INTERNATIONAL JOURNAL

of

COMPUTERS, COMMUNICATIONS & CONTROL

ISSN 1841-9836

ISSN-L 1841-9836

A Bimonthly Journal
With Emphasis on the Integration of Three Technologies

Year: 2013 Volume: 8 Issue: 4 (August)

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

Agora University Editing House

CCC Publications

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

International Journal of Computers, Communications & Control

EDITORIAL ADDRESS:
R&D Agora Ltd. / S.C. Cercetare Dezvoltare Agora S.R.L.
Piaţa 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/

International Journal of Computers, Communications & Control

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: MILLENNIUM BANK, Bank address: Piata Unirii, str. Primariei, 2, Oradea, Romania
IBAN Account for EURO: RO73MILB0000000000932235
SWIFT CODE (eq.BIC): MILBROBU

International Journal of Computers, Communications & Control

Short Description of IJCCC

Title of journal: International Journal of Computers, Communications & Control
Acronym: IJCCC
Abbreviated Journal Title: INT J COMPUT COMMUN
International Standard Serial Number: ISSN 1841-9836, ISSN-L 1841-9836
Publisher: CCC Publications - Agora University
Starting year of IJCCC: 2006
Founders of IJCCC: Ioan Dzitac, Florin Gheorghe Filip and Mişu-Jan Manolescu
Logo:

Publication frequency: Bimonthly: Issue 1 (February); Issue 2 (April); Issue 3 (June); Issue 4 (August); Issue 5 (October); Issue 6 (December).

Coverage:

- Beginning with Vol. 1 (2006), Supplementary issue: S, IJCCC is covered by Thomson Reuters - SCI Expanded and is indexed in ISI Web of Science.
- Journal Citation Reports(JCR)/Science Edition:
  - Impact factor (IF): JCR2009, IF=0.373; JCR2010, IF=0.650; JCR2011, IF=0.438; JCR2012, IF=0.441.
- Beginning with Vol. 2 (2007), No.1, IJCCC is covered in EBSCO.
- Beginning with Vol. 3 (2008), No.1, IJCCC, is covered in Scopus.

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 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:

- Integrated solutions in computer-based control and communications;
- Computational intelligence methods (with particular emphasis on fuzzy logic-based methods, ANN, evolutionary computing, collective/swarm intelligence);
- Advanced decision support systems (with particular emphasis on the usage of combined solvers and/or web technologies).

International Journal of Computers, Communications & Control
ISSN 1841-9836, ISSN-L 1841-9836, Volume 8, Issue 4, August, 2013.

Contents

Combining Feature Methods for Content-Based Classification of Mammogram Images
K. Chikamai, S. Viriri, J-R. Tapamo **499**

Processing Capacity and Response Time Enhancement by Using Iterative Learning Approach with an Application to Insurance Policy Server Operation
M. Ercan, T. Acarman **514**

Parkinson's Disease Prediction based on Multistate Markov Models
O. Geman, H. Costin **525**

Enlarging the Domain of Attraction in Nonlinear Polynomial Systems
F. Hamidi, H. Jerbi, W. Aggoune, M. Djemaï, M.N. Abdelkrim **538**

Multithreaded Application for Real-Time Visualization of ECG Signal Waveforms and their Spectrums
S.S. Ilić, A.Č. Žorić, P. Spalević, Lj. Lazić **548**

A Model for Collaborative Filtering Recommendation in E-Commerce Environment
Y. Jing, H. Liu **560**

PyBNEq - A Tool for Computing Bayes-Nash Equilibria
I. Joldeş, B. Pârv, I. Parpucea, V. Lupşe **571**

Information Security Engineering: a Framework for Research and Practices
M. Li, M. Tang **578**

Smart and Safe Vehicle Monitoring with Fuzzy Integral and Haar-like Features
Se-bin Oh, Yeon-ho Chung **588**

A Fuzzy Data Envelopment Analysis Approach based on Parametric Programming
S.H. Razavi, H. Amoozad, E.K. Zavadskas, S.S. Hashemi **594**

Toward a Holistic Delay Analysis of EtherCAT Synchronized Control Processes
M. Sung, I. Kim, T. Kim **608**

Using Augmented Reality in Remote Laboratories
H. Vargas, G. Farias, J. Sanchez, S. Dormido, F. Esquembre **622**

Knapsack-model-based Schemes and WLB Algorithm for the Capacity and Efficiency Management of Web Cache
A.B. Xiang, L. Xu, B.Z. Niu **635**

Author index **647**

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):499-513, August, 2013.

Combining Feature Methods for Content-Based Classification of Mammogram Images

K. Chikamai, S. Viriri, J-R. Tapamo

Keith Chikamai
Serestina Viriri*
School of Mathematics, Statistics and Computer Science
University of KwaZulu-Natal, Westville Campus
Durban, South Africa
viriris@ukzn.ac.za
*Corresponding author

Jules R. Tapamo
School of Engineering
University of KwaZulu-Natal, Howard College
Durban, South Africa
tapamoj@ukzn.ac.za

Abstract: Breast cancer is among the leading cause of death among females. Studies show that early detection allows for a better prognosis. Mammography is one of the successful ways for early detection of breast cancer. It mostly involves manual reading of mammograms, a process that is difficult and error-prone. This paper discusses a classification model for mammograms based on microcalcification characteristics, as a way of helping radiologists make quick and accurate diagnostic decisions by availing to them similar past cases. The images are pre-processed by Gaussian smoothing and median filtering with 5×5 and 3×3 kernels respectively. Gabor and Haralick features are then extracted to form the image signatures over which similarity measurements are made. Experimental results show an average precision value between 0.5 and 0.61 using Haralick features, 0.49 and 0.57 using Gabor features, and 0.51 and 0.78 using combination of Gabor and Haralick features.
Keywords: Mammogram, Classification, Gabor filters, Grey Level Co-occurrence Matrix, Haralick Features.

1 Introduction

Breast cancer is a high-mortality disease, and one of the leading causes of death among women [1] . If detected early, this disease is manageable, and can be cured in some instances. Mammography is one of the methods used to detect early signs of cancer, and has proven to be very effective [23]. It involves generating images of the breast through X-ray photography, enabling the visualization of the internal breast structure for analysis that can expose any abnormality. Traditionally, mammogram analysis has been manually done by radiologists through visual inspection. The amount of medical images being generated is increasing exponentially. According to Geneva radiology, at Geneva University and Hospitals, images in excess of 30000 are being produced daily [1]. A large image database increases the referential space, providing a solid foundation for solving new cases easily. However, the large size increases the time needed for processing of the images. Speed of decision making is important, especially in medical diagnosis.

Computer Aided Diagnosis (CADx) employs techniques in image processing and artificial intelligence to assist pathologists arrive at objective conclusions about a given image [3]. It is commonly used for identification of suspicious regions in a mammogram, as well as for determination of malignancy. A major part of malignancy determination involves extracting and computing features that are used to characterise the image. Accurate characterization of these features is important to the overall performance the CAD system and can significantly reduce the rate of unnecessary biopsies. Researchers have explored feature extraction methods to characterize breast pathology that include: morphological [3], wavelets [12], fractal and histogram-based measures. However, efficient and accurate retrieval of images based on their content as a field of computer vision is still an open problem [2, 4]. This research work implements the Gabor filter and the Grey Level Co-occurrence Matrix (GLCM) in an attempt to enable efficient and accurate retrieval of mammogram images containing microcalcifications, as described by a pathologist.

The rest of the paper is organized as follows. Section 2 looks at related relevant work in the literature. Section 3 discusses the proposed system including the features used. Experimental setup and results are discussed in section 4 and the paper is concluded in section 5.

2 Related work

Mammogram images lack color information, and usually exhibit low intensity ranges as well as noise occlusions. Overlying vessels and tissues also present a lot of challenges to the detection of malignant objects. This restricts the type of applicable features to those that exploit shape and textural characteristics of objects, with the requirement that these features be stable and robust against the mentioned limitations.

Researchers have attempted Haralick features for the determination of malignancy in mammograms. Hamid et al. [12] attempted a comparison between wavelet, Haralick and shape features for classification of benign and malignant tumors in mammograms. Pre-processing phase included segmentation using adaptive filtering banks described in [14]. Martins et al. [13] combined Haralick features with shape features as input to the K-means and SVM classifier, achieving considerable success rate of 85%.

Muller et al. [4] look at developments in content-based image retrieval (CBIR) in medical domain and present some future promising research directions. The authors note that speed as an evaluation parameter is rarely mentioned yet is important for an interactive system. They also propose that performance comparison for different feature sets needs to be done to identify well performing visual features and their optimal applications. In pointing out future research directions, the study revealed that availability of good quality features could increase accuracy in data mining and related applications. Specialization is also proposed as a means of including the domain knowledge as a measure of improving accuracy.

Wei et al. [15] analytically look at the potential of CBIR in Medical image database retrieval, and discuss the benefits and feasibility of applying it, or extending the current techniques in order to apply them to daily medical practice. They review the limitations of the current non-CBIR approach, as well as obstacles of the application of CBIR to medical image retrieval. They propose a textural analysis approach based on Grey-level Co-Occurrence Matrices for CBIR in Mammography as a case study. The method involves two stages: image analysis and image retrieval. Image analysis determines the discriminating textural features that best act as descriptors for the image, later to be used for the subsequent image retrieval process. Twelve GLCMs were constructed in four directions at three distances. Eleven Haralick features [11] were then calculated for the 12 GLCMs giving a total of 132 features for each Region of Interest (ROI). The $L2$ norm was used for similarity measurement, with the smallest distance indicating most similarity. A total of 329 ROIs were used for evaluation from images sourced from the Mammographic Images Analysis Society (MIAS). Precision and recall were used to test accuracy, with the system achieving 51% and 19% as the highest scores respectively. Both these values were scored at a GLCM distance of 5. The study identified a number of research issues, which include: semantic gap, systems integration, usability and performance evaluation. The major problems identified with current retrieval systems include subjectivity, financial and time costliness and inefficiency of image data representation. Obstacles to CBIR application include image noise and many image formats available.

3 General Structure of Proposed Model

The proposed model follows the typical CBIR model (Figure 1). It is generally composed of two stages: off-line feature extraction and on-line image retrieval. Off-line feature extraction involves extraction of features from each of the database images and their storage into the feature domain space. Features are stored in feature vectors which are descriptors of their corresponding images. During on-line image retrieval, a user supplies a query example image whose features are also extracted and used by subsequent algorithms against the database features.

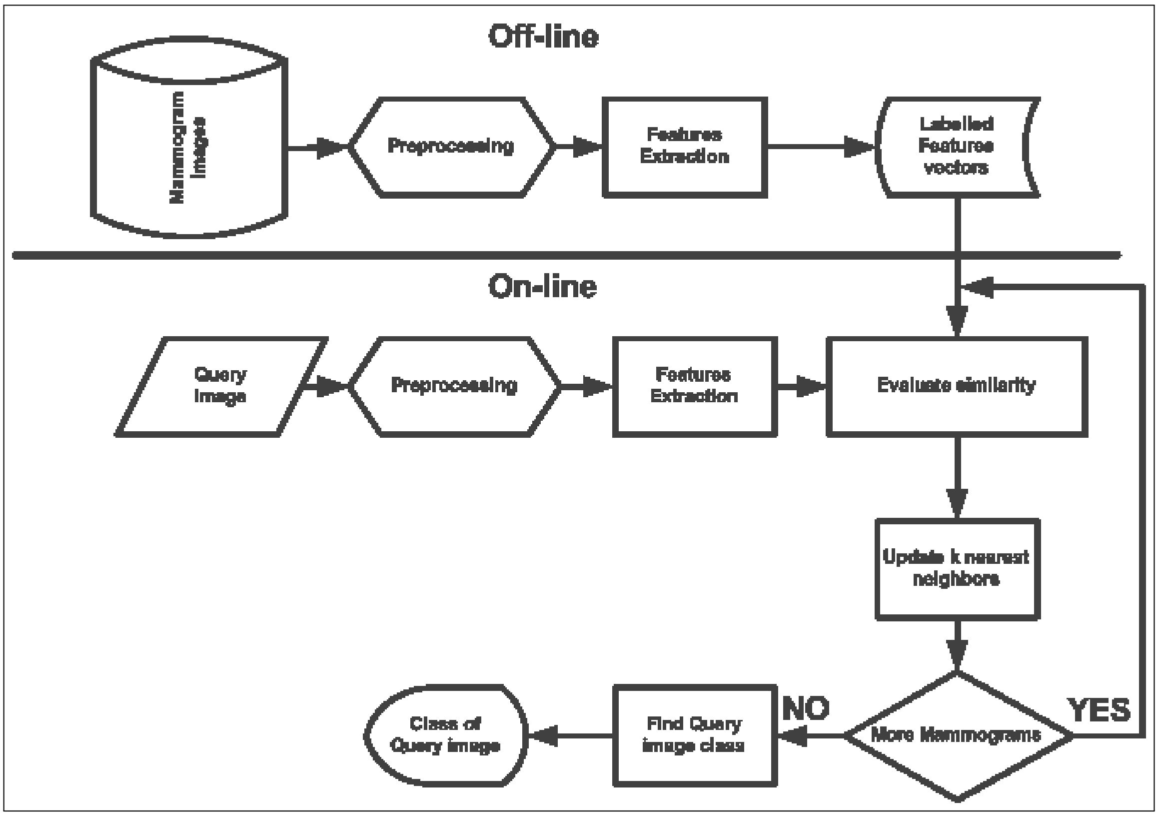


Figure 1: Different steps of a content based classification of a candidate mammogram

3.1 Pre-processing

Breast ROI extraction and denoising

The three techniques applied prior to features extraction are: median smoothing, gaussian smoothing, and region growing. Given an image I with R rows and C columns, the preparation of this image for further processing is carried out in the three following phases:

Phase 1:
The image is first processed using a median filter to remove sporadic sharp frequencies that are characteristics of digital noise. This filter has some edge-preserving characteristics since it does not adversely blur edges. Assuming $\mathcal{N}$ is the kernel, made of neighborhood

pixels around the target pixel ,(x_0, y_0), defined as

$$\mathcal{N} = \{(x_{-n}, y_{-m}), (x_{-n+1}, y_{-m}), ..., (x_n, y_m)\} \tag{1}$$

where $n \times m$ is the kernel size. The median filtering of the (x_0, y_0) is calculated as follows,

- Sort $\mathcal{N}$ in the sequence $S = (S_i)_{i=0,1,...,(n-1)\times(m-1)}$,
- Assign the median value of the sorted sequence to the target coordinates i.e. $I(x_0, y_0) = S_{\frac{(n\times m)}{2}}$

In this case, $n = m = 3$

Phase 2: Gaussian smoothing is then applied on the output image using a kernel $g(x, y)$ of dimension 5×5. The kernel used is of the form,

$$g(x, y) = \frac{1}{2\pi\sigma^2} e^{\frac{x^2+y^2}{2\sigma^2}} \tag{2}$$

Phase 3: After denoising, region growing is applied to detect the breast region of interest and isolate artificial labels. In order to increase the efficacy of the process by insensitizing the algorithm against small random signal perturbations characteristic of noise, the region growing considers blocks of 8 pixels at a time instead of individual pixels.

Local Gradient and Contrast Enhancement

The image is further enhanced using gradient and contrast enhancement techniques [20]. Gradient enhancement increases the intensity of pixels in an adaptive manner. Taking $I(x, y)$ as the intensity function of a 2D image, the gradient at a pixel (x, y) in a mammogram image is given by,

$$g(x, y) = \frac{1}{n \times n} \sum_{i=-\frac{m}{2}}^{\frac{m}{2}} \sum_{j=-\frac{n}{2}}^{\frac{n}{2}} |I(x+i, y+j) - I(x, y)| \forall x, y \in S \tag{3}$$

The calculated gradient values are then added to the original image to give the gradient enhanced image $I\prime(x, y)$,

$$I\prime(x, y) = I(x, y) + g(x, y) \tag{4}$$

where

$S = \{(x, y) \mid 0 \le x \le C-1,\ 0 \le y \le R-1\}$ is the set of all image pixels

m, n - are the vertical and horizontal spatial dimensions of the kernel m and n determine the extent over which the gradient value is calculated and by implication, the size of objects that are enhanced. A square kernel is used in this work i.e. $m = n = 3$. This technique increases the intensity of pixels relative to the gradient of their local neighborhood. Those areas presenting a higher gradient will thus have their intensities increased more, as determined by the kernel size. The kernel size is intuitively chosen to approximate the spatial extent of microcalcifications in order to enhance their gradient.

