were experimented. Thus, the selected architectures (RNN and FFNN) with one hidden layer from the previous section were tested with two hidden layers. Also, different adaptive learning functions were used: gradient descent with momentum weight and bias and gradient descent weight and bias, and different number of neurons in the second hidden layer (from three to twelve).

A trial and error method was applied in order to identify the best ANN forecasting model architecture. The best ANN architecture selection was determined using the same statistical indices as in the section 3.

It is interesting to analyze how is influenced the forecasting accuracy (error) of the ANN model by the number of hidden layers and the number of hidden neurons in each hidden layer. More hidden layers involve a deeper learning ability of the ANN model. However, the number of hidden layers is dependent on the application specific data sets (e.g. in our case $PM_{2.5}$ concentration hourly measurements) and must be determined by experiment until optimum prediction (i.e. minimum RMSE).

Table 5 presents a selection of the statistical indices values for an ANN architecture with two hidden layers, which has four neurons in the first hidden layer and 3 to 12 neurons in the second hidden layer.

Table 5: Statistical indices for ANN model with two hidden layers (München))

ANN Structure		**MAE [$\mu g/m^3$]**	**RMSE [$\mu g/m^3$]**	**IA**	**R**
4x4x4x1/LGDM	FF	1.9319	3.2041	0.9803	0.9617
	LR	1.9491	3.2159	0.9801	0.9614
4x4x4x1/LGD	**FF**	**1.9303**	**3.1923**	**0.9804**	**0.9620**
	LR	1.9338	3.2120	0.9802	0.9615
4x4x5x1/LGDM	FF	1.9397	3.2065	0.9802	0.9616
	LR	1.9494	3.2228	0.9800	0.9612
4x4x5x1/LGD	FF	1.9475	3.2192	0.9801	0.9613
	LR	1.9491	3.2258	0.9800	0.9612
4x4x7x1/LGDM	FF	1.9460	3.2292	0.9800	0.9611
	LR	1.9776	3.2450	0.9796	0.9607
4x4x7x1/LGD	FF	1.9291	3.2028	0.9803	0.9617
	LR	**1.9102**	**3.1890**	**0.9805**	**0.9621**
4x4x8x1/LGDM	FF	1.9562	3.2264	0.9799	0.9611
	LR	1.9428	3.2356	0.9800	0.9610
4x4x8x1/LGD	FF	1.9472	3.2131	0.9801	0.9615
	LR	1.9320	3.2061	0.9803	0.9617
4x4x9x1/LGDM	FF	1.9350	3.2149	0.9801	0.9314
	LR	1.9310	3.2062	0.9802	0.9616
4x4x9x1/LGD	FF	1.9353	3.2165	0.9801	0.9614
	LR	1.9630	3.2346	0.9799	0.9609
4x4x10x1/LGDM	FF	1.9359	3.2105	0.9802	0.9615
	LR	1.9481	3.2137	0.9801	0.9615
4x4x10x1/LGD	FF	1.9377	3.2149	0.9802	0.9614
	LR	1.9516	3.2327	0.9799	0.9610

Changing the number of neurons in the first hidden layer from 4 to 5 the results from Table 6 are obtained. The number of neurons in the second hidden layer is the same as in the previous case (from 3 to 12 neurons).

Table 6: Statistical indices for ANN model with two hidden layers (München)

ANN Structure		**MAE [$\mu g/m^3$]**	**RMSE [$\mu g/m^3$]**	**IA**	**R**
4x5x4x1/LGDM	FF	1.9455	3.2076	0.9802	0.9616
	LR	1.9491	3.2159	0.9801	0.9614
4x5x4x1/LGD	FF	1.9512	3.2137	0.9801	0.9614
	LR	1.9338	3.2120	0.9802	0.9615
4x5x5x1/LGDM	FF	1.9627	3.2302	0.9799	0.9610
	LR	1.9489	3.2265	0.9799	0.9611
4x5x5x1/LGD	FF	1.9389	3.2012	0.9802	0.9617
	LR	1.9835	3.2450	0.9796	0.9607
4x5x7x1/LGDM	FF	1.9313	3.2147	0.9802	0.9614
	LR	1.9556	3.2257	0.9799	0.9612
4x5x7x1/LGD	FF	1.9314	3.1957	0.9804	0.9619
	LR	1.9791	3.2499	0.9796	0.9606
4x5x8x1/LGDM	FF	1.9256	3.1926	0.9804	0.9620
	LR	1.9350	3.1961	0.9803	0.9619
4x5x8x1/LGD	FF	1.9380	3.2025	0.9803	0.9617
	LR	1.9446	3.2249	0.9800	0.9612
4x5x9x1/LGDM	FF	1.9505	3.2251	0.9800	0.9612
	LR	**1.9184**	**3.1926**	**0.9804**	**0.9620**
4x5x9x1/LGD	**FF**	**1.9181**	**3.1678**	**0.9807**	**0.9626**
	LR	1.9658	3.2414	0.9797	0.9608
4x5x10x1/LGDM	FF	1.9336	3.2041	0.9802	0.9617
	LR	1.9429	3.2145	0.9801	0.9614
4x5x10x1/LGD	FF	1.9164	3.2053	0.9803	0.9617
	LR	1.9441	3.2136	0.9801	0.9614

Analyzing the results from Tables 5 and 6 it can be observed that the best ANN architecture with two hidden layers is feedforward with five neurons in the first hidden layer, nine neurons in the second hidden layer and gradient descent weight and bias as the adaptive learning function (Figure 4). In this case, MAE and RMSE have the smallest values and IA and R have the biggest values.

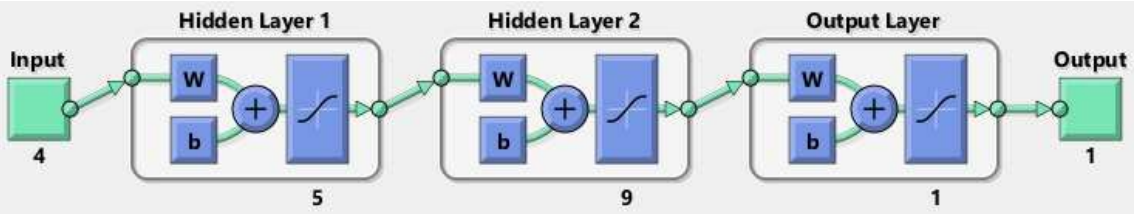


Figure 4: Best ANN architecture

For the best architecture mentioned above are presented in Figures 5 and 6 the testing error evolution and a partial view of the comparison between testing and forecasted data for München data set.

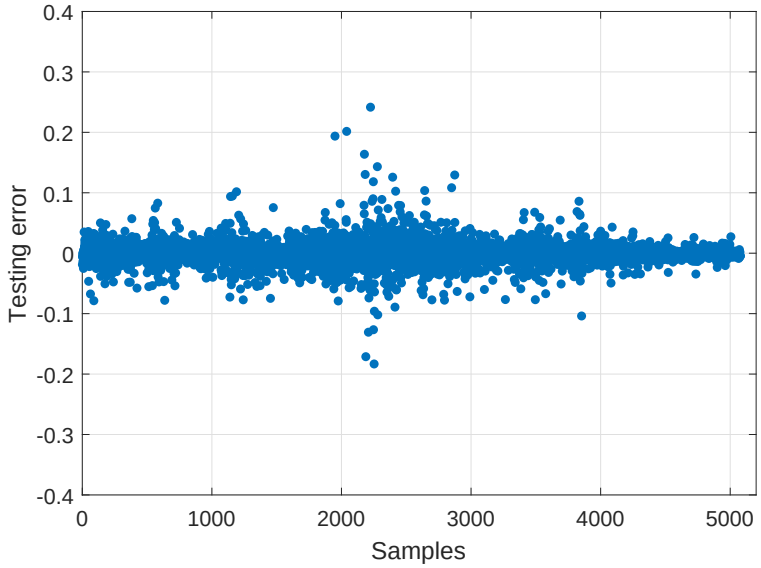


Figure 5: Testing error for best ANN architecture (FF - 4x5x9x1)

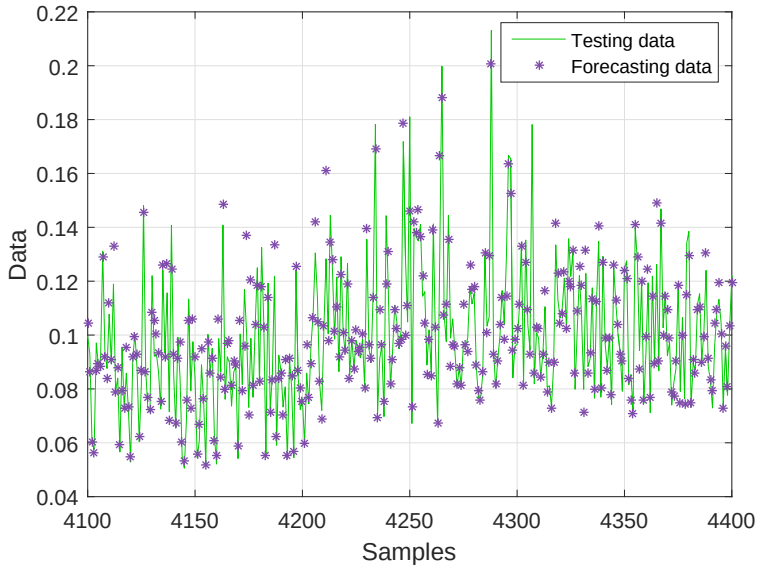


Figure 6: Comparison of testing vs forecasted data for best ANN architecture (FF - 4x5x9x1)

Table 7: Best ANN Architecture

Best ANN architecture	**MAE [$\mu g/m^3$]**	**RMSE [$\mu g/m^3$]**	**IA**	**R**
with one hidden layer	1.9278	3.1931	0.9804	0.9619
with two hidden layers	**1.9181**	**3.1678**	**0.9807**	**0.9626**

Table 7 concludes that using an ANN architecture with two hidden layers improves the results obtained with the ANN architecture one hidden layer suggesting a possible use of this type of structure for $PM_{2.5}$ forecasting.

5 Conclusions and future works

This paper is an extension of [27] which proposed $PM_{2.5}$ air pollution forecasting models based on computational intelligence techniques. Starting from the study presented in [27], in this paper we extend the comparative study between artificial neural networks (ANNs) and adaptive neuro-fuzzy inference system (ANFIS) which are the most accurate computational intelligence models used for particulate matter (fraction $PM_{2.5}$) air pollution forecasting by increasing the number of neurons in the hidden layer of the ANN architecture. We also present an extended overview of computational intelligence techniques based on neural networks approach with up to date solutions of air pollution forecasting.

The data used for training and validation were from a München (Germany) urban traffic station available from the Airbase database with hourly $PM_{2.5}$ concentrations (for 4 years). For the testing of the model data from Ploieşti (Romania), a city which presents interest as part of a larger project (ROKIDAIR project - www.rokidair.ro) were also used. The proposed models have four $PM_{2.5}$ hourly concentrations as inputs and the prediction of the next hour $PM_{2.5}$ concentration. We compared the performances of the models using statistical indices such as mean absolute error, root mean square error, index of agreement and correlation coefficient. The simulation results obtained with ANN are better than the ANFIS results, and corroborated with the fact that for Ploieşti data the statistical indices are much better in the case of ANN and the time for training is much smaller (with at least 10 times) for ANN, it can be concluded that the ANN forecasting model is most suitable for the prediction of $PM_{2.5}$ concentrations. Therefore, the next part of the paper concentrates on finding the best neural network architecture by increasing the number of hidden layer of the ANN architecture. In this new approach there are performed numerous simulations with different architectures starting from the best architectures with one hidden layer obtained in Section 3. Therefore we kept the number of the neurons in the first hidden layer and varied the number of neurons in the second different layers. The paper proposes an ANN model for $PM_{2.5}$ forecasting, whose architecture was identified via trial and error method, during an extended analysis that was performed. The simulation results pointed out that introducing another hidden layer is beneficial in terms of statistical indices that assess the forecasting performance. This is a good premise to find other solutions using deep learning as future work. The model can be used to real time forecasting and is incorporated in the ROKIDAIR DSS, being used by the ROKIDAIR Early Warning System in case some $PM_{2.5}$ air pollution episodes arise in different areas from the two pilot cities: Ploieşsti and Târgovişte. The selection of the most proper computational intelligence based $PM_{2.5}$ forecasting model was made during the comprehensive comparative study of applying the ANN and ANFIS models, which proved to be efficient in real time forecasting systems. The ANN $PM_{2.5}$ forecasting model can detect concentrations above the standard limit values for the human health protection.

As future work we will extend the study for the improvement of the forecasting model performance through deep learning or combining neural networks with wavelets or data mining techniques.

Acknowledgment

The research leading to these results has received funding from EEA Financial Mechanism 2009-2014 under the project ROKIDAIR "Towards a better protection of children against air pollution threats in the urban areas of Romania" contract no. 20SEE/30.06.2014.

Bibliography

[1] Alfaro M.D.; Sepúlveda J.M.; Ulloa J.A. (2013); Forecasting Chaotic Series in Manufacturing Systems by Vector Support Machine Regression and Neural Networks, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 8(1), 8-17, 2013.

[2] Bezdek J.C. (1994); What is computational intelligence?, *Computational Intelligence - Imitating Life*, IEEE World Congress on Computational Intelligence (WCCI), ISBN: 978-0780311046, 1-12, 1994.

[3] Carbajal-Hernández J.J.; Sánchez-Fernández L.P.; Carrasco-Ochoa J.A.; Martínez-Trinidad J.F. (2012); Assessment and prediction of air quality using fuzzy logic and autoregressive models, *Atmospheric Environment*, 60, 37-50, 2013.

[4] Cherif A.; Cardot H.; Boné R. (2011); SOM time series clustering and prediction with recurrent neural networks, *Neurocomputing*, ISSN: 0925-2312, 74(11), 1936-1944, 2011.

[5] Cherkassky V., Krasnopolsky V.; Solomatine D.P., Valdes J. (2006); Computational intelligence in earth sciences and environmental applications: Issues and challenges, *Neural Networks*, 19, 113-121, 2006.

[6] Connor J.T., Martin R., Atlas L.E. (1994); Recurrent neural networks and robust time series prediction, *IEEE Transactions on Neural Networks*, 5(2), 240-254, 1994.

[7] Domańska D., Wojtylak M. (2012); Application of fuzzy time series models for forecasting pollution concentrations, *Expert Systems with Applications*, 39, 7673-7679, 2012.

[8] Elangasinghe M.A.; Singhal N.; Dirks K.N.; Salmond J.A.; Samarasinghe S. (2014); Complex time series analysis of PM_{10} and $PM_{2.5}$ for a coastal site using artificial neural network modeling and k-means clustering, *Atmospheric Environment*, 94, 106-116, 2014.

[9] Feng X.; Li Q.; Zhu Y.; Hou J.; Jin L.; Wang J. (2015); Artificial neural networks forecasting of $PM_{2.5}$ pollution using air mass trajectory based geographic model and wavelet transformation, *Atmospheric Environment*, 107, 118-128, 2015.

[10] Fernando H.J.S., Mammarella M.C., Grandoni G., Fedele P., Di Marco R.; Dimitrova R., Hyde P. (2012); Forecasting PM_{10} in metropolitan areas: Efficacy of neural network, *Environmental Pollution*, ISSN 0269-7491, 163, 62-67, 2012.

[11] Fu M., Wang W., Le Z., Khorram M.S. (2015); Prediction of particulate matter concentrations by developed feed-forward neural network with rolling mechanism and gray model, *Neural Computing & Applications*, 26(8), 1789-1797, 2015.

[12] Gardner M.W., Dorling S.R. (1998); Artificial neural networks (the multilayer perceptron) - a review of applications in the atmospheric sciences, *Atmospheric Environment*, 32, 2627-2636, 1998.

[13] Hertz J., Krogh A., Palmer R.G. (1991); *Introduction to the theory of neural computation*, Addison Wesley, ISBN: 978-0201515602, 1991.

[14] Hornik K., Stinchcombe M., White H. (1989); Multilayer feedforward networks are universal approximators, *Neural Networks*, 2, 356-366, 1989.

[15] Jang R. (1993); ANFIS: Adaptive-Network-Based Fuzzy Inference System, *IEEE Transactions on Systems, Man and Cybernetics*, 23(3), 665-685, 1995.

[16] Karatzas K.D., Voukantsis D. (2008); Studying and predicting quality of life atmospheric parameters with the aid of computational intelligence methods, *International Congress on Environmental Modelling and Software (iEMS)*, 2, 1133-1139, 2008.

[17] Kumar Chandar S., Sumathi M., Sivanadam S.N. (2016); Forecasting Gold Prices Based on Extreme Learning Machine, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 11(3), 372-380, 2016.

[18] Kurt A., Oktay A.B. (2010); Forecasting air pollutant indicator levels with geographic models 3 days in advance using neural networks, *Expert Systems with Applications*, 37, 7986-7992, 2010.

[19] Mandic D., Chambers J. (2001); *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability*, ISBN: 978-0-471-49517-8, Wiley, 2001.

[20] Marquardt D. (1963); An algorithm for least squares estimation of nonlinear parameters, *SIAM Journal of Applied Mathematics*, ISSN: 0036-1399, 11, 431-441, 1963.

[21] McKendry I.G. (2002); Evaluation of artificial neural networks for fine particulate pollution (PM_{10} and $PM_{2.5}$) forecasting, *Journal of the Air & Waste Management Association*, 52, 1096-1101, 2002.

[22] Mihalache S.F., Popescu M.; Oprea M. (2015); Particulate Matter Prediction using ANFIS Modelling Techniques, *Proc. of 19th International Conference on System Theory, Control and Computing (ICSTCC)*, October 14-16, Cheile Gradistei, Romania, ISBN: 978-1-4799-8481-7, 895-900, 2015.

[23] Nebot A., Mugica F. (2014); Small-Particle Pollution Modeling Using Fuzzy Approaches, M.S. Obaidat et al. (eds.), *Simulation and Modeling Methodologies, Technologies and Applications, Advances in Intelligent Systems and Computing*, ISBN: 978-3-319-03580-2, Springer International Publ. 239-252, 2014.

[24] Ong B.T., Sugiura K., Zettsu K. (2016); Dynamically pre-trained deep recurrent neural networks using environmental monitoring data for predicting $PM_{2.5}$, *Neural Computing & Applications*, ISSN: 1433-3058, 27, 1553-1566, 2016.

[25] Oprea M., Buruiana V., Matei A. (2010); A Microcontroller-based Intelligent System for Real-time Flood Alerting, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 5(5), 844-851, 2010.

[26] Oprea M.; Dragomir E.G.; Mihalache S.F.; Popescu M. (2014); Prediction methods and techniques for $PM_{2.5}$ concentration in urban environment (in Romanian), S. Iordache and D. Dunea (eds.), *Methods to assess the effects of air pollution with particulate matter on children's health* (in Romanian), MatrixRom, ISBN: 978-606-25-0121-1, 387-428, 2014.

[27] Oprea M.; Mihalache S.F., Popescu M. (2016); A comparative study of computational intelligence techniques applied to $PM_{2.5}$ air pollution forecasting, *Computers Communications and Control (ICCCC), 2016 6th International Conference on*, ISBN: 978-1-5090-1735-5, 103-108, 2016.

[28] Ordieres J.B., Vergara E.P., Capuz R.S., Salazar R.E. (2005); Neural network prediction model for fine particulate matter ($PM_{2.5}$) on the US-Mexico border in El Paso (Texas) and Ciudad Juárez (Chihuahua), *Environmental Modelling & Software*, ISSN: 1364-8152, 20, 547-559, 2005.

[29] Palit A.K., Popovic D. (2005); *Computational Intelligence in Time Series Forecasting - Theory and Engineering Applications*, Springer, ISBN 978-1-85233-948-7, 2005.

[30] Pasero E., Mesin L. (2010); Artificial Neural Networks to Forecast Air Pollution, In: *Air Pollution*, Vanda Villanyi (Ed.), InTech, ISBN: 978-953-307-143-5, 221-240, 2010.

[31] Peréz P., Trier A., Reyes J. (2000); Prediction of $PM_{2.5}$ concentrations several hours in advance using neural networks in Santiago, Chile, *Atmospheric Environment*, ISSN 1352-2310, 34, 1189-1196, 2000.

[32] Polat K., Durduran S.S. (2012); Usage of output-dependent data scaling in modeling and prediction of air pollution daily concentration values (PM_{10}) in the city of Konya, *Neural Computing and Applications*, ISSN: 1433-3058, 21, 2153-2162, 2012.

[33] Savić M., Mihajlović I., Arsić M., Živković Z. (2014); Adaptive-network-based fuzzy inference system (ANFIS) model-based prediction of the surface ozone concentration, *Journal of the Serbian Chemical Society*, ISSN 1820-7421, 79(10), 1323-1334, 2014.

[34] Schäfer A.M., Zimmermann H.-G. (2006); Recurrent neural networks are universal approximators, *ICANN, 16th International Conference, Proceedings*, ISBN 978-3-540-38627-8, 1, 632-640, 2006.

[35] Thomas S., Jacko R.B. (2007); Model for forecasting expressway fine particulate matter and carbon monoxide concentration: application of regression and neural network model, *Journal of Air Waste Management Association*, ISSN: 2162-2906, 57(4), 480-488, 2007.

[36] Voukantsis D.; Karatzas K.; Kukkonen J.; Räsänen T.; Karppinen A.; Kolehmainen M. (2011); Intercomparison of air quality data using principal component analysis and forecasting of PM_{10} and $PM_{2.5}$ concentrations using artificial neural networks, in Thessaloniki and Helsinki, *Science of the Total Environment*, ISSN: 0048-9697, 409, 1266-1276, 2011.

[37] Zhang G., Patuwo B.E.; Hu M.Y. (1998); Forecasting with artificial neural networks: the state of the art, *International Journal of Forecasting*, ISSN: 0169-2070, 14(1), 35-62, 1998.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 381-392, June 2017.

A Conceptual Framework for Artificial Creativity in Visual Arts

D. Sirbu, I. Dumitrache

Daniela Sirbu
Department of New Media
University of Lethbridge
Lethbridge, Canada
daniela.sirbu@uleth.ca

Ioan Dumitrache
Department of Automatic Control and Systems Engineering
University Politehnica Bucharest
Bucharest, Romania
ioan.dumitrache@acse.upb.ro

Abstract: The present paper introduces the conceptual framework for an artificial system for visual creativity addressing the idea of niche creativity that is domain specific and non-anthropocentric in its conceptual approach. We think that the visual creative output of the system reflects the artificial medium and the specific artificial processes engaged in its production and, therefore, it is an expression of the idea of embodied creativity with the proposed system offering in this sense an example of digital embodiment of creativity. Although our approach to artificial creativity is non-anthropocentric, the system design is inspired by processes in the natural world that lead to the production of new and useful structures in both living and non-living systems with human creative cognition being included among these processes. The main problem raised by this abstract approach to artificial creativity in visual arts is the compatibility of its artistic production with human aesthetics, the ultimate goal of the proposed system being to produce visual output that would aesthetically engage human visual perception.
Keywords: artificial creativity, computational creativity, stochastic processes, evolutionary computing, multiagent systems.

1 Introduction

Most approaches in computational creativity are concerned with emulating human creative cognition or processes of natural structure formation and growth in living organisms and non-living systems. This goes hand in hand with an equally great effort to understand in more depth these processes and the material base from which they operate and which conditions them. These approaches raise the question if human creative cognition can be emulated in an artificial medium and if such a transfer is the most effective approach to computational creativity. The same question is raised when transferring processes of structure formation and growth from the natural world into the computational medium with the purpose to produce visual output of artistic value.

Our conceptual approach to the design of an artificial system for visual creativity (ASVC) is different from this general trend in the sense that it is more holistic and is abstract in the interpretation of creativity, but remains more specific to the computational medium that embodies it. The purpose of this research is to demonstrate that no matter the form in which creativity is embodied, it is based on stochastic processes at its core, although the nature of its embodiment may affect the creative product due to the nature of influences exercised throughout its development.

We aim to demonstrate that the stochastic nature of processes involved in creativity is what makes this process unpredictable, mysterious, and intimidating and we hope to bring a small contribution in debunking the myth of creativity and accept creativity as a natural part of everyday life just as much as intelligence and the very fact of being into the world are natural processes that unfold in a general and continuous process of change.

We recognize the importance of approaching creativity research in interdisciplinary manner considering perspectives from cognitive psychology, neurobiology, learning, and complex systems as it has been previously suggested [13]. This approach allowed identifying some of the main aspects involved in creativity like: memory, divergent thinking, convergent thinking, and flow [13], which appear in previous theories and models of creativity [9], [1], [3], [10], [11], [14], [15], [23], [27], [37], although terminology and descriptions may vary. We also take into consideration Limb's viewpoint [13] that a general theory of creativity is hard to define because creativity is viewed as a "complex" with a multitude of facets involved. However, we think that there are some general principles and processes across many forms, if not all forms, of creativity in living and non-living systems. Our purpose is to identify these general principles and processes and then test their validity empirically through an artificial system for visual creativity. We can then go back to particular forms of embodied creativity, among which human creativity is most important, and analyze their specifics while backed by principles identified through our holistic approach. We also hope this approach provides the basis for a practical approach to developing systems of specialized creativity which are less complex, but effective in their application domains.

We present in this paper the fundamental concepts at the basis of ASVC design and the hypothesis that underlies our approach. We briefly review the means to implement these concepts into an artificial system for visual creativity, and synthetically discuss our empirical approach in testing and interpretation of results and how these support our initial hypothesis.