Contrast enhancement uses the mean of a region to alter its pixels intensities. The mean of a pixel neighborhood is iteratively calculated as follows,

$$\mu^k(x, y) = \frac{1}{m \times n} \sum_{i=-\frac{m}{2}}^{\frac{m}{2}} \sum_{j=-\frac{n}{2}}^{\frac{n}{2}} \mu^{k-1} I(x+i, y+j) \forall x, y \in \sum * \tag{5}$$

The contrast enhanced image is represented as follows,

$$I''(x,y) = \frac{\mu^k(x,y)}{L-1} I'(x,y) \tag{6}$$

where k specifies the iterations over which the mean is calculated, μ^k is the mean at the kth iteration, L is the highest intensity in the image * the other variables are defined as for gradient enhancement.

While gradient enhancement increases the intensities of high gradient areas without affecting the rest, contrast enhancement diminishes the effect of low contrast areas by lowering their intensity values. The neighborhood determines the objects (by size) that are enhanced. The number of iterations, k, determines how much the original signal is attenuated. A value of $k = 2$ is empirically chosen for the experimental runs in this work. Similar to the gradient kernel, a square mask of dimension 9 (i.e. $m = n = 3$) is used for the contrast kernel. These filters are useful in reducing the effect of the monotonous pelvic muscle which mimics the gradient levels of microcalcifications.

3.2 Feature extraction

The Gabor filter and Haralick features are used for textural analysis of the pre-processed image. The Gabor filtering is an intermediate stage, with the first and second moments being calculated from the Gabor filtered image to give the final features. The techniques and the specific parameters used are discussed in the following sections.

Gabor Filtering

Gabor filter is a transform function related to the Fourier transform which can be used to convey spatial information in addition to frequency properties of a signal. It is commonly applied as a band-pass filter in signal processing where it is used to determine the sinusoidal frequency and phase content of local sections of a time varying input signal, and has been found useful in image compression. Among other useful properties, the Gabor filter has been found to better minimize the conjoint time-frequency information resolution of a signal [4].

A Gabor filtering is obtained by multiplying a complex sinusoidal plane wave of a certain frequency with a Gaussian envelope as follows [5]:

$$g(t) = ke^{j\theta} w(at)s(t) \tag{7}$$

where
$w(t) = e^{-\pi t^2}$ is the Gaussian envelope,
$s(t) = e^{j(2\pi f t)}$ is the sinusoidal function, where f is the frequency of the sinusoidal plane wave,
k is the constant, and $e^{j\theta}$ determines the orientation

The strength of the Gabor filter response depends on the filter's congruence with the local signal; where the filter's sensitivity is determined by tuning of the parameters that include: orientation, phase and frequency [5]. Given that Gabor filters are not inherently orthogonal [9], this section determines an optimal set of parameters for designing Gabor filter jets that will detect the desired range of object characteristics with minimum redundancies. Based on the work in [6], this project implements the following 2-D Gabor filter:

$$g_{\lambda\theta\varphi}(x,y) = \frac{1}{\sqrt{2\pi\sigma_x\sigma_y}} e^{-\left(\frac{x'^2}{\sigma_x^2} + \frac{\gamma y'^2}{\sigma_y^2}\right)} cos(2\pi x'/w + \varphi) \tag{8}$$

$$x\prime = xcos(\theta) + ysin(\theta) \tag{9}$$

$$y\prime = ycos(\theta) - xsin(\theta) \tag{10}$$

where

λ, θ and φ are configurable input parameters specifying wavelength, orientation and phase, respectively; γ specifies the aspect ratio of the gaussian envelope. Its value is empirically set to 0.5; σ_x and σ_y define the convergence of the gaussian envelope along the $x-$ and $y-$ axes respectively. They are defined as follows,

$$a = \left(\frac{U_h}{U_l}\right)^{\frac{1}{S-1}}$$

$$\sigma_u = \frac{U_h\left(a-1\right)}{\left(a+1\right)\sqrt{2log2}}$$

$$\sigma_v = tan\left(\frac{\pi}{2k}\right)\left[U_k - 2log\left(\frac{2\sigma_u^2}{U_h}\right)\right]\left[2log2 - \frac{\left(2log2\right)^2\sigma_u^2}{U_h^2}\right]^{-\frac{1}{2}}$$

$$\sigma_x = \frac{1}{2\pi\sigma_x}, \qquad \sigma_y = \frac{1}{2\pi\sigma_y}$$

The spatial width of the filter (FW) is then linearly scaled from the derived standard deviation as follows,

$$FW = 4\sigma + 0.5 \tag{11}$$

This filter width calculation is empirically established to give a good compromise since it does not greatly affect border pixels. The pre-processed input image $I(x, y)$ is convolved with Gabor filter $g(s, t)$ to give the response image $r(x, y)$ as shown in Equation (12),

$$r(x, y) = \int\int_{(x,y)} I(s, t)g(x - s, y - t)\delta s\delta t \tag{12}$$

This implementation (Equation 12) is computationally expensive in the spatial domain, making it impractical for large input images and Gabor kernels. This study thus takes advantage of the convolution theorem to implement filtering in the frequency domain as defined in Equation 14. Since convolution in the spatial domain is equivalent to multiplication in the frequency domain, the computational complexity is reduced using the latter method [21]. It uses the comparatively optimal FFTW library to carry out the fourier-spatial transformations.

$$\mathfrak{F}\left(r(x, y)\right) = \mathfrak{F}\left\{I(x, y)\right\}\mathfrak{F}\left\{g(x, y)\right\} \tag{13}$$

$$g(x, y) = \mathfrak{F}^{-1}\left\{\mathfrak{F}\left\{I(x, y)\right\}\mathfrak{F}\left\{g(x, y)\right\}\right\} \tag{14}$$

This work considers a set of Gabor filters configured with four orientations: 0, 45, 90 and 135. These values are calculated according to Equation 15 considering recommendations in [[7, 8]].

$$\theta_k = \frac{k\pi}{n},\ k = \{0, \ldots, n-1\} \tag{15}$$

where n is the number of orientations, k represents the kth orientation.

This method ensures equidistant spacing in the orientation field. Furthermore, the orientation space is chosen from the range $\theta_k \in [0, \pi]$, which provides sufficient coverage since it has been established that response values in the range $[\pi, 2\pi]$ only differ from those in $[0, \pi]$ by phase shift [7, 8].

GLCM and Haralick features

In this work, Haralick features are extracted to describe the Mammogram's textural characteristics. Textural characteristics are described by patterns of pixel intensities [11]. Practically, these intensities are described by a distance-angular relationship model using a Grey Level Co-occurrence Matrix (GLCM) as proposed by Haralick et al. [11]. GLCMs are second-order statistics that define relationships between distinct tonal intensities by measuring the frequency with which they occur together at certain directions (θ) and distances d, and fall under statistical textural classification approaches.

Grey Level Co-occurrence Matrix feature method is also used to model grey level dependencies of mammogram images. Similar to the Gabor wavelets, set of matrices are defined over four directions: 0, 45, 90 and 135, at a distance d and is represented as $P(i, j, \theta, d)$. A subset of nine Haralick features [11] are calculated over the GLCM to describe the mammograms' textural characteristics. For notational convenience, lets denote $P(i, j)$ as the probability of i occurring alongside j, the Haralick features are then defined as follows:

$$Energy = \sqrt{\sum_{i,j} P(i,j)^2} \tag{16}$$

$$Entropy = -\sum_{i,j} P(i,j) log(P(i,j)) \tag{17}$$

$$Contrast = \sum_{i,j} P_{i,j}(i-j)^2 \tag{18}$$

$$Homogeneity = \sum_{i,j} \frac{P_{i,j}}{1+(i-j)^2} \tag{19}$$

$$\text{Max prob} = MAX(P(i,j)) \tag{20}$$

$$Correlation = \sum_{i,j} P_{i,j} \left[\frac{(i-\mu_i)(j-\mu_j)}{\sqrt{\sigma_i^2 \sigma_j^2}} \right] \tag{21}$$

$$Dissimilarity = \sum_{i,j} P_{i,j}|i-j| \tag{22}$$

$$idm = \sum_{i,j} \frac{1}{1+(i-j)^2} P(i,j) \tag{23}$$

3.3 Classification of a Candidate Mammogram

The CBIR system designed, returns results ranked in order of relevance to the query image. The number of returned images (represented by k in this context) impacts on accuracy or precision of the system if factored during system evaluation. The k Nearest Neighbor (k-NN) classifier is used for classification; we used the version described in [9, 22]. In our context there are two classes M_0, and M_1:

Algorithm 1 :K-NN Algorithm to Classify a mammogram. Given a mammogram c_m, classify c_m based on a database of mammogram grouped into two clusters; one with mammograms containing mammograms microcalcifications and the other one with the ones without microcalcifications.

Require: M, c_m, k ▷ M, is the set of mammograms in the database (each mammogram is labeled belonging to M_0 or M_1) and c_m is the candidate mammogram to be classified. k is the number of nearest neighbors to be considered.

Ensure: $m \in M_l$, ▷ means M_l $(l = 0, 1)$ is the class to which belongs the mammogram c_m

Let S be a sequence k elements, ▷ s_i will be the i^{th} element of S

$S = \emptyset$ ▷ Initialize S to an empty sequence

for all element $y \in M$ **do**

    insert(S, y, c_m) ▷ insert y in S, in increasing order of distance between y and c_m,

end for

$l = \arg \max_{v \in \{0,1\}} \sum_{i=1}^{k} I_{M_v}(s_i)$ ▷ This means $c_m \in M_l$

- M_0: class of mammograms without microcalcifications
- M_1: class of mammograms with microcalcifications

insert(S, y, c_m) inserts y in S, in increasing order of distance between y and c_m. Given a set A, $I_A(.)$ is and indicator function defined as follows:
$I_A(x) = 1$ if $x \in A$ and $I_A(x) = 0$ otherwise.

The classifier in **Algorithm 1** finds the minimum distance between the given query vector c_m and all mammograms in M, builds a sequence S of k vectors representing the mammogramms with the minimum distance. The query vector is finally assigned to the class $M_l(l = 0, 1)$ that has the majority of elements of S.

Experiments are conducted with values of k taken from an set of 5 elements, $\{1, 3, 5, 7, 9\}$.

4 Results and Discussion

4.1 Image dataset

Experimental tests were conducted on a set of 60 images sourced from the Mammographic Image Analysis Society (MIAS) database [10]. This database contains 322 mammogram images of mixed pathologies, and is accompanied by ground truth that has been verified and marked by radiologist. The ground truth gives the severity of the pathology (Malignant/benign) as well as the spatial locality and extent of the pathology. The pathology classes are:

- Calcification
- Well-defined/circumscribed masses
- Spiculated masses
- Other, ill-defined masses
- Architectural distortion
- Asymmetry

- Normal

The breast tissue of each mammogram has been classified depending on density. The density of a breast tissue alludes to the amount of fat present in that tissue. Fatty tissues appear as relatively darker areas in mammograms. In order to reduce the amount of fatty tissue, the mammograms are classified under any of these three types: Fatty, Fatty-glandular and Dense glandular.

All dataset images have been quantized to 256 grey levels, and were digitized at a resolution of 200 microns. They are padded and clipped to occupy a standard size of 1024×1024 pixels.

4.2 Experimental setup

Sixteen images were selected from the database to form the query image set. These images were taken from both classes of pathology, i.e., normal and malignant. Normal images in this experiment were defined as those images not containing Microcalcifications. This definition covers images diagnosed as positive for other malignancy indicators such as circumscribed masses and asymmetries. Results were then collected for each round for every query image. For generalized results, precision values calculated are averaged over all query images instead of one. The query process was repeated ten times (ten iterations) using a randomized set of 8 images from the normal class, the average precision values obtained in each round were then averaged over the ten iterations to give the statistical base for reporting. This process was done for every value of $k \in [1, 3, 5, 7, 9]$.

4.3 Performance Metric

The average precision curve [18] was used to evaluate the performance of the system. Precision gives the general classification performance of the system. It measures the ability to correctly classify both sample sets. Sixteen images from both classes are used as query images, and results collected after every round as explained in section 4.2. For every returned result set, precision is calculated as follows,

$Precision = \frac{R}{k}$, where R is the number of accurate predictions and k the number of neighbors

Algorithm 2 is used to compute precisions. The average of precision values for both sets of query images is then taken as the precision value for the round. The precision values are then used to sketch the precision curve for diagrammatic representation. The precision value ranges from a minimum of 0 to a maximum of 1. A high precision value implies that the system has a commensurately high ability of correctly classifying a given sample.

4.4 Discussion

The first results of the experimental runs are given in Figures 3, 4 and 5, show the Haralick results for single, double and combined class query image sets, respectively. The figures 6, 7 and 8, show the Gabor filter results for single, double and combined class query image sets, respectively. The figures 9, 10 and 11 show the results for single, double and combined class query image sets, respectively, using combined Gabor and Haralick features.

The first results (Figures 3, 4 and 5), show precision values obtained by querying the database using 8 images randomly selected from the "Normal" class. The querying process (section 4.2) is evaluated over five rounds. The lowest precision value of 0.71 is scored at distance $k = 3$, with the highest value of 0.88 scored at the distance $k = 1$. The system gives a low score for images identified positive for microcalcifications . The highest score of 0.375 is recorded at distance $k = 1$, and the lowest score of 0.13 scored at distance $k = 9$. For mixed class query images (Figure 4), $1-$Nearest Neighbor gives the best precision score at 0.69, with the $9-$Nearest

Algorithm 2 Retrieval performance benchmarking

$Query\_Set \leftarrow getRandomImages(Database)$
for all kNN $distances$ k **do**
  $Precision(k) \leftarrow 0$
  for $i = 1$ **TO** $numOfIterations$ **do**
    for all $Query\_image$ q in $Query\_Set$ **do**
      $Result \leftarrow getNearestNeighbors(q)$
      $Precision(k) = Precision(k) + getPrecision(Result)$
    end for
    $Precision(k) \leftarrow Precision(k)/no.ofqueryimages$
  end for
  $Precision(k) = Precision(k)/no.ofIterations$
end for

Figure 2: Query result using Normal class images

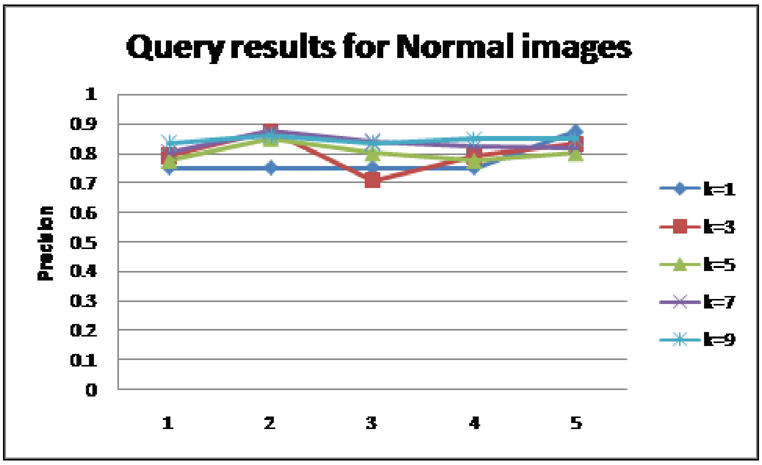


Figure 3: Query result using Normal class images

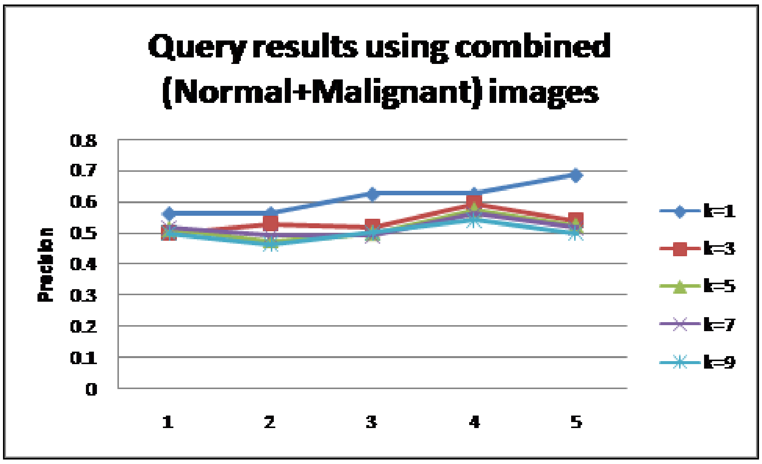


Figure 4: Query results using mixed class images

Neighbor giving the lowest score at 0.47. Overally, the performance degrades with an increasing value of k (Figure 5). The highest value scored is 0.61 at $k = 1$, with the lowest value of 0.50

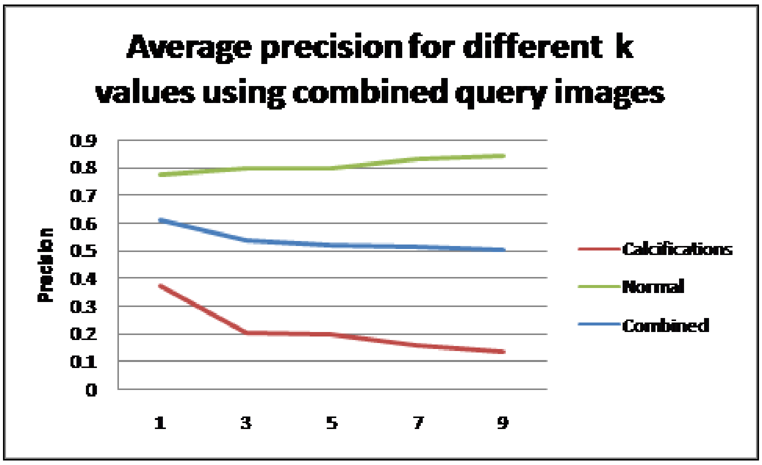


Figure 5: Average Precision for Combined class query image set

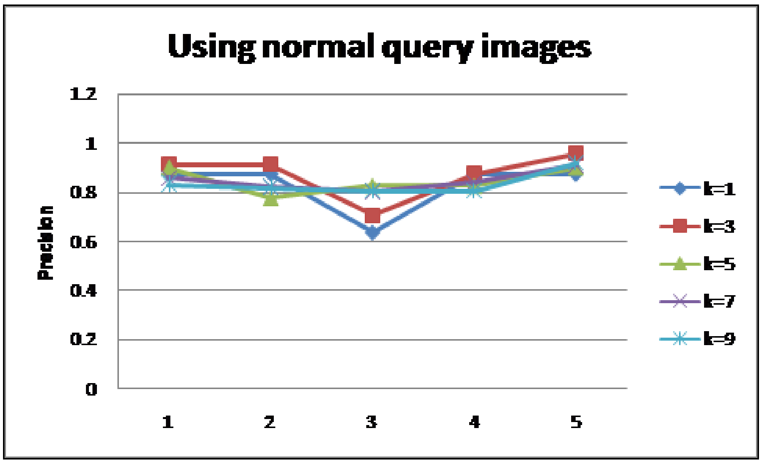


Figure 6: Query result using Normal class images

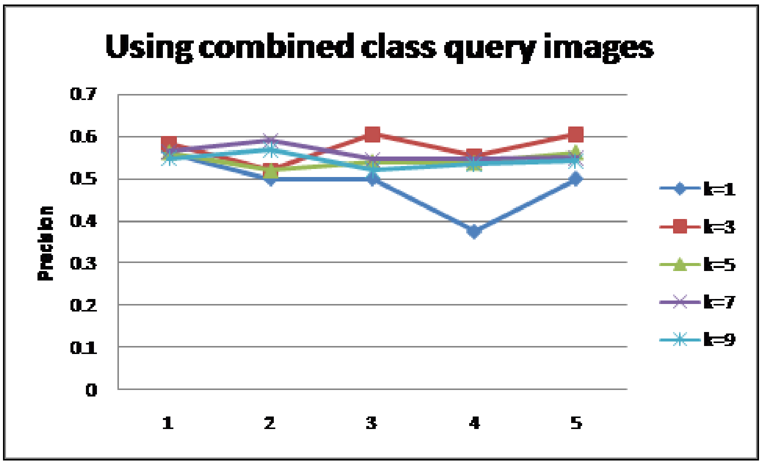


Figure 7: Query results using mixed class images

being scored at $k = 9$.

Compared to Haralick features, the Gabor feature set gives slightly higher average values

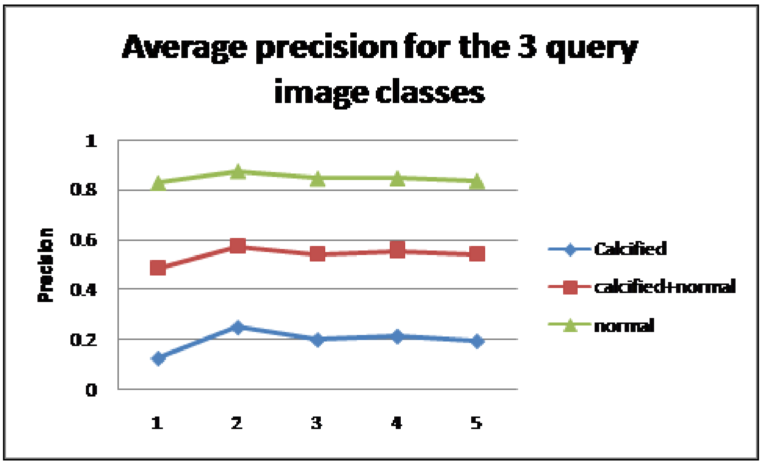


Figure 8: Average Precision for Combined class query image set

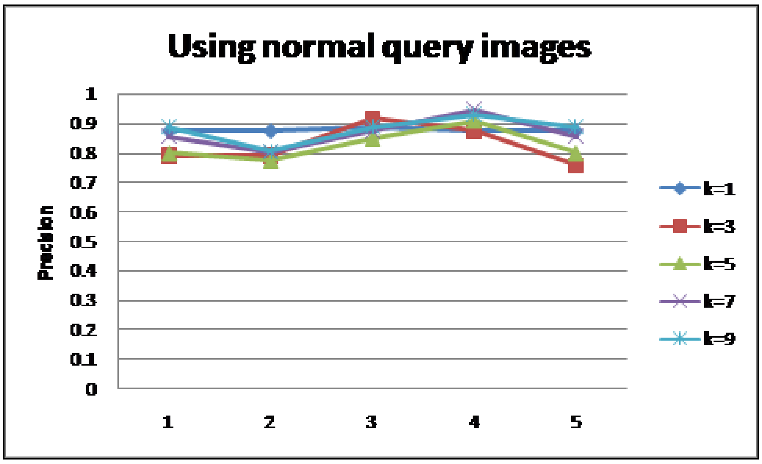


Figure 9: Query result using Normal class images

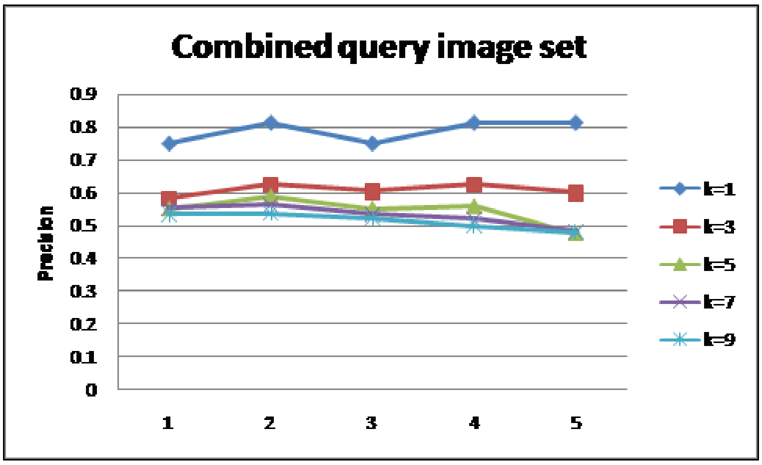


Figure 10: Query results using mixed class images

(Figures 6, 7 and 8) for queries involving benign classified images with a high score of 0.85 (considering all values of k). However, the Gabor vector gives relatively lower high scores for the

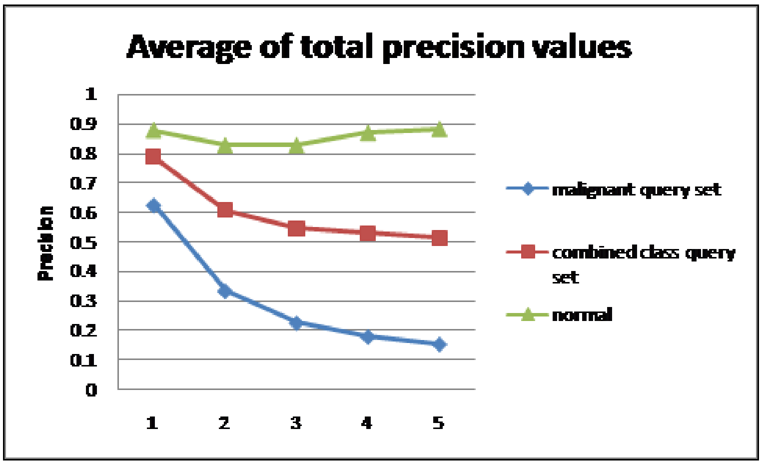


Figure 11: Average Precision for Combined class query image set

calcification and combined class query image set, at 0.25 and 0.57 respectively. It however gives consistently better average results considering all values of k for all three query classes.

Comparison of retrieval performance was done considering a mixed Gabor and Haralick feature set (Figures 9, 10 and 11), which gave mixed results for various values of k. The combined feature set gave the highest score for queries involving normal classified images, with a score of 0.88 at $k = 9$ (Figure 9). It also registered the highest scores for $k \in [1, 7]$, with the Gabor vector giving the highest values for the remaining values of k. This set also gave high values for the calcification and mixed class query images at 0.625 and 0.79 respectively, both values attained at $k = 1$. Summarily, the combined feature set of Gabor and Haralick features enhances retrieval performance for all classes of query images. The best consistent performance is achieved at $k = 1$ for all query classes.

For comparison, Wei et al. [19] implement a GLCM-based mammogram retrieval system for comparison with their algorithm. Regions of interest (ROIs) are cropped from mammogram images of multiple pathologies. The images are then Gabor-filtered before calculation of Haralick features. The GLCM matrix is calculated over three distances and four orientations. Their average precision values range between 0.33 and 0.64. Our system proves to be more discriminating towards images not containing microcalcifications than for those containing microcalcifications. A possible explanation is that a lot of unnecessary breast information is being included for similarity calculations. This means that the algorithm needs to be enhanced more to reduce the impact of non-calcification regions. The high dimension of features might also have a negative effect on the accuracy of the algorithm by introducing redundancies. Our precision value is however not far-off the one obtained in [19]. This adds to the fact that the proposed model automates ROI selection.

5 Conclusions and Future Work

Mammography allows the detection of breast cancer in its early stages, which makes possible early remedial measures that can reduce the high mortality rates associated with the disease. This paper discussed a content-based classification model for mammogram images, with the objective of availing a second opinion to a radiologist for reference during diagnosis. It implemented the Gabor filter and Haralick features for textural analysis and description for similarity assessment. This work evaluated Haralick features at five distances, $k \in [1, 3, 5, 7, 9]$ and four orientations East, North, South and West. The best value is attained using a combined Gabor and Haralick feature set, with a score of 0.79 at the distance of $k = 1$, with lowest value as 0.49 at the distance $k = 9$ using Gabor features only. The moderate precision value could be attributed to the impact of non-calcification breast areas, as well as redundant and less discriminating features. Work is underway to remove redundant features as well as those features that do not discriminate well with respect to microcalcifications.

Bibliography

[1] Rangayyan, R.M. (2005); Breast Cancer and Mammography, *Biomedical Image Analysis, Springer-verlag*: 22-27.

[2] Chu, K.C. et al (1996); Recent Trends in US Breast Cancer Incidence, Survival, and Mortality rates, *Journal of the National Cancer Institute*, ISSN 1460-2105, 88(21): 1571-1579.

[3] Oliver, A.; Freixenet, J.; Marti, R.; Zwiggelaar, R. (2006); A Comparison of Breast Tissue Classification Techniques, *MICCAI, CRC Press*: 872-879.

[4] Hamid, S.Z.; Farshid, R.R.; Siamak, P.N. (2004); Comparison of Multiwavelet, Wavelet, Haralick, and Shape Features for Microcalcification Classification in Mammograms, *Pattern Recognition* 37: 1973-1986.

[5] Li, C.T.; Wei, C.H; Wei, C.H.; Li, C. (2005); A Content-based Approach to Medical Image Database Retrieval, *Database Modeling for Industrial Data Management: Emerging Technologies and Applications* 10(6): 681-685.

[6] Muller, H.; Michoux, N.; Bandon, D.; Geissbuhler, A. (2004); A Review of Content-based Image Retrieval Systems, Medical Applications-clinical Benefits and Future Directions, *Int J Med Inform*: 1-23.

[7] Miyamoto, E.; Merryman, T.; Fast Calculation of Haralick Texture Features, *Technical Report, Carnegie Mellon University.*

[8] Martins, O.L. et al (2009); Detection of Masses in Digital Mammograms using K-Means and Support Vector Machine, *Electronic Letters on Computer Vision and Image Analysis*, 8(2): 39-50.

[9] Wei, C.H.; Chang-Tsun, L.; Roland, W. (2009); A Content-based Approach to Medical Image Database Retrieval, *Database Technologies: Concepts, Methodologies, Tools, and Applications*: 1062-1083.

[10] Haralick, R.M.; Shanmugan, K.; Dinstein, I. (1973); Textural Features for Image Classification, I*EEE Transactions on Systems, Machine, and Cybernetics SMC*, 3(6): 610-621.

[11] Lee, S.K. et al (2000); A Computer Aided Design Mammography Screening System for Detection and Classification of Microcalcifications, *International Journal of Medical Informatics* 60: 29-57.

[12] Lee, T.S. (1996); Image Representation using 2D Gabor Wavelets, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10: 959-971.

[13] Turner, M.R. (1986); Texture Discrimination by Gabor Functions, *Biological Cybernetics*, 10: 71-82.

[14] Hamamoto, Y. et al (1998); A Gabor Filter-based Method for Recognizing Handwritten Numerals, *Pattern Recognition*, 31(4): 395-400.

[15] Kruizinga, P. et al (1999); Comparison of Texture Features based on Gabor Filters, *ICIAP*: 142-147.

[16] Liu, C.; Wechsler, H. (2002); Gabor Feature based Classification using the Enhanced Fisher Linear Discriminant Model for Face Recognition, *IEEE Trans. Image Processing*, 11(4): 467-476.

[17] Kyrki, V.; Kamarainen, J.; Kalviainen, H. (2004); Simple Gabor Feature Space for Invariant Object Recognition, *Pattern Recognition Letters*, 10: 311-318.

[18] Dimai, A. (1999); Rotation Invariant Texture Description using General Moment Invariants and Gabor Filters, *Proc. of the 11th Scandinavian conference on Image analysis*: 391-398.

[19] Steinbach, M.; Tan, P.N. (2009); KNN: k-Nearest Neighors, Data Mining and Knowledge Discovery, *Taylor Francis Group*, Ch. 8: 151-161.

[20] Suckling, J. et al (1994); The Mammographic Image Analysis Society Digital Mammogram Database, Exerpta Medica, *International Congress Series*, 1069: 375-378.

[21] Luo, J.; Nascimento, M.A. (2004); Content-based Sub-Image Retrieval using Relevance Feedback, *Proc. of MMDB*: 2-9.