2 Background

Based on the general problematics identified in creativity research, we synthesize several aspects that are more relevant to our conceptual approach. There is a model of creativity based on four main stages [47] identified as preparation, incubation, illumination, and verification. This model is largely based on narratives from creative people [31] and therefore it is rejected by some authors [48], [49]. However, the model has continued to gain recognition in creativity research [23], [24], [25], [16], [14], [30] up to the present day. In our perspective, the Wallas [47] model is important in emphasizing stochastic processes at the core of incubation and illumination stages when the creative product is conceived. This view received support from psychoanaltic theories of creativity [17], [18] and through experimental data from more recent research [23], [24]. In this sense, we point out that the concept of adaptive regression [17], [18] describing the generation of new ideas in a process of shifting cognition on a continuum between consciousness and subconsciousness. This shift during creative cognition leads to a state where rules over the knowledge domain are weakened allowing free associations and combinations of mental structures. Therefore the generation of new ideas takes place through stochastic processes under some relaxed influences from the knowledge domain. The domain influence is manifested through the nature of the pre-existing mental structures and the limitations on the associations and combination between them.

Furthermore, behaviorist [4], [37], historiometric [38], [39], [44], and systemic [7], [8] approaches in creativity research link creativity to external factors and lead to evolutionary views of creativity. Natural evolution is intrinsically based on stochastic processes under environmental influences and therefore creativity is viewed as such an adaptive process.

In summary, we emphasize that interpretations of creativity as both intrapsychic and extrapsychic process link creativity to stochastic processes under influences that ultimately express in weak form the domain knowledge. This aspect is taken into consideration in computational models of creativity. It has been recognized [2], [30]- [35] that computational models of creativity based on evolutionary computing paradigms typically build on the Wallas model [47]. Some of these models emulate natural evolution processes without particular reference to human cognition [41], [42]. Other models in this category focus on certain known or assumed processes in human creative cognition e.g. analogy making [28], [29], [14], [29], [22], or curiosity [34], while other models focus on the evolutionary paradigm and define a systematic theory for the development of self-improving algorithms as models of innovation [12].

3 Fundamental concepts in ASVC design

In our approach to artificial creativity, a central idea is that creativity reflects in its processes and output the organization of matter and processes that lead to structure formation in living and non-living natural systems.

We believe that most processes in the natural world are stochastic in nature and pressures exercised due to physical phenomena lead to the formation of structures at all dimensional scales and complexities. These structures have intrinsic usefulness within the systems where they emerge. However, the notion of usefulness is a human construct and responds to human value systems. From a very holistic standpoint, there is nothing more or less useful in nature in the process of change that unfolds on a continuous basis. Forms that seem to be destroyed or disappear under the influence of natural forces are, in fact, simply changing to become part of new structures. Sometimes, or most of the times, we cannot even comprehend these new structures and the processes that lead to their formation due to their very small or very large scales in relation to our plan of observation. This is best described by the fractal theory advanced by Mandelbrot [20] and further developed by other authors [5], and which describes features of self-similarity in natural forms at different scales in the visible world.

Therefore, we believe that creativity, in its strict definition as a process that produces new and useful artifacts, is intrinsic throughout the entire visible living and non-living systems as a process of change and structure formation. As similar ideas are expressed in most areas of human exploration, it is natural to turn to practical means made available in control engineering and computer science in order to empirically explore artificial forms of creativity in visual arts.

The fundamental concept in our approach to ASVC design is to create artificial stochastic processes which are then exposed to the influence of forces within the system leading to structure formation in the artificial world. Our hypothesis is that if influences manifested on stochastic processes are integrated into an aesthetic system that governs loosely the artificial world, then there is a high probability that emerging structures are organized in visual compositions with aesthetic value. This way we unify in ASVC design the stochastic nature of artificial processes with a system of influences that respond to human aesthetics and, therefore, to the idea of usefulness of the ASVC output deeming the system to be creative.

The research approach in exploring these questions is empirical. It is based on applying the principles that we hypothesize to underlie creativity to the development of artificial systems that are expected to have creative capabilities expressed in the production of new and aesthetically valuable visual compositions. These systems are then tested experimentally and results are analyzed based on an aesthetic system adapted to the specifics of artificial creativity.

While the proposed system for artificial creativity is abstract in the sense that natural processes are not closely emulated, the system design is, however, inspired by principles and processes, which are effective in producing new forms in the visible living and non-living systems in

the real world.

In doing so, we approach the design of the artificial system for visual creativity using stochastic models based on random walk algorithms and evolutionary computation and create artificial ecosystems in which aesthetic principles are manifested through relaxed pressures on the artificial stochastics. The main research contribution is based on the idea of embedding aesthetic knowledge in the system in a relaxed way that allows great stochastic freedom. This is key for the ASVC capability to generate a large number of new visual compositions with aesthetic value. In this approach creativity is viewed as an adaptive process embodied in an artificial ecosystem with stochastic substrate and functioning for aesthetic performance.

4 General requirements for an artificial system for visual creativity

Based on the problematics identified in creativity research and computational creativity systems, we synthesize the following set of requirements for the ASVC design:

- The system must incorporate in some form specialized domain knowledge, which, in this case, is visual aesthetics.
- ASVC processes that lead to the production of new visual compositions must reflect the knowledge domain.
- The same processes that lead to the formation of new visual structures must incorporate stochastic aspects.
- The system must implement a digital process that effectively executes the drawing/painting as a counterpart of the drawing process in real life.
- ASVC must develop its own visual concepts.
- The system must integrate the computational concept development process and the artificial drawing process.
- The system must develop new visual output through computational processes that are adaptive in response to influences from a given artificial environment.
- The ASVC creative process must be best adapted to the computational medium that embodies it.

These requirements define ASVC creativity as an abstract process that liberally uses a blend of examples of structure formation in the living and non-living systems in the real world, but it is focused on the final product in the sense of synthetic creativity [19], [6] that is typical in the artistic domain.

5 ASVC general architecture

Considering the requirements formulated above for an artificial system for visual creativity, we suggest a generic architecture (Figure 1) that can be particularized for many versions of ASVC depending on computational paradigms employed for the implementation of various components in its structure. This architecture is based on two main components as follows:

- The kinetic drawing systems (KDS).

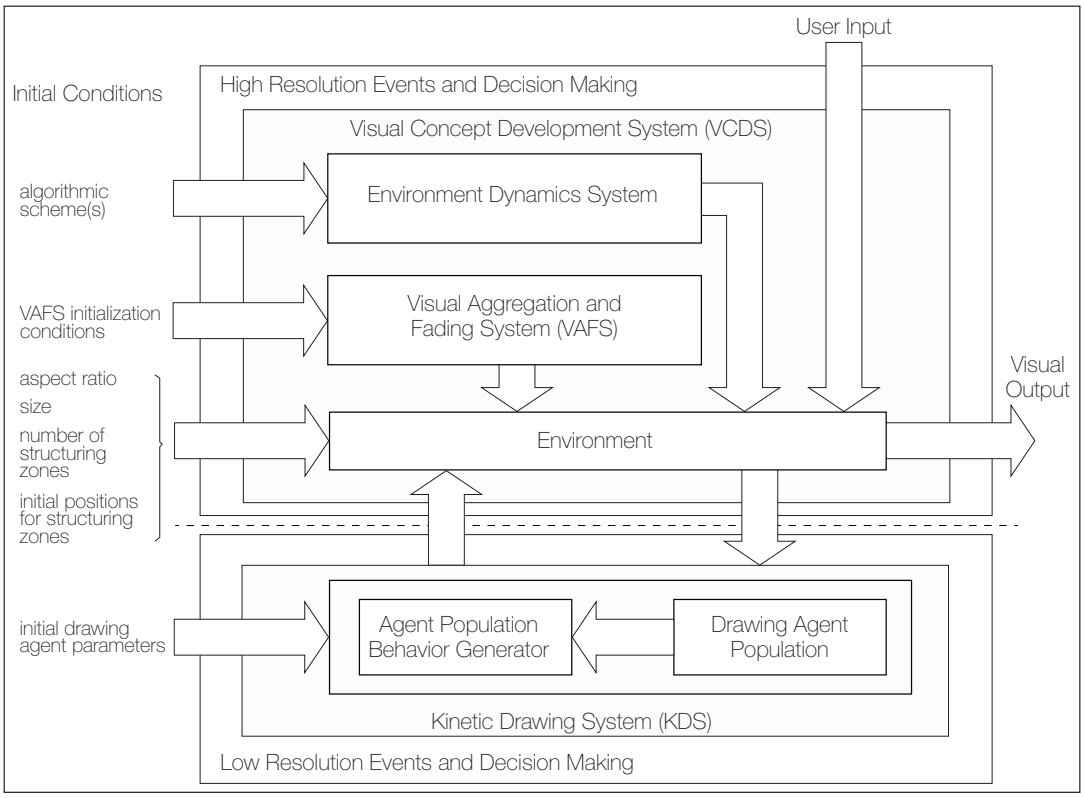


Figure 1: Generic ASVC architecture showing main system components and their functional correlation.

- The visual concept development system (VCDS).

These two main components suggest a hierarchical structure based on generating visual concepts at higher hierarchical level and their implementation at lower hierarchical level. We emphasize that the most important aspect in this architectural organization is, in fact, the level of resolution involved defining ASVC as multiresolution system in the sense described in [26]. In this sense, the low resolution corresponding to KDS is identified through local actions with corresponding small scale or micro influence in the system. The high resolution is related to actions that influence the visual development at large or macro level and this is associated with VCDS (see Figure 1 and Figure 2).

The important aspect is that both the virtual drawing through the KDS and the artificial concept development through VCDS are exploratory in nature being based to a large degree on stochastic processes that are under environmental influences.

6 Kinetic drawing through a virtual ecosystem

To further particularize the description of the generic architecture, ASVC can be described as a virtual ecosystem populated by drawing agents. The motion of the drawing agents in the environment takes place at micro level in the system, which can be associated with a primary perceptual level where drawing agents seem to be independent in their actions, but responsive to environmental influences. The drawing process takes place through recording the motion trajectories of the drawing agents in the system. The motion of drawing agents is self-generated.

Therefore, the KDS output at low perceptual level is influenced by the computational paradigm

that underlies agent motion. There are a number of pre-determined parameters that decide the agent appearance, which has a role as a unit form in visual structure aggregation and therefore has impact on the visual output from the system. This aspect is described in more detail with reference to a particular ASVC implementation based on random walk algorithms in [43], but other computational paradigms can be employed.

7 Design principles and visual concept development in ASVC

Aesthetic principles are manifested in ASVC through environmental dynamics. Traces generated by the drawing agents in motion are steered towards aesthetic organization of form aggregations through interactions between the drawing agents and the environment. Therefore, visual concept development in ASVC depends on the structural elements in the environment configuration and the overall dynamics of the environment.

If we employ an evolutionary computing paradigm, which is not mandatory, but is very intuitive, then defining a certain configuration and dynamics for the artificial environment means defining an environmental niche. In this case, the development of a visual concept in ASVC can be described through an environmental niche in the virtual world. We re-emphasize that once the environmental niche is configured and the underlying computational paradigm is defined, this does not mean that the environmental niche is static. By design, the dynamics of the environmental components is largely based on stochastic processes that continuously unfold. This is meant to facilitate within aesthetic constraints a large variety of possible compositional developments.

8 Artificial creativity as a feedback loop system

Starting from the idea that both intelligence and creativity are processes of adaptation to the environment and both are functions of the brain, we can discuss an interpretation of creativity as overlapping in many respects with intelligence as an adaptive system functioning in interaction with a given environment. The main difference between intelligence and creativity is that intelligence operates within the knowledge domain, while creativity operates to expand the knowledge domain. Based on this interpretation, we can adapt the representation of intelligent systems as adaptive feedback loop systems [26] to incorporate aesthetic knowledge and relaxation of aesthetic principles through randomized algorithms at various levels in the ASVC system as presented in Figure 2.

9 Implementations

The conceptual framework discussed in the paper provides the basis for the development of several systems for assisted and autonomous artificial creativity based on random walk algorithms, genetic algorithms, and hybrid systems, which combine several computing paradigms through various components of the ASVC generic architecture presented in Figure 2.

9.1 Random walk ASVC

A Random Walk ASVC (RWASVC) has been developed based on KDS with the motion of the drawing agents based on random walk algorithms. This is integrated with an environment with randomized areas of interaction that fracture the random walk continuity and reinitialize the algorithm with the agent placed in high recurrence areas. RWASVC systems have been extended

into hybrid systems that incorporate drawing agents engaged in physical simulation systems in addition to the random walk drawing agent population (Figure 3).

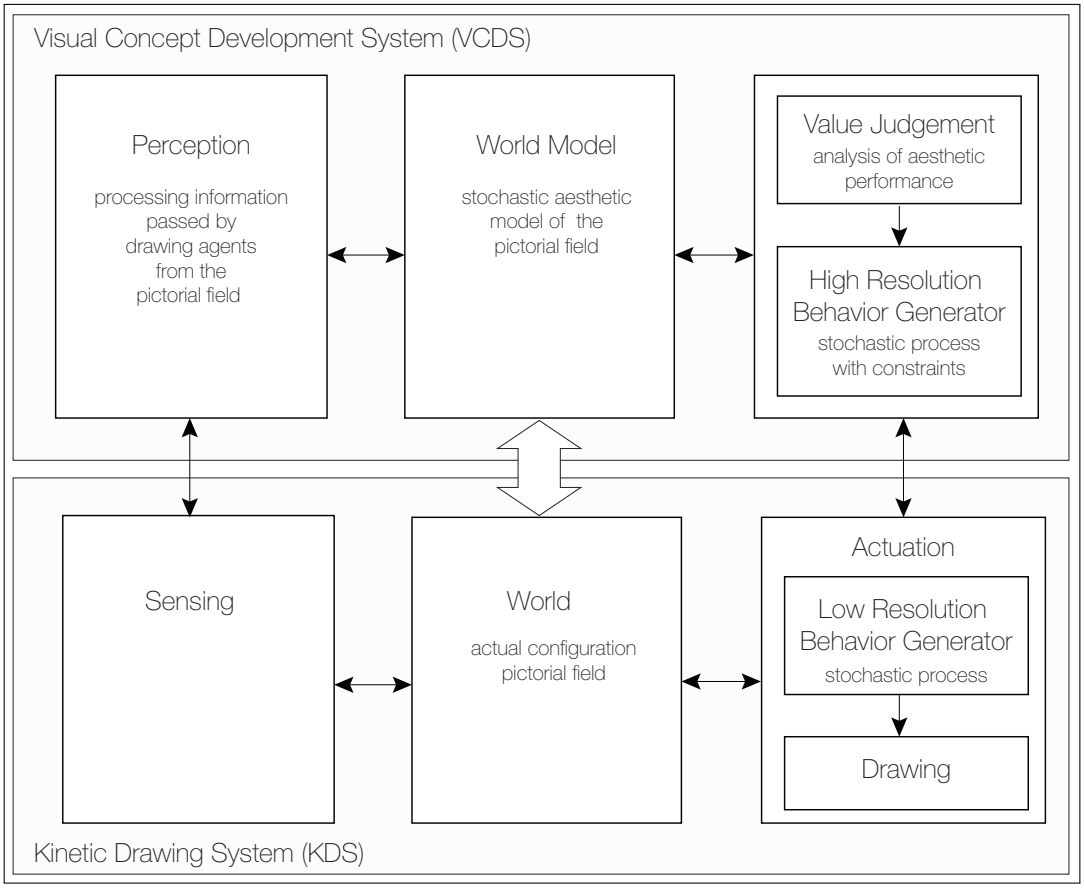


Figure 2: ASVC generic architecture adapted to emphasize computational creativity components and functionality in a feedback loop. This expands computational intelligent system representation proposed in [26] to accommodate rules relaxation over aesthetic knowledge domain in the world dynamics and drawing agents behavior in ASVC.

With this system we obtained the appearance of a natural garden fence with vegetation in continuous growth and having a distribution in the pictorial field that responds to aesthetic principles of visual composition organization. Experiments with RWASVC systems show that for a defined size of the frame of reference and a certain range of proportional relationships with the agent size, these systems are very reliable in producing a large number of visual compositions in a reasonable amount of time. Some recommendations for effective setups have been synthesized from these experiments. Hybrid RWASVCs incorporating physical systems extend the range of visual styles, but more refined correlations of the agent size and numbers must be performed.

9.2 Evolutionary ASVC

A number of Evolutionary ASVC (EASVC) systems have been developed with drawing agents in movement under an evolutionary computing paradigm and with the environment being characterized by non-deterministic dynamics. Experiments with these systems provide very interesting and consistent results based on experimental setups that allow a large degree of stochastic freedom in the environment dynamics and therefore in the visual concept development system of the EASVC. These experiments emphasize that the systems creativity and the output quality

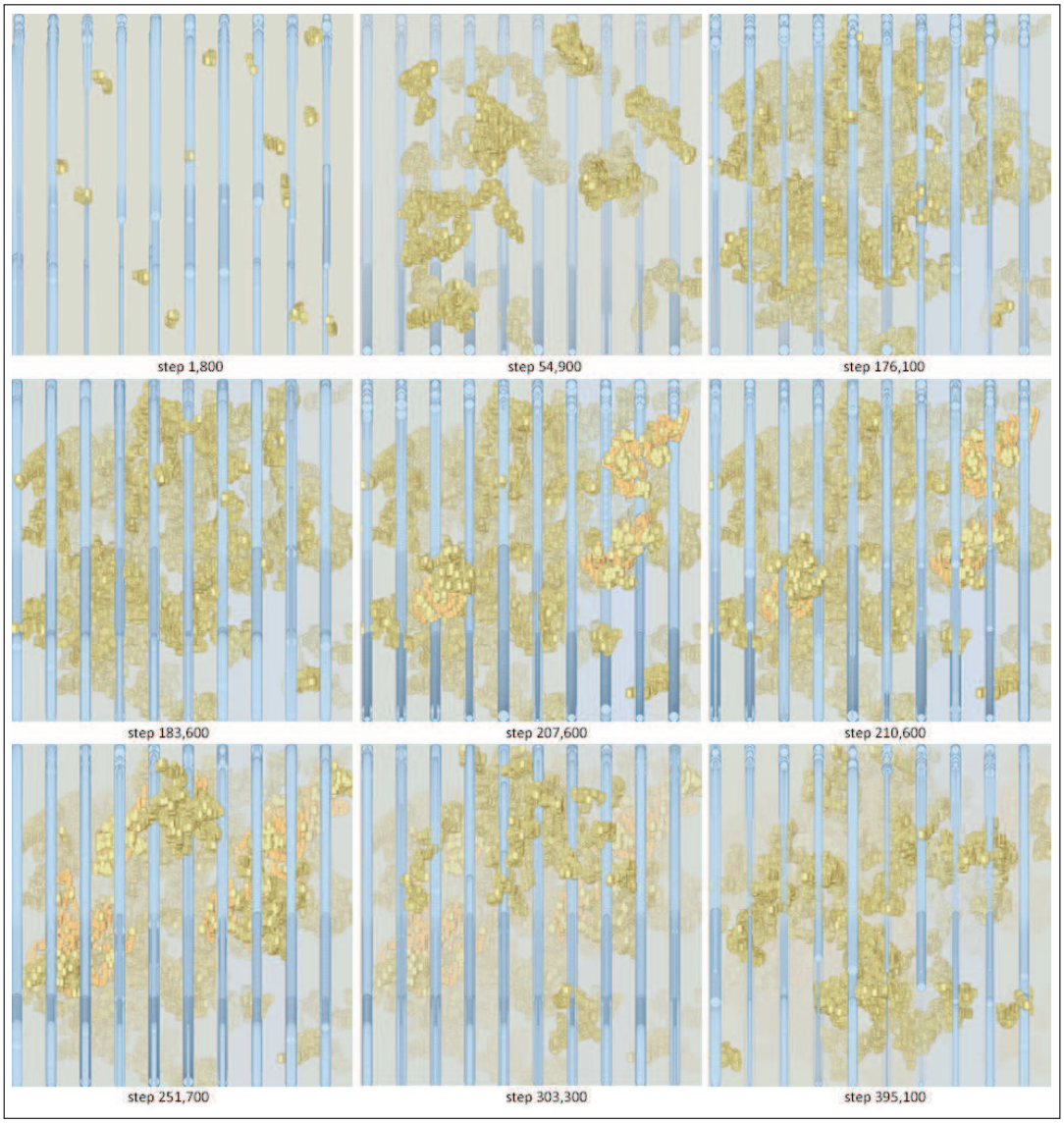


Figure 3: Selected cluster of sampled compositions generated by the hybrid ASVC - version 8 combining a random walk with 14 drawing agents and a physical simulation component with 10 drawing agents. Random walk drawing agent: rectangle, size 15 pixels x 25 pixels.

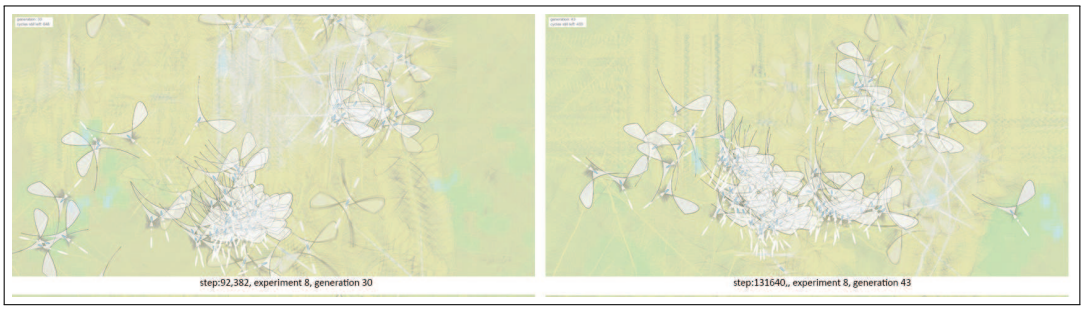


Figure 4: Dominant compositions with dense forms created during experiment number 8 with the EASVC system.

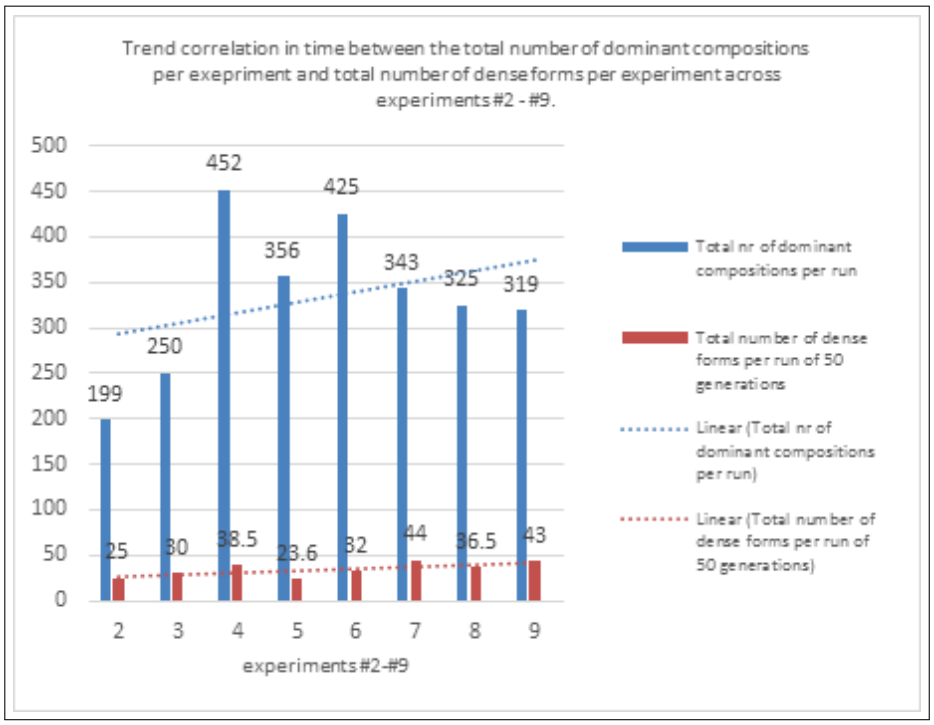


Figure 5: Correlations between total the number of dense forms per run and the total number of dominant compositions per run across experiments 2-9, case study 1. Each run is of 50 generations of drawing agents.

increases when aesthetic rules are loose allowing more freedom in creative combinations and associations of forms. Sampled images from this experiment are presented in Figure 4.

Experimental results are synthesized in Figure 5 showing correlations between the total number of dense forms, which express better quality through better form definition, and total number of dominant compositions, which show quantitative increase of system productivity over generations.

10 Conclusion

We conclude emphasizing that experimental results support our hypothesis that stochastic freedom catalyzes ASVC creative behavior when manifested both at conceptual level (visual concept development through the environment) and execution level (drawing through agents), provided that stochastic processes are under a system of aesthetic influences within ASVC. Therefore, the synthesis of a number of guidelines that allow the computational implementation of general creativity principles allow specialized forms of creative behavior in visual arts to be manifested within a non-anthropocentric ASVC. This can provide a basis for the development of specialized forms of computational creative behavior related to a large range of other application domains.

Acknowledgements

This research is based on grants from the Canadian Foundation for Innovation (CFI), West Grid Program phase II for Collaboration and Visualization, projects MARVIS I and II and

I-HEARD I and II.

Bibliography

[1] Amabile T.M. (1983); *Social Psychology of Creativity*, New York, NY: Springer-Verlag, 1983.

[2] Bentley P.J., Corne D.W. (2002); *Creative Evolutionary Systems*, San Diego, CA: Academic Press.

[3] Boden M.A. (2004); *The Creative Mind. Myths and Mechanisms*, 2nd ed., London, UK: Routledge/Taylor and Francis, 2004.

[4] Cautilli J.(2004); Toward a behavioral theory of creativity: A preliminary essay, *The Behavior Analyst Today*, 5(1), 126-140.

[5] Coleman P.H., Pietronero L. (1992); The fractal nature of the universe, *Physica A: Statistical Mechanics and its Applications*, 185(1), 45-55, 1992.

[6] Cross N. (2006); *Designerly Ways of Knowing*, London, UK: Springer, 2006.

[7] Csikszentmihalyi M. (1996); *Creativity. Flow and the Psychology of Discovery and Invention*, New York, NY: Harper Collins Publishers, 1996.