[22] Wei, C.H.; Li, Y.; Li, C.T. (2007); Effective Extraction of Gabor Features for Adaptive Mammogram Retrieval, *Proc. of ICM*: 1503-1506.

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):514-524, August, 2013.

Processing Capacity and Response Time Enhancement by Using Iterative Learning Approach with an Application to Insurance Policy Server Operation

M. Ercan, T. Acarman

Mutlu Ercan
Technology Management Department
AvivaSA Retirement and Life Insurance Company, İstanbul, Turkey
mutlu_ercan@yahoo.com

Tankut Acarman*
Computer Engineering Department
Galatasaray University, İstanbul,Turkey
tacarman@gsu.edu.tr
*Corresponding author

Abstract: In this study, computing system performance enhancement by using iterative learning technique is presented. Computational response time and throughput of the computing system is improved by introducing computational cost model and selection probability for each individual job. Excepted gain by enforcing dynamic caching is maximized in terms of classifying the arriving computing jobs on an selective manner and dynamically replacing them in a limited memory space. Gain maximization is performed by tuning the window size which helps to compare the computing jobs in terms of their individual selection and occurence probabilities. Fairly special computing work in insurance risk investigation is chosen for experimental validation of the proposed approach. Aspect Oriented Programming (AOP) methodology on Java platform is used for the experimental setup. AOP allows to identify computational jobs and their parameters based on the marked annotations. Experimental results show that the developed iterative learning based caching algorithm performs better than the other well known caching techniques.
Keywords: iterative learning, caching, computational cost model.

1 Introduction

Information processing requests and distributed applications are increasing along the web-based server-client interaction development cycle, and the systems with much more powerful processing capabilities are needed. Despite the increase in hardware's processing and transmission busses' speed, performance enhancement of the overall computing system may require simultaneous resource allocation depending on the computing task. To enhance the overall processing performance, research is focussed on software control and dynamic resource allocation of the computing and storage units. Static and dynamic scheduling algorithms have been developed on a real-time basis systems operations. Static scheduling algorithms have low costs, but incapable to adapt scheduling rules to enhance efficiency of the limited resources subject to unpredicted computing jobs' arrival whereas dynamic scheduling algorithms may adapt and respond in a timely-fashion. The dynamic scheduling algorithms are mainly based on selection criteria driven approach such as Earliest Deadline First, Highest Value First, Highest Value Density First, [1]. Data access operations may degrade computing system performance. Unnecessary data reading may be avoided by caching the same and frequently occurring computational jobs, [2]. Caching is one of the common performance improvement patterns which may be applied in different layers of the systems, for example web browser caches static web content, database caches blocks of data from disk for performance, the SAN controller will cache read/write data from disk and only complete when efficient to do so, (see for instance different caching approaches in [3]). There are some initial studies to model and control resources allocations in multi-tier virtualized applications which are considered to be part of the green and cloud computing. The applications are running in isolation by runtime resource allocation. In virtual execution environments, CPU utilization is controlled versus workload changes to achieve performance criterion based on mean response time, [4], [5]. A preliminary version of this paper is presented in [6] where the cost of processing a job in the Central Processor Unit (CPU) is derived and dynamic caching methodology is studied via simulation. Iterative Learning Control (ILC) is a methodology, which can be applied to repetitive systems, tries to improve system performance based on the knowledge of the previous experiences. System learns from previous repetitions to improve next repetitions to minimize the tracking error, [7]. In factories, industrial manipulators performs same cycles repetitively, by using ILC energy consumption and operation time is minimized, [8].

In this paper, ILC-based caching scheme is introduced to enhance response time and processing capacity of the computing systems. Computational jobs are processed by the CPU which uses other storage devices such as memory and disk etc,. The service time of the CPU is minimized by preventing reprocessing the same jobs and results by enforcing learning scheme, time cost of obtaining the outputs of the jobs is minimized. Due to the limited caching memory resources, caching size is limited and a reference value for the iterative caching procedure is derived. The proposed dynamic caching scheme is experimentally tested on the insurance policy server operating on a real-time basis and its computational performance is presented. This paper contributes to dynamic caching for computing performance enhancement by using iterative learning technique.

This paper is organized as follows. In Section 2, computer process modeling and cost diagram of allocating the memory for caching and computational purposes is presented. The expected gain of the individual computational job is described. In Section 3, the reference value generation, the output value of the computing system and the error term is proposed towards autonomous management. Experimental studies to validate the effectiveness of the proposed approach are presented in Section 4. Finally, some conclusions are given.

2 Computing Process Modeling

From the view point of our study, the cost of processing a job in the CPU instead of caching its previously obtained result is considerably higher. The memory is used to store the repeated job results. Accessing to the memory from the CPU is fast enough. For example, the average time of getting an output from memory is approximately 12.10^{-7} milliseconds. A priori, cost of a simple request from the memory is cheaper than complex memory usages. CPU is the bottleneck where all jobs are processed, and mainly the memory is used to perform the jobs. There are two kinds of jobs: I/O jobs and computational jobs. The computational jobs are deterministic: they always give the same output with respect to the same input. "$3+x$" is a computational job, because when a value is set for "x" the output is independent from time. The other jobs are the I/O jobs for example writing to a file system. An I/O job such as data writing to a disk is not a deterministic job from this point of view. Random or time dependent jobs are considered to be I/O jobs. In this study, to minimize the time cost of processing the repeating jobs by a single CPU, some amount of the previously processed job results are cached in the memory. Obtaining the results from memory is a faster operation compared with running a complex job in CPU which may use other storage devices such as memory and disk etc,. Some partition of the memory is used to cache the job results and this partition versus the total memory space is

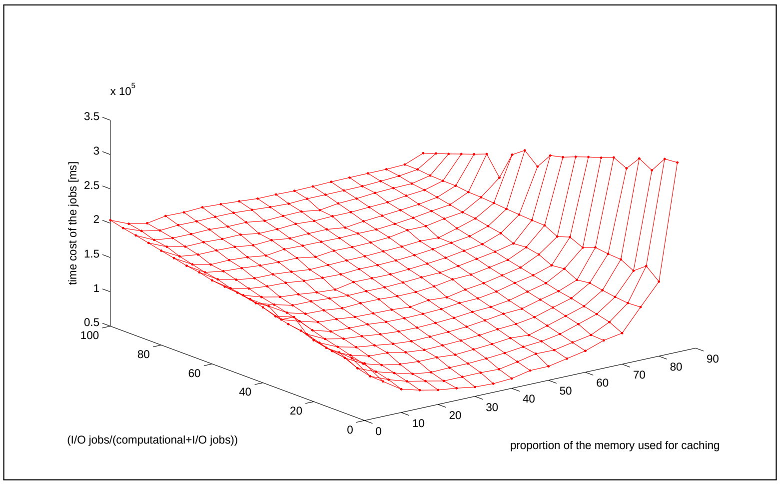


Figure 1: Cost diagram of memory allocation subject to static size to cache the jobs' results

denoted by α,

$$\alpha = \frac{m}{m_T} \tag{1}$$

where m denotes the total footprint of the job outputs and m_T is the total memory of the system. Each job produces an output object that occupies some footprint in the memory. The tradeoff of regulating the partition of the memory to cache some amount of the job results versus the request of memory space to process the job whose output has not been cached in the memory is illustrated by the following time cost diagram in Fig.1. In this figure, a set of I/O and computational jobs has been generated to investigate the computing performance of the system. The jobs are mainly constituted by I/O and computational tasks. The characteristics of the individual jobs such as footprints of the outputs, service and arrival rates are chosen randomly. The service rates are chosen between 1 and 10000, the footprints are between 1 and 1000 and all jobs are distributed uniformly. Time cost with respect to the memory caching partition and the ratio of the I/O jobs versus the total number of the I/O and computational jobs are plotted. When the small partition of the memory is reserved for caching purposes, the possibility of finding the previously processed result is low and the job needs to be reprocessed by the CPU which costs some time. If most of the jobs are I/O jobs, caching the computational job results does not affect the CPU processing time. When the number of computational jobs is increased versus the overall jobs, caching may reduce CPU's processing load by using the previously processed identical job result. On the other hand, to increase the memory caching partition may cause slower computing system response due to low memory space left for processing the new jobs or whose results have not been cached, *i.e.,* when the system becomes unable to find enough memory to process the new jobs, time cost of the jobs increases. The slope of the cost function presents a valley form and its slope in the left side illustrates this effect.

3 Real-time Dynamic Caching

The time cost of computation for each individual job may be described by,

$$t_{CPU}(\alpha) = \begin{cases} t_r & \text{if the job is operated by the CPU} \\ t_c & \text{if the job result exists in the memory} \end{cases} \tag{2}$$

where t_r and t_c denotes the time cost of the job run by the CPU (runtime) and the time cost of getting its output from the memory (cache time), respectively. The jobs are assumed to be in the processing queue and the total size of the jobs is restricted by the window dimension. Initially, the last $n+m$ jobs can be supposed in the sliding window. When one job is processed, a new job enters in the sliding window and the last job is left out of the window as illustrated in Fig.2.

The total time cost of the computational jobs in the *kth* window is shown as;

$$\Delta t^k_{CPU}(\alpha) = \sum_{i=1}^{m} b_{ki} t^{ki}_{CPU}(\alpha) \qquad \forall b_{ki} \in \{0,1\} \tag{3}$$

b_{ki} denotes the existence of the *ith* job in the *kth* window. The total time cost of the I/O jobs in the same window,

$$\Delta t^k_{I/O}(\alpha) = \sum_{j=1}^{n} a_{kj} t^{kj}_{I/O}(\alpha) \qquad \forall a_{kj} \in \{0,1\} \tag{4}$$

a_{kj} denotes the existence of the *jth* job, m defines the number of computational jobs, n defines the number of I/O jobs in the *kth* window. Δt^k_{CPU} and $\Delta t^k_{I/O}$ are dependent on the reserved memory partition, denoted by α, to cache the previous job results as illustrated in Figure 2.

$$T^k(\alpha) = \Delta t^k_{CPU}(\alpha) + \Delta t^k_{I/O}(\alpha) \tag{5}$$

T^k is the total time cost of the *kth* window.

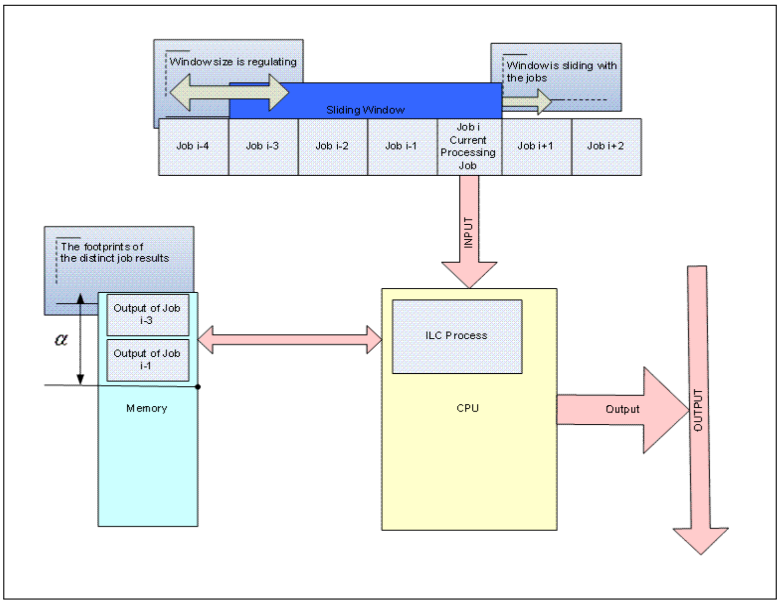


Figure 2: ILC-based caching scheme.

The proposed ILC-based dynamic caching scheme is illustrated in the Fig.2. ILC runs on the CPU as a memory allocator program and it selects some processed job results in the sliding window and caches them in the given partition of the memory. The sliding window size is adjusted in an optimal way to enable a performance criterion between forgetting the jobs with lower criteria and accepting the new jobs with higher selective criteria based on their recall or arrival frequency. In Fig. 3, selection scenario is illustrated where the window size is equal to 4, but the memory contains only 2 jobs which are selected based on the individual job's parameter. The results of the *(i-1)th* and *(i-3)th* computing job are selected and their results are cached in the memory to be used in their next recall. The selection parameter of the *(i-2)th* job is assumed to be lower compared to the cached ones. The selection parameter of the *(i+1)th* job is going to be calculated in the next sampling period while the selection window is sliding towards the new coming jobs.

Probability of caching the individual job result depends on both of the probability on the selection criterion and the individual job's occurrence in the selection window which are denoted by $p_i^{sc}(\tau)$ and $\widetilde{p}_i(\tau)$, respectively.

$$p_i(\tau) = p_i^{sc}(\tau)\,\widetilde{p}_i(\tau) \tag{6}$$

The selection probability, denoted by p_i^{sc}, is calculated for each individual job and it is ranked among the other jobs to decide about caching the most suitable job results in the memory. The selection probability is derived by,

$$p_i^{sc}(\tau) = e^{-(\frac{am_i\mu_i\Gamma\tau}{\lambda_i C})} \tag{7}$$

where $\Gamma = \sum_{i=1}^{\Omega} \frac{1}{\mu_i}\tau$ is the selection (sliding) window size, m_i is the size of the individual job output, μ_i is the individual job service rate provided by the CPU, Ω is the number of distinct jobs in the computing system, λ_i is the arrival rate of the *ith* job, C is the capacity of the memory and a is the scaling factor to assure that the selective criterion takes values between 0 and 1. The job parameters are assumed to be determined *a priori.*

A simple analysis may show that when the individual size of the job output, denoted by m_i, increases, the selection probability p_i^{sc} decreases. This leads to cache as many as job results instead of caching few larger results. When the individual service rate, μ_i, increases, p_i^{sc} decreases leading to the choice of caching larger service time requiring jobs results instead of caching the jobs that can be processed faster. The selection criterion may increase as the individual arrival rate λ_i increases leading to cache the jobs with higher occurrence probability to avoid frequent computation efforts. When the sliding window size, τ is increased, the caching selection criterion p_i^{sc} for the individual job may be decreased due to the augmented number of the distinct jobs in the selection window.

The ratio of the *ith* individual computational job's arrival rate versus the sum of the arrival rate of the other individual jobs existing in the sliding window is given,

$$N_i = \frac{\lambda_i}{\sum\limits_{j=1}^{\Omega} \lambda_j} \tag{8}$$

For the individual job in the window, the probability of being selected by the ILC-based caching scheme is normalized with respect to the total number of the individual computational job's arrival rate ratio, $p_i = \frac{N_i}{N_{Total}}$ where $\mathrm{N}_{Total} \equiv \sum\limits_{j=1}^{N} \mathrm{N_j}$ and N is the total number of the processed jobs *i.e.,* the sum of the distinct jobs and the reprocessed ones. When the specified individual job is not in the sliding window, it may not be selected by the proposed ILC scheme. Through straight-forward analysis, the probability of not being selected due to absence of the individual job in the window,

$$\overset{*}{p}_i = 1 - p_i = 1 - \frac{\mathrm{N_i}}{\mathrm{N}_{Total}} = \frac{\mathrm{N}_{Total} - \mathrm{N_i}}{\mathrm{N}_{Total}} \tag{9}$$

Since the number of the jobs whose results may be nominated for caching is equal to the size of the window denoted by τ, the probability of being absent in the window is given by,

$$p_i(\tau) = \prod_{k=1}^{\tau} \frac{\mathrm{N}_{Total} - \mathrm{N_i} - k + 1}{\mathrm{N}_{Total} - k + 1} \tag{10}$$

Gain of the job is defined as the time cost of CPU to process its output result. Time cost of getting from memory is ignored, because it is small and invariant (almost same time cost value for all jobs). The relative gain is important for the system, the proportion of the absolute gain of the individual job versus the sum of the absolute gains of the total jobs. The relative gain of the *ith* job in the window is defined by Ψ_i,

$$\Psi_i = \frac{\frac{1}{\mu_i}}{\Gamma} = \frac{1}{\mu_i \Gamma} \tag{11}$$

Total expected gain is obtained in terms of the average of all possible relative individual job gain weighted by its probability on existence in the sliding window and selection criterion, ([9]),

$$E(\Psi) = \sum_{i \ in \ \tau} \Psi_i \widetilde{p}_i(\tau) p_i^{sc}(\tau) \tag{12}$$

By using the derived probabilities given by and (7), (10) and (11), total expected gain for dynamic caching procedure may be given by,

$$E(\Psi) = \sum_{i=1}^{\Omega} \left(\frac{1}{\mu_i \Gamma} \left(\left(1 - \prod_{k=1}^{\tau} \frac{\mathrm{N}_{Total} - \mathrm{N}(i) - k + 1}{\mathrm{N}_{Total} - k + 1} \right) e^{-\left(\frac{am_i \mu_i \Gamma \tau}{\lambda_i C}\right)} \right) \right) \tag{13}$$

where Ω is the number of distinct jobs in the computing system. Total expected gain (13) is constituted by summing all the individual expected gain of the distinct jobs in the computing system. Towards ideal caching procedure, only the distinct job results are counted for the total expected gain, recalled and reprocessed jobs are not considered for the derivation of the expected gain.