[8] Csikszentmihalyi M. (1999); Implications of a systems perspective for the study of creativity, *Handbook of Creativity*, R. J. Sternberg (ed.), Cambridge, UK: Cambridge University Press, 313-335, 1999.

[9] Dumitrache I. (2016); Problems of Brain Modeling, Plenary Speech at *20th International Conference on Systems Theory, Control, and Computing*, Sinaia, Romania, 2016.

[10] Findlay C.S., Lumsden C.J. (1988); The creative mind. Towards an evolutionary theory of discovery and innovation, *Journal of Social and Bilological Structures*, 11, 3-55, 1988.

[11] Finke R.A., Ward T.B., Smith S.M. (1992); *Creative Cognition. Theory, Research, and Applications*, Cambridge, MA: MIT Press, 1992.

[12] Goldberg D.E. (2002); *The Design of Innovation. Lessons from and for Competent Genetic Algorithms*, Boston, MA: Kluwer Academic Press, 2002.

[13] Gute D., Gute G. (2015); *How Creativity Works in the Brain. Insights from a Santa Fe Institute Working Group*, Santa Fe Institute, National Endowment for the Arts Office of Research and Analysis, Washington DC, 2015, [Online], Available: https://www.arts.gov/sites/default/files/how-creativity-works-in-the-brain-report.pdf.

[14] Hofstadter D.R. and the Fluid Analogies Research Group (1995); *Fluid Concepts and Creative Analogies: Computer Models of Fundamental Mechanisms of Thought*, New York, NY: Basic Books, 1995.

[15] Hofstadter D., Mitchell M. (1995); The Copycat Project: A Model of Mental Fluidity and Analogy Making, *Fluid Concepts and Creative Analogies. Computer Models of the Fundamental Mechanisms of Thought*, New York, NY: Basic Books/ Harper Collins, 205-268, 1995.

[16] Jung R.E., Mead B.S., Carrasco J., Flores R.A. (2013); The structure of creative cognition in the human brain, *Frontiers in Human Neuroscience*, [Online], Available at: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3703539/, doi: 10.3389/fnhum.2013.00330, 7, 330, 2013.

[17] Kris E. (1952); *Psychoanalytic Exploration in Art*, New York, NY: International University Press, 1952.

[18] Kubie L.S. (1958); *The Neurotic Distortion of the Creative Process*, Lawrence, Kansas: University of Kansas Press, 1958.

[19] Lawson B. (1980); *How Designers Think*, London, UK: Architectural Press, 1980.

[20] Mandelbrot B. (1983); *The Fractal Geometry of Nature*, 2nd - revised ed., New York, NY: W.H. Freeman and Comp., 1983.

[21] Marshall J.B.(1999); *Metacat: A Self-Watching Cognitive Architecture for Analogy-Making and High-Level Perception*, Ph.D. Dissertation, Department of Computer Science, Indiana University Bloomington, 1999.

[22] Marshall J.B. (2006); A self-watching model of analogy-making and perception, *Journal of Experimental and Theoretical Artificial Intelligence*, 18(3), 267-307, 2006.

[23] Martindale C. (1981); *Cognition and Consciousness*, Homewood, IL: Doresey Press, 1981.

[24] Martindale C. (1999); Biological basis of creativity, *Handbook of Creativity*, R. J. Sternberg (ed.), Cambridge, UK: Cambridge University Press, 137-152, 1999.

[25] Martindale C. (2007); Creativity, primordial cognition, and personality, *Personality and Individual Differences*, 43(7), 1777-1785, 2007.

[26] Meystel A.M., Albus J.S. (2002); *Intelligent Systems: Architecture*, Design and Control, New York, NY: Wiley, 2002.

[27] Minsky M. (2006); *The Emotion Machine. Commonsense Thinking, Artificial Intelligence, and the Future of the Human Mind*, New York, NY: Simon and Schuster, 2006.

[28] Mitchell M. (1990); *COPYCAT: A Computer Model of High-Level Perception and Conceptual Slipage in Analogy Making*, Ph.D. Thesis, University of Michigan, Michigan, 1990.

[29] Mitchell M. (1993); *Analogy-Making as Perception. A Computer Model*, Cambridge, MA: The MIT Press, 1993.

[30] Partridge D., Rowe J. (1994); *Computers and Creativity*, Oxford, UK: Intellect, 1994.

[31] Poincaré H. (1910); "Mathematical Creation," *The Monist*, 20(3), 321-335, 1910.

[32] Saunders R., Gero J. (2001); Artificial creativity: A synthetic approach to the study of creative behaviour, *Proc. Computational and Cognitive Models of Creative Design, Key Centre of Design Computing and Cognition*, University of Sydney, Sydney, Australia, 113-139, 2001.

[33] Saunders R. (2002); *Curious Design Agents and Artificial Creativity. A Synthetic Approach to the Study of Creative Behavior*, Ph.D. Dissertation, School of Architecture, Design Science and Planning, Faculty of Architecture, University of Sydney, Sydney, Australia, 2002.

[34] Saunders R. (2006); Towards a computational model of creative societies using curious design agents, *Proc. Conf. Engineering Societies in the Agents World 2006 (ESAW06)*, 18-31, 2006.

[35] Saunders R.(2012); Towards autonomous creative systems: A computational approach, *Cognitive Computation*, 4(3), 216-225, 2012.

[36] Schmajuk N.,Aziz D.R., Bates M.J.B. (2009); Attentional Associative Interactions in Creativity, *Creativity Research Journal*, 21(1), 92-103, 2009.

[37] Simonton D.K. (1999); *Origins of Genius. Darwinian Perspectives on Creativity*, New York, NY: Oxford University Press, 1999.

[38] Simonton D.K. (1999); Creativity from a Historiometric Perspective, *Handbook of Creativity*, R. J. Sternberg (ed.), Cambridge, UK: Cambridge University Press, 116-133, 1999.

[39] Simonton D.K. (2013); Evolution as Phenomenon and Evolution as Process, *Proc. Int. Symp. - Investigations of Cultural Life: Quantitative Aspects*, Ekaterinsburg, Russian Federation, 21-23, 2013.

[40] Simonton D.K. (2011); "Historiometry," *Encyclopedia of Creativity*, S. R. Pritzker (ed.), San Diego: Academic Press, 617-622, 2011.

[41] Sims K. (1991); Artificial evolution for computer graphics, *Computer Graphics*, 25(4), 319-328, 1991.

[42] Sims K. (1994); Evolving 3D morphology and behavior by competition, *Proc. Artificial Life IV*, 28-39, 1994.

[43] Sirbu D. (2013); Emerging visual structures from a random walker, *Proc. of 16th International Conference on Generative Art*, Politecnico di Milano University, Milan, Italy, 66-76, 2013.

[44] Sternberg R.J. (1999); The concept of creativity: Prospects and paradigms, *Handbook of Creativity*, R.J. Sternberg (ed.), Cambridge, UK: Cambridge University Press, 3-15, 1999.

[45] Sternberg R.J. (1999); *Handbook of Creativity*, R.J. Sternberg (ed.), Cambridge, UK: Cambridge University Press, 1999.

[46] Ulrich D. (2002); *Seven Stages of Creativity* Hillsboro, OR: Beyond Words, 2002.

[47] Wallas G. (1926); *The Art of Thought*, London: J. Cape, 1926.

[48] Weisberg R.W. (1993); Creativity: Beyond the myth of genius, *Handbook of Creativity*, R.J. Sternberg (ed.), New York, NY: W.H. Freeman, 1993.

[49] Weisberg R.W.(1999); Creativity and knowledge: A challenge to theories, *Handbook of Creativity*, R.J. Sternberg (ed.), Cambridge, UK: Cambridge University Press, 1999.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 393-401, June 2017.

Genetic Algorithm with Modified Crossover for Grillage Optimization

M. Ramanauskas, D. Sesok, R. Belevicius, E. Kurilovas, S. Valentinavicius

Mikalojus Ramanauskas, Dmitrij Sesok
Rimantas Belevicius, Saulius Valentinavicius
Vilnius Gediminas Technical University,
Dept. of Information Technology, Vilnius, Lithuania
Vilnius, LT-10223 Sauletekio al. 11, Lithuania
mikalojus.ramanauskas@vgtu.lt

Eugenijus Kurilovas
1. Vilnius Gediminas Technical University,
Dept. of Information Technology, Vilnius, Lithuania
Vilnius, LT-10223 Sauletekio al. 11, Lithuania
jevgenij.kurilov@vgtu.lt
2. Vilnius University Institute of Mathematics and Informatics
Vilnius, LT-08663 Akademijos str. 4, Lithuania
*Corresponding author: jevgenij.kurilov@vgtu.lt

Abstract: Modified genetic algorithm with special phenotypes' selection and crossover operators with default specified rules is proposed in this paper thus refusing the random crossover. The suggested crossover operator enables wide distribution of genes of the best phenotypes over the whole population. During selection and crossover, the best phenotypes of the newest population and additionally the genes of the best individuals of two previous populations are involved. The effectiveness of the modified algorithm is shown numerically on the real-life global optimization problem from civil engineering - the optimal pile placement problem under grillage-type foundations. This problem is a fair indicator for global optimization algorithms since the ideal solutions are known in advance but with unknown magnitudes of design parameters. Comparison of the proposed algorithm with 6 other stochastic optimization algorithms clearly reveals its advantages: at similar accuracy level the algorithm requires less time for tuning of genetic parameters and provides narrower confidence intervals on the results than other algorithms.
Keywords: genetic algorithm; crossover operator; grillage optimization.

1 Introduction

In this paper, we propose a new genetic algorithm with modified crossover operator and compare it with other well-known stochastic optimization algorithms. As a benchmark the problem of pile placement optimization under grillage-type foundations is chosen. The grillage foundations consist of piles driven to the ground, and connected on the top with girders. The in-plane configuration of girders may be complex, and the number of piles may reach several tens (few examples of grillages are provided in the Appendix). Here we consider that the optimal grillage is the grillage with minimal number of piles of given stiffness characteristics, and the reactive forces in piles are equal. This can be achieved changing the positions of piles under girders. In ideal case the reactive forces in all piles are distributed evenly and are equal to the bearing capacity of pile. Practically this is hardly possible, since the theoretical number of piles which is obtained dividing the total loading on the foundation (i.e., all active forces plus the dead weight of the erection) by bearing capacity of pile, is common case not the integer number. Also, some technological requirements can hinder achieving the ideal scheme, e.g., the given minimal

allowable distance between adjacent piles due to pile-driver characteristic, or the presence of the given immovable piles that are introduced into placement scheme by a designer (usually at the corners of girders) and do not change their position in the optimization process.

The pile placement problem is ideal for comparison of global optimization algorithms. Firstly, the global solution - the reactive force that should be evened out in all piles is known in advance. Secondly, the practical solution of the problem shows that the landscape of objective function is complex and has many local extremes. Usually, the objective function is very sensitive to the pile positions: even small changes position of one pile sometimes leads to a large alteration of objective function. All this makes the problem a complex global optimization problem.
The mathematical models of optimization of grillage-type foundations were formulated and the solution algorithms were suggested in [4], [3]. Three problems were solved: pile placement seeking for even distribution of reactive forces in piles, pile placement seeking for least bending moments in the connecting girders, and integrated problem for minimization of reactive forces and bending moments. In case of two last problems the global solution cannot be obtained in advance, therefore it is not a right choice for comparison of optimization algorithms. The first problem was solved in [10] employing all popular at that time stochastic optimization algorithms. In all these algorithms a phenotype, or an individual, is the approximate mathematical model of the whole grillage. The fitness of a phenotype is measured by a maximum reactive force magnitude among all piles, i.e., the fittest individual has the least reactive force. In [15], combination of the sizing and topology optimization is observed, however the piles are aggregated to special groups of pile. Exhaustive technical details on the design of grillages can be found, e.g. in [13]. Also, this problem was solved by the new several dimension optimization method BAcoor [8], [9]. The last method outperformed all other algorithms for the grillages where the piles have to be placed at very uneven distances. At more even distribution of piles the classical stochastic algorithms provided better results.

In all cases genetic algorithm (GA) with carefully tuned genetic parameters provided best results or results close to the best solutions. Among these parameters, the crossover operator plays significant role [4], [7] since it combines the information contained in the previous individuals in order to obtain fitter phenotypes. Therefore, in this paper, the main attention is given to the crossover operator. The crossover was investigated, e.g., in [5] where a gender was assigned to each individual and the crossover was performed only between individuals of opposite genders. In many cases the crossover operator is dedicated for a particular type of problems, e.g., operator EAX for traveling salesman problems [15], [11].

There are five chapters and appendices in this paper. In the second chapter, the ideas of new crossover operator and its implementation are described. The third chapter depicts the problem under consideration - the analysis of grillage-type foundations via finite element method. In the fourth chapter, the optimization problem along with all requirements for fair comparison of optimization algorithms are given, and the obtained results are discussed. In the final section, some general conclusions are drawn.

2 Modified crossover operator

In the classical genetic algorithm, the initial population of a given length N is created. Then, using selection and crossover operators the new individual generations are generated. During the selection, the fitter individuals have better chances to be included into the next generation. The probability of an individual to be selected for a next generation is usually directly proportional to the ratio of its objective function value with the best function value in the present population. Instead of individuals that do not enter into the new generation, the new random individuals are created. After the selection, the random individuals interchange genes between them during

the crossover operation. This does not guarantee that after the selection all best genes will survive and will get into the new generation. If the non-elitist strategy is employed, always a certain probability exists that even the individual representing the global solution will not pass the selection. It also should be noted, that in classical algorithm the crossover operator is applied only to the individuals of newest generation. Thus, the individuals of previous generations are completely lost. We suggest the following modified selection and crossover operations:

- The new population is created not only of crossbred individuals, but also of the best individuals of two previous generations. Thus in the current population coexist parents and grandparents.
- One-third of the best individuals participate in the crossover, and each individual is interbred twice with in advance definite individuals.
- Thus, less random genetic parameters must be chosen for the algorithm; only the breeding point and mutation probability have to be selected.

The crossover rules are summarized as follows:
1. Initial population of N individuals $A_0 = \{a_0, a_1, ..., a_{N-1}\}$.
2. The individual a'_i of a succeeding population A_j $(j \neq 0)$ is obtained breeding N best individuals of the generation A_{j-1} according to the equations:

- if $i < \frac{N}{2}, a'_i = a_i \times a_{i+\frac{N}{2}}$
- if $i \geqslant \frac{N}{2}, a'_i = a_i \times a_{N-i-1}$

where the $\times$ denotes the crossover operations.
3. Population A_j $(j > 1)$ consists of $3 \cdot N$ individuals: N crossbred individuals of generation A_{j-1} and two groups of size N of the best individuals from generations A_{j-1} and A_{j-2} .
Thus, the recursive function for creation of populations is: $A_0 = \{a_0, a_1, a_2, ..., a_{N-1}\}$,
$A_1 = A_0 \bigcup A_0^{\times}$,
$A_2 = A_0 \bigcup TOP_N(A_1) \bigcup TOP_N\left(A_1^{\times}\right)$,
$A_3 = TOP_N(A_1) \bigcup TOP_N(A_2) \bigcup TOP_N\left(A_2^{\times}\right)$,
...
$A_i = TOP_N(A_{i-2}) \bigcup TOP_N(A_{i-1}) \bigcup TOP_N\left(A_{i-1}^{\times}\right)$,
where $TOP_N(A_i)$ is the set of N individuals of the i^{th} generation with the best objective function values $A_i^{\times}$, is the crossover operator breeding the individuals of the i^{th} generation according to the rules shown. Thus, the proposed rules do not require tuning of selection parameters. The third rule also guaranties a special elitist strategy, i.e., the individual with the best objective function value is retained from generation to generation. On the other hand, passing into the next generations only the fittest individuals of preceding populations may narrow the search space. Sufficient diversity of a population therefore is achieved by a classical mutation with a rather high probability that should be tuned for the problem under consideration.

3 Grillage optimization model

One individual of a routine population is the approximate mathematical model of the grillage-type foundation. Grillage is discretized by the finite element method into 2D beam element mesh with out-of-plane boundary conditions instead of piles. Out-of-plane active forces consist of the self-weight of the erection, plus all active forces according to Eurocodes. The girders of

a grillage are approximated as two-node beam elements with fixed cross-section and material characteristics. The piles are represented as the supports with specified displacements (zero displacements are the most common case). Alternatively, piles are regarded as supports with specified stiffness characteristics. Generally, the maximum reactive force among all piles is treated as the objective function.

Supports of the first type are rather non-realistic representations and sometimes yield misleading analysis results. For example, when multiple supports are needed to carry large concentrated load, this kind of supports will lead to a logjam. If odd number of supports is placed under load, the central support will be located just beneath the load and will take all the force. In case of even number of supports the "saw-teeth" like distribution of reactions is observed, and the more supports will be installed, the larger in absolute value reactions will arise.

The optimization problem is defined as in

$$\min_{x \in D} f(x). \tag{1}$$

Here $f(x)$ is the objective function, D is the feasible shape of structure, which is defined by the type of certain supports, the given number and layout of different crosssections as well as different materials in the structure.

$f(x)$ is defined by the maximum difference between vertical reactive force at a support and allowable reaction for this support, thus allowing us to achieve different reaction at supports on different beams, or even at particular supports on the same beam:

$$f(x) = \max_{x \in D} \max_{1 \leqslant i \leqslant N_i} |R_i - c_i R_{allow}|. \tag{2}$$

Here N_s denote the number of supports, R_{allow} is allowable reaction, c_i are factor to this reaction and R_i are reactive forces in each support.

Finite element matrices and sensitivity analysis

The problem has to be solved in statics and in linear stage

$$[K]\{u\} = \{F\}. \tag{3}$$

Here $[K]$ is the stiffness matrix of grillage, $\{u\}$ are the displacement of grillage nodes, and $\{F\}$ - the loadings. The reactive forces at a rigid supports are obtained using equation

$$R_i = \sum_j K_{ij} u_j, i = 1, 2, ..., N_S, \tag{4}$$

where a part of nodal displacements (displacements of free nodes) are already obtained via, and the displacements of nodes representing the rigid supports are specified (usually - zero). If the supports have finite stiffness k_i ,

$$R_i \approx k_i u_i, i = 1, 2, ..., N_S. \tag{5}$$

If the local search around the certain solution obtained by stochastic optimization algorithm is implemented, the sensitivity information is the must. The sensitivity analysis is performed using the pseudo-load approach; thus, the numerical calculation of derivatives can be avoided. Denoting the support positions by $x_i, i = 1, 2, ..., N_S$

$$R_{i,x_i} = [K]_{,x_i}\{u\} + [K]\{u\}_{,x_i}. \tag{6}$$

Here the derivative of stiffness matrix is obtained analytically, while the derivative of displacements supposes solution of the general sensitivity equation:

$$[K]\{u\}_{,x_i} = \{F\}_{,x_i} - [K]_{,x_i}\{u\}. \tag{7}$$

The derivatives of load vector are obtained also in a closed form, analytically. A simple two-node beam element with 6 d.o.f's at a node (three displacements and three rotations about local element axes) is employed in the analysis. Details on the stiffness matrices of an element may be found in many textbooks, e.g., [16].

Program. Original Fortran program is used for obtaining the objective function value. First of all, the finite element mesh along with all needed data is prepared by a special pre-processor. Some initial data for the pre-processor are constant and do change in the optimization process - the configuration of the grillage, self-weight of the structure and active loadings, material characteristics. The locations of piles are obtained from the guess of optimization algorithm. Since supports have to be placed under the girders of grillage, first of all the grillage is "unfolded" to one-dimensional construct, and locations of all supports are freely chosen along the whole length of this construct. After that, the initial configuration is restored. From all this input, the pre-processor automatically prepares the finite element mesh introducing nodes at support points, discontinuities of material and cross-sections properties, etc. The third independent program analyses the finite element results and provides the objective function value. Model transformation patterns are obtained by using the Formal Concept Analysis [10], where relations and element meta-classes of target and source models are linked together based on model classification group links that have similarities between them.

4 Numerical results and discussion

10 different grillages possessing from 17 to 55 piles, i.e., the optimization variables were optimized. Data for these problems (see Appendix 1) are obtained from several Dutch design bureaus which use the professional software package MatrixFrame (http://www.matrix-software.com/Uk/structuralengineering/matrixframe/index.html) for structural engineering. Seven different optimization algorithms are compared. 28 independent numerical experiments were performed with each algorithm. In order to have fair comparison, the objective function was evaluated 5000 times in each experiment. The following algorithms were employed [10]: modified random search (MRS), simulated annealing (SA), simplex (SM), the variable metric method NEWUOA [10], [12], [14], BAcoor, and the proposed genetic algorithm with modified crossover (MCGA).

Comparison of all algorithms is provided in the Fig.1. Since total number of objective function evaluations is 5000, small populations of 15 individuals were created in 333 generations. The mutation probability after few numerical experiments was set to 15%. All numerical results of 28 independent experiments are rendered in Appendix 2. Thus, in three cases the proposed algorithm outperforms all other algorithms. Compared to the classical genetic algorithm, the results of MCGA is better for almost all problems considered. The ideal solution was not found for any problem, however, the differences compared to the ideal solutions do not exceed 5% for problems No. 2, 5 and 7. Results for problems No. 3, 4, 6, 7 and 9 differ from global solutions till 8%. Summarizing, the SA showed the performance for the grillage optimization problems, however, the NEWUOA, MCGA and GA (excluding the simplest structures No. 1 and 3) are not far behind. GA outperforms the MCGA only for most complex grillages No. 8 and 10 with 34 and 55 design parameters. Better results may be expected with larger populations. However, in this case we cannot fairly compare results with other algorithms. More important is, the

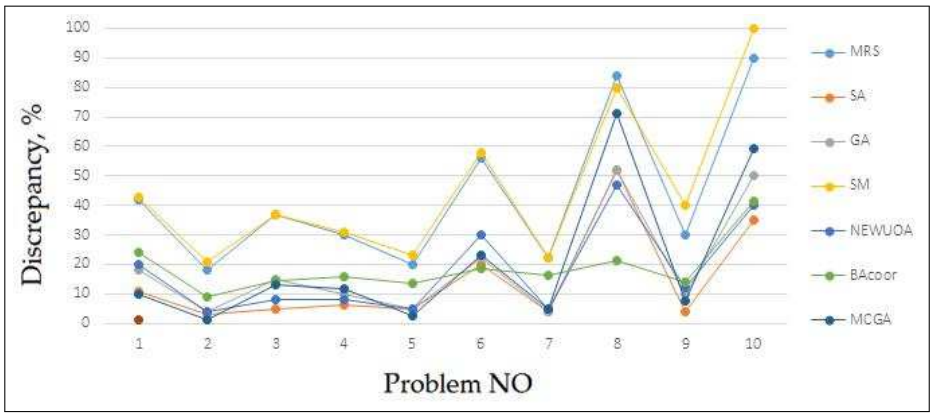


Figure 1: The best objective function values in 28 runs obtained by MRS, SA, GA, SM, NEWUOA, BAcoor and MCGA, normalized to Rideal .

confidence intervals of objective function value are much narrower for results of MCGA than for results of BAcoor (we do not have the confidence intervals for other algorithms) (see in Fig. 2). Also, for all problems the upper values of MCGA confidence intervals are better than ones of BAcoor. In some problems the upper values are even better than the lower values of confidence intervals of BAcoor. All numerical results - objective function values and confidence intervals are provided in the Tables 1 and 2.

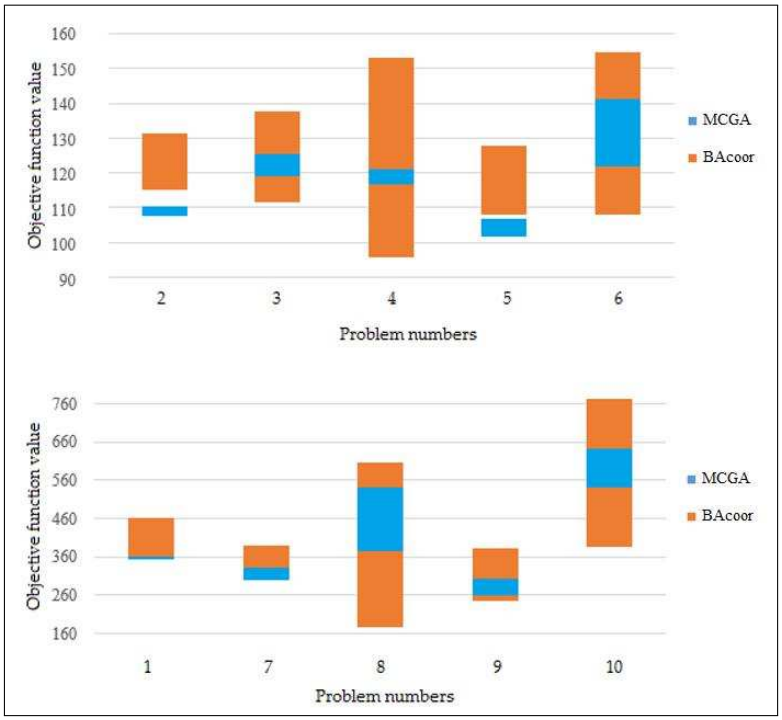


Figure 2: Confidence intervals.

Table 1: Summary of numerical results

Problem No	Best value found in MCGA	Best value found in [10]	Exact solution
1	337.15	339.30	307.47
2	105.34	106.36	104.12
3	115.02	107.25	101.85
4	113.29	106.80	101.24
5	100.00	101.05	97.51
6	120.28	115.45	97.53
7	301.48	298.11	287.35
8	404.29	286.22	236.28
9	263.12	253.00	244.71
10	556.11	463.34	349.05

Table 2: Summary of confidence intervals

Problem No	MCGA lower	MCGA upper	BAcoor lower	BAcoor upper
1	351.98	361.85	360.29	461.26
2	107.46	110.48	115.29	131.41
3	119.01	125.32	111.67	137.62
4	116.65	121.18	95.61	153.22
5	101.72	106.96	107.92	127.86
6	121.87	141.36	108.08	154.73
7	300.02	332.12	319.70	390.43
8	375.90	541.41	174.99	606.98
9	260.63	302.23	246.69	384.22
10	539.95	642.83	387.85	772.54

5 Conclusions

In rather simple configurations of grillages where the optimal distribution of piles due to the geometry and given loadings is more or less even, the proposed method MCGA outperforms other popular stochastic optimization algorithms. Additionally, this algorithm requires less effort in tuning of algorithm parameters. The crossover and selection parameters are forecasted in advance by recursive crossover rules. Also the confidence intervals for results of MCGA are always narrower than those of BAcoor. Only for grillage configurations requiring larger number of supports and their uneven distributions, it loses to NEWUOA and BAcoor. Thus, the proposed method can be treated as a new global optimization algorithm that is simpler to use as the classical genetic algorithm but providing better or similar level of accuracy.