The window size, denoted by τ, is regulated to maximize the expected gain of the sliding window by comparing the incoming jobs and caching the most profitable results for their possible future recalls which may lead to reduce computational efforts. The error variable is derived in terms of the attainable desired value, which is unknown *a priori* due to unknown job arrival, and the sum of the individual gains in the sliding window,

$$e_k = \frac{\max\limits_{\tau \in N}(E(\Psi)) - \sum_{i=k-\tau+1}^{k} \frac{\Psi_i p_i}{\tau}}{\max\limits_{\tau \in N}(E(\Psi))} = 1 - \frac{\sum_{i=k-\tau+1}^{k} \frac{\Psi_i p_i}{\tau}}{\max\limits_{\tau \in N}(E(\Psi))} \tag{14}$$

where N denotes the set of the previously arrived computing jobs, including the recalled and reprocessed ones, whose parameters such as occurrence frequencies, size, arrival rates may be learnt. The reference value is obtained by maximizing the derived total expected gain versus the sliding window size. The sliding window size is adjusted at each iteration step based on the previously processed and new arrival jobs to maximize the reference value for caching. The second term in the error variable is the sum of the individual job gains existing in the current sliding window standing for the actual output of the proposed caching system.

ILC methodology is based on the previously processed results and the job parameters such as their size in the memory, arrival and service rate, etc., and they will be used for the next memory scheduling process. The goal is to minimize the error and to regulate the window size to ensure caching the most suitable job results in the memory to avoid unnecessary computing efforts. To achieve this goal, the window dimension may be adjusted at each iteration step. The proposed ILC-based caching methodology is, (see for instance iterative learning regulator in [10] for different application)

$$\tau_{k+1} = \tau_k - sign\left(\frac{e_k - e_{k-1}}{\tau_k - \tau_{k-1}}\right) \tag{15}$$

Following (15) the window size is regulated at the next iteration based on the difference of two consecutive error terms and the window sizes denoted by e_k and τ_k at the kth iteration step, e_{k-1} and τ_{k-1} at the $(k-1)th$ iteration step, respectively. Ideally, the error variable in (14) may tend to zero if the given memory caching space is enough to cache all distinct job results.

4 Experimental Study

The proposed iterative learning based dynamic caching scheme is applied to the insurance policy server. This insurance company server executes more than 10.000 remote policy inquiries on a daily basis. To benefit from the individual retirement product, a customer must be more than 18 years old and 56 years old, and must be contributing to the insurance financial system more than 10 years. Most of the remote policy inquiring customers are between 18 and 46. About 350 customers are at the same age. At each age interval, half of them are male and female and they have to select their incomes and professions etc., in a limited set of the given query data. Many calculations are executed for each request coming from different individual customer having the similar input data.

In insurance and retirement enterprises area, the calculations of retention and premium need heavy actuarial computations. Like different types of mortality tables, disability, assurance, commission calculation tables, funds grow expectations and many other ones are used to perform these calculations. The tables are not changed frequently. The tables used for life insurance calculations are updated when death proportions or born rates are changed (i.e. a disaster or an epidemic disease occurs). Individual retirement tables are changed when enterprise's strategy is updated which may occur on a yearly basis, [11]. It can be assumed that for a life insurance policy with same input values always same premium is calculated. Aspect Oriented Programming

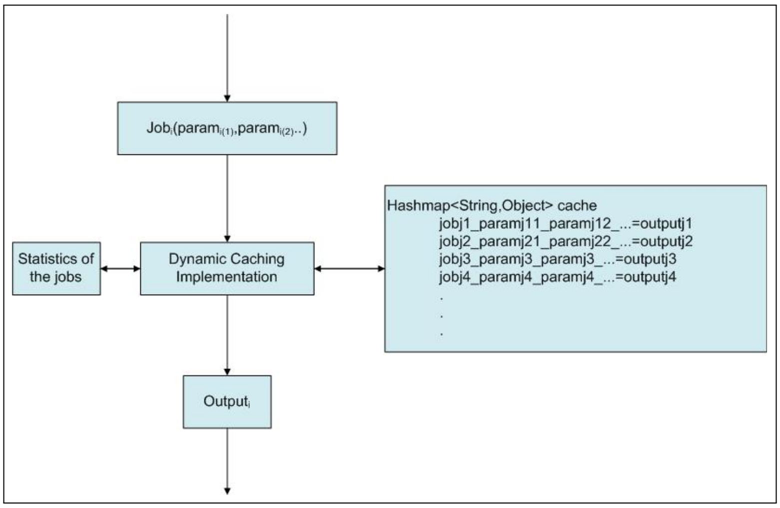
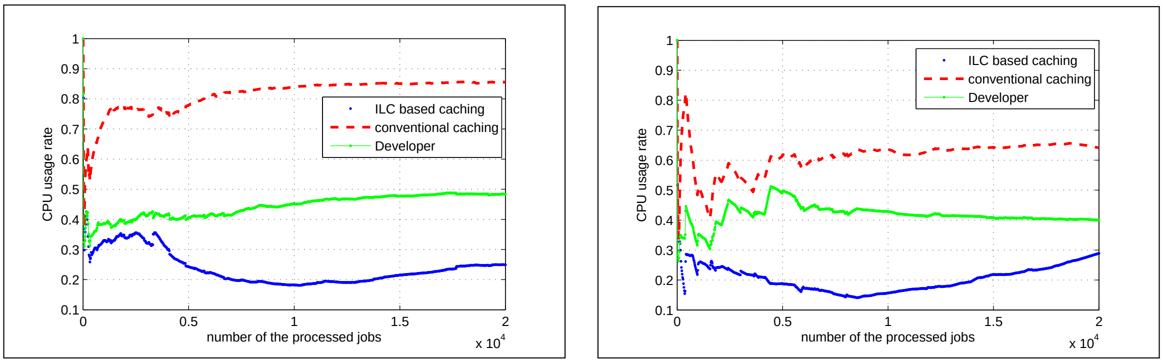


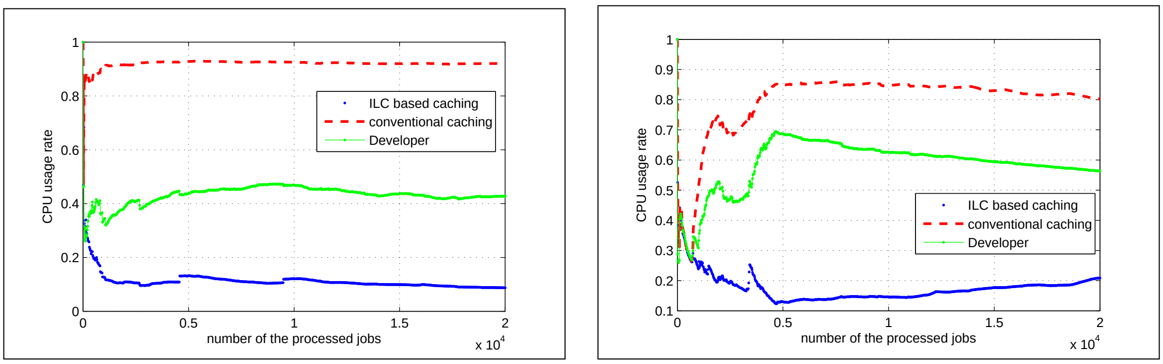
Figure 3: Implementation of process flow

(AOP) methodology is used to separate business logic and other programming requirements such as logging, transactions etc. AOP works on join points like method calls, object instantiations with advice types; before, after, around etc. And also AOP is used to identify the individual jobs and their similar results. The developer marks computational jobs by annotations or regular expressions as join points. Each join point has its own signature, when a method is called by same parameters the signature does not change. If the output of the previous job is cached, it is accessible by the signature of the job, (see for instance [12] or more information on AOP methodology). AOP has been designed to be implemented in many ways, including source-weaving or bytecode-weaving and directly in the virtual machine. Each method has its own signature including package and class name. In this study, a map is used to cache methods, key objects of the map are string implementation of the signature of the method concatenated with string implementation of the parameters (supposed that if and only if the parameters that have the same string implementation produce the same output). Implementation process is illustrated in Fig.3. The individual *ith* job is arrived with its parameters. The process searches whether the cache map has an object with string representation of the job (string representation consists of the signature of the method and the parameters separated by underscores). If the result has been obtained and cached for this string representation, the output is read from the cache and the parameters of the jobs are updated. Otherwise the job is processed by the CPU and the parameters are updated accordingly. In a real-time experiment, all of the jobs characteristics



(a) (b)

Figure 4: a) Process time of CPU to execute the incoming jobs on a real-time basis: The first day performance comparison of the CPU usage rate. b)The fifth day performance comparison.



(a) (b)

Figure 5: a) The tenth day performance comparison. b) The fifteenth day performance comparison.

are not known in advance. To overcome this situation, the system is designed as the average of the data characteristics cumulated in the previous days. The steady-state probabilities of the previous day distinct job parameters are used by the ILC-based caching scheme to calculate the overall job probabilities of the following day. In Fig.4 through Fig.5, CPU usage rate versus the large number of the jobs processed is plotted for different days. The performance of the proposed dynamic caching scheme is compared with other caching schemes. CPU usage rate responses are slightly changed dependent on the random arrival order of the remote policy inquiries. For

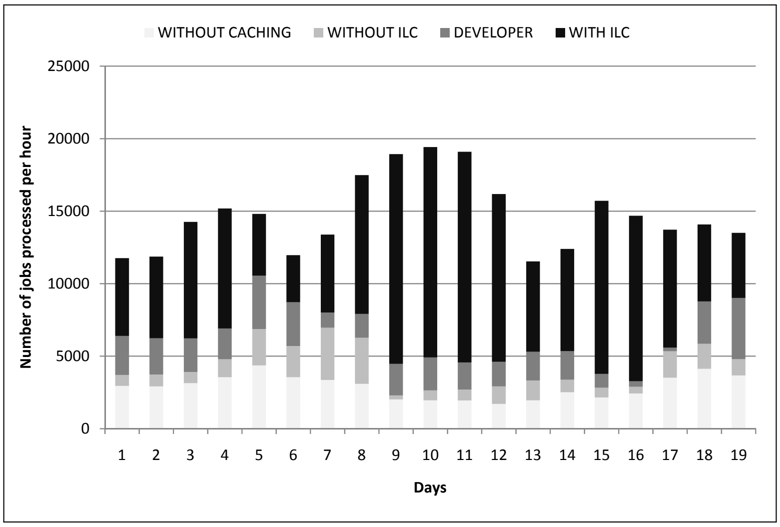


Figure 6: Comparison of the number of the jobs processed per hour for different caching schemes

each day, when CPU is enforced to process each individual job without considering even if their results are the same, CPU is loaded by %80 of its nominal capacity. When ILC is applied, the jobs are processed by using approximately 20% of the CPU computing capacity leading almost to %80 of free runtime. When the CPU has some free runtime, more jobs may be processed by

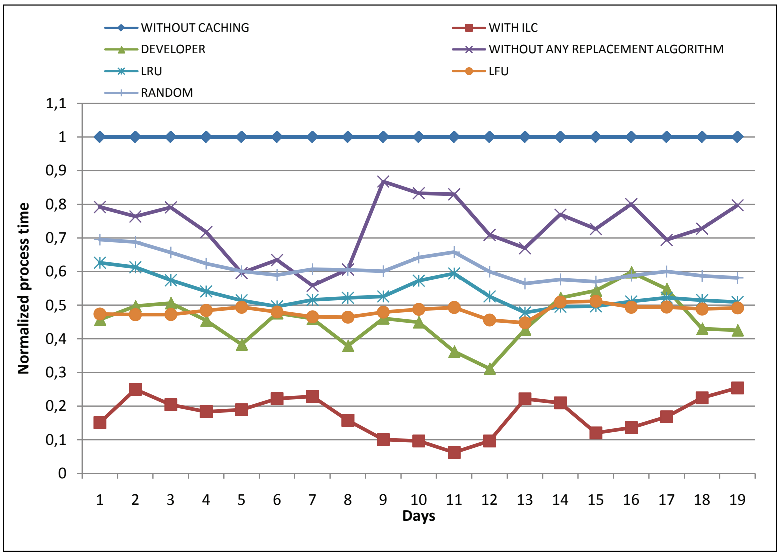


Figure 7: Comparison of the response time of the caching schemes

the proposed scheme.The number of the jobs that may be processed per hour by using the full CPU capacity is compared in Fig.6. For each day, ILC-based caching assures more jobs being processed per hour by utilizing the CPU's full capacity. Dark gray bars represent the number of the processed jobs by ILC-based caching. The lightest gray bar shows the number of the jobs processed without caching. When the first-in first-out caching scheme is applied, the number of the jobs processed is slightly increased, plotted with lighter gray bar. Even when the developer knows all parameters of the jobs, the job processing capability of the system per hour is less than the ILC-based system. The number of the processed jobs is significantly higher when the proposed learning and sliding window based caching algorithm is applied.

Different cache replacement algorithms such as random, least frequently used (LFU) or least recently used (LRU) are performed to compare with the introduced real-time ILC-based caching methodology. Different caching schemes react to the arrival job in a different manner such as when the computing job arrives, random algorithm chooses a random job result and replaces it in the cache, LRU chooses the least frequently arriving job or LRU chooses the job which has not been recalled for a long time. In Fig.7, processing time of the CPU to handle the incoming jobs per day is plotted for different caching systems. Without caching, the process time is considered to be the nominal and the process time of the other methodologies are normalized according to this nominal time.

The introduced ILC-based scheme, which avoids the unnecessary CPU computational efforts and permits the CPU react to the computational requests faster, is plotted with red-square line. The ILC-based system response time is compared with the system developer who is assumed to know all job characteristics, marks them and selects some of the results towards caching. The developer methodology time-response is around 40%, which is plotted with green-triangle line, and it is two times slower compared with ILC-based scheme. When random function is used for dynamic cache replacement method, process time is around 60% and it is plotted with solid and plus marked line. LRU, plotted with solid and x-marked line and LFU time responses, plotted with circle marked line, lead to respond two times slower compared to ILC-based caching, slightly slower than the developer's performance. The first-in first out caching methodology, which means without any replacement algorithm, causes fluctuations in the CPU response time and performs almost 20% faster at average compared to the case without caching.

5 Conclusion

In this paper, ILC-based caching scheme is introduced to enhance response time and processing capacity of the computing systems. The proposed scheme introduces the reference value for caching regulation. This reference value is introduced by summing possibly higher individual expected gains among the previously processed job results for the given window size. The average sum of the individual gain existing in the current window constitutes the actual output of the proposed caching system. Along the random arrival of the computational jobs, the size of the window helps to compare some amount of the computational jobs based on their parameters enabling a performance criterion about replacing the jobs with lower criteria by the jobs with higher selective criteria in the dedicated caching memory space. At each iteration step, the sliding window size is regulated to cache the most suitable job results in the memory based on the individual job selection probabilities. The sliding window size is incrementally increased and kept on inquiring the new arriving computational jobs until reaching to the operating point where the expected gain is maximized and the window size is adequately large to capture the most suitable job results subject to the given limited memory caching space.