Bibliography

[1] Belevicius R., Ivanikovas S., Sesok D., Zilinskas J., Valentinavicius S. (2011); Optimal placement of piles in real grillages: experimental comparison of optimization algorithms, *Information Technology and Control*, 40(2), 123-132, 2011.

[2] Belevicius R., Valentinavicius S. (2001); Optimisation of grillage-type foundations, *Proceedings of 2nd European ECCOMAS and IACM Conference Solids, Structures and Coupled*

Problems in Engineering, 416-421, Cracow, Poland 26-29 June, 2001.

[3] Belevicius R., Valentinavicius S., Michnevi E. (2002); Multilevel optimization of grillages, *Journal of Civil Engineering and Management*, http://dx.doi.org/10.1080/13923730.2002.10531259, 8(2), 98-103, 2002.

[4] De Jong K.A., Spears W.M. (1992); A formal analysis of the role of multi-point crossover in genetic algorithms, *Annals of Mathematics and Artificial Intelligence*, 5(1), 1-26, 1992.

[5] Garci C.M., Lozano M., Herrera F., Molina, D., Sanchez A.M. (2008); Global and local real-coded genetic algorithms based on parent centric crossover operators, *European Journal of Operational Research*, 185, 1088-1113, 2008.

[6] Kim K.N., Lee S.-H., Kim K.-S., Chung C.-K., Kim M.M., Lee H.S. (2001); Optimal pile arrangement for minimizing differential settlements in piled raft foundations, *Computers and Geotechnics*, http://dx.doi.org/10.1016/S0266352X(01)00002-7, 28(4), 235-253, 2001.

[7] Kita H. (2001); A comparison study of self-adaptation in evolution strategies and real-coded genetic algorithms, *Evolutionary Computation Journal*, 9(2), 223-241, 2001.

[8] Mockus J., Belevicius R., Sesok D., Kaunas J., Maciunas D. (2012); On Bayesian approach to grillage optimization, *Information Technology and Control*, 41(4), 332-339, 2012.

[9] Mockus J., Eddy W., Mockus A., Mockus L., Reklaitis G. (1997); *Bayesian Heuristic Approach to Discrete and Global Optimization*, Kluwer Academic Publishers, ISBN 0-7923-43227-1, Dordrecht-London-Boston, 1997.

[10] Nelder J. A., Mead R. (1965); A simplex method for function minimization, *Computer Journal*, 7, 308-313, 1965.

[11] Oliver I.M., Smith D.J., Holland J.R.C. (1987); A study of permutation crossover operators on the traveling salesman problem, *In Proceedings of the Second International Conference on Genetic Algorithms*, Mahwah, NJ, USA, 1987. Lawrence Erlbaum Associates, Inc. sd., 224-230, 1987.

[12] Powell M.J.D. (2006); The NEEWUOA software for unconstrained optimization without derivatives, *Di Pillo, G., Roma, M. (eds) Large-Scale Nonlinear Optimization. Vol. 83 of Nonconveex Optimization and Its Applications*, Springer, 255-296, 2006.

[13] Reese L.C., Isenhhower W.M., Wang S.-T. (2006); *Analysis and Design of Shallow and Deep Foundations*, John Wiley & Sons, 2006.

[14] Ros R. (2009); Benchmarking the NEWUOA on the BBOB-2009 noisy testbed, *F. Rothlauf, editor, GECCO (Companion)*, ACM, 2429-2434, 2009.

[15] Watson J.-P., Ross C., Eisele V., Denton J., Bins J., Guerra C., Whitley L.D., Howe A.E. (1998); The traveling salesrep problem, edge assembly crossover, and 2-opt, *PPSN V: Proceedings of the 5th International Conference on Parallel Problem Solving from Nature*, London, UK, Springer-Verlag, 823-834, 1998.

[16] Zienkiewicz O.C., Taylor R.L., Nithiarasu P. (2005); *The Finite Element Method for Fluid Dynamics*, Butterworth-Heinemann, Oxford, 6th edition, 2005.

Appendices

Table 3: Appendix 1: Characteristics of problems

Problem No	Number of supports	Foundation length	R_{allw}	R_{ideal}
1	25	172,90	325	307,47
2	18	52,90	110	104,12
3	31	84,10	105	101,85
4	31	84,90	105	101,24
5	30	63,90	100	97,51
6	37	80,10	100	97,53
7	23	129,10	300	287,35
8	34	137,90	250	236,28
9	17	97,60	250	244,71
10	55	315,61	350	349,05

Table 4: Appendix 2: Optimization results for all 10 problems in 28 independent numerical experiments

Probl/Exper	1	2	3	4	5	6	7	8	9	10
1	375,87	108,33	122,88	119,18	105,21	136,06	308,04	540,96	283,54	614,04
2	359,09	111,01	124,15	118,72	107,04	129,00	312,08	520,77	277,54	591,94
3	391,24	107,81	127,80	121,81	110,12	129,23	313,24	434,80	275,06	580,37
4	356,93	111,85	117,99	119,09	102,96	122,32	321,58	414,26	282,97	606,22
5	341,47	107,69	123,78	118,68	105,85	126,39	313,00	404,29	287,37	579,35
6	368,95	106,81	120,46	113,29	105,79	120,28	319,00	425,89	265,36	563,57
7	352,35	107,28	124,94	119,06	101,28	133,34	315,42	419,62	309,76	658,62
8	358,81	105,86	121,88	123,05	105,82	127,83	310,84	538,27	268,19	593,13
9	340,49	108,43	124,31	119,79	107,36	122,32	305,63	407,56	290,50	576,15
10	348,00	112,49	119,78	117,50	103,19	138,99	319,01	523,01	299,97	556,11
11	352,52	107,82	123,30	120,11	105,54	141,05	327,97	532,61	273,09	590,31
12	349,46	107,75	118,99	119,69	101,86	125,69	341,05	432,63	284,85	567,98
13	376,92	110,01	126,53	118,31	105,97	126,22	301,49	525,76	279,00	607,32
14	351,34	105,45	118,58	121,63	102,85	136,55	315,65	420,27	286,73	563,72
15	341,32	106,93	115,70	118,35	103,23	131,61	312,04	429,79	276,59	606,96
16	352,12	117,91	123,82	119,26	106,82	143,18	318,38	425,28	270,80	602,00
17	337,15	108,61	123,21	121,01	107,00	127,17	305,69	442,52	263,12	638,64
18	345,64	107,62	118,43	114,78	105,90	138,09	311,00	439,41	276,29	613,01
19	364,22	109,22	117,90	121,67	101,88	143,99	313,38	408,72	298,48	578,94
20	344,76	111,06	128,80	115,69	103,36	134,57	354,81	445,75	268,88	612,41
21	369,08	110,98	115,02	113,43	100,00	124,37	316,54	457,68	290,61	566,76
22	370,84	106,44	117,24	119,15	103,18	147,41	319,63	449,23	266,89	593,30
23	354,59	105,34	131,69	117,51	105,93	132,50	307,67	537,75	279,97	557,41
24	361,88	110,66	118,33	120,48	103,45	136,64	311,88	446,24	278,44	590,55
25	355,11	108,59	127,67	118,72	103,24	122,41	315,93	441,27	290,06	563,00
26	376,52	106,98	120,09	122,02	102,57	126,20	309,44	534,65	280,92	578,71
27	341,46	112,65	120,74	120,05	103,12	136,93	315,33	423,97	290,76	621,52
28	355,41	109,58	126,61	117,59	101,11	124,86	314,31	419,46	284,29	586,92

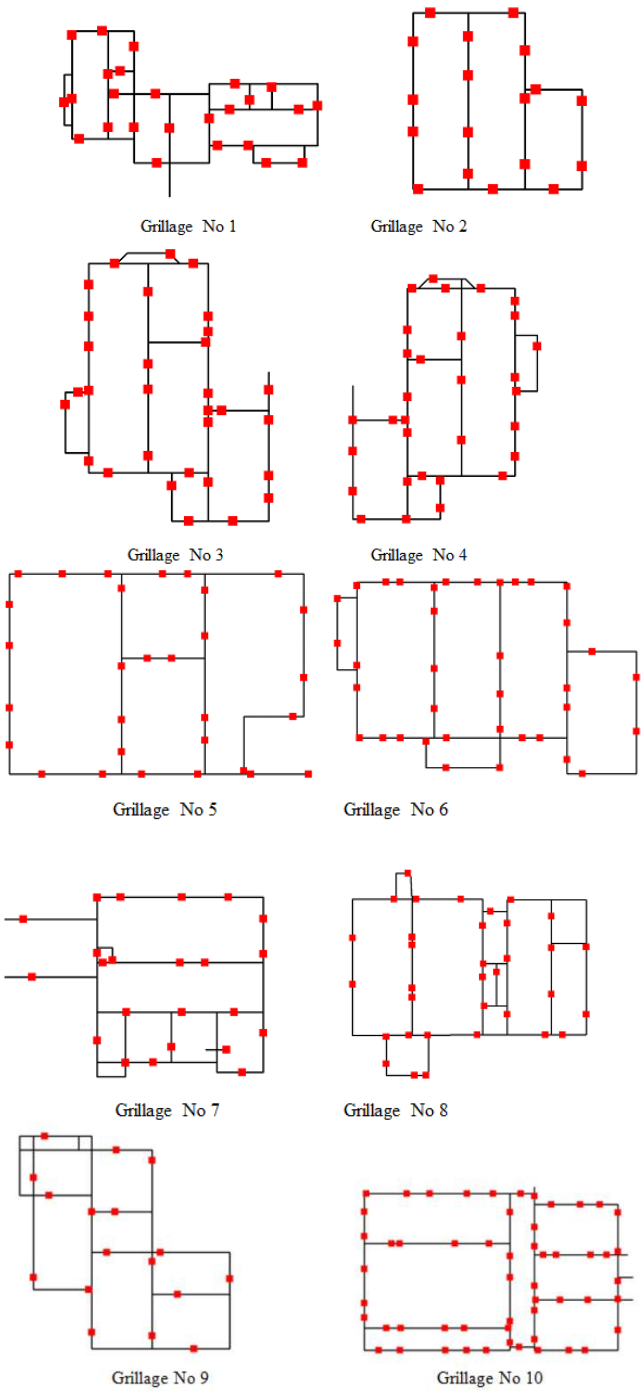


Figure 3: Appendix 3: Best pile placement schemes found with MCGA

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 403-414, June 2017.

An Extension of the VSM Documents Representation

L. Vinţan, D. Morariu, R. Creţulescu, M. Vinţan

Lucian Vinţan, Daniel Morariu,
Radu Creţulescu*, Maria Vinţan
Lucian Blaga University of Sibiu
Romania, Sibiu, Emil Cioran, 4
lucian.vintan@ulbsibiu.ro, daniel.morariu@ulbsibiu.ro
radu.kretzulescu@ulbsibiu.ro, maria.vintan@ulbsibiu.ro
*Corresponding author: radu.kretzulescu@ulbsibiu.ro

Abstract: In this paper we will present a new approach regarding the documents representation in order to be used in classification and/or clustering algorithms. In our new representation we will start from the classical "bag-of-words" representation but we will augment each word with its correspondent part-of-speech. Thus we will introduce a new concept called hyper-vectors where each document is represented in a hyper-space where each dimension is a different part-of-speech component. For each dimension the document is represented using the Vector Space Model (VSM). In this work we will use only five different parts of speech: noun, verb, adverb, adjective and others. In the hyper-space each dimension has a different weight. To compute the similarity between two documents we have developed a new hyper-cosine formula. Some interesting classification experiments are presented as validation cases.
Keywords: documents representation, vector space model, hyper-vectors, documents similarity, classification, clustering.

1 Introduction

One of the main goals of information retrieval is organizing and retrieving information from a large number of text-based documents. Typically, an information retrieval problem is to locate relevant documents based on the user's input, such as keywords or sample documents. Usually information retrieval systems include on-line library catalog systems and on-line document management systems.

An information retrieval system [8] based on similarity finds similar documents using a set of common keywords. The output for this system is based on the degree of relevance measured according to keywords closeness and the relative frequency of the keywords [2, 5]. In some cases, it is difficult to give an accurate measure of the relevance between the keywords set. In modern information retrieval systems, keywords for document representation are automatically extracted from the documents. This system often associates a stopword list with the set of documents. A stopword list is a set of words that are considered "irrelevant" for the representation of the document set and can vary when the document set varies. Another issue is the extraction of the word *stem*. A group of different words may share the same word stem. A text retrieval system needs to identify groups of words where the words in a group have small syntactic variants, and collect only the common word stem per group.

The most common method used in document representation for classification or clustering algorithms is the vector of word frequencies [5]. This method is quite rapid and reliable because it does not require huge computing power. However, the latest approaches for the representation of documents in current applications (which require speed and fast response time such as the majority of the online applications) are not always feasible due to computational cost and the waiting time needed for processing large results.

In this article, we intend to present a new approach for the representation of text documents, which augments the classical VSM (Vector Space Model) representation with some semantic information such as the parts of speech [6] of the words (or other classification of the corresponding words). This new representation we have called H-VSM (Hyper - Vector Space Model). The reason is natural because in this case the new document representation, instead of containing just a single vector of scalars (representing its words occurrences), like in the VSM representation, it contains a hyper-vector containing parts of speech vectors. Such a vector might contain, for example, all the noun-words; another one contains all the verb-words and so on. This idea brings in a new light, a more general VSM representation form, containing new information (the words classes - parts-of-speech in our case). It may open a new effective approach to document classification or clustering using computational techniques. Despite the semantic approaches emergence, these computational methods are still of great interest in documents classification research. The reasons are multiple: they are simpler, easier to implement, faster, independent of the research field, more feasible in some implementations (for example in internet browsers) than the semantic approaches (ontology, NLP, etc.), whose complexities are enormous and the run time, too.

2 The classical VSM representation model for text documents

In the classical VSM representation, a text document is represented as a vector of frequencies of its words [2, 5]. Thus, in this approach for each set of data a dictionary of words (typically the stem-words) will be generated.

Let us consider a set of d documents and a set of t terms used for representing the documents into the information retrieval system. We can represent each document as a vector v in the t -dimensional space $\mathbb{R}^t$. The i^{th} coordinate of $v[]$ is a number that measures the association of the i^{th} term with respect to the given document: it is generally defined as 0 if the document does not contain the term, and nonzero otherwise. The v_i element from a $v[]$ vector, can indicate the frequency of the word in the document. For defining the frequency of the terms (term-weighing) for the nonzero entries in such a vector there are also used many methods [5, 7]. For example, it can be simply defined $v_i = 1$ if the i^{th} term occurs in the document, or let v_i be the term frequency, or normalized term frequency. The term frequency is the number of occurrences of the i^{th} term in the document.

There are different ways to weight the term frequency for the nonzero entries in such a vector. We can simply set value 1 if the term t occurs in the document (binary normalization), or use term frequency or the relative them frequency, that is, the term frequency divided by total number of occurrences of all the terms in the document (nominal normalization). Another way to normalize the term frequency is Cornell SMART representation that uses the following formula:

$$TF(d,t) = \begin{cases} 0 & \text{if } freq(d,t) = 0 \\ 1 + \log\left(1 + \log\left(freq(d,t)\right)\right) & \text{othervise} \end{cases} \tag{1}$$

where $freq(d,t)$ is the frequency of term t in the document d. Based on this representation there is another measure called inverse document frequency that represents the scaling factor for a term t relative to the frequency of term in all documents from the dataset.

$$IDF(t) = \log \frac{1 + |d|}{|d_t|} \tag{2}$$

where d is the document collection and d_t is the set of documents containing term t in a complete vector space model TF and IDF are combined together forming the $TF - IDF$ measure:

$$TF - IDF(d,t) = TF(d,t) \times IDF(t) \tag{3}$$

We expect that similar documents will have similar relative terms frequencies, and we can measure the similarity among a set of documents. There are many metrics for measuring the document similarity. A common used measure is the Euclidean distance but the most used in the literature is the cosine similarity defined as:

$$sim(\overline{v_1}, \overline{v_2}) = \frac{\overline{v_1} \cdot \overline{v_2}}{\|\overline{v_1}\| \ \|\overline{v_2}\|} \tag{4}$$

3 The new H-VSM model

3.1 The rationale for a new vector model representation

Let $\overline{V}$ be a vector, which represents a document in the VSM. Let us agree that all related words to $\overline{V}$ are from the syntactic point of view only of two types (x and y). Keeping the VSM representation, the vector $\overline{V}$ can be written as $\overline{V} = \overline{V_x}\&\overline{V_y}$ where & is the concatenation operator.

But if we consider $\overline{V_x}$ and $\overline{V_y}$ being two orthogonal axes, we could represent $\overline{V}$ as:

$$\overline{V} = V_x \cdot \overline{h_i} + V_y \cdot \overline{h_j} \tag{5}$$

Where $\overline{h_i}$ and $\overline{h_j}$ are the "hyper-unit vectors" of this "plane", because V_x and V_y are vectors (although in formula 5 they are denoted as simple scalars just to simplify the notations). We note the vector $\overline{V}$ in this representation a "hyper-vector" (representing a vector of vectors).

3.2 A first particular model of hyper-vector similarity

Let consider two hyper-vectors represented in a 2D hyper-plane as:

$$\begin{array}{l} \overline{HV_1} = V_{x1} \cdot \overline{h_i} + V_{y1} \cdot \overline{h_j} \\ \overline{HV_2} = V_{x2} \cdot \overline{h_i} + V_{y2} \cdot \overline{h_j} \end{array} \tag{6}$$

where (V_{x1}, V_{y1}) and (V_{x2}, V_{y2}) are the projections of $\overline{HV}_1$ respectively $\overline{HV}_2$ on the orthogonal axes. In text documents representation the vectors V_{x1}, V_{x2} could be, for example, the verb vectors for the document 1 and document 2 and the vectors V_{y1}, V_{y2} could be the noun vectors for the same documents (syntactic extensions are immediate).

As we already pointed out, a common metric used to measure the similarity between two vectors in the VSM representation is the cosine distance. The problem is to consistently define a representation of this distance for 2 "hyper-vectors". We will call this distance "hyper-cosine" and we will note it $hcos(\overline{HV}_1, \overline{HV}_2)$, even if it is, in fact, the cosine between two vectors.

To solve this problem, we will write the "scalar" product of the "hyper-vectors" as:

$$\begin{array}{rcl} \overline{HV_1} \cdot \overline{HV_2} & = & (V_{x1} \cdot \overline{h_i} + V_{y1} \cdot \overline{h_j}) \cdot (V_{x2} \cdot \overline{h_i} + V_{y2} \cdot \overline{h_j}) = \\ & = & V_{x1} \cdot V_{x2} + V_{y1} \cdot V_{y2} \end{array} \tag{7}$$

But:

$$\overline{V_{x1}} = V_{x11} \cdot \overline{i} + V_{x12} \cdot \overline{j} \quad and \quad \overline{V_{x2}} = V_{x21} \cdot \overline{i} + V_{x22} \cdot \overline{j} \tag{8}$$

where V_{x11}, V_{x12} represent the frequencies of occurrence of the syntactic part x (verbs for example) in the vectors $\overline{V_{x1}}$ respectively $\overline{V_{x2}}$. Analogous for the vectors $\overline{V_{y1}}$ respectively $\overline{V_{y2}}$ (representing nouns, for example). Here $\bar{i}$ and $\bar{j}$ are the well-known orthogonal unit vectors.

Substituting the 8 equalities in formula 7 we obtain:

$$\overline{HV_1} \cdot \overline{HV_2} = V_{x11} \cdot V_{x21} + V_{x12} \cdot V_{x22} + V_{y11} \cdot V_{y21} + V_{y12} \cdot V_{y22} \tag{9}$$

It follows that:

$$\begin{aligned} hcos\left(\overline{HV_1}, \overline{HV_2}\right) &= \frac{\overline{HV_1} \cdot \overline{HV_2}}{|HV_1| \cdot |HV_2|} = \\ &= \frac{V_{x11} \cdot V_{x21} + V_{x12} \cdot V_{x22} + V_{y11} \cdot V_{y21} + V_{y12} \cdot V_{y22}}{\sqrt{V_{x1}^2 + V_{y1}^2} \cdot \sqrt{V_{x2}^2 + V_{y2}^2}} \end{aligned} \tag{10}$$

But:

$$\overline{V_{x1}^2} = \left(V_{x11} \cdot \bar{i} + V_{x12} \cdot \bar{j}\right)^2 = V_{x11}^2 + V_{x12}^2 \tag{11}$$

We represent $\overline{V_{y1}^2}$, $\overline{V_{x2}^2}$, $\overline{V_{y2}^2}$ in a similar way. Replacing in 10 we obtain:

$$hcos(\overline{HV_1}, \overline{HV_2}) = \frac{V_{x11} \cdot V_{x21} + V_{x12} \cdot V_{x22} + V_{y11} \cdot V_{y21} + V_{y12} \cdot V_{y22}}{\sqrt{V_{x11}^2 + V_{x12}^2 + V_{y11}^2 + V_{y12}^2} \cdot \sqrt{V_{x21}^2 + V_{x22}^2 + V_{y21}^2 + V_{y22}^2}} \tag{12}$$

We notice that $V_{x11}^2 + V_{x12}^2 = \left|\overline{V_{x1}}\right|^2$ and analogues.

Based on this notation, 12 relationship can be written more concisely:

$$hcos(\overline{HV_1}, \overline{HV_2}) = \frac{\overline{V_{x1}} \cdot \overline{V_{x2}} + \overline{V_{y1}} \cdot \overline{V_{y2}}}{\sqrt{\left|\overline{V_{x1}}\right|^2 + \left|\overline{V_{y1}}\right|^2} \cdot \sqrt{\left|\overline{V_{x2}}\right|^2 + \left|\overline{V_{y2}}\right|^2}} \tag{13}$$

Note: The formulas 12 and 13 are "different" representations for the cosine between two vectors. Indeed, if all the words belong to a single syntactic category (and not two, x and y) formula 13 becomes the well-known formula for the cosine:

$$\cos(\overline{V_1}, \overline{V_2}) = \frac{\overline{V}_1 \cdot \overline{V_2}}{\left|\overline{V_1}\right| \cdot \left|\overline{V_2}\right|} \tag{14}$$

$\cos(\bar{V}_1, \bar{V}_2) = 1$ being equivalent with $\bar{V}_1 = k\bar{V}_2$.

3.3 A generalization of the similarity between hyper-vectors

Considering two hyper-vectors having "n" orthogonal dimensions:

$$\begin{aligned} \overline{HV_1} &= \sum_{k=1}^{n} V_{xk1} \cdot \overline{h_{ik}} \\ \overline{HV_2} &= \sum_{k=1}^{n} V_{xk2} \cdot \overline{h_{ik}} \end{aligned} \tag{15}$$

Formula 13 becomes:

$$hcos(\overline{HV_1}, \overline{HV_2}) = \frac{\sum_{k=1}^{n} \overline{V_{xk1}} \cdot \overline{V_{xk2}}}{\sqrt{\sum_{k=1}^{n} \left|\overline{V_{xk1}}\right|^2} \cdot \sqrt{\sum_{k=1}^{n} \left|\overline{V_{xk2}}\right|^2}} \tag{16}$$

Further, in order not to complicate the notation, we are considering in this first model that all vectors $\overline{V_{xk}}, \cdots k = \overline{1,n}$ have the same length "n". Of course, different lengths for each vector "m_k" can also be considered (as we do in the next section).

In order to generalize formula 12 we can use the following notation:

$$\overline{V_{xk1}} = \sum_{p=1}^{m} V_{xk1}^{p} \cdot \vec{i}_p, \ \forall k = \overline{1,n} \tag{17}$$

(where "p" is considered an index, not a power).