To illustrate the effectiveness of the ILC-based caching system, the proposed dynamic caching scheme is applied to the insurance policy server. The proposed dynamic caching scheme is capable of processing more computing jobs for the given computing system capacity while the server being more responsiveness. By using the proposed autonomous caching and regulation methodology, the developers will not have to decide about the job results to be cached in the memory. Computing system performance, processing capacity and responsiveness to the new and unpredicted computational jobs will be autonomously improved without any expert intervention.

Bibliography

[1] Liu,C.L.; Layland, J.W. (1973); Scheduling Algorithms For Multiprogramming In A Hard-Real-Time Environment, *Journal of Association of Computing Machinery*, ISSN 1556-4665, 20(1): 46 - 61.

[2] Nock, C. (2003); *Data Access Patterns: Database Interactions In Object-Oriented Applications*,Addison Wesley.

[3] Ford, C. et al (2008) *Patterns For Performance And Operability – Building And Testing Enterprise Software*, Auerbach Publications.

[4] Padala, P. et al (2007), Adaptive Control of Virtualized Resources In Utility Computing Environments, *Proc. of the European Conference on Computer Systems*, Portugal:289-302.

[5] Kalyvianaki,E. et al (2009), Self-Adaptive And Self-Configured CPU Resource Provisioning For Virtualized Servers Using Kalman Filters, *Proc. of the 6th International Conference on Autonomic Computing*, New York, USA: 117-126.

[6] Ercan, M.; Acarman, T. (2010) Iterative Learning Control of Dynamic Memory Caching To Enhance Processing Performance On Java Platform, *International Conference on Computational Science*, Amsterdam, The Netherlands: 2664-2669.

[7] Butcher,M. et al (2008), A Statistical Analysis of Certain Iterative Learning Aontrol Algorithms,*International Journal of Control*, 81:156-166.

[8] Moon, J. et al (1997) An Iterative Learning Control Scheme For Manipulators, *Intelligent Robots and Systems*, 2:759-765.

[9] Pitman, J. (1999), *Probability*, Springer.

[10] Ahn, H. et al (2007), Iterative Learning Control: *Robustness And Monotonic Convergence For Interval Systems*, Springer.

[11] Booth, P. et al (2005), *Modern Actuarial Theory And Practice*, Chapman & Hall/CRC.

[12] Clarke, S.; Baniassad, E. (2005) *Aspect-Oriented Analysis And Design: The Theme Approach*, New Jersey: Pearson Education.

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):525-537, August, 2013.

Parkinson's Disease Prediction based on Multistate Markov Models

O. Geman, H. Costin

Oana Geman
Faculty of Electrical Engineering and Computer Science,
Stefan cel Mare University of Suceava
Development and Human Health Department
13, University St., 720229 Suceava, Romania
E-mail: geman@eed.usv.ro

Hariton Costin
1. Faculty of Medical Bioengineering,
Grigore T. Popa University of Medicine and Pharmacy, Iasi, Romania
2.Institute of Computer Science of Romanian Academy Iasi Branch, Romania
E-mail: hncostin@mail.umfiasi.ro

Abstract: In the real medical world, there are many symptoms or chronic diseases that cannot be characterized in a deterministic way, and which must be examined in a random way. In the study of these stochastic processes, Markov chains are used. There is a wide variety of phenomena that suggest a behavior in a Markov process manner such as: the probability that a patient's health to improve, to get worse, to remain stable or to progress to death within a certain time slot, depending on what happened in the previous time window. Our goal is to show that the Markov chains can be applied to the patients with Parkinson's disease in order to predict the evolution of the disease over time. So the doctor may decide a therapeutic solution that is adapted to the patient's needs, and that can improve the quality of the patient's life with Parkinson's disease in terminal stage.
Keywords: Parkinson's disease, Markov chains, Multistate Markov Models, Prediction

1 Introduction

Parkinson's disease (PD) is a neurodegenerative disease that occurs due to loss of dopamine that is a neurotransmitter and due to slow and inexorable destruction of neurons. Brain area affected by progressive destruction of neurons is responsible for movements controlling [1]. For this reason, patients with Parkinson's disease have rigid and uncontrollable gestures, postural instability, tremor, and speech disorders. Although Parkinson's disease is considered specific old age, the average age is 50 years and can be confused with the normal aging process of the individual [2]. When first symptoms are manifested, it is believed that between 60% and 80% of the cells for the control of motor activity are destroyed [3]. Parkinson's disease is a progressive disease, with signs and symptoms accumulated over time. Although this is potentially an invalidity disease, it progresses slowly so that most patients benefit from many years of active life after diagnosis. Moreover, unlike other serious neurological disorders, Parkinson's disease is treatable. Treatment is surgical or based on drugs, but may also consist of an implanted device for brain stimulation [4]. Worldwide, the disease is diagnosed in 300,000 people each year [5]. Disease incidence and prevalence increase with age. Parkinson's disease affects 1% of people aged over 65. Rarely, the disease occurs in childhood or adolescence. The incidence is 1.5 times higher among males than among women [6]. If Parkinson's disease would be detected in an early stage, the physician may interfere with a proper treatment in order to slow the disease's progression. Unfortunately, currently there is no screening test or biomarker that can be highlighted in Parkinson's disease. The three cardinal signs of Parkinson's disease are resting tremor, rigidity and bradykinesia. Among them, two are essential for diagnosis. Postural instability is the fourth cardinal sign, but occurs late, usually after 8 years of disease evolution. In 70% of cases, uncontrollable rhythmic gestures of the hands, head and feet are the first symptoms and occur mainly at rest and during the stress' periods (see [7]- [9]). Tremor is diminished during movements, disappears during sleep, and is exacerbated by stress and fatigue. Tremor becomes less evident as disease progression. This tremor, in the absence of other characteristic signs, indicates an early stage of disease or another diagnosis (see Table 1) [10]- [14].

Table 1: Neurological disorders characteristic signs

Moment	Speed	Location	Neurological Disorders
Rest tremor	4-6 Hz	arms, legs	Parkinson's disease
Postural tremor	7-12 Hz	hands	Essential tremor
Intention tremor	2-5 Hz	arms,legs	Cerebellar lesions

From the many symptoms or diseases that cannot be characterized in a deterministic way, but in a random way, PD is a prominent example. In our study, we used Markov chains as they characterize very well stochastic processes like diseases evolutions. So, our goal is to show that the Markov chains can be applied to the patients with PD in order both to predict the evolution of the disease over time, and illustrate the response to the specific treatment. In this way the doctor may decide upon a therapeutic solution that is adapted to the patient's needs, and can improve the quality of the patient's life in terminal stage.

2 Multistate Markov models

In mathematics, a Markov process is a stochastic process having the property that, given its present state, the future states are independent of the past. This property is called the Markov property [15]- [19]. In a Markov process, the system can change or keep its state, according to a certain probability distribution. Changes of its state are called transitions. A random experiment that consists of a series of random sub-experiments is called a stochastic process. Such a special class of these processes is made by the Markov chains [20]- [26].

The evolution of a Markov process can be described by a transition matrix. We can consider the evolution of the health status of a patient as a Markov process that passes through the following states: Well, Suspicious, Ill (PD), or Dead, as is illustrated in Figure 1. For the Markov process illustrated in Figure 1, we can write the general matrix (1), where m = 4 (possible mutually exclusive results: E1 well, E2 suspicious, E3 ill/PD, E4 dead).

$$P = \begin{bmatrix} p_{ww} \; ... \; 0 \\ ... \; ... \; ... \\ 0 \; ... \; p_{dd} \end{bmatrix} \tag{1}$$

As it can be seen, the transition matrix consists of p_{ij} elements, which represent the conditional probability that the system will change from the initial state (well) to next state j. The probability that the system remains in the same state after the experiment is given by p_{ij} with i = j, and the probability for the system to move from one state to another is given by p_{ij} with i≠j. The transition matrix for the proposed system is a square matrix of order m = 4. The elements of the transition matrix must satisfy the following properties [19]:

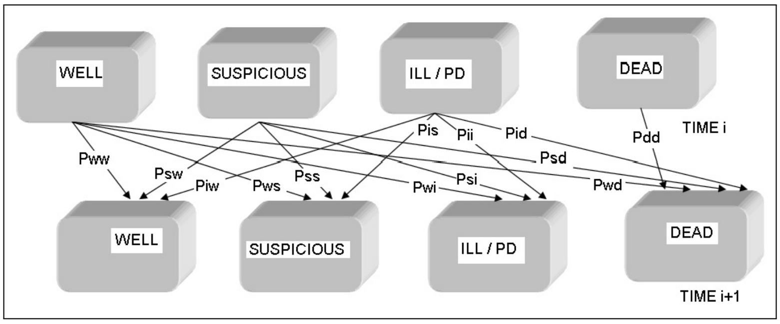


Figure 1: Four-state Markov Model for Parkinson's disease stage (well, suspicious, ill/PD, dead)

1. $0 \leqslant p_{ij} \leqslant 1, i, j = 1, ..., m$.
2. $\sum_{j=1}^{m} p_{ij} = 1, i = 1, 2, ..., m$. The sum of the elements of each line must be 1 because $E_1, ..., E_m$ is a complete system of events.
3. $p_{dd} = 1$, for our application.

Information about the transitions from one state to another in a Markov chain can be represented by a transition matrix. It consists of elements p_{ij} - probability of crossing a step from state i to state j (i,j = 1,..., m, where m=4). We can talk about the transition probability of exactly k steps and a matrix formed by them. So, multistate Markov models in continuous time may be used to model the course of Parkinson's diseases. Since Markov chains are stochastic processes, we cannot know exactly what it is happening on each state, so the system must be described in terms of probability.

Definition 1. [19]: Consider a Markov chain with m states. A state vector for Markov chain is a probability vector $X = [x_1, x_2, ..., x_m]$. The x_i coordinates of the state vector X should be interpreted as the probability that the system be in the state i.

The behavior of a Markov chain can be described by a sequence of state vectors. The initial state of the system can be described by a state vector noted X_0. After a transition, the system can be described by a vector X_1 and after k transitions the system is described by the state vector X_k. The relationship between these vectors can be summarized by the following theorem [19]:

Consider a Markov process with the transition matrix P. If X_k and X_{k+1} are vectors that describe a process state after k and k +1 transitions respectively, then $X_{k+1} = X_k * P$.

We represent structural elements as a vector $S = [s_t^1, ..., s_t^i, ..., s_t^m]$, that for each t = 1,...,n and for each i = 1,...,m, s_t^i varies between 0 and 1, and the sum of structural elements is 1 for any t. In order to model a Markov process, we must respect the following steps [19]:

1. First-order differences of the vector S_t will be calculated, thus $\Delta S_{t/t-1} = S_t - S_{t-1}$.
2. For each pair $t/t-1$ of consecutive periods of time we will buid the partial transition matrices (MTP), as $MTP_{t/t-1}(m*m)$ form. The elements of the $MTP_{t/t-1}(m*m)$ matrix can be determined as follows:

$$MTP_{t/t-1}^{ij} = min(s_{t-1}^i, s_t^j) if i = j \tag{2}$$

$$MTP_{t/t-1}^{ij} = \left| \Delta s_{t/t-1}^j * \frac{\Delta s_{t/t-1}^j}{\sum_{i=2}^{m}(+\Delta s_{t/t-1}^{ij})} \right|, \tag{3}$$

if i $\neq j$ and $\Delta s^i_{t/t-1} < 0$ and $\Delta s^j_{t/t-1} > 0$.

$$MTP^{ij}_{t/t-1} = 0, \tag{4}$$

for the other elements, where i,j =1,...,m.

In formula (3) the expression $\sum_{i=2}^{m}(+\Delta s^{ij}_{t/t-1})$ denotes the sum of positive values of the difference vector $\Delta s_{t/t-1}$.

3. $MTP(m*m)$, total transition matrix is determined by summing the elements of partial transition matrixes.

4. $MP(m*m)$, transition probability matrix is calculated by ratio between each element of the total transition matrix and the sum of the line on which is located than item.

5. In the final stage of the algorithm, we obtain a forecast of the structural elements for future p periods by multiplying transposed of the matrix $MP(m*m)$ raised to the k power with the vector of structural elements for the last period.

3 Intelligent system for health status prediction using a Markov chain

The architecture of the proposed system is shown in Figure 2. It consists from three modules. The first module will handle with the signal acquisition from patients suspected of Parkinson disease. In terms of software, this module is a software application that can acquire biomedical signals from Wii^{TM} Remote device or other devices that can acquire signals generated by tremor. All data acquired from these devices are analyzed using the method presented in Section 2. Furthermore, the data are saved on a server. On this server, physicians can access data in order to establish a long history of patient evolution.

The second module of this system is represented by the extracting knowledge from biomedical signals acquired from the patients. This module consists of a software application that runs on the server where there are kept biomedical signals acquired. The third module is the application that is executed in the doctor's office. This application performs an interfacing of the doctor with the intelligent system, and presents the medical treatment and rehabilitation options. It must be said that bio-signals can be acquired in the doctor's office but also at home if the patient has a PC and an internet connection. The design and development of this intelligent system used the newest technologies for distributed application development (WCF, SOAP), and the observations received from patients and specialists.

3.1 Database

For the database we used the proposed methodology in previous papers [27], [28]. Database with affected patients has been provided by Suceava Emergency Hospital (Neurology Clinic).

This dataset is composed of a range of biomedical tremor measurements from 88 people, 28 with Parkinson's disease (PD), 30 "normal" tremor and 30 "suspicious" PD (undiagnosed). Each column in the table is a particular tremor measure, and each row corresponds one of 2500 tremor recordings from these individuals ("name" column). The main aim of the data is to discriminate healthy people from those with PD, according to "status" column which is set to 0 for healthy and 1 for PD or "Suspicious". All patients are suffering of moderate to severe postural tremor. This postural tremor cannot be differentiated on clinical features (frequency,

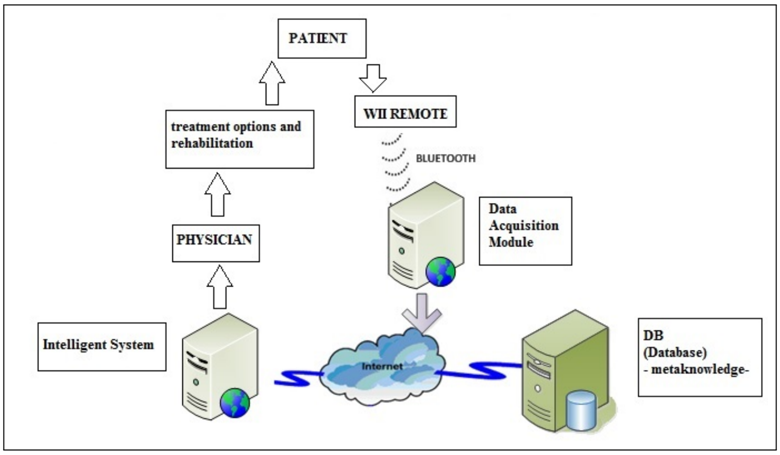


Figure 2: Intelligent system for health status prediction of a patient using a Markov chain

Table 2: Data: size, age, gender, and disease duration distribution of PD, SPD, and NT subjects

	PD	SPD	NT
Number of patients	28	30	30
Mean age	64.54	63.24	64.52
(range in years)	(40-90)	(27-94)	(24-86)
Gender (male/female)	18/10	16/8	19/11
Mean disease duration	16,4	5,3	

amplitude). Patients were kept under observation and investigation for 2 years, and data were acquired at 6 months, 1 year and 2 years (see Table 2).

The mean disease duration (time for disease to install, in years), age and sex of PD patients were compared with the SPD or NT in Table 2. Notice in Table 2 that the mean age of PD, SPD and NT populations is similar, but the age ranges are different. This could be considered as an indicator that the PD starts years before actual diagnosis.