For instance: $\overline{V_{x11}}$ represents the "syntactic" vector of verbs from document number 1 (k=1), $\overline{V_{x21}}$ represents the "syntactic" vector of nouns from document number 1 (k=2), etc.

$$\overline{V_{xk1}} \cdot \overline{V_{xk2}} = \sum_{p=1}^{m} V_{xk1}^{p} \cdot V_{xk2}^{p} \tag{18}$$

Also:

$$\left|\overline{V_{xk1}}\right|^2 = \sum_{p=1}^{m} \left(V_{xk1}^{p}\right)^2 \tag{19}$$

$$\left|\overline{V_{xk2}}\right|^2 = \sum_{p=1}^{m} \left(V_{xk2}^{p}\right)^2 \tag{20}$$

Replacing 18, 19 and 20 in 16 we obtain:

$$hcos(\overline{HV_1}, \overline{HV_2}) = \frac{\sum_{k=1}^{n}\sum_{p=1}^{m} V_{xk1}^{p} \cdot V_{xk2}^{p}}{\sqrt{\sum_{k=1}^{n}\sum_{p=1}^{m} \left(V_{xk1}^{p}\right)^2} \cdot \sqrt{\sum_{k=1}^{n}\sum_{p=1}^{m} \left(V_{xk2}^{p}\right)^2}} \tag{21}$$

Formula 21 represents the generalization of formula 12. The formulas 16 and 21 represent similarities between documents that are represented in a space "syntactically richer" than the VSM, namely a hyper-space which generalizes consistently the VSM representation. Normalized weights of classes of words $(x_1, x_2, \ldots x_n)$ are possible and they lead to authentic generalizations of cosines, meaning:

$$\begin{aligned} \overline{HV_1} &= \sum_{k=1}^{n} \alpha_k \cdot \overline{V_{xk1}} \cdot \overline{h_{ik}} \\ \overline{HV_2} &= \sum_{k=1}^{n} \alpha_k \cdot \overline{V_{xk2}} \cdot \overline{h_{ik}} \quad with \quad \sum_{k=1}^{n} \alpha_k = 1 \end{aligned} \tag{22}$$

$\alpha_k > 0$, will be chosen larger or smaller depending on the greater or lower "semantic importance" of a certain (k) hyper-dimension. Unlike the cases presented so far, in this case we would obtain a *h-cos* type similarity formula, different from the classical cos type, which could have positive consequences in improving the accuracy of document classification algorithms (see formula 23).

$$hcos(\overline{HV_1}, \overline{HV_2}) = \frac{\sum_{k=1}^{n}\left(\alpha_k^2 \cdot \sum_{p=1}^{m} V_{xk1}^{p} V_{xk2}^{p}\right)}{\sqrt{\sum_{k=1}^{n}\left(\alpha_k^2 \cdot \sum_{p=1}^{m} \left(V_{xk1}^{p}\right)^2\right)} \cdot \sqrt{\sum_{k=1}^{n}\left(\alpha_k^2 \cdot \sum_{p=1}^{m} \left(V_{xk2}^{p}\right)^2\right)}} \neq \cos(\overline{HV_1}, \overline{HV_2}) \tag{23}$$

The α_k coefficient can be simplistically computed for instance using:

$$\alpha_k = \frac{n_k}{\sum_{j=1}^{m} n_j}, \quad for\ k = \overline{1,m} \tag{24}$$

n_k represents the length of a certain dimension k ($k = \overline{1,m}$). In this case we consider that a longer vector for a given type of a part of speech could be more important than a shorter one. Also there can be used other formulas for computing the α_k coefficient.

It remains to be proven by experiments that this new representation of the text documents with the new similarity metrics will improve the accuracy of the document classification in accordance with our scientific hypothesis. This hypothesis is based on a rational intuition: including in document representation some new "morphologic" information offers a greater discriminatory power.

3.4 Generalization for "m_k" length different for each $\overline{V_{xk}}$ vector

We are considering in this paragraph that our document is represented in a hyper-space with m dimensions (e.g. the document is represented for m different parts of speech). Each space of this hyper-space has n_k dimensions (any 2 spaces can have different sizes $n_i \neq n_j$). For example n_1 represents the dimension of the nouns vector; n_2 represents the dimension of the verbs vector, etc.

Any two documents are represented in the same hyper-space m and have the same dimension in each space. Thus:

$$\begin{aligned} \overline{HV_1} &= \sum_{k=1}^{m} \overline{V_{k1}} \cdot \overline{h_k} \\ \overline{HV_2} &= \sum_{k=1}^{m} \overline{V_{k2}} \cdot \overline{h_k} \end{aligned} \tag{25}$$

We try to simplify the notations as much as possible. In the formula 21 we have replaced x_k directly with k.

The scalar product of two documents vectors in this hyper-space will be (generalizing formula 9):

$$\overline{HV_1} \cdot \overline{HV_2} = \sum_{k=1}^{m} \sum_{p=1}^{n_k} V_{k1p} \cdot V_{k2p} \tag{26}$$

The norm of the hyper-vector becomes:

$$\left|\overline{HV_1}\right| = \sqrt{\sum_{k=1}^{m} \sum_{p=1}^{n_k} V_{k1p}^2} \tag{27}$$

Using the formula 14 representing the cosine between two vectors and generalizing formula 21 the hyper-cosine in the hyper-space defined in this way becomes:

$$hcos(\overline{HV_1}, \overline{HV_2}) = \frac{\sum_{k=1}^{m} \sum_{p=1}^{n_k} V_{k1p} \cdot V_{k2p}}{\sqrt{\sum_{k=1}^{m} \sum_{p=1}^{n_k} V_{k1p}^2} \cdot \sqrt{\sum_{k=1}^{m} \sum_{p=1}^{n_k} V_{k2p}^2}} \tag{28}$$

This formula is similar with formula 21 (using a different notation style).

Under these circumstances if we consider that the hyper-vector is a single vector with the dimension equal with $n_1 + n_2 + \cdots + n_m$, than the cosine value should be identical.

The initial idea from which we have started was that the vectors are represented as follows:

$$\begin{aligned} \overline{HV_1} &= \sum_{k=1}^{m} \left(\alpha_k \cdot \overline{V_{k1}} \cdot \overline{h_k}\right) \\ \overline{HV_2} &= \sum_{k=1}^{m} \left(\alpha_k \cdot \overline{V_{k2}} \cdot \overline{h_k}\right) \end{aligned} \tag{29}$$

Where $\sum \alpha_k = 1$ (equilibrium relationship).

Then the hyper-cosine becomes:

$$hcos(\overline{HV_1}, \overline{HV_2}) = \frac{\sum\limits_{k=1}^{m} \left(\alpha_k^2 \cdot \sum\limits_{p=1}^{n_k} V_{k1p} \cdot V_{k2p}\right)}{\sqrt{\sum\limits_{k=1}^{m} \left(\alpha_k^2 \cdot \sum\limits_{p=1}^{n_k} V_{k1p}^2\right)} \cdot \sqrt{\sum\limits_{k=1}^{m} \left(\alpha_k^2 \cdot \sum\limits_{p=1}^{n_k} V_{k2p}^2\right)}} \tag{30}$$

In this case the value for the *hcos* can be different from the value obtained by a simple concatenation of the part-of speech vectors.

Again the α_k coefficient may be computed for instance as in the following formula:

$$\alpha_k = \frac{n_k}{\sum\limits_{j=1}^{m} n_j}, \quad for\ k = \overline{1, m} \tag{31}$$

This idea could bring in a new light the VSM representation. The alpha coefficients for each part of speech depend more on the semantics of the documents than on the percentage for a certain part of speech (as we have previously suggested in 24 formula). Therefore choosing the optimal alpha weights remains an open problem. In this article, we have tried finding, in an empirical way, the coefficients' values so that the classification accuracy increases, without claiming to have found the optimal values.

3.5 Other generalized formulas used to measure the similarity between two vectors

Considering that all the $\overline{V_{xk}}, \cdots k = \overline{1, n}$, vectors have the same length "n", the Euclidian hyper-distance becomes:

$$hd_{Eucl}(\overline{HV_1}, \overline{HV_2}) = \sqrt{\sum_{k=1}^{m} \sum_{p=1}^{n} \left(V_{xk1}^p - V_{xk2}^p\right)^2} \tag{32}$$

As well it would be possible to weight classes of words, depending on their importance, leading to consistent generalizations of the Euclidean distance, with potentially positive influences on the document classification algorithms. In this case the Euclidean distance for different space dimension m_k becomes:

$$hd_{Eucl}(\overline{HV_1}, \overline{HV_2}) = \sqrt{\sum_{k=1}^{m} \left(\alpha_k^2 \sum_{p=1}^{n_k} \left(V_{k1p} - V_{k2p}\right)^2\right)} \tag{33}$$

The dot product between two vectors can be expressed as:

$$\overline{HV_1} \cdot \overline{HV_2} = \sum_{k=1}^{m} \left(\alpha_k^2 \sum_{p=1}^{n_k} V_{k1p} \cdot V_{k2p} \right) \tag{34}$$

Similarly other distances can be expressed using weight classes generalization (City-block distance, etc.).

$$hd_{CB}\left(\overline{HV_1}, \overline{HV_2}\right) = \sum_{k=1}^{m} \left(\alpha_k^2 \sum_{p=1}^{n_k} |V_{k1p} - V_{k2p}| \right) \tag{35}$$

4 Improvements brought to document classification

4.1 Tagging the Reuters dataset

For validating the above presented theoretical results, based on a set of documents, we have tried a separation of words according to their part of speech. After that step, we have then performed a vector representation of documents as vectors of frequencies of words but we have taken into account also the part of speech for the given words. In this way, we have obtained an augmented VSM representation. In this new representation each vector in the hyper-vector space represents a part of speech component. We want to compare the results of the new representation with the results obtained by us in previous experiments where documents were represented only by frequencies of words vectors [9]. We have used the Support Vector Machine (SVM) algorithm for classifying text documents. Because SVM is a supervised learning algorithm, we need a set of data that is tagged in terms of parts of speech and classified in terms of documents belonging to classes. These classes are defined axiomatically according to the content of documents. In initial experiments, we have used the Reuters 2000 dataset [10] which contains documents that are pre-classified by the Reuters news agency but without having any information about the part of speech of words contained. In our experiments, regarding the parts of speech, we have used the Brown Corpus [1] that is a corpus labeled in terms of part of speech but not classified into categories (classes) by document contents.

Therefore, in the first step we have tried to label the words from the documents contained in the Reuters database with their corresponding parts of speech. For this purpose, using the Brown Corpus we have evaluated several known tagging (labeling the part of speech) applications and also some tagging applications which were developed by us [3], in order to find the best suited tagging application / applications for labeling the documents. The entire experiment for the Tagger selection was explained in our article published in [4].

In [4] we have performed a number of experiments for selecting the "best" tagger in order to tag the Reuters dataset. Analyzing the obtained results we have decided to use, in the first experiments, only the Tree Tagger tool [11]. This tool has obtained the best results for three of the tested parts of speech: noun, verb and adverb. Hopefully, in further experiments, we'll combine the results from more taggers (through meta-tagging) in order to increase the probability to obtain the correct part of speech for a word in a given context.

After tagging our Reuters dataset and representing each document as a vector of words frequencies, where each word is separated based on its part of speech, we have obtained 27240 different words (for all 5 parts-of speech taken into consideration). In the next table we present for each part of speech the number of words discovered by the tagger in the Reuters Dataset.

After the prediction of the part of speech using the Tree Tagger tool, we have extracted the stem of the words and after that we have recorded the words' frequencies. We have obtained

Table 1: POS for the Reuters data

Part of Speech	**# of words**	**% of total word**
Nouns	16820	61.75
Verbs	3668	13.47
Adjectives	5555	20.39
Adverbs	1001	3.67
Others	196	0.72
Total	27240	

27240 different words, which mean in the hyperspace to have a representation with five different spaces, each space having a different dimension. For example, the space of nouns has 16820 dimensions. It is remarkable that the Tree tagger has labeled in the "other" category only 196 stem of words that represents less than 1% from all stem of words extracted from the Reuters dataset.

4.2 Obtained results with SVM Classifier

We have started some initial experiments using our tagged Reuters data set in order to determine if there are some improvements in the classification accuracy using this new representation. In the SVM classifier, we have decided to use both polynomial and Gaussian kernels as it was presented in [9]. The results obtained using the SVM classifier and all 27240 features are presented in the Table 2. These results were compared with previous results obtained using a vector with 18424 features but without POS. (The new document representation has a bigger dimension because a certain word could belong to multiple parts of speech). This was the biggest dimension obtained for the words frequencies vectors representation. In [9] it was demonstrated that if the vector dimension decreases and fewer features are chosen through some feature selection techniques, than the noise introduced in the classification algorithm is smaller and the learning is better.

With this new representation it was interesting to observe that although the number of features was higher, which would theoretically induce more noise in the document representation, the classification results are at average with 0.85% better for the polynomial kernel and with 1.1% at average better for the Gaussian kernel. These results give us hope for future better results especially for the Gaussian kernel.

In the Table 2 we have marked with bold the highest values for the two developed experiments. When the part of speech was introduced, the results improved (even if we have weights equal to 1 for the hyper-vectors) especially for small degrees of the kernels (so there is no need for a shift into a higher space). Also the results are more equilibrate for different kernel dimensions that mean that we can obtain better results searching only in few spaces.

Further we have selected only the best 1309 features from all 27240 features using the information gain as feature selection method [7]. In Table 3 we present the number of words obtained, according to their part of speech.

In the Table 4 we present the results of our classification experiments where we have applied the formula 34 for computing the dot product between two hyper-vectors, considering that each hyper-vector consists of five vectors with different sizes for each part of speech. In these experiments we have weighted with 0.8 the verbs and the adverbs, with 0.9 the nouns and the adjectives and with 0.6 the "other" category. The obtained results are shown in the Table 4. The values of coefficients were chosen close to 1 (value that is used in case of computing the classical dot product between two vectors for example, the formula 14).

Table 2: VSM versus VSM-with-POS for all features using SVM classifier

Polynomial Kernel				**Gaussian Kernel**			
Degree of Kernel	Data Representation	for 18428 features (VSM)	for 27240 features (VSM with POS)	Degree of Kernel	Data Representation	for 18428 features (VSM)	for 27240 features (VSM with POS)
P1.0	BIN	83.03	**83.41**	C1.0	BIN	82.01	**82.22**
P1.0	NOM	86.22	**86.98**	C1.0	SMART	81.75	**84.22**
P1.0	SMART	82.52	**84.01**	C1.3	BIN	82.69	**82.86**
P2.0	BIN	85.79	85.16	C1.3	SMART	82.39	**84.43**
P2.0	NOM	85.50	**85.67**	C1.8	BIN	82.86	**83.03**
P2.0	SMART	85.92	**86.39**	C1.8	SMART	82.60	**84.26**
P3.0	BIN	83.96	76.05	C2.1	BIN	82.56	**82.77**
P3.0	NOM	84.94	**85.92**	C2.1	SMART	82.43	**84.18**
P3.0	SMART	77.16	**85.03**	**Average**		**82.41**	**83.50**
P4.0	BIN	53.64	**64.44**				
P4.0	NOM	82.99	82.56				
P4.0	SMART	59.34	55.47				
Average		**75.25**	**80.09**				

Table 3: Number of words selected according to their POS

Part of Speech	**# of words**	**% of total word**
Nouns	683	52.18
Verbs	289	22.08
Adjectives	188	14.369
Adverbs	63	4.81
Others	86	6.57
Total	1309	

Table 4: VSM versus VSM-with-POS for 1309 features using SVM classifier

Polynomial Kernel				**Gaussian Kernel**			
Degree of Kernel	Data Representation	for 1309 features (VSM)	for 1309 features (VSM with POS)	Degree of Kernel	Data Representation	for 1309 features (VSM)	for 1309 features (VSM with POS)
P1.0	BIN	**81.45**	80.99	C1.0	BIN	82.99	**84.05**
P1.0	NOM	**86.69**	86.39	C1.0	SMART	82.99	**84.77**
P1.0	SMART	80.99	**81.07**	C1.3	BIN	83.74	**84.05**
P2.0	BIN	**86.64**	85.11	C1.3	SMART	83.57	**84.56**
P2.0	NOM	**85.03**	84.94	C1.8	BIN	83.24	**83.96**
P2.0	SMART	**87.11**	85.03	C1.8	SMART	84.30	**84.47**
P3.0	BIN	**85.79**	85.33	C2.1	BIN	83.11	**84.05**
P3.0	NOM	**84.35**	82.52	C2.1	SMART	83.83	**84.13**
P3.0	SMART	**86.51**	85.71	**Average**		**83.47**	**84.25**
P4.0	BIN	74.61	**83.03**				
P4.0	NOM	**81.54**	79.20				
P4.0	SMART	71.84	**81.67**				
Average		**82.71**	**83.41**				

As a first observation, it is interesting to point out that the average classification accuracies for the 1309 features representation are significantly better (up to 7.46%) than those obtained using all features representation (see Table 4 vs. Table 2).

Regarding the polynomial kernel, most results are slightly lower than that obtained without the POS. Interesting, all the VSM with POS results are close to a certain value for both small degrees and high degrees kernel values suggesting that the new representation would not be affected so much by shifting into another higher space. However, at average, the new representation gets better results with 0.70% for the polynomial kernel. Using the new representation with the Gaussian kernel, we have obtained constantly slightly better results than with the classical VSM representation. More precisely, at average the SVM with Gaussian kernel obtained, using the new representation, an improvement of 0.78% using only 1309 features, compared to the simple frequency of words vector representation.

As we have already mentioned, the selection for the weighting values for each vector from the hyper-space remains an open problem because it is difficult to determine what contribution has each part of speech regarding the quality of the classification accuracy. In future experiments we will try to find different methods and strategies for computing the optimal values for the weights.

5 Conclusions and future works

In our paper, we have presented a possible improvement for the VSM representation used in text documents classification adding some new morphological information, transforming the vectors of documents into hyper-vectors, which contain information about the part of speech of the words (just in our case presented here). We have also developed a new formula for the cosine between two hyper-vectors starting from the well-known formula for the cosine distance between two vectors. In fact, the proposed model is more general, because it tries to augment the classical

representation which leads to a separation of the representations that can have different weights.

Considering these first experiments we have observed that such a representation which adds supplementary information helps to achieve better classification results. We intend to improve this information trying to find the most optimal representation. Anyway, even if the classification algorithms will have no improvement with this representation, this does not necessarily mean that the idea of "representation by hyper-vectors" is bad; this means that the used hyper-space is inappropriate for the purpose (classification). The chosen representations might not lead to a better discrimination. If we choose otherwise, it may lead to a better classification. What representation should be used to obtain a better classification? Well, that nobody knows, we can only make assumptions based on intuition (common-sense).

As a further work idea, instead of using the parts of speech for words we can consider to use the parts of the sentence (subject, predicate, attribute, etc.) or, more generally, we can consider to group information in other quasi-orthogonal categories (these categories could be whatever: syntactic, morphologic, etc.) and weighted each category separately and compute the similarity. (For example we can have "beautiful" words and "ugly" words. If this involves better classification accuracy, why not?)

Bibliography

[1] Brown University Standard Corpus of Present-Day American English (Brown Corpus), [Online] http://icame.uib.no/brown/bcm.html, accessed in April 2014.

[2] Chakrabarti S.(2003); *Mining the Web- Discovering Knowledge from Hypertext Data*, Morgan Kaufmann Press, 2003.

[3] Cretulescu R., David A., Morariu D., Vintan L. (2014); Part of Speech Tagging with Naive Bayes Methods, *Proceedings of The 18-th International Conference on System Theory, Control and Computing*, Sinaia (Romania), doi: 10.1109/ICSTCC.2014.6982457, 446-451, 2014.

[4] Cretulescu R., David A., Morariu D., Vintan L. (2015); Part of Speech Labeling for Reuters DataBase, *Proc. of The 19-th International Conference on System Theory, Control and Computing*, Gradistea (Romania), doi: 10.1109/ICSTCC.2015.7321279, 117-122, 2015.

[5] Han J., Kamber M. (2001); *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, 2001.

[6] Manning D., Schütze H. (1999); *Foundations of Statistical Natural Language Processing*, MIT Press, ISBN: 987–0–262–133360–9, 1999.

[7] Mitchell T. (1999); *Machine Learning*, McGraw Hill Publishers, 1997.

[8] Mitkov R. (2005); *The Oxford Handbook of Computational Linguistics*, Oxford University Press, 2005.

[9] Morariu D. (2008); *Text Mining Methods based on Support Vector Machine*, MatrixRom, Bucharest, 2008.

[10] Reuters Corpus, [Online] http://about.reuters.com/researchandstandards/corpus/, Released in November 2000.

[11] Tree tagger, [Online] http://www.cis.uni-muenchen.de/ schmid/tools/TreeTagger, accessed in April 2014.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 415-428, June 2017.

Compensation of Time-Varying Delay in Networked Control System over Wi-Fi Network

H.-C. Yi, C.-J. An, J.-Y. Choi

Hyun-Chul Yi, Cheol-Jin An, Joon-Young Choi*
Department of Electronics Engineering
Pusan National University, Pusan, Korea
2, Busandaehak-ro 63beon-gil, Geumjeong-gu,
Pusan, 46241, Korea
hcy@pusan.ac.kr, cja@pusan.ac.kr
*Corresponding author: jyc@pusan.ac.kr

Abstract: In this study, we design a state predictor-based output feedback controller that compensates for unavoidable time-varying network delays in networked control systems (NCSs) over Wi-Fi networks. We model time-varying network delays as time-varying input delays of NCSs over Wi-Fi networks. The designed controller consists of a linear quadratic regulator (LQR), a full-order observer, and a time-varying step-ahead state predictor. The state predictor plays a key role in compensating for the time-varying input delay by providing the LQR with an estimation of future states ahead by the current network delay time. The time-varying network delays are acquired in real time by measuring the time differences between sent and received control data packets. We verify the stability and compensation performance of the designed controller by performing extensive experiments for an NCS in which a rotary inverted pendulum is controlled over Wi-Fi networks.
Keywords: networked control system (NCS), Wi-Fi network, time-varying delay, state predictor, rotary inverted pendulum.

1 Introduction

Networked control systems (NCSs) are spatially distributed systems in which sensors, actuators, and controllers exchange I/O information through a shared band-limited digital communication network. NCSs have been applied to a broad range of areas such as wireless sensor networks (WSNs), remote surgery, haptics collaboration over the Internet, automated highway systems, and unmanned aerial vehicles (UAVs) [4, 10, 25]. In particular, wireless networked control systems (WNCSs) have been increasingly applied in different fields because of the need for mobile operations, flexible installations, and rapid deployment in many applications.

As an alternative solution to wired NCSs, WNCSs are considered as primary solutions of networked control applications because of their simple configuration and mobility. However, the reliability and real-time performance of WNCSs are lower than for wired NCSs because with wireless networks, the sizes of network delays abruptly changes over time owing to the dynamic state variation of wireless networks. It is commonly known that even a small time delay in control system feedback loops can make the whole system oscillating or unstable [4, 15, 25], and it is obvious that fluctuating time-varying delays severely degrade the stability and performance in WNCSs.

In order to deal with the network delays, various control schemes for WNCSs are proposed in [1–3, 5–9, 11–14, 16–23, 26, 27]. These include: fuzzy-based control [6, 22], predictor-based control [13, 27], PID-based control [7, 14, 19, 20], H_∞ filter-based control [1, 5, 17, 21, 23], fault-tolerant-based control [3, 9], linear quadratic regulator (LQR)-based control [8], linear matrix inequality (LMI)-based control [18], Kalman filter-based control [12, 16, 26], and observer-based control [11].

When constructing these control schemes, it is necessary to model the network delays. In [1,3,8,16,22], the network delays are modeled as constant-valued delays or bounded time-varying delays that do not suitably describe the rapidly fluctuating time-varying delays in WNCSs. In [2, 5–7, 9, 11–14, 17–21, 23, 26, 27], the network delays are modeled as time-varying delays, but the stability and performance of proposed control schemes are demonstrated only by performing network or numerical simulations using OPNET++, TRUETIME network simulators, or Matlab/Simulink. Considering dynamically changing states of wireless networks, the stability and performance of controllers designed for WNCSs must be verified by performing experiments over real wireless networks.

Moreover, all of the controllers in [1–3, 5–9, 11–14, 16–23, 26, 27] were designed under the assumption that the network delays are known in advance in the form of constant-valued or time-varying delays for all operation times. However, this is not the case in real operations of WNCSs, but time-varying network delays must be measured in real time, and should be used for the control action to compensate for the time-varying delays. Hence, a real-time measurement method for time-varying network delays must be designed and be a part of WNCSs.

In order to cope with the problems in the existing results of WNCSs, we design a state predictor-based output feedback controller to compensate for the time-varying network delays of NCSs over Wi-Fi networks. We model Wi-Fi network delays as time-varying input delays, and we design a control scheme consisting of an LQR, a full-order observer, and a time-varying step-ahead predictor. Moreover, we design a real-time measurement method for time-varying network delays, and we use the measured delays to construct the state predictor in real time. To apply the designed control scheme, we construct a WNCS hardware platform where the rotary inverted pendulum is controlled over Wi-Fi networks, and we conduct extensive experiments to verify the stability and performance of the designed predictor-based controller.

This paper is organized as follows. In Section 2, we formulate the control problem. In Section 3, we propose the design of a state predictor-based output feedback controller. In Section 4, we propose a delay-measurement method in a WNCS hardware platform. In Section 5, we conduct experiments and discuss the results. In Section 6, we conclude the paper.

2 Problem formulation

We consider a WNCS over a Wi-Fi network, where the plant belongs to the class of single-input multi-output systems, and the controller output is transferred via a Wi-Fi network, as depicted in Fig. 1. In order to precisely describe the dynamic changes in the network state of WNCS, we model the network delays as time-varying delays, $d(k)$, and we assume that the measured values of the plant output are directly available for the construction of the controller. With respect to the models of the plant and controller, we choose discrete-time models because modern controllers are usually implemented using digital computers [2], and continuous-time controllers inevitably involve a degrading discretization process during implementation. The discrete-time linear plant including the network delays is described by the following model.

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k - d(k)) \\ y(k) &= Cx(k), \end{aligned} \tag{1}$$

where $x(k) \in \mathbb{R}^n$ is the system state, $u(k) \in \mathbb{R}$ is the control input, $d(k)$ is the time-varying input delay, $y(k) \in \mathbb{R}^q$ is the plant output, and it is assumed that the pair (A, B) are controllable and (A, C) are observable.

The goal of this paper is summarized as follows. First, we design a state predictor-based output feedback controller for a discrete-time linear system (1) in order to compensate for the

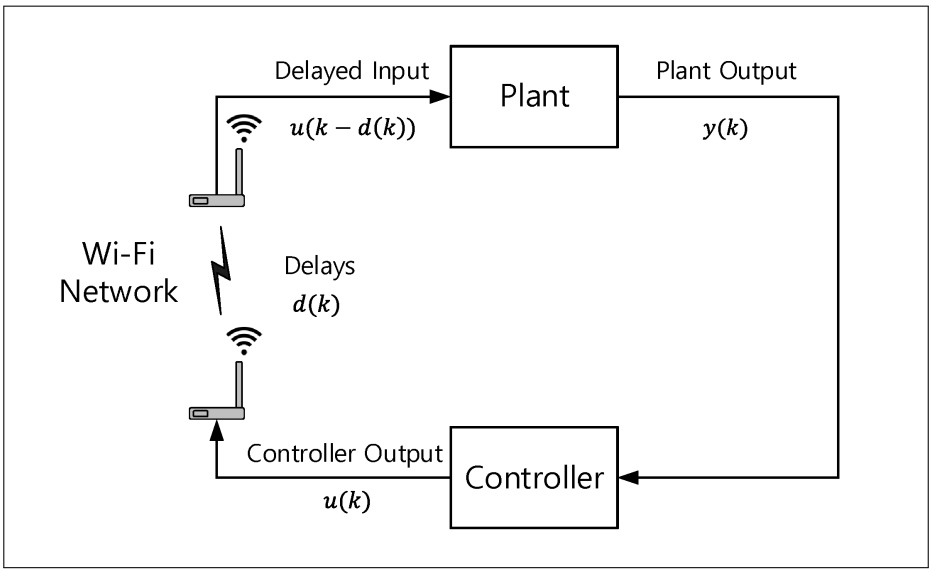


Figure 1: Structure of WNCS over Wi-Fi network.

time-varying input delay and regulate the system output to a desired reference. Second, we develop a real-time method to measure the time-varying network delays in Wi-Fi networks. The measured network delays are used for the implementation of the designed controller in real time. Finally, we construct the hardware platform of a WNCS with a rotary inverted pendulum as the plant for experiments of the designed controller.

3 Design of predictor-based feedback controller

We design the predictor-based feedback controller by performing the following three steps. In the first step, we design a full-state feedback controller for the plant with no input delay by applying discrete-time LQR theory. In the second step, we design a full-order observer that estimates the plant states. In the last step, we design a state predictor for the compensation of the time-varying input delay.

First, we design a static state feedback controller $u(k) = -Kx(k)$ for the plant (1) with no input delay (i.e., with $d(k) = 0$) such that the designed controller can stabilize the plant (1); that is, the matrix $(A - BK)$ becomes Schur stable. The existence of the stabilizing state feedback gain, K, is guaranteed by the assumption that the pair (A, B) is controllable [24]. By applying the discrete-time LQR theory, we can easily obtain the optimal state feedback controller

$$u(k) = -Kx(k), \tag{2}$$

which stabilizes the plant system (1) with no input delay and minimizes the following quadratic cost function:

$$J(u) = \sum_{k=1}^{\infty} \left(x(k)^T Q x(k) + R u^2(k) \right), \tag{3}$$

where the state-cost matrix, Q, and the performance index constant, R, are design parameters [24].

The implementation of the state feedback controller (2) requires knowledge of all state variables, and it is necessary to estimate the system state, $x(k)$, from the measurement of the system

output $y(k)$. For this purpose, we design the full-order observer as follows.

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + Bu(k-d(k)) + L\left(y(k) - \hat{y}(k)\right) \\ \hat{y}(k) &= C\hat{x}(k), \end{aligned} \tag{4}$$

where $\hat{x} \in \mathbb{R}^n$ is the estimated state, $\hat{y} \in \mathbb{R}^q$ is the estimated output, and $L \in \mathbb{R}^{n\times q}$ is the observer gain, which is designed such that the matrix $(A - LC)$ becomes Schur stable; that is, the designed observer becomes exponentially stable [24]. Then, for the designed observer (4), it holds that $\hat{x}(k)$ exponentially converges to $x(k)$, and it is proven that the estimated state, $\hat{x}(k)$, can be used to construct the state feedback controller (2) instead of the unavailable actual state, $x(k)$.

Finally, we seek the final controller that is constructed with the estimated state as

$$u(k - d(k)) = -K\hat{x}(k), \tag{5}$$

which can be alternatively written as

$$u(k) = -K\hat{x}(k + d(m)), \tag{6}$$

where m satisfies $m - d(m) = k$, and it is non-implementable because it requires future values of state. However, the $d(m)$-step-ahead predictor is designed in [2] as

$$\hat{x}(k + d(m)) = A^{d(m)}\hat{x}(k) + \sum_{j=k-d(m)}^{k-1} A^{k-j-1}Bu(j), \tag{7}$$

which yields the implementable predictor-based feedback controller

$$u(k) = -K\left[A^{d(m)}\hat{x}(k) + \sum_{j=k-d(m)}^{k-1} A^{k-j-1}Bu(j)\right]. \tag{8}$$

The closed-loop system with the designed controller (8) is globally exponentially stable in the sense of the norm $\left(|\hat{x}(k)|^2 + \sum_{j=0}^{D-1} |u(k+j-d(m+j))|^2\right)^{1/2}$, where D is an upper bound of the time-varying input delay [2].

Remark 1. In order to implement the designed controller (8), we must find m satisfying $m - d(m) = k$ at each time k, which requires some knowledge of future input delay, $d(m)$. Practically, it is difficult to know the future time-varying input delay in advance. We overcome this problem by measuring the time-varying delay at the plant input, and adopting the measured delay as an estimate of $d(m)$.

The overall structure of the WNCS with the designed controller is depicted in Fig. 2. As shown in Fig. 2, the delayed input value is applied to the plant input, and the measured delays in real time are used as the time-varying input delay, $d(m)$, for the state predictor. Once the time-varying input delay, $d(m)$, is estimated, the state predictor can compensate for the time-varying delay and stabilize the entire closed-loop system.

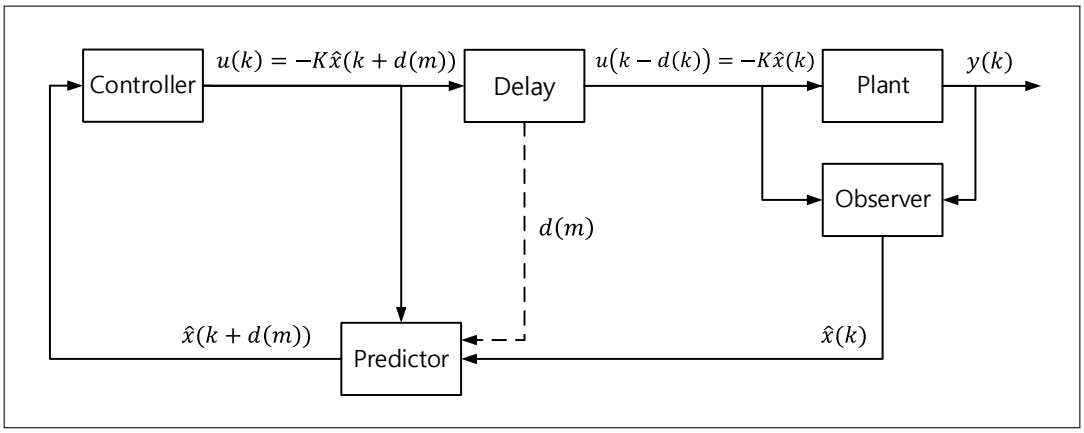


Figure 2: WNCS with the predictor-based feedback controller.

4 Delay measurement method in WNCS hardware platform

As emphasized in Remark 1, in order to implement the designed controller (8), we need to measure the time-varying delay in real time that occurs in WNCS. In this section, we design a delay-measurement method for WNCS over Wi-Fi networks by extending the method proposed in [25] to measure WSN (Wireless Sensor Network) delays in real time.

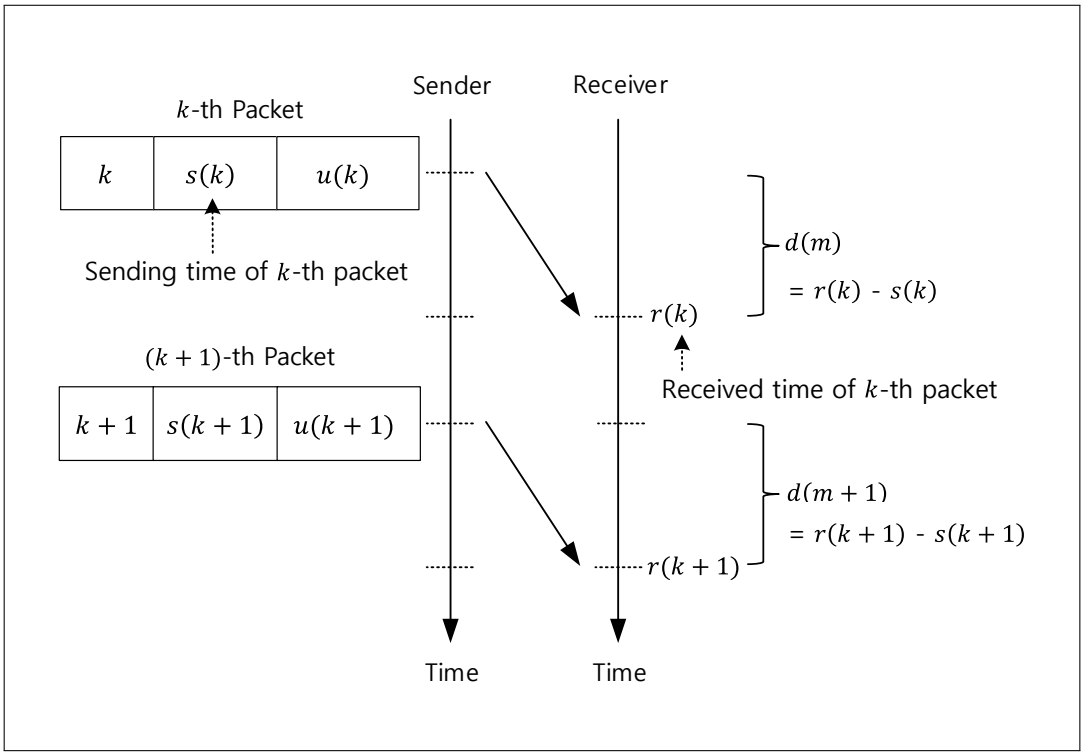


Figure 3: Measurement method for time-varying network delay.

The basic idea of the delay-measurement method is to measure the transmission delay of signal packets from the controller output to the plant input. When controller output packets are sent, the time of sending is appended to the packets; when they are received at the plant input, the transmission delay is calculated by subtracting the time of sending from the present time.

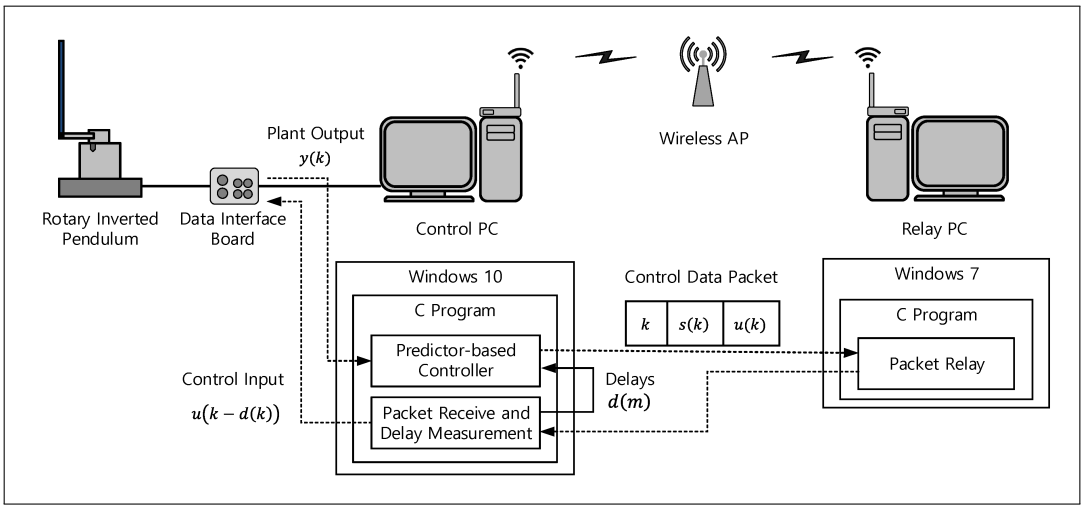


Figure 4: Structure of WNCS hardware platform.

Then, the calculated transmission delay is used to generate the next time-step controller output as the estimated delay of $d(m)$ in (8). This basic operation principle is depicted in Fig. 3 in the level of transmission packets.

In order to implement and verify the designed controller and the delay-measurement method, we construct the WNCS hardware platform, as depicted in Fig. 4, where the rotary inverted pendulum is chosen as the physical plant to be controlled. The rotary inverted pendulum is well known to be nonlinear and unstable, and it is commonly accepted as a test plant to demonstrate the stability and performance of designed controllers [4].

As shown in Fig. 4, the pendulum is connected through a data interface board to an Intel i7-6700 3.4 GHz desktop PC (control PC), which acquires the pendulum position data, computes the control input, and transmits the control input to another Intel i5-3570 3.4 GHz desktop PC (relay PC) over a Wi-Fi network. The relay PC retransmits received packets from the control PC to the control PC over the Wi-Fi network, which results in the time-varying delay in the WNCS.

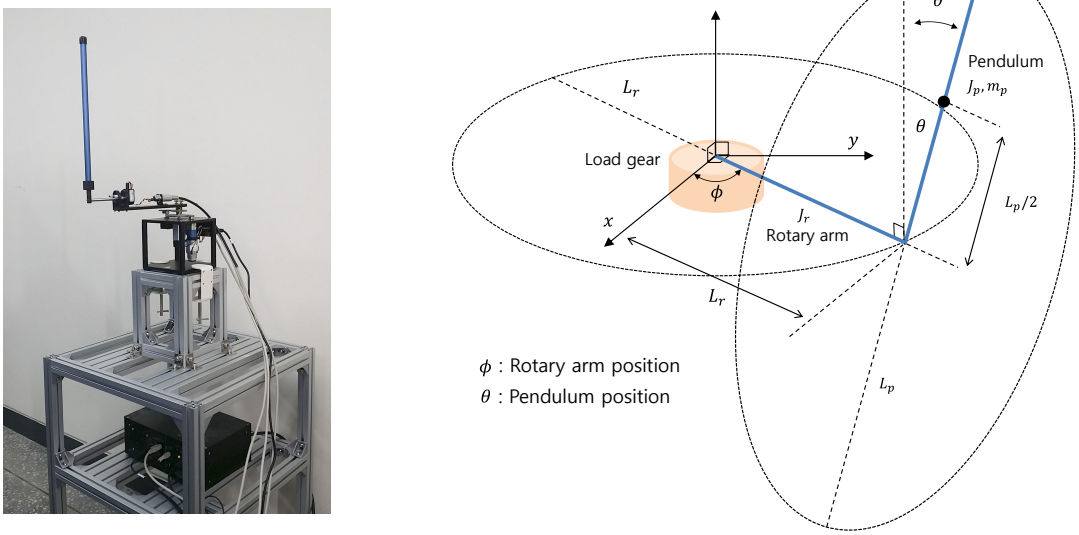
The delay-measurement method is implemented on the control PC such that the sending time is added to each sending packet, and the received time of each packet from the relay PC is recorded. Then, by subtracting the sending time from the received time, the current network delay is obtained and used as an estimate of $d(m)$ in (8).

5 Experiment

5.1 Implementation of designed controller

As mentioned in Chapter 4, we choose the rotary inverted pendulum as the physical plant; Fig. 5 shows the photograph and schematic diagram of the pendulum's movement, and the actual parameters of the pendulum are listed in Table 1.

In order to apply the designed controller (8) to the pendulum over the Wi-Fi network, we first linearize the continuous-time nonlinear model of the pendulum. Then, we convert the continuous-time linear model into the discrete-time linear model because the controller is designed based on a discrete-time linear system. We compute the linearized equations using the Taylor series expansion, and the discretized equations using the zero-order hold discretization method. Based on the parameters in Table 1 and the sampling period of $2ms$, we obtain the following discrete-



(a) Photograph of rotary inverted pendulum (b) Schematic diagram and coordinate system

Figure 5: Rotary inverted pendulum.

Table 1: Parameters of rotary inverted pendulum.

Parameters	**Description**	**Value**
L_r	The Length of the Rotary Arm	$21.6cm$
J_r	The Moment of Inertia of the Rotary Arm	$9.98 \times 10^{-4} kg.m^2$
m_r	The Mass of the Rotary Arm	$0.257kg$
L_p	The Length of the Pendulum	$33.7cm$
J_p	The Moment of Inertia of the Pendulum	$0.0012kg.m^2$
m_p	The Mass of the Pendulum	$0.127kg$
η_g	The Efficiency of the Gearbox	$0.90(\pm 10\%)$
K_g	The Total Gear Ratio of the High-Gear	70
η_m	The Efficiency of the Motor	$0.69(\pm 5\%)$
k_t	The Current-torque Constant of the Motor	$7.68 \times 10^{-3} N - m/A$
k_m	The Back-EMF Constant of the Motor	$7.68 \times 10^{-3} V/(rad/s)$
R_m	The Armature Resistance of the Motor	$2.6\Omega(\pm 12\%)$
B_p	The Viscous Damping Coefficient of the Pendulum	$0.0024Nm(s/rad)$
B_r	The Viscous Friction Torque of the Pendulum	$0.0024Nm(s/rad)$

time linear model for the rotary inverted pendulum system.

$$\begin{aligned} x(k+1) &= \begin{bmatrix} 1 & 0.0002 & 0.0019 & 0 \\ 0 & 1.0002 & -0.0001 & 0.0020 \\ 0 & 0.1554 & 0.9125 & -0.0016 \\ 0 & 0.2368 & -0.0842 & 0.9975 \end{bmatrix} x(k) + \begin{bmatrix} 0.0002 \\ 0.0002 \\ 0.1594 \\ 0.1533 \end{bmatrix} u(k) \\ y(k) &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x(k). \end{aligned} \tag{9}$$

where the state vector $x(k)$ and output vector $y(k)$ consist of the rotary arm position ϕ, the pendulum position θ, and their velocities as

$$\begin{aligned} x(k) &= \begin{bmatrix} \phi(k) & \theta(k) & \dot{\phi}(k) & \dot{\theta}(k) \end{bmatrix}^T \\ y(k) &= \begin{bmatrix} \phi(k) & \theta(k) \end{bmatrix}^T, \end{aligned} \tag{10}$$

and the input, $u(k)$, is the motor input voltage.

Now, we are ready to construct the designed controller (8). We obtain the state feedback gain by applying the LQR method to the plant model (9) using the following parameters in the cost function (2):

$$Q = \begin{bmatrix} 30 & 0 & 0 & 0 \\ 0 & 30 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{ and } R = 1, \tag{11}$$

which yields the state feedback gain as

$$K = \begin{bmatrix} 5.3456 & -24.8038 & 2.9490 & -3.3639 \end{bmatrix}, \tag{12}$$

which assigns the eigenvalues of $(A - BK)$ to $\{0.9114,\, 0.9699, 0.9902 + 0.0039j, 0.9902 - 0.0039j\}$ and makes $(A - BK)$ Schur stable.

With respect to the observer, the eigenvalues of $(A - LC)$ must be selected much faster than the eigenvalues of $(A - BK)$. Considering the eigenvalues of $(A - BK)$ from the state feedback, we select the desired eigenvalues of $(A - LC)$ as $\{-0.2000, -0.2100, -0.2200, -0.2300\}$, which yields the following observer gain:

$$L = \begin{bmatrix} 2.3441 & -0.0100 \\ -0.0955 & 2.4262 \\ 666.0953 & -6.2360 \\ -77.0342 & 735.4454 \end{bmatrix}. \tag{13}$$

Using (9), (11), and (13), we can implement the predictor-based feedback controller as

$$u(k) = -K \left[A^{d(m)} \hat{x}(k) + \sum_{j=k-d(m)}^{k-1} A^{k-j-1} B u(j) \right]. \tag{14}$$

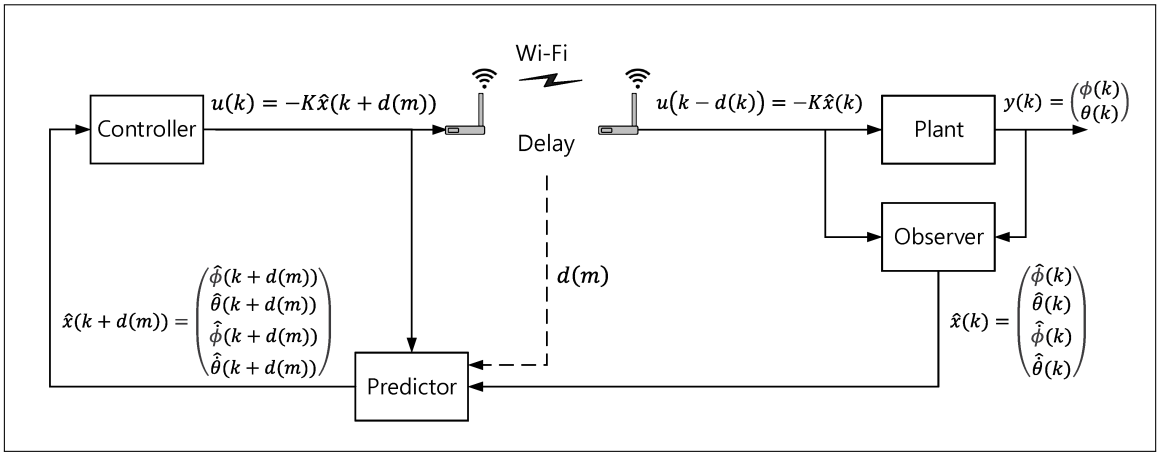


Figure 6: WNCSs in the second and the third case.

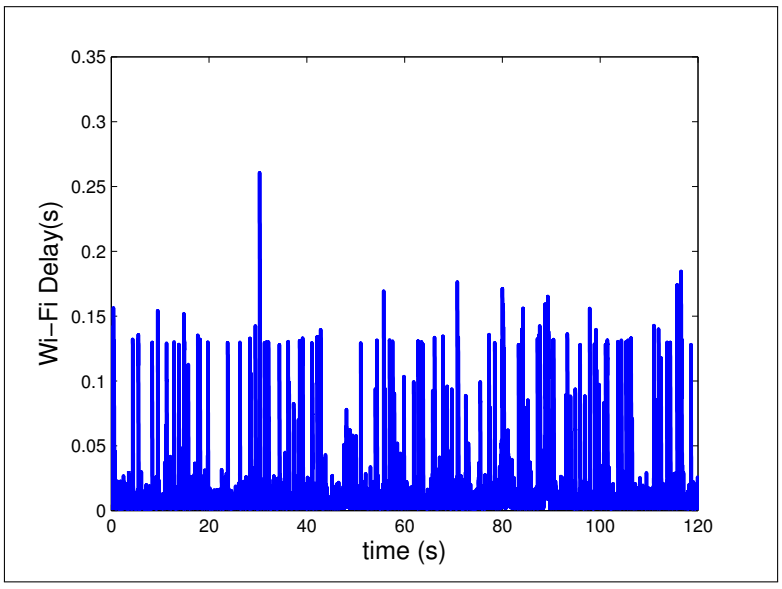
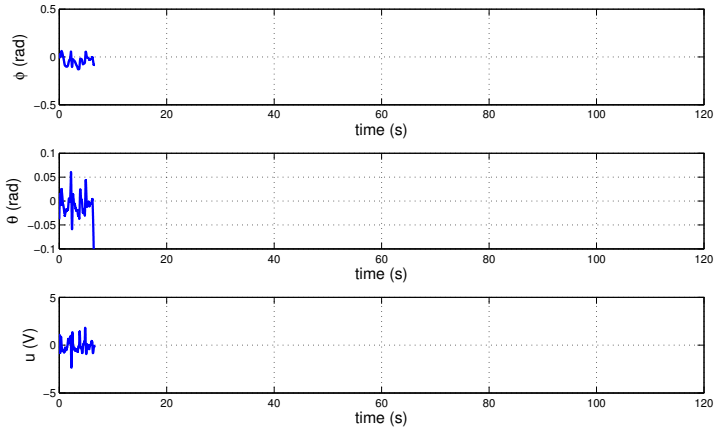
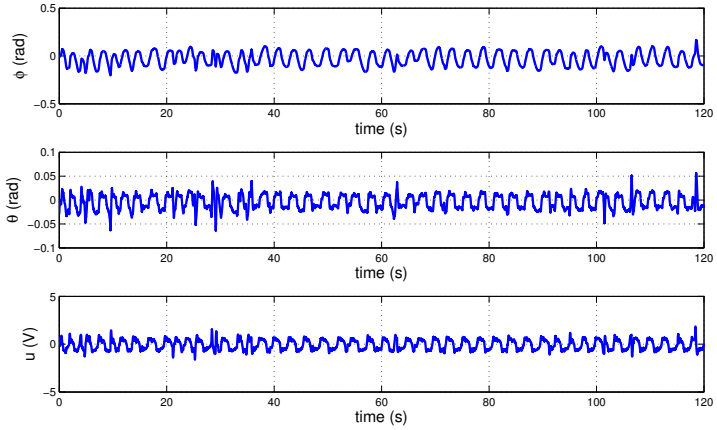


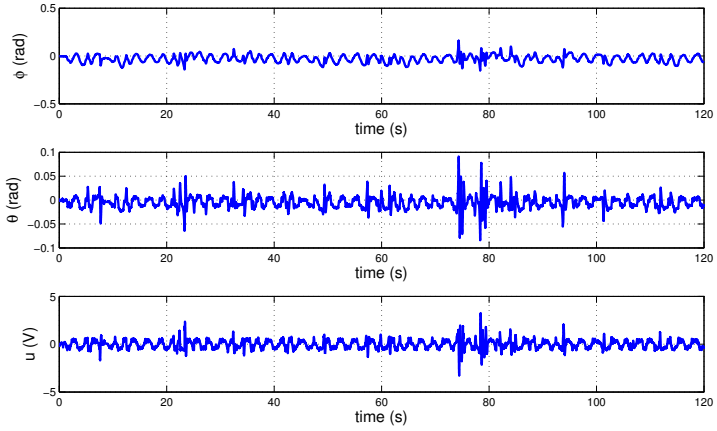
Figure 7: Time-varying delay in Wi-Fi network.