3.2 Tremor recoding

Yet, some researches have been made (including in Romania) in order to early diagnose the PD and its progress by means of the tremor or the gait analysis or other symptoms [29]- [34]. The tremor time series were acquired using an accelerometer sensor from a Wii^{TM} console [35], connected via Bluetooth to a PC. The data were analyzed using an application implemented in Visual C 2010 Professional. The Wii^{TM} Remote is the primary controller for Nintendo's Wii^{TM}i console. A main feature of the Wii^{TM} Remote is its motion sensing capability, which allows the user to interact with and manipulate items on screen through the use of accelerometer and optical sensor technology [35]. Nintendo works on three axes: x - lateral, y - anteroposterior, and z - vertical. The device records both acceleration induced by hand movement and the component of gravitational force. If the controller is rotated, the gravity accelerometer affects the values on the x, y, and z axes (see Figure 3).

This system using a Wii^{TM} Remote is capable of analyzing frequency and estimated amplitude of tremor between 3 - 15 Hz (N tremor is between 5 - 12 Hz, and PD tremor is between 4-6 Hz). The Wii^{TM}i Remote and PC are connected by Bluetooth - Human Interface Device Profile. The tremor analysis program was developed using Visual C 2010 Professional. The acceleration sampling period was set at 10 ms in the Nintendo device. Because the transmission rate through

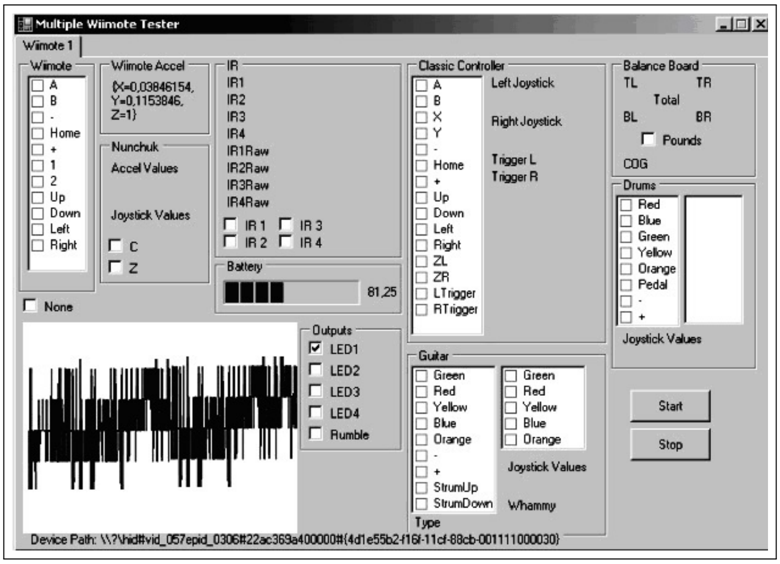


Figure 3: Interactive GUI using Wii^{TM} Remote (tremor application)

the Bluetooth device is limited, the sampling period of the tremor analysis was 40 ms. The accelerometer built into Wii^{TM} Remote (Nintendo) measures gravitational and non-gravitational acceleration. The results of this paper suggest that Nintendo is useful for measurement and analysis of tremor using the methodologies described in [28], [29], [31]. We defined the following linguistic variables (for instance for X axis):

- If x is between -0.10 mm and -1 mm then x is minimum X_{min};
- If x is between -0.10 mm and 0.10 mm then x is medium X_{med};
- If x is between 0.10 mm and 1 mm then x is maximum X_{max}.

We counted the number of spikes for each interval, and we used these values to describe the state vector.

Next we proposed to predict the state of a patient using Markov chains. In this analysis the state vector is defined as:

$$S = X_{min}, X_{med}, X_{high}, Y_{min}, Y_{med}, Y_{high}, Z_{min}, Z_{med}, Z_{high} \tag{5}$$

. Table 3 presents the number of spikes in each category for "normal" subjects, while Table 4 presents the values of the state vector for a subject with diagnosed Parkinson's disease.

For this paper we chose to exemplify the calculation of transition matrices from T0 to T1 and from T1 to T2 only for patients with Parkinson's disease, by following the methodology presented in Section 2 (here notations Ti were used instead of ti and we illustrated the method only for PD patients and normal patients).

In the first step we computed, according to the methodology, the deviations $\Delta S_{T1/T0} = S_{T1} - S_{T0}$ and $\Delta S_{T2/T1} = S_{T2} - S_{T1}$. We illustrate this in Tables 5 and 6 only with data acquired from a patient with PD.

We computed next the transition matrices from T0 to T1 and from T1 to T2 (Tables 7 and 8, respectively). For example, the transition matrix from T0 to T1, $MTP_{T1/T0}(m*m)$ is computed as follows:

1. the elements from the main diagonal are (S^i_{T1}, S^i_{T0});

Table 3: "Normal" subject vector, spikes number at T0, T1 and T2 for 60 seconds each record

Features Vector	T0	T1=6 months after T0	T2=12 months after T0	Total spikes
X_{min}	284	257	286	827
X_{med}	1524	1458	1511	4511
X_{max}	651	687	558	1896
Y_{min}	1289	1439	1435	4163
Y_{med}	1283	1247	1257	3787
Y_{max}	664	657	557	1878
Z_{min}	392	382	378	1152
Z_{med}	768	865	789	2422
Z_{max}	2031	1998	1875	5904

Table 4: Data: size, age, gender, and disease duration distribution of PD, SPD, and NT subjects

Features Vector	T0	T1=6 months after T0	T2=12 months after T0	Total spikes
X_{min}	382	358	379	1119
X_{med}	785	758	688	2231
X_{max}	897	857	912	2666
Y_{min}	578	547	524	1649
Y_{med}	457	479	487	1423
Y_{max}	354	349	357	1060
Z_{min}	257	282	253	792
Z_{med}	578	549	754	1881
Z_{max}	1300	1329	1348	3977

Table 5: The deviations $\Delta S_{T1/T0} = S_{T1} - S_{T0}$ (PD patient) T1 vs. T0, for the state vector S

Time	X_{min}	X_{med}	X_{max}	Y_{min}	Y_{med}	Y_{max}	Z_{min}	Z_{med}	Z_{max}	SUM
$T1$	358	758	857	547	479	349	282	549	1329	
$T0$	382	785	897	578	457	354	257	578	1300	
Deviation	-24	-27	-40	-31	22	-5	25	-29	29	
Deviation+					22		25	549	29	76

Table 6: The deviations $\Delta S_{T1/T0} = S_{T1} - S_{T0}$ (PD patient) T2 vs. T1, for the state vector S

Time	X_{min}	X_{med}	X_{max}	Y_{min}	Y_{med}	Y_{max}	Z_{min}	Z_{med}	Z_{max}	SUM
$T2$	379	688	912	524	487	357	253	754	1348	
$T1$	358	758	857	547	479	349	282	549	1329	
Deviation	21	-70	55	-23	8	8	-29	205	19	
Deviation+	21		55		8	8		205	19	316

2. if $i \neq j$, $\Delta S^{i}_{T1/To} < 0$ and $\Delta S^{j}_{T1/To} > 0$, so the matrix equals the absolute value of $\Delta S^{i}_{T1/T0} * \frac{\Delta S^{j}_{T1/T0}}{\sum \Delta S^{ij}_{T1/T0} > 0}$;

3. the rest of elements equals 0.

Table 7: The transition matrix from T0 to T1

Features Vector	Xmin	Xmed	Xmax	Ymin	Ymed	Ymax	Zmin	Zmed	Zmax
Xmin	358	0	0	0	0	0	0	0	0
Xmed	2.548	758	1.625	10.244	0	0	4.345	0	0
Xmax	0	0	857	0	0	0	0	0	0
Ymin	0	0	4.548	547	0	0	0	0	0
Ymed	7.413	0	1.021	12.547	479	0	0	12.457	0
Ymax	1.124	0	1.245	6.333	0	349	0	1.125	0
Zmin	1.354	0	0	6.687	0	0	257	2.548	0
Zmed	0	0	0	0	0	0	0	549	0
Zmax	4.211	0	0	24.442	0	0	0	8.457	1.300

Table 8: The transition matrix from T1 to T2

Features Vector	Xmin	Xmed	Xmax	Ymin	Ymed	Ymax	Zmin	Zmed	Zmax
Xmin	358	0	2.387	0	0	0	0	0	0
Xmed	0	688	0	6.257	0	0	4.345	6.211	0
Xmax	1.250	0	857	0	4.587	0	0	0	5.244
Ymin	0	0	1.287	524	0	0	0	0	0
Ymed	0	5.687	1.021	8.985	479	0	0	6.258	0
Ymax	1.124	0	1.245	0	0	349	0	1.125	0
Zmin	1.354	0	0	6.154	5.698	2.542	257	2.548	0
Zmed	0	0	0	0	0	0	0	549	2.241
Zmax	4.211	0	0	3.587	2.587	2.325	0	9.237	1.329

In the third step we calculated the total transition matrix (Table 9), which is the sum of partial transition matrices computed in the previous stage.

In the fourth stage we computed the probability transition matrix by the ratio of each element of the total transition matrix to the sum of the line where the element is located.

In the final stage of the algorithm we obtained the forecast of the structural elements for next year by multiplying the transposed matrix of transition probabilities with the vector of the structural elements for T2, i.e. the vector corresponding to T2 = 12 months. We get the following transition probabilities between the 9 elements of the features vector $X_{min}, ..., Z_{max}$.

The values of the main diagonal are the probabilities that the patient progress to state that is described by the features vector (which corresponds to a stage of the disease). The forecast of the $X_{min}...Z_{max}$ for the next year is obtained by multiplying the two matrices (transposed and

Table 9: The total transition matrix (in %).

Features Vector	Xmin	Xmed	Xmax	Ymin	Ymed	Ymax	Zmin	Zmed	Zmax	Total %
Xmin	100	0	2.387	0	0	0	0	0	0	100
Xmed	9.59	90	0.0005	0.0051	0	0	0	0.0015	0	100
Xmax	8.54	0	91.46	0	0	0	0	0	0	100
Ymin	1.07	0	0	98.93	0	0	0	0	0	100
Ymed	12.91	0	0.17	1.69	84.71	0	0	0.5	0	100
Ymax	7.96	0	0.0004	0.47	0	91.37	0	0.14	0	100
Zmin	8.94	0	0.0009	0.90	0	0	89.79	0.26	0	100
Zmed	8.1	0	0	0	0	0	0	91.9	0	100
Zmax	15.19	0	0.0008	0.85	0	0	0	0.25	83.6	100

elements for T2). Thus we obtain the patient's evolution for next year, for "normal" and "PD" (Table 10 and Table 11).

Table 10: The "normal" subject's evolution for the next year (no. of spikes)

Features Vector	T0	T1=6 months after T0	T2=12 months after T0	T3=24 months sfter T0 (with Markov chain)	T4=24 months after T0(recorded)
X_{min}	244	257	286	295	299
X_{med}	1442	1458	1511	1657	1656
X_{max}	651	687	688	689	694
Y_{min}	1412	1439	1442	1420	1421
Y_{med}	1233	1247	1257	1243	1240
Y_{max}	614	627	665	688	686
Z_{min}	392	399	410	412	414
Z_{med}	768	788	789	786	785
Z_{max}	1992	1998	1999	1995	1994

From the last two tables one can see, by using Markov chains, the tremor symptom evolution of certain patients. Also we may note the very good prediction power of this method, as the features vector elements for the predicted tremor signal after 24 months from the first recording are very similar with the same vector elements, but acquired and measured by means of Wii^{TM} Remote and the appropriate software. The maximum error between prediction and measured values was 1.33%.

Similar judgement was used and corresponding good results concerning the prediction of disease evolution were obtained in the case of "suspicious PD" patients, for whom some early signs were found (insomnia, constipation, loss of smell, equilibrium and postural impairment, tremor symptom or speech difficulties) and they became to be attentively monitored. Also, another remark may be made related to the similarity between features vectors measured for "suspicious PD" patients and "diagnosed PD" patients, when using the same Markov chains for status prediction.

Table 11: The PD patient's evolution for the next year (no. of spikes)

Features Vector	T0	T1=6 months	T2=12 months	T3=24 months sfter T0 (with Markov chain)	T4=24 months after T0(recorded)
X_{min}	382	385	396	398	399
X_{med}	785	792	784	796	795
X_{max}	837	857	912	944	946
Y_{min}	518	537	544	586	585
Y_{med}	457	459	467	489	488
Y_{max}	354	359	373	382	380
Z_{min}	257	282	278	310	308
Z_{med}	528	549	558	568	566
Z_{max}	1300	1329	1348	1399	1398

4 Conclusions

In this paper we describe a general purpose model of PD prognosis based on Markov process and show how this simple mathematical tool may be used to generate detailed and accurate assessments of Parkinson's disease stage and therefore may be applicable in medical screening for PD. Markov models consider a patient to be in one of a finite number of discrete states of health. All clinically important events are modeled as transitions from one state to another. Thus, the use of Markov models has the potential to allow the development of decision models that more faithfully represent clinical problems.

Our study used a database where there are subjects who are considered normal, but with some tremor symptoms, and subjects considered "suspects", for whom we can apply the above methodology and can see if certain subjects move to the "normal" state or the first symptoms of Parkinson's disease will appear. Thus, medical staff can intervene with specific medication for Parkinson's disease.

Using Markov chain is an efficient way to find the features vector for an individual patient at a given time, and this state vector may be used to predict and identify a stage in Parkinson' disease. So, the physician can choose a treatment, based on this forecast with an appropriate level of medication.

The system was validated for 88 patients under observation: 28 with PD tremors, 30 with SPD ("Suspicious" PD tremor), and 30 with NT (Normal tremor), and we plan to expand the study to more patients with PD. Already results interpretation and discussions with involved neurologists are directed to the validation of the study. The next step will be the creation of an expert or decision-support system based on fuzzy logic for Parkinson's disease screening, which will help a physician to diagnose PD in its early stages, especially of individuals in the class "Suspicious" of PD. So, future research approaches will include the testing and validation of a screening test, in order to detect Parkinson's disease or other neurological disorders in their early stages.

Acknowledgment

The authors would like to thank Radu Vasilcu, MD, and Prof. Mihai Covasa, PhD, MD, for their help in providing data for PD patients and for clinical evaluation of results during experiments.

Important notice: the experiments were not prejudicial in any way to the health of human subjects investigated and they were not subject to any invasive maneuvers. All the subjects were free to decide whether or not they wish to participate in this study. They did not lose any benefits to which they are entitled, if they did not accept the participation. The duration of this study was 3 years. All personal information was and will be kept confidential. Medical information may be made available to the institution that houses the research, Ethics Commission, or other persons/institutions where the law requires. The benefits will be strictly medical. The information obtained in this study may help physicians to find a method of early diagnosis for those suffering from PD and to identify the best options for their treatment. It was no financial compensation during the study for the participants.

Bibliography

[1] National Parkinson Foundation, http://www.parkinson.org (2013).

[2] EuroPa, http://www.europarkinson.net (2013).

[3] National Institute of Neurological Disorders and Stroke, http://www.ninds.nih.gov/ (2013).

[4] Helmich, R.C.; Hallett, M.; Deuschl, G.; Toni, I.; Bloem, B.R. (2012); Cerebral Causes and Consequences of Parkinsonian Resting Tremor, *Brain*, 135(11): 3206-3226.

[5] [Abdo, W.F.; Van de Warrenburg, B.P.; Quinn, N.P.; Bloem, B.R. (2010); The clinical approach to movement disorders, *Nat. Rev. Neurol.*, 6: 29-37.

[6] Benninger, D.H.; Thees, S.; Kollias, S.S.; Bassetti, C.L.; Waldvogel, D. (2009); Morfological differences in Parkinson's disease with and without rest tremor, J. Neurol., 256: 256-263.

[7] Lees, A.J.; Hardy, J.; Revesz, T. (2009); Parkinson's disease, *Lancet*, 373: 2055-2066.

[8] Moran, R.J.; Mallet, N.; Litvak, V.; Dolan, R.J.; Magill, P.J.; Friston, K.J.; Brown, P. (2011); Alteration in brain connectivity underlying beta oscillations in Parkinsonism, *Plos. Comput. Biol.*, 7: 102-124.

[9] Mure, H.; Hirano, S.; Tang, C.C.; Isaias, I.U.; Antonini, A.; Ma, Y.; Dhawan, V.; Eidelberg, D. (2011); Parkinson's disease tremor-related metabolic network : characterization, progression, and treatment effects, *Neuroimage*, 54: 1244-1253.