(a) Without the state predictor



(b) With the state predictor using the constant delay



(c) With the state predictor using the time-varying delay

Figure 8: Response of rotary inverted pendulum (ϕ is the rotary arm position, θ is the pendulum position, and u is the input voltage)

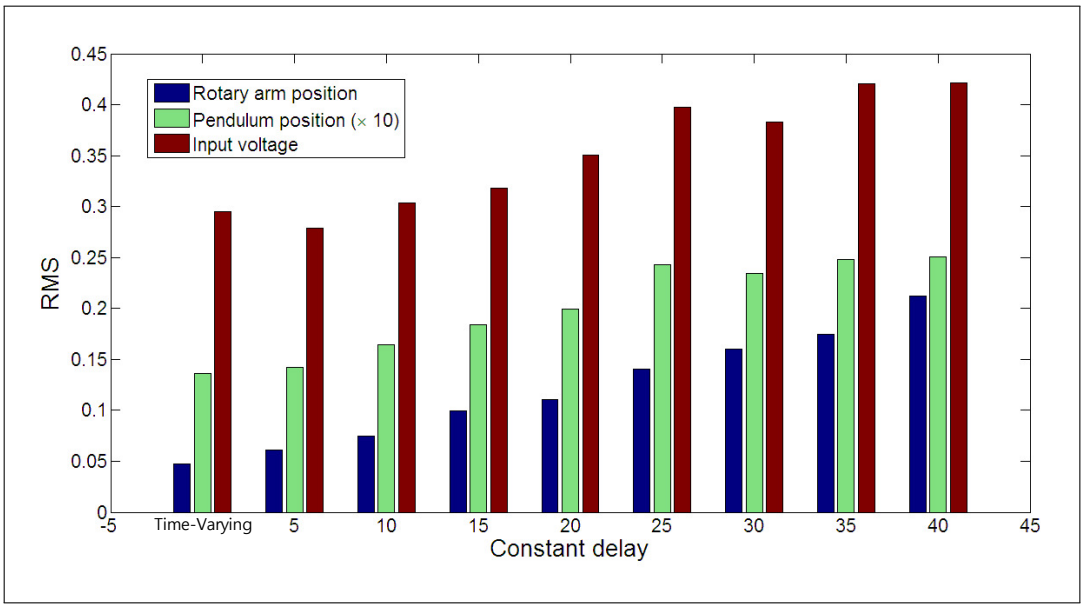


Figure 9: RMS values of state and input variables.

5.2 Discussion of experiment results

In our experiments, we consider three cases. The first case is the WNCS only with the state feedback (12) and the observer (13) without the state predictor (14). The experimental result of the first case is shown in Fig. 8(a), which demonstrates that the pendulum cannot be controlled because it falls down at around $6s$. This is because the time-varying delay is not compensated and destroys the stability.

In order to verify the stability and performance of the designed controller for WNCSs over Wi-Fi networks, we conduct several extensive experiments on the hardware platform constructed in Chapter 4. A detailed configuration of the experimental set-up is depicted in the block diagram of Fig. 6.

Before presenting the experiment results for the controller performance, we discuss the true extent of time-varying delays that occurred in the WNCS by measuring the real delays in Wi-Fi networks for a $120s$ period with a $2ms$ sampling rate. The measurement delays are shown in Fig. 7 and it is obvious that the delays in Wi-Fi networks fluctuate extensively and are relatively large, and they must be compensated to realize stable operation of the WNCS. The maximum, average, and minimum delays in Fig. 7 are $260.5ms$, $14.1ms$, and $1.3ms$, respectively.

The second case is the WNCS with the state feedback (12), the observer (13), and the state predictor (14), but the time-varying delay is approximated to a constant delay such that $d(m)$ is selected as a constant in (14). That is, the state predictor (14) with a constant $d(m)$ is applied to the WNCS with time-varying delays, and Fig. 8(b) shows the experimental result when $d(m) = 10$, which demonstrates that the pendulum is adequately controlled, and is kept in the upright position with small oscillations of the state variables, with only a constant-delay compensation. However, we observe that when we conduct the same experiments with various constant-delay values, only the delay within the range of $5 \leq d(m) \leq 41$ can stabilize and control the pendulum kept in the upright position.

The third case is the WNCS with the state feedback (12), the observer (13), and the state predictor (14) using the measured time-varying delay, $d(m)$. Fig. 8(c) shows the experimental result for the third case, which demonstrates that the pendulum is adequately controlled, and is kept in the upright position with small oscillations of the state variables, as expected.

From Fig. 8(b) and 8(c), we observe that the amplitudes in Fig. 8(b) are larger than those in Fig. 8(c) for both state and input variables, which demonstrates that the time-varying delay compensation exhibits better performance than the constant-delay compensation. In order to highlight the better performance quantitatively, in Fig. 9, we display the root mean square (RMS) values of state and input variables for both the time-varying delay compensation and several constant-delay compensations that are computed for the experiment interval ranging from $0s$ to $120s$. Considering the performance in terms of both the state and input variables, Fig. 9 demonstrates that the time-varying delay compensation achieves the best performance.

6 Conclusion

We designed a state predictor-based output feedback controller for NCSs over Wi-Fi networks, and we constructed the WNCS hardware platform for many different experiments. In order to design a feedback controller to compensate for time-varying network delays, we proposed a three-step design process. We designed a state feedback controller based on the LQR theory, the observer estimates the full-state of the system, and the predictor predicts the future state for the plant input. We acquired the time-varying network delays in real time by measuring the time difference between sending and received control data packets. We used the measured delays for the designed controller in real time, which allows the controller to precisely compensate time-varying delays. A distinct feature compared to other studies involving WNSCs is that we verified the stability and performance of WNCSs even by considering the time-varying delays in Wi-Fi networks and conducting real experiments on the WNCS hardware platform.

The designed controller can be applied to a specific class of WNCSs over Wi-Fi networks, where only the controller output is connected to the plant input via Wi-Fi networks. This limitation provides the motivation for carrying out a challenging future study, in which we aim to extend the results obtained in this paper to realize the control of more general WNCSs over Wi-Fi networks, where the plant input and output are connected to the controller output and input, respectively, through Wi-Fi networks. In this case, the network delays between the plant output and the controller input can be modeled as the time-varying output delays of the plant, which would require a new observer design scheme that estimates system states from the time-varying delayed output.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2015R1D1A1A01056843).

Bibliography

[1] Bai J. et al. (2015); Modeling and H_∞ Control of Wireless Networked Control System with Both Delay and Packet Loss, *Journal of The Franklin Institute*, ISSN 0016-0032, 352, 3915-3928, 2015.

[2] Choi J.Y., Krstic M. (2015); Compensation of Time-varying Input Delay for Discrete-time Nonlinear Systems, *International Journal of Robust and Nonlinear Control*, DOI: 10.1002/rnc.3382, 1-22, 2015.

[3] Ding S.X. et al. (2013); An Integrated Design Framework of Fault-Tolerant Wireless Networked Control Systems for Industrial Automatic Control Applications, *Industrial Informatics, IEEE Transactions on*, ISSN 1551-3203, 9(1), 462-471, 2013.

[4] Drew M. et al. (2005); Networked Control System Design over a Wireless LAN, *Proceedings of the 44th IEEE Conference on Decision and Control*, ISBN 0-7803-9567-0, 6704-6709, 2015.

[5] Du D. et al. (2015); Multiple Event-triggered H_2 / H_∞ Filtering for Hybrid Wired-wireless Networked Systems with Random Network-induced Delays, *Information Sciences*, ISSN 0020-0255, 325, 393-408, 2015.

[6] Du Z., Lin T.C., Zaho T. (2015); Fuzzy Robust Tracking Control for Uncertain Nonlinear Time-Delay System, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 10(6), 812-824, 2015.

[7] Ferrari P., Flammini A., Sisinni E. (2013); Improving Simulation of Wireless Networked Control Systems Based on WirelessHART, *Computer Standards & Interfaces*, ISSN 0920-5489, 35(6), 605-615, 2013.

[8] Girolamo G.D.D., D'Innocenzo A., Benedetto M.D.D. (2015); Co-design of Controller and Routing Redundancy over a Wireless Network, *5th IFAC Workshop on Distributed Estimation and Control in Networked Systems NecSys 2015*, ISSN 2405-8963, 48(22), 100-105, 2015.

[9] Halawa H.H. et al. (2013); Performance Optimization for Reliable Wireless Networked Control Systems in the Presence of Interference, *Emerging Technologies and Factory Automation (ETFA), 2013 IEEE 18th Conference on*, ISSN : 1946-0740, 1-4, 2013.

[10] Hespanha J.P., Naghshtabrizi P., Xu Y. (2007); A Survey of Recent Results in Networked Control Systems, *Proceedings of the IEEE*, ISSN 0018-9219, 95(1), 138-162, 2007.

[11] Hong Z., Gao J.F., Wang N. (2014); Output-Feedback Controller Design of a Wireless Networked Control System with Packet Loss and Time Delay, *Mathematical Problems in Engineering*, ISSN 1563-5147, 1-7, 2014.

[12] Kilinc D., Ozger M., Akan O.B. (2015), On the Maximum Coverage Area of Wireless Networked Control Systems With Maximum Cost-Efficiency Under Convergence Constraint, *Automatic Control, IEEE Transactions on*, ISSN 0018-9286, 60(7), 1910-1914, 2015.

[13] Kuzu A., Bogosyan S., Gokasan M. (2016); Predictive Input Delay Compensation with Grey Predictor for Networked Control System, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 11(1), 67-76, 2016.

[14] Leng Q. et al. (2014); Improving Control Performance by Minimizing Jitter in RT-WiFi Networks, *Real-Time Systems Symposium (RTSS), 2014 IEEE*, ISSN 1052-8725, 63-73, 2014.

[15] Li H., Li X., Wang J., Ma H. (2009); Design of Networked Control System Based on Wireless Sensor Networks, *Information and Automation, 2009, ICIA '09. International Conference on*, ISBN 978-1-4244-3607-1, 1140-1143, 2009.

[16] Li L.L., Lemmon M. (2014); Weakly Coupled Event Triggered Output Feedback System in Wireless Networked Control Systems, *Discrete Event Dynamic Systems*, ISSN 0924-6703, 24(2), 247-260, 2014.

[17] Ling R. et al. (2016); Energy-efficient H_∞ Filtering over Wireless Networked Systems-A Markovian system approach, *Signal Processing*, ISSN 0165-1684, 120, 495-502, 2016.

[18] Mahmoud M.S., Sabih M., Elshafei M. (2015); Event-triggered Output Feedback Control for Distributed Networked Systems, *ISA Transactions*, ISSN 0019-0578, 18, 1-9, 2015.

[19] Moghadam H.F., Vasegh N. (2014); Robust PID Stabilization of Linear Neutral Time-Delay Systems, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 9(2), 201-208, 2014.

[20] Quang N.V.A., Yoo M. (2014); Performance Analysis of Packet Loss on Wireless Network Control Systems, *Information and Communication Technology Convergence (ICTC), 2014 International Conference on*, ISBN 978-1-4799-6786-5, 185-186, 2014.

[21] Shi A., Liu B. (2015); Robust H_∞ Control of Wireless NCS with Delay and Packet Dropouts, *Control and Decision Conference (CCDC), 2015 27th Chinese*, ISBN 978-1-4799-7016-2, 6022-6025, 2015.

[22] Tong S., Qian D., Liu G.P. (2014); Networked Predictive Fuzzy Control of Systems with Forward Channel Delays Based on a Linear Model Predictor, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 9(4), 471-481, 2014.

[23] Wang Y. et al. (2014); An H_∞ Fault Estimation Scheme of Wireless Networked Control Systems for Industrial Real-Time Applications, *Control Systems Technology, IEEE Transactions on*, ISSN 1063-6536, 22(6), 2073-2086, 2014.

[24] William S. Levine (1996); *The Control Handbook*, 2nd ed., CRC Press, ISBN : 978-0-8493-8570-4, 1996.

[25] Yi H.C. et al (2012); Design of Networked Control System using RTT Measurement over WSN, *Wireless Information Technology and Systems (ICWITS), 2012 IEEE International Conference on*, ISBN 978-1-4673-0947-9, 1-4, 2012.

[26] Zhang W. et al. (2014); Time Delay Compensation in a Wireless Tracking Control System with Previewed Reference, *2014 American Control Conference (ACC)*, ISSN 0743-1619, 3293-3298, 2014.

[27] Zhang W. et al. (2015); Robust Time Delay Compensation in a Wireless Motion Control System with Double Disturbance Observers, *2015 American Control Conference(ACC)*, ISBN 978-1-4799-8685-9, 5294-5299, 2015.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 12(3), 429-441, June 2017.

A New Adaptive Elastic Net Method for Cluster Analysis

J. Yi, P. Zhao, L. Zhang, G. Yang

Junyan Yi
Beijing Key Laboratory of Robot Bionics and Function Research,
Department of Computer Science and Technology
Beijing University of Civil Engineering and Architecture
1 Zhanlanguan Road
Beijing, 100044, China
yijunyan@bucea.edu.cn

Peixin Zhao*
School of Management,
Shandong University, Jinan, China
27 Shanda Nan Road
Jinan, 250100, China
*Corresponding author: pxzhao@sdu.edu.cn

Lei Zhang
Department of Computer Science and Technology
Beijing University of Civil Engineering and Architecture
1 Zhanlanguan Road
Beijing, 100044, China
lei.zhang@bucea.edu.cn

Gang Yang
School of Information
Renmin University of China
59 Zhongguancun Street
Beijing, 100872, China
yanggang@ruc.edu.cn

Abstract: Clustering is inherently a highly challenging research problem. The elastic net algorithm is designed to solve the traveling salesman problem initially, now is verified to be an efficient tool for data clustering in n-dimensional space. In this paper, by introducing a nearest neighbor learning method and a local search preferred strategy, we proposed a new Self-Organizing NN approach, called the Adaptive Clustering Elastic Net (ACEN) to solve the cluster analysis problems. ACEN consists of the adaptive clustering elastic net phase and a local search preferred phase. The first phase is used to find a cyclic permutation of the points as to minimize the total distances of the adjacent points, and adopts the Euclidean distance as the criteria to assign each point. The local search preferred phase aims to minimize the total dissimilarity within each clusters. Simulations were made on a large number of homogeneous and nonhomogeneous artificial clusters in n dimensions and a set of publicly standard problems available from UCI. Simulation results show that compared with classical partitional clustering methods, ACEN can provide better clustering solutions and do more efficiently.
Keywords: self-organizing neural network, elastic net, adaptive, cluster analysis.

1 Introduction

Cluster analysis is useful in exploratory data analysis. It is an important but challenging task in unsupervised learning. The essence of the task is to partition a set of objects into a number of

clusters, minimizing the within-cluster variability and maximizing the between-cluster variability [10]. Data clustering is a common technique for statistical data analysis. It can be widely used in a variety of scientific and engineering disciplines such as biology [2, 19, 22], computer vision [6, 8, 9, 17] and optimizations [18, 24]. Many clustering algorithms have been proposed. Generally they can be divided into the following categories: hierarchical clustering method, partitional clustering method, density based clustering method, grid based clustering method, and neural network based clustering method and so on [23].

In the last 20 years also, neural networks have been shown to be powerful tools for solving clustering problems. At present, neural networks-based clustering has been dominated by SOFM (Self-Organization Feature Map) and ART (Adaptive Resonance Theory). SOFM is a powerful tool for data analysis, and used widely in data mining, pattern recognition, clustering analysis [1, 11, 12, 16]. But when applied in real practice a number of user-dependent parameters cause problems. Like the K-means algorithm, SOMF need to predefine the size of the lattice, i.e., the number of the clusters, which is unknown for most circumstances. In addition, trained SOMF may be suffering from input space density misrepresentation, where areas of low pattern density may be over-represented, and areas of high density under-represented. Some researchers have proposed modifications of SOMF algorithm [4, 14].

In this work we propose a new Self-Organizing NN clustering method based on the elastic net, and attempt to alleviate the problems mentioned above. Elastic net is evolved from SOFM, proposed by R. Durbin and D. Willshaw [5] as a heuristic method. Like SOFM, the elastic net tries to find a topology preserving map between two spaces. It was originally proposed to solve the Traveling Salesman Problem (TSP) [20]. Later it was given a mechanical statistic foundation [15]. Then elastic net can be used as a tool for data clustering in n-dimensional space and for diverse of problems in pattern research [3, 7, 13, 21].

The advantages of the elastic net method for cluster analysis are as follows: firstly, elastic net is an unsupervised learning network, and the manual training is not required. Secondly, elastic net does not need the prior knowledge about the number of clusters, and its effectiveness is not effected by the input sequence of data and isolated data. Thirdly, elastic net's geometry corresponds well to the problem definition, which makes it suitable for solving high-dimensional clustering problems. And its convergence is governed by well-established physics theories leading to a sensible solution. Lastly, when solving the clustering problems, the number of neurons needed in the elastic net is linearly related to the number of spatial data points. That is, the complexity of the elastic network algorithm is $O(N)$, where N is the number of spatial data points [20]. The performance of the proposed method is evaluated by simulations on a number of homogeneous and nonhomogeneous synthetic clustering problems in n dimensions and some sets of publicly standard test problems available from UCI (accessible via the web at http://archive.ics.uci.edu/ml/datasets.html). Simulation results show that the proposed method can consistently and efficiently provide better clustering solutions compared with the classical partitional clustering algorithms.

2 The new adaptive clustering elastic net method

In the section, a new Self-Organizing NN approach for clustering ACEN is presented. ACEN consists of two phases: the adaptive clustering elastic net phase and local search preferred phase. Let $x_i(i = 1, 2, ..., n)$ be a data set X with n objects; kŁş the number of clusters; m_jŁş the centroid of cluster $C_j(j = 1, 2, ..., K)$. Then ACEN tries to minimize the cost function - Sum of

Euclidean Distance (SED):

$$SED = \sum_{i=1}^{n} \sum_{j=1,\ x_i \in C_j}^{k} d(x_i, m_j) \tag{1}$$

The details of two phases are described below.

2.1 The adaptive clustering elastic net phase

Given m unit coordinates $Y = \{y_1, y_2, ..., y_m\}$ defining the rubber band of the elastic net, n data points $X = \{x_1, x_2, ..., x_n\}$ defining the clustering data set. The new energy function of the adaptive clustering elastic net is defined as below:

$$E = -\alpha(t)K \sum_{i=1}^{n} ln \sum_{j=1}^{m} e^{-|x_i - y_j|/2K^2} + \beta(t) \sum_{j=1}^{m} |y_j - y_{j+1}| \tag{2}$$

The positions of the points defining the rubber band are updated according to the following formula:

$$\Delta y_j = \alpha(t) \sum_{i=1}^{n} w_{ij}(x_i - y_j) + \beta(t)K(y_{j+1} - 2y_j + y_{j-1}) \tag{3}$$

$$\begin{cases} \alpha(t+1) = (1 - \Psi)\alpha(t) & if\ \alpha(t) > \alpha_{min}, \\ \alpha(t+1) = \alpha_{min} & otherwise. \end{cases} \tag{4}$$

$$\begin{cases} \beta(t+1) = (1 + \vartheta)\beta(t) & if\ \beta(t) < \beta_{max}, \\ \beta(t+1) = \beta_{max} & otherwise. \end{cases} \tag{5}$$

where
Ψ = damping factor of $\alpha(t)$ $(0 \leq \Psi < 1)$.
ϑ = promoting factor of $\beta(t)$ $(0 \leq \vartheta < 1)$.

Here w_{ij} is the same as the original elastic net. The parameters Ψ and ϑ in Eqn.(4) and (5) are small positive parameters, which are selected empirically.

This phase helps the elastic net to find a cyclic permutation of the points so as to minimize the total distance of adjacent points in shorter time. After the computation of the adaptive clustering elastic net phase, each element of the data set is matched with the points of the elastic band. Therefore the permutation of data points can be obtained from the sequence of the elastic net points. Then the distances of adjacent elements of the data set can be used to divide the clusters.

Each point dotted in Fig.1 represents the distances between the adjacent elements of the permutation. The value of the x-axis is the sequence of elements of the data set. The value of the y-axis is the distance between the adjacent elements of the data set. Therefore, each pick in Fig.1 represents a larger distance between the adjacent elements, meaning a cluster change.

For large data sets, some elements may be lost in the final permutation. These elements are regarded as noise elements in our algorithm. We proposed the nearest neighbor learning method to handle the noise elements. The learning method is described as below. For each noise element, find its nearest data x_i in all of the matched elements of the data set, and calculate the distances between the noise element and the two neighbors x_{i-1} and x_{i+1} in the path. Compare the two distances and insert the noise element between x_i and the nearer neighbor element x_{i-1} or x_{i+1}. Then update the sequence of the data set.

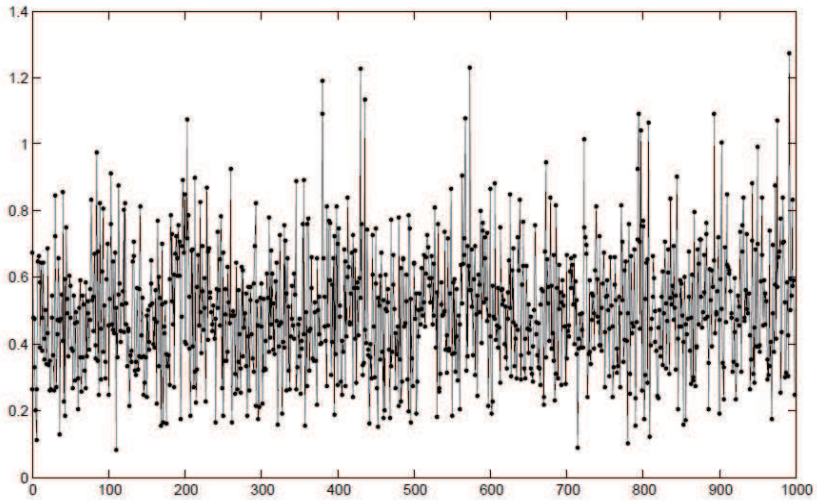


Figure 1: The distances between the adjacent elements of the data set.

2.2 The local search preferred phase

In the first stage, we could achieve initial clusters according to the distances between the adjacent points. For most of the clustering problems, the first phase could efficiently identify high quality solutions. According to the requirements of different practical applications, the clustering solutions sometimes still need to be further optimized. Then the local search preferred phase is needed to help the network to search for the better solutions. Mathematically the steps of the second phase can be stated as below.

Step 1. Select the number of the nearest neighbors p and get the centroids m_j of each cluster C_j in the first phase.

Step 2. Assign each element in the data set X to the clusters C_j with the closest centroid m_j using the Euclidean distance as the criteria.

Step 3. Update k medoids. For $j = 1$ to the number of clusters k do

(a) Find a subset C_{subset} in the cluster C_j. C_{subset} corresponds to m_j and its p nearest neighbors that have not been evaluated before current iteration.

(b) Calculate the new medoid

$$q = \arg \min_{x_i \in C_{subset}} \sum_{x_i \in C_j} d(x_k, x_i) \tag{6}$$

after that the old medoid m_j is replaced by q if it is different from m_j.

(c) Repeat steps (a) and (b) until the medoid does not change any more.

Step 4. Repeat steps 2 and 3 until the k medoid does not change any more.

The local search preferred phase uses the Euclidean distance as the criteria to assign each point like many other partitional clustering algorithms. The algorithm chooses the objects in the data set as the medoids of the clusters. The selection of the initial centroids in the first phase helps the algorithm to overcome the disadvantage of the instability of the other partitional algorithms, and could achieve better solutions.

3 Algorithm

For the clustering problem, the proposed algorithm ACEN is presented as below:

Step 1. Given the initial number m of dynamic nodes $Y = \{y_1, y_2, ..., y_m\}$ and establish the nodes in a small circle with the center of the circle at the centroid of the clustered points. $m = 2.5n$.

Step 2. Decide the sequence of the nodes.

Step 3. For each data points $x_i(i = 1, 2, ..., n)$, choose a node and move the node and its neighbors on the circle towards the points.

Step 4. Repeat until all the points $x_i(i = 1, 2, ..., n)$ have been matched with the nodes.

Step 5. Get the sequence of the $x_i(i = 1, 2, ..., n)$ according to the sequence of the nodes.

Step 6. Use the nearest neighbor learning method to update the sequence of data set.

Step 7. $d_i(i = 1, 2, ..., n)$ (the distance of the adjacent data points) is used to divide the groups $C_j(j = 1, 2, ..., k)$, where larger distances d_i mean cluster changes.

Step 8. Assign each point in the data set X to the clusters C_j with the closest centroid m_j under Euclidean distance metric.

Step 9. Update k medoids. For $j = 1$ to the number of clusters k do

(a) Find a subset C_{subset} in the cluster C_j. C_{subset} corresponds to m_j and its p nearest neighbors that have not been evaluated before current iteration.

(b) Calculate the new medoid

$$q = \arg \min_{x_i \in C_{subset}} \sum_{x_i \in C_j} d(x_k, x_i) \tag{7}$$

after that the old medoid m_j is replaced by q if it is different from m_j.

(c) Repeat steps (a) and (b) until the medoid does not change any more.

Step 10. Repeat steps 8 and 9 until k medoid does not change any more.

Our algorithm improves the original elastic net method by introducing the adaptive clustering parameter strategy and the nearest neighbor learning method. Combining the new elastic net method with the local search preferred strategy, the proposed algorithm could obtain better solutions for cluster analysis problems.

The advantages of ACEN mainly reflected in the following aspects:

- ACEN does not need the prior knowledge about the number of clusters.
- Algorithm's effectiveness is not effected by the input sequence of data and isolated data.
- Given n values of the number of clusters k, ACEN could obtain the different solutions for different numbers of clusters just running the network one time, which effectively improve the computation speed. On the contrary, given n values of the number of clusters k, the traditional partitional clustering algorithms such as K-means and K-medoids need to run n times.
- Due to the geometrical and physical characteristics of the elastic net, ACEN could express and calculate high-dimensional data set simply and efficiently, and display the data intuitively.
- By introducing the adaptive parameter strategy and the nearest neighbor learning method, combined with the local search preferred method, ACEN could consistently and efficiently identify high quality clustering solutions.
- The algorithm is geometric in nature, and the evolution of the algorithm can be tracked visually.

It has been pointed out that rearrangement clustering is equivalent to the TSP. And it can be solved to optimality by solving the TSP [3]. But there is a serious defect in previous approaches when applied to data which falls into natural clusters as shown in Fig.2. Objects in Fig.2 only have two features: horizontal and vertical coordinates. When these objects are clustered by the method of rearrangement clustering, although x and y are very close the objects x and y will be separated by 16 objects in two different clusters. It is clear that generally clusters may be broken into pieces so as to minimize the dissimilarity to adjacent clusters. Using the ACEN method these problems could be solved effectively. In ACEN the partition is taken based on the larger distances between the adjacent elements in the permutation, which is more suitable for the definition of the natural cluster problem, which could help the algorithm to achieve high quality clustering solutions.

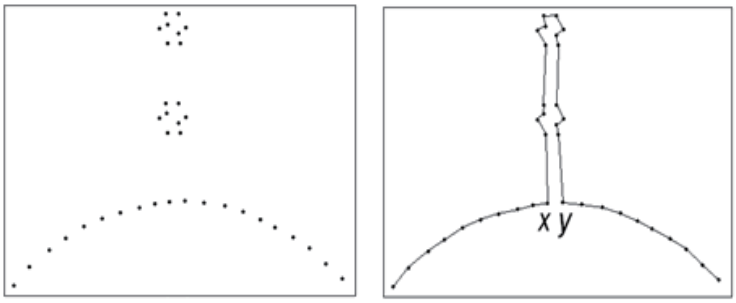


Figure 2: (a) Three clusters; (b) The TSP path specifying the optimal rearrangement. Although x and y are very close, their placement in the rearrangement is far apart.

4 Simulation results

Extensive simulations were implemented on a large number of homogeneous and nonhomogeneous synthetic clusters in n dimensions. And the value of SED shown in Eqn. (1) was adopted to judge the quality of the different methods. Parameters were set as follows: $\alpha(0) = 0.6$; $\beta(0) = 1.5$; $\Psi = 0.001$; $\vartheta = 0.002$; $k = 0.5$.

We tested the ACEN on some nonhomogeneous clusters with different density regions. First we compared the ACEN with K-means and K-medoids methods to solve the problem shown in Fig.2. These data falls into natural clusters. ACEN identified high quality clustering solution for this simple problem. On the contrary the other two partitional clustering methods did not cluster the data well as predicted. The typical results produced by three methods were shown in Fig.3. In the following figures, the different shapes and colors of points represent different clusters.

The second of the comparative computational tests we report used the nonhomogeneous cluster points in 2D and 3D respectively. The problems both involved 200 points. To compare the performance of the three methods, we recorded the average SED values obtained from 20 simulations with different initial cluster centroids for k-means and k-medoids. Because the results of ACEN were not effected by the initial cluster centroids, we only took one trial for ACEN. The comparisons of ACEN with K-means and K-medoids were shown in Fig.4 and Fig.5.

From Fig.4 and Fig.5 we can see clearly:

- ACEN could identify high-quality solutions consistently for all of the tested problems. The

other two partitional clustering methods both get poor solutions.

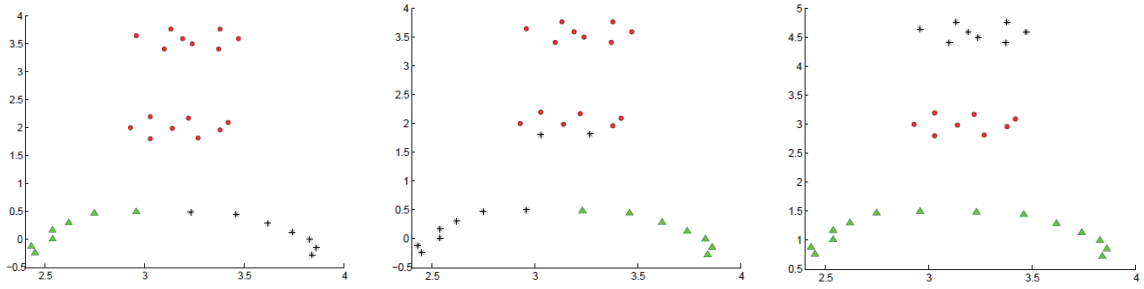
- The values of SED produced by ACEN were 29.58% and 41.74% less than the average values of SED produced by K-means over 20 trials, and 28.18% and 45.54% less than the average values of SED produced by K-medoids over 20 trials respectively for the tested 2D and 3D clustering problems.

The proposed method consists of two phases: the adaptive clustering elastic net phase and the local search preferred phase. The first phase can identify clusters according to the distance between the adjacent points. The aim of the second phase is to minimize the cost function - Sum of Euclidean Distance (SED) in Eqn.(1). For most of the cases, solutions produced by the first phase are good enough, then the second phase is not necessary.

In the following, we clustered 500 points into 4 clusters in 2D using the proposed algorithm. The result of the first phase and the optimized result produced by the second phase were shown in Fig.6. From Fig.6 we can see clearly that the solution produced by the first phase was the same as the optimized result produced by the second phase, and ACEN could achieve high quality solution for the clustering problem without using the local search preferred phase.

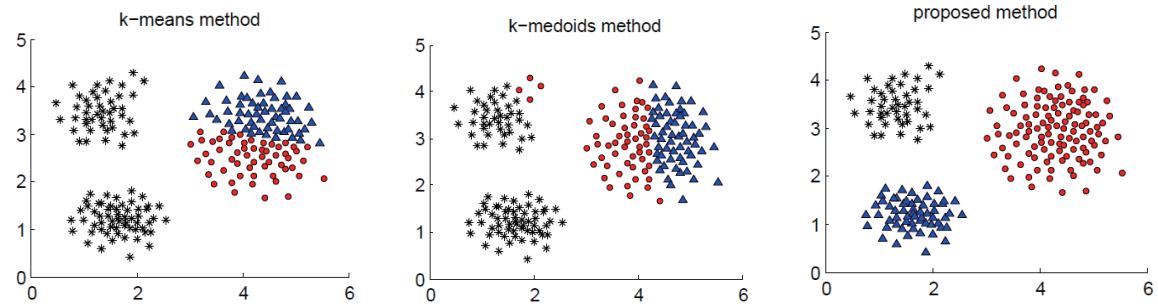
To meet the requirements of different practical applications, the local search preferred phase is used to improve the solution quality. We tested the proposed method on a homogeneous artificial clustering problem involve 150 data points in 2D. Fig.7 shows the evolutions of clustering process using the proposed approach for this homogeneous artificial problem. Fig.7(a) shows the result of the adaptive clustering elastic net phase. Fig.7(b) and Fig.7(c) display the intermediate processes of clustering. Fig.7(d) shows the final solution produced by the local search preferred phase of the ACEN for the clustering problem. From Fig.7 we can see clearly that the proposed algorithm could identify high quality solutions of giving clustering problem, and the decrease of the value of SED showed that the local search preferred phase could optimize the result effectively.

In the third of our experimental tests we selected a large number of instances of 50, 100, 300, 500 and 1000 homogeneous and nonhomogeneous simulated cluster points in 2D, 3D, 4D, 7D 10D and 13D to show the solution qualities of ACEN, K-means and K-medoids. In the following experiments, to compare performance of the three methods, we recorded the average SED values obtained from 20 simulations with different initial cluster centroids for k-means and k-medoids. As described before, ACEN was performed only once. In the proposed method we use another set of parameter values as follows: $\alpha(0) = 0.6$; $\beta(0) = 1.5$; $\Psi = 0.0025$; $\vartheta = 0.0025$. Comparisons of the proposed method with the K-means and K-medoids methods were shown in Table 1 and Table 2.



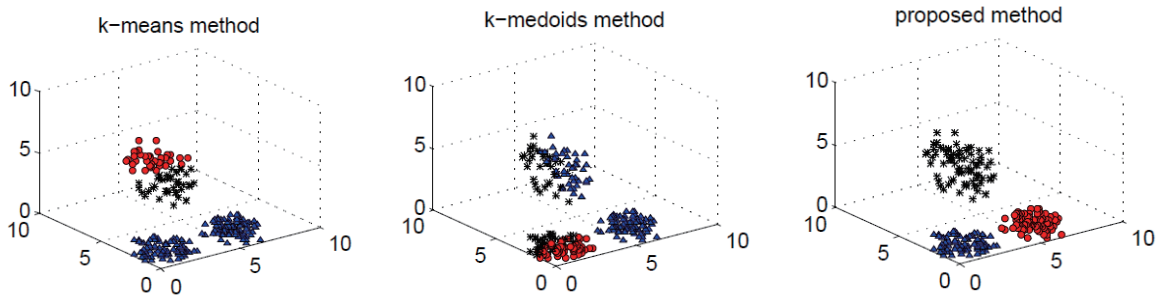
(a) Typical clustering result of K-means (b) Typical clustering result of K-medoids (c) Clustering result of the ACEN

Figure 3: The typical clustering results of three methods for the natural cluster problem described in Fig. 2: (a) K-means method; (b) K-medoids method; (c) The proposed method.



(a) SED = 189.92 (b) SED = 186.21 (c) SED = 133.74

Figure 4: The typical solutions and the average values of SED obtained from 20 simulations produced by three methods for the nonhomogeneous data in 2D: (a) K-means method; (b) K-medoids method; (c) The proposed method.



(a) SED=525.56 (b) SED=562.17 (c) SED=306.15

Figure 5: The typical solutions and the average values of SED obtained from 20 simulations produced by three methods for the nonhomogeneous data in 3D: (a) K-means method; (b) K-medoids method; (c) The proposed method.

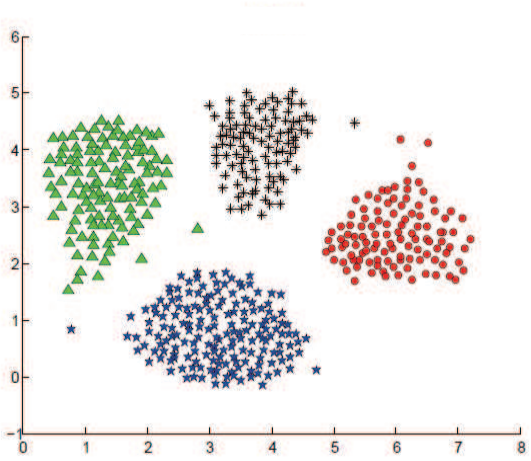
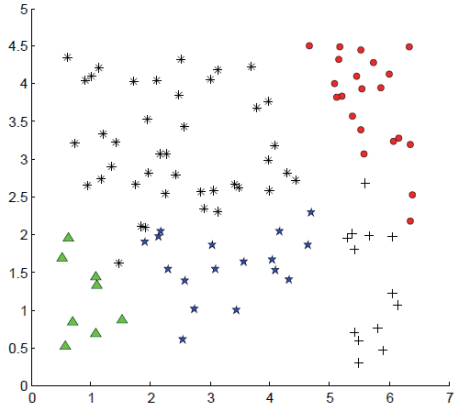
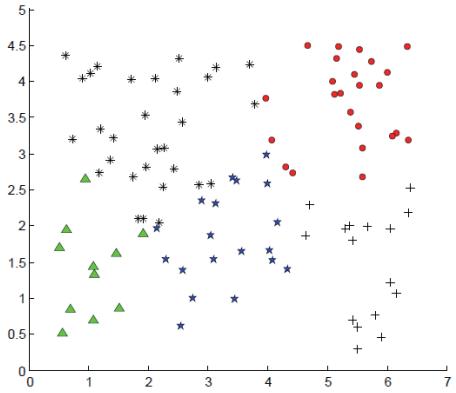


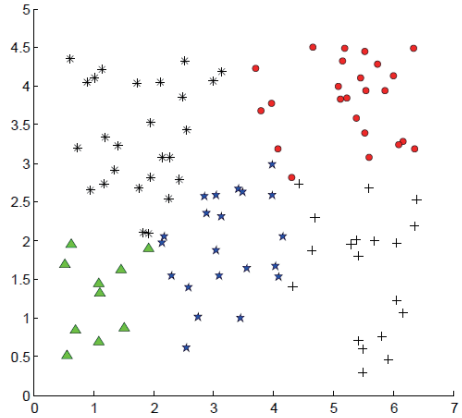
Figure 6: The result produced by the first phase and the optimized result produced by the second phase of ACEN. The two results were the same.



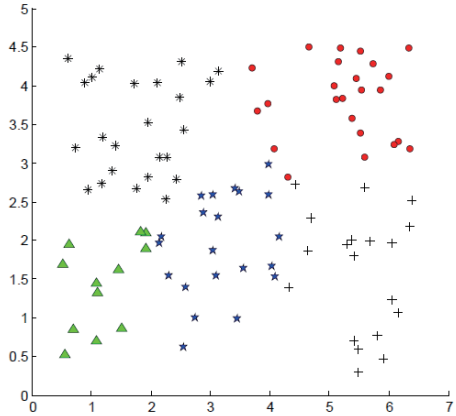
(a) The result of the adaptive clustering elastic net phase: SED=88.16



(b) The intermediate processes of clustering: SED=84.46



(c) The intermediate processes of clustering: SED=82.98



(d) The final result of the local search preferred phase: SED=82.86

Figure 7: Evolutions of ACEN and the values of SED of the clustering process.

Table 1: The average values of SED over 20 trials produced by K-means and K-medoids and the value of SED produced by ACEN on sets of 50, 100, 300, 500 and 1000 synthetic data in 2D, 3D and 4D.

Data	Number	Dimension 2			Dimension 3			Dimension 4		
	of	Kmeans	Kmedoids	ACEN	Kmeans	Kmedoids	ACEN	Kmeans	Kmedoids	ACEN
Size	Clusters	SED	SED	SED	SED	SED	SED	SED	SED	SED
50	3	32.33	29.8	20.85	46.82	42.19	29.66	44.28	44.27	36.83
	5	31.87	29.37	15.59	26.17	25.72	19.47	37.48	38.04	29
	10	19.35	16.85	7.29	23.16	21.4	11.39	24.54	24.7	17.4
100	3	76.85	73.98	45.21	75.41	74.14	67.7	109.27	109.38	109.27
	5	68.76	64.38	39.71	104.37	96.42	83.05	69.8	68.77	54.09
	10	37.72	31.45	20.88	67.02	55.91	35.95	62.36	59.54	44.91
300	3	316.14	288.72	156.74	252.86	253.44	252.32	381.8	351.98	247.73
	5	186.23	178.6	99.63	249.96	234.22	191.74	271.27	269.27	233.84
	10	114.38	109.69	80.17	146	14.25	127.76	184.75	183.44	173.1
500	3	645.76	505.98	221.87	564.6	595.15	388.7	654.73	698.28	448.91
	5	473.01	346.35	208.93	433.16	410.04	336.76	474.99	488.12	426.57
	10	263.33	225.86	158.97	301.64	288.96	261.16	346.28	365.4	309.63
1000	3	739.43	714.95	430.77	984.59	809.36	797.51	1216.3	1331.27	916.36
	5	639.8	616.95	396.37	781.41	724.79	693.75	969.57	926.82	869.2
	10	419.75	405.18	345.37	646.31	655.78	628.33	806.21	811.8	692.55

An examination of Table 1 and Table 2 yields the following observations:

- For all of the 30 homogeneous and nonhomogeneous clustering challenges, ACEN could obtain better clustering solutions than K-means and K-medoids.
- The values of SED calculated by the proposed method around 50, 100, 300, 500 and 1000 points for different dimensions were 35.03%, 31.2%, 30.25%, 30.19% and 29.53% less than the average values of SED calculated by the K-means over 20 trials; and 33.89%, 29.33%, 27.34%, 27.23% and 25.87% less than the average values of SED calculated by the K-medoids over 20 trials respectively.
- Like many partitional clustering algorithms, the results of K-means and K-medoids depended on the selection of the initial cluster centroids. On the contrary, the results of ACEN were stable and were not influenced by the selection of initial centroids.
- ACEN's effectiveness was not effected by the input sequence of data and isolated data, and could produce stable clustering solutions .

For the last of our experiments we took some sets of standard test problems available from UCI (accessible via the web at http://archive.ics.uci.edu/ml/datasets.html). The standard problems we solved include Iris data set and Zoo data set. Comparisons of ACEN with K-means and K-medoids on standard problems were summarized in Table 3. For this comparison, the value of SED and the accuracy were adopted to judge the quality of the clustering solutions. In Table 3, we also performed 20 simulations for K-means and K-medoids, and the average values of SED and accuracy were calculated for each standard problem. Because the results of ACEN were stable and were not effected by the initial centroids, it was performed only once.

The Iris Plants Database is the best known database found in the literature of cluster analysis and referred frequently in cluster analysis texts. The data set contains 3 classes of iris plant with

Table 2: The average values of SED over 20 trials produced by K-means and K-medoids and the value of SED produced by ACEN on sets of 50, 100, 300, 500 and 1000 synthetic data in 7D, 10D and 13D.

Data	Number	Dimension 7			Dimension 10			Dimension 13		
	of	Kmeans	Kmedoids	ACEN	Kmeans	Kmedoids	ACEN	Kmeans	Kmedoids	ACEN
Size	Clusters	SED	SED	SED	SED	SED	SED	SED	SED	SED
50	3	60.17	55.51	41.5	67.22	69.71	55.42	78.22	78.66	61.4
	5	57.21	67.12	37.61	58.7	60.28	45.58	72.84	69.75	52.91
	10	38.55	30.03	27.15	44.34	47.28	29.03	57.13	56.53	30.42
100	3	129.37	110.65	107.62	145.55	138.62	123.93	150.66	137.54	141.61
	5	119.03	118.4	98.99	147.81	133.84	128.27	157.99	135.34	133.59
	10	88	8740	63.89	108.39	108.49	75.16	121.19	116.77	91.71
300	3	468.58	492.8	376.95	594.06	539.17	444.53	707.93	621.61	513.57
	5	386.18	377.22	331.98	498.24	452.06	443.21	504.93	472.72	423.52
	10	335.11	332.33	292.95	420.15	383.47	352.41	421.29	419.52	376.32
500	3	925.75	887.49	726.5	918.19	864.07	759.74	1071.6	1035.17	945.52
	5	691.72	676.57	614.96	865.06	859.41	716.71	862.08	876.33	707.35
	10	618.28	586.41	530.67	774.71	660.53	615.03	834.16	761.27	694.91
1000	3	1603.92	1547.7	1327.82	2082.18	1812.84	1563.24	1945.86	1822.22	1713.09
	5	1374.88	1306.2	1166.27	1638.95	1599.03	1391.72	1981.69	1743.76	1507.95
	10	1289.32	1287.17	956.41	1565.89	1404.95	1135.8	1657.39	1520	1371.93

Table 3: Computational results- standard UCI clustering problems.

Algorithm	Iris		Zoo	
	SED	Accuracy	SED	Accuracy
K-means	125.74	69%	133.63	66%
K-medoids	123.66	71%	127.52	68%
ACEN	98.95	91%	102.38	90%

50 instances each. The Zoo data set contains 7 classes and each animal is made up of 16 properties, 15 of them as boolean and 1 numeric property. In the tests we set the parameter values as described before.

Considering Table 3 we can make the point: ACEN was much more efficient than the other two partitional clustering methods. The accuracies of the ACEN algorithm for standard UCI clustering problems Iris and Zoo were 22% and 24% higher than K-means , and 20% and 22% higher than K-medoids respectively.

Our experiments clearly illustrate the improvement of the clustering solution quality using the ACEN compared with the traditional partitional clustering methods. Moreover the results produced by ACEN are not effected by the input sequence of data and isolated data, which ensure it consistently and efficiently identify high quality solutions for different problems.

5 Conclusions

In this paper, we proposed a new Self-Organizing neural network approach-the Adaptive Clustering Elastic Net (ACEN) for cluster analysis. Our approach has two phases, the adaptive clustering elastic net phase and the local search preferred phase. The first phase is proposed by

introducing an adaptive clustering parameter strategy and the nearest neighbor learning method into the original elastic net. This phase is used to find a cyclic permutation of the points so as to minimize the total distances of the adjacent points, and assign each point to different clusters using the Euclidean distance as the criteria. The local search preferred phase aims to minimize the total dissimilarity within each clusters.

We tested the ACEN method and some classical partitional clustering algorithms on the same data sets include homogeneous and nonhomogeneous synthetic clusters in n dimensions and some sets of publicly standard problems available from UCI. Experiments show that compared with the other traditional partitional clustering algorithms, ACEN could provide better and stable clustering solutions and do more efficiently for all of the tested problems. ACEN algorithm's effectiveness is not effected by the input sequence of data and isolated data. In addition ACEN does not need the prior knowledge about the number of clusters, which makes it more suitable for solving clustering problems.

Acknowledgment

This work is funded by National Natural Science Foundation of China (No. 61402032), Beijing Municipal Natural Science Foundation (No. 4144072), Project Supported by the Beijing Excellent Talent (No. 2013D005017000017), Scientific Research Foundation of Beijing University of Civil Engineering and Architecture (No. 00331613002 and 00331616023, Y13-23), Open Research Fund Program of Beijing Key Laboratory of Robot Bionics and Function Research,the Independent Innovation Foundation of Shandong University, IIFSDU(IFW12109), the Key Project of Philosophy and Social Science Plan of Jinan(2014), Shandong Province Natural Science Foundation(ZR2015GM012),the China Postdoctoral Science Foundation (No.2014T70624),the National Statistical Science Research Programm(2013LZ38) and Beijing Municipal Commission of Education (No. KM201410016007 and 21271413117).

Bibliography

[1] Alahakoon D., Halgamuge S.K. (2000); Dynamic Self-organizing Maps with Controlled Growth for Knowledge Discovery, *IEEE Trans. on Neural Networks*, 11(3), 601-614, 2000.

[2] Baldi P., Hatfield G.W.(2002); *DNA Microarrays and Gene Expression*, Cambridge University Press, 2002.

[3] Climer S., Weixiong Zhang W. (2006); Rearrangement Clustering: Pitfalls, Remedies, and Applications, *Journal of Machiner Learning Research*, 7, 919-943, 2006.

[4] Dittenbach M., Rauber A. (2002); Uncovering Hierarchical Structure in Data Using the Growing Hierarchical Self-organizing Map, *Neurocomputing*, 48, 199-216, 2002.

[5] Durbin R., Willshaw D. (1987); An Analogue Approach to the Traveling Salesman Problem Using an Elastic Net Approach, *Nature*, 326, 689-691, 1987.

[6] Frigui H., Krishnapuram R. (1999); A Robust Competitive Clustering Algorithm with Applications in Computer Vision, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(5), 450-465, 1999.

[7] Gorbunov S., Kisel I. (2006); Elastic Net for Standalone RICH Ring Finding, *Proceedings-published in NIM*, 559, 139-142, 2006.

[8] Jain A.K., Flynn P. (1996); Image Segmentation Using Clustering, *Advances in Image Understanding*, 65-83, 1996.

[9] Jain K. (2010); Data Clustering: 50 Years Beyond K-Means, *Pattern Recognition Letters*, 31(8), 651-666, 2010.

[10] Kantardzic M. (2011); *Data Mining: Concepts, Models, Methods, and Algorithms*, Wiley-IEEE Press, 2011.

[11] Kohonen T. (1982); Self-organized Formation of the Topologically Correct Feature Maps, *Biological Cybernetics*, 43, 59-69, 1982.

[12] Kohonen T. (2001); *Self-Organizing Maps*, 3rd Ed. New York: Springer-Verlag, 2001.

[13] Levano M.; Hans Nowak H. (2011); New Aspects of the Elastic Net Algorithm for Cluster Analysis, *Neural Comput and Applic*, 20, 835-850, 2011.

[14] Raube A., Merkl D. (2002); The Growing Hierarchical Self-Organizing Map: Exploratory Analysis of High-Dimensional Data, *IEEE Transactions on Neural Networks*, 13(6), 1331-1340, 2002.

[15] Rose K., Gurewitz E., Fox G. (1990); Statistical Mechanics and Phase Transitions in Clustering, *Phys Rev Lett*, 65, 945-948, 1990.

[16] Saric T.; Simunovic, G. (2016); Estimation of Machining Time for CNC Manufacturing Using Neural Computing, *International Journal of Simulation Modelling*, 15(4), 663-675, 2016.

[17] Shi J., Malik J. (2000); Normalized Cuts and Image Segmentation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8), 888-905, 2000.

[18] Tang M., Gong D., Liu S., Zhang H. (2016); Applying Multi-phase Particle Swarm Optimization to Solve Bulk Cargo Port Scheduling Problem, *Advances in Production Engineering and Management*, 11(4), 299-310, 2016.

[19] Tavazoie S., Hughes D.; Campbell M.J., Cho R.J., Church G.M. (1999); Systematic Determination of Genetic Network Architecture, *Nature Genetic*, 22, 281-285, 1999.

[20] Vakhutinsky A.I., Golden B.L. (2003); The Co-adaptive Neural Network Approach to the Euclidean Traveling Salesman Problem, *Neural Networks*, 16(10), 1499-1525, 2003.

[21] Wang J., Tang Z., Qiping Cao, Xinshun Xu (2003); An Elastic Net Learning Algorithm for Edge Linking of Images, *IEICE Trans. Fundamentals*, E86-A(11), 2879-2886, 2003.

[22] Wu S., Liew A.W.C., Yan, H., Yang M. (2014); Cluster Analysis of Gene Expression Database on Self-Splitting and Merging Competitive Learning, *IEEE Trans. Information Technology in Biomedicine*, doi: 10.1109/TITB.2004.824724, 8(1), 5-15, 2014.

[23] Xu R. (2005); Survey of Clustering Algorithm, *IEEE Transaction On Neural Networks*, 16, 645-678, 2005.

[24] Yang K. W. (2015); A Variables Clustering Based Differential Evolution Algorithm to Solve Production Planning Problem, *International Journal of Simulation Modelling*, 14(3), 525-538, 2015.

Author index