[10] Shtilbans, A.; Henchcliffe, C. (2012); Biomarkers in Parkinson's Disease, *Curr. Opin. Neurol.*, 25(4): 460-465.

[11] Dorsey, E.R.; Constantinescu, R.; Thomson, R.P.; et al. (2007); Projected number of people with Parkinson's disease in most populous nations, 2005 through 2030, *Neurology*, 68: 384-386.

[12] Chahine, L.M.; Stern, M.B. (2012); Diagnostic markers for Parkinson's disease, *Curr. Opin. Neurol.*, 24: 309-317.

[13] Stern, M.B.; Siderowf, A. (2010); Parkinson's at risk syndrome: can Parkinson's disease be predicted?, *Mov. Disord.*, 25: 89-93.

[14] Beck, J.R.; Pauker, S.G. (1983); The Markov Process in Medical Prognosis, *Med. Decis. Making*, 3(4): 419-435.

[15] Magni, P., Quaglini, S.; Marchetti, M.; Barosi, G. (2000); Deciding when to intervene: a Markov decision process approach, *International Journal of Medical Informatics*, 60: 237-253.

[16] Jackson, C.H.; Duffy, S.W.; Couto, Elisabeth (2003); Multistate Markov models for disease progression with classification error, *The Statistician*, 52(2): 193-209.

[17] Mackel, T.; Rosen, J.;Pugh, C. (2007); Application of Hiden Markov Modeling to Objective Medical Skill Evaluation, *Medicine Meets Virtual Reality*, 15: 316-318.

[18] Rosen J., Chang, L.; Brown, J.; Sinanan, M.; Hannaford, B. (2006); Generalized Approach for modeling Minimally Invasive Surgery as a Stochastic Process Using a Discrete Markov Model, *IEEE Transaction on Biomedical Engineering*, 53: 399-413.

[19] Rabiner, L. (1989); A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Proceedings of the IEEE*, 77(2).

[20] Sonnenberg, F.A.; Beck, J.R. (1993)., Markov models in medical decision making: a practical guide, *Med. Decis. Making*, 13: 322-338.

[21] Bricq, S.; Collet, C.; Armspach, J.P. (2008); MS Lesion Segmentation based on Hidden Markov Chain, pp.2-6.

[22] Key, R. (1986); A Markov model for analyzing cancer markers and disease states in survival studies, *Biometrics*, 42: 855-865.

[23] LeStrat, Y.; Carrat, F. (1999); Monitoring epidemiologic surveillance data using hidden Markov models, *Statistics in Medicine*, 18: 3463-3478.

[24] Zipkin, E.F.; Jennelle, C.S.; Cooch, E.G. (2010); A primer on the application of Markov chains to the study of wildlife disease dynamics, *Methods in Ecology & Evolution*, 1: 192-198.

[25] Craig, B.A.; Sendi, P.P. (2002); Estimation of the transition matrix of a discrete-time Markov chain, *Health Economics*, 11: 33-42.

[26] Faddy, M.J.; McClean, S.I. (2005); Markov Chain Modelling for Geriatric Patient Care, *Methods Inf. Med.*, 44: 369-373.

[27] Geman, O. (2011); Data Processing for Parkinson's Disease Tremor, Speech and Gait Signal Analysis, *EHB'2011, IEEE E-Health and Bioengineering Conference*, ISBN: 978-1-4577-0292-1, pp. 53-57.

[28] Pohoata, S.; Geman, Oana; Graur, A. (2012); Dual tasking: gait and tremor in Parkinson's disease acquisition, processing and clustering, *Proc. of the National Symposium of Theoretical Electrical Engineering, SNET 2012*.

[29] Marimorita, M.; Iizuka, T.; Takeuchi, A.; Shirataka, M.; Ikeda, N. (2009), Development of a System for Measurement and Analysis of Tremor Using a Three-axis Accelerometer, *Methods Inf. Med.*, 6: 589-594.

[30] Nicolau, V.; Andrei, M. (2012); Estimation Aspects of Signal Spectral Components Using Neural Networks, *Book Series: Advances in Intelligent Systems and Computing, Book: Soft Computing Applications*, 195: 521-530.

[31] Clark, R.A.; Bryant, A.L.; Pua, Y.; McCrory, P.; Bennell, K.; Hunt, M. (2010); Validity and reliability of Nintendo Wii^{TM}i Balance Board for assessment of standing balance, *Gait & Posture*, 31: 307-310.

[32] Dobrea, D.M.; Sîrbu, A. (2007); Movement and Postural Noncontact Bioinstrumental Sensor a DSP Implementation, *Proceeding of the 8-th International Symposium on Signals, Circuits and Systems*; pp. 369-372.

[33] Geman, Oana; Turcu, C.O.; Graur, A. (2013); Parkinson's disease Screening Tools using a Fuzzy Expert System, *Advances in Electrical and Computer Engineering*, 13(1): 41-46.

[34] Ungurean, I., Gaitan, C. (2012); Speech Analysis for Medical Predictions based on Cell Broadband Engine, 20th European Signal Processing Conference (EUSIPCO 2012), pp. 1733-1736.

[35] Nintendo Wii^{TM}, http://www.nintendo.co.uk/

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):538-547, August, 2013.

Enlarging the Domain of Attraction in Nonlinear Polynomial Systems

F. Hamidi, H. Jerbi, W. Aggoune, M. Djemaï, M.N. Abdelkrim

Faiçal Hamidi, Mohamed Naceur Abdelkrim
Unité de Recherche Modélisation,
Analyse et Commande des Systèmes, ENIG, 6029 Gabes, Tunisie.
Email: faicalhamidi@yahoo.fr, Naceur.Abdelkrim@enig.rnu.tn

Houcem Jerbi
University of Hail, College of Engineering
P.O.Box 2440, Hail, Kingdom of Saudi Arabia.
Email: h.jerbi@uoh.edu.sa

Wahiba Aggoune
ECS-Lab, Equipe Commande des Systèmes,
ENSEA,6 avenue du Ponceau, 95014 Cergy, France.
Email: aggoune@ensea.fr

Mohamed Djemaï
Univ. Lille Nord de France, F-59000 Lille,
France. UVHC, LAMIH, CNRS, UMR 8201,
Campus du Mont Houy F-59313, Valenciennes, France.
Email: mohamed.djemai@univ-valenciennes.fr

Abstract: This paper addresses the problem of enlarging the Domain of Attraction **(DA)** based on a Generalized Eigenvalue Problem **(GEVP)** approach. The main contribution is the maximization of the **(DA)** while characterizing the asymptotic stability region by a Lyapunov Function. Such result is obtained using a Genetic Algorithm **(GA)**. A theoretical proof of the validity of the obtained domain is developed. An illustrative example ends the paper.
Keywords: Nonlinear Polynomial Systems, LMI, Genetic Algorithm, Stability.

1 Introduction

The problem of enlarging the Domain of Attractions **(DA)** has been the topic of an important number of research works (see for example [2] [4], [5], [6] [11], [13] and the references cited therein). The **DA** is defined as the set of initial conditions from which the states converge to the asymptotically stable equilibrium point [7]. As a result, it is essential to identify the shape of this region whenever one has to study the stability of a system. For this purpose, we exploit the fundamental theory of Lyapunov stability (see [3], [12]). Indeed, for a particular Lyapunov function, the largest estimated region of asymptotic stability can be defined as the largest level set of the Lyapunov function included in the region where its derivative is negative. In a recent work [3], the author proposes a static nonlinear feedback input, which allows enlarging the **DA**. The proposed controller is polynomial in the measurable output; it exploits relaxations based on the sum of squares of polynomials in order to prove that the lower bound of the maximum achievable largest estimated domain of attraction and a corresponding controller can be computed via a generalized eigenvalue problem. The main advantage of the methodology is that the problem is formulated as a quasi-convex Linear Matrix Inequalities **(LMI)** ([1], [3], [10]).

The main purpose of this work is to develop an exact method allowing the maximization of the **DA**. The objective of this work is to improve the approach adopted in [3] by combining the

Genetic Algorithm **(GA)** as an advanced optimization strategy to the LMI technique in order to maximize the **DA**. The parameter optimizing approach will simultaneously deal with the Lyapunov Function and the control input parameters. Based on the Reverse Trajectory Method **(RTM)** one can accurately determine a preliminary maximal region of asymptotic stability and thereafter define the parameter of the maximal Lyapunov Function ([8], [9], [12]). This allows to define precisely the initial values and the constraints related to the required parameters of the investigated Lyapunov function.

2 Problem Statement and Notations

Consider the continuous-time polynomial system

$$\dot{x} = f(x) + g(x)u; \qquad y = h(\tilde{x}) \tag{1}$$

where $f(.), g(.)$ and $h(.)$ are polynomial functions such that $f(0) = 0$, $g(0) = 0$ and $h(0) = 0$. (The equilibrium point is the origin). In what follows, we assume that $x \in R^n$ is the state vector, $u \in R^p$ is the input vector and $y \in R^q$ is the measurable output. The control input is supposed to be a polynomial function of the form

$$u = U\phi\left(h\left(\tilde{x}\right)\right) \tag{2}$$

where $\phi(.)$ is a given polynomial function of the output and $h\left(\tilde{x}\right)$ and u are defined by :

$$h\left(\tilde{x}\right) = \left[\tilde{x}^{[1]}, \tilde{x}^{[2]}, \cdots, \tilde{x}^{[q]}\right]; \qquad u = \sum_{i=1}^{q} u_i \tilde{x}^{[i]} \tag{3}$$

with $\tilde{x}^{[i]}$ is the non-redundant Kronecker power of the state vector x to the i^{th} order, q is a truncation order and $U \in R^{p \times r}$ is a matrix belonging to the interval matrix:

$$\mathcal{U} = \left\{ \; U = [U_1, \ldots, U_q] \;\; : U_i \in \left(U_i^-, U_i^+\right), i = 1, .., q \right\}$$

.

For the seek of simplicity and for explaining our approach, we consider the case of $q = 2$. We obtain $U_1 = [u_1, u_2]$, $U_2 = [u_3, u_4, u_5]$ and consequently

$$u = u_1 x_1 + u_2 x_2 + u_3 x_1^2 + u_4 x_1 x_2 + u_5 x_2^2$$

The domain of attraction of the controller is the set of states which can be steered towards the terminal region. This paper is devoted to enlarge this domain. The size of the **DA** depends on the control parameters, and the chosen Lyapunov function. A wise and optimal choice of both of these may yield a bigger domain of attraction. The size of the region depends on the computed controller, the constraints on the system and the procedure used to compute it. Thus, the most used procedure to enlarge the domain of attraction are based on a polynomial control. This leads to a greater number of parameters and therefore, to a greater computational effort. In this paper, a formulation of the problem, focused on enlarging the domain of attraction without increasing the computational effort is presented. The optimization problem formulation, and hence the computational effort is similar to the original one but with a larger domain by using the Genetic Algorithm approach.

For this purpose, we proceed in three steps :
-First, we exploit the method described in [3] to derive an initial **DA**.

-Second, we implement a **GA** combined with a Linear Matrix Inequalities **LMI** approach to determine explicitly a maximal parameterized Lyapunov Function.
-Third, the implementation of the **RTM** leads to a maximized asymptotic stability region, while giving an accurate idea on numerical values of the Lyapunov function parameters. The second step is finally reapplied in order to define the maximal Lyapunov function and its corresponding **DA**. A parameter optimization will cover in this step both the Lyapunov function parameters and those of the polynomial control input.

3 Preliminaries results

Before proceeding further, we will give some preliminary results.

Let $V(x) \in R$ be a positive definite, radially unbounded and continuously differentiable function. The bounded set

$$\Omega(c) = \{x \in R^n / V(x) \leq c\} \tag{4}$$

is an estimate of the region of attraction if $\Omega \subset \mathcal{D}$ where $\mathcal{D} = \{x \in R^n / \dot{V}(x, U) < 0\} \cup \{0\}$. The time derivative of $V(x, U)$ along trajectory of system (1) is given by

$$\begin{aligned} \dot{V}(x, U) &= \tfrac{\partial V(x)}{\partial x} f(x) + \tfrac{\partial V(x)}{\partial x} g(x)\ U\phi(h(\tilde{x})) \\ &= L_f V(x) + L_g V(x) U \phi(h(\tilde{x})) \end{aligned} \tag{5}$$

where $L_f V(x)$ (resp. $L_g V(x)$) is the Lie derivative of $V(x)$ along the polynomial function vector $f(x)$ (resp. $g(x)$). In what follows, we shall denote $L_g V(x) U \phi(h(\tilde{x}))$ by $L_{(g,U)} V(x)$ for simplicity; i.e. $L_{(g,U)} V(x) = L_g V(x) U \phi(h(\tilde{x}))$.

The largest estimate of the **DA** is given by $\Omega(c^*(U))$ where :

$$c^*(U) = \inf_{x \in R^n} V(x) \quad \text{such as} \quad \dot{V}(x, U) = 0 \quad \text{for each matrix} \quad U \in \mathcal{U}. \tag{6}$$

The optimal value of $c^*(U)$ is obtained by

$$c^* = \sup_{U \in \mathcal{U}} c^*(U). \tag{7}$$

In [3], it has been proven that for any given $c \in R$, $c \leq c^*$ if there exists $U \in \mathcal{U}$ and $s(x)$ a positive definite polynomial such that

$$\dot{V}(x, U) + (c - V(x))\, s(x) < 0 \tag{8}$$

then the polynomial degrees of $V(x)$ and $\dot{V}(x, U)$ are $2\delta_V$ and δ_L respectively. If we choose $s(x)$ degree to be $2\delta_s$ such that

$$\delta_s \geq \frac{\delta_L}{2} - \delta_V \tag{9}$$

it follows that the degree of the polynomial

$$t(x, U, c, s(x)) = \dot{V}(x, U) + (c - V(x)) s(x) \tag{10}$$

is equal to $2\delta_m$ where $\delta_m = \delta_V + \delta_s$.

An approach based on both Square Matricial Representation **(SMR)** and Complete Square Matrix Representation **(CSMR)** of polynomials is used in order to determine an appropriate optimization problem [3]. The CSMR provides all the possible representations of a polynomial in terms of a quadratic form.

The CSMR matrix of $t(x, U, c, s(x))$ is given by

$$T(\alpha, U, c, S) = D_f(\alpha) + D_g(U) + cW_1(S) - W_2(S) \tag{11}$$

where $D_f(\alpha)$ is the CSMR of $L_f V(x)$, $\alpha \in R^{\tau(n,\delta_m)}$, $D_g(U)$ is the SMR of $L_{(g,U)}V(x)$, $W_1(S)$ and $W_2(S)$ are the SMR of $s(x)$ and $V(x)s(x)$.

The condition (8) with (11) implies that if

$$\hat{c}^* = \sup_{U\in\mathcal{U},\alpha,S>0} c \quad \text{such that} \quad T(\alpha, U, c, S) < 0; \qquad \text{then} \qquad \hat{c}^* \leq c^* \tag{12}$$

Theorem 1. ([3]) *The lower bound $\hat{c}^*$ is given by*

$$\hat{c}^* = \frac{-\lambda^*(U)}{1 + \mu\lambda^*(U)} \tag{13}$$

where $\lambda^*(U)$ is the solution of the following GEVP

$$\lambda^*(U) = \inf_{U\in\mathcal{U},\alpha,S>0,\lambda} \lambda \tag{14}$$

$$\textit{such that} \begin{cases} 1 + \mu\lambda > 0; \qquad U \in \mathcal{U}; \qquad S > 0 \\ \lambda W(S) > D_f(\alpha) + D_g(U) - W_2(S) \end{cases}$$

with μ being any positive scalar and

$$W(S) = K^T\left(\begin{pmatrix} 1 & 0 \\ 0 & \mu V \end{pmatrix} \otimes S\right)K. \tag{15}$$

The symbol $\otimes$ is the Kronecker's product and the matrix K satisfies

$$\begin{pmatrix} 1 \\ x^{\{\delta_V\}} \end{pmatrix} \otimes x^{\{\delta_s\}} = Kx^{\{\delta_m\}} \tag{16}$$

where $x^{\{\delta_m\}} \in R^{\varsigma(n,\delta_m)}$, $\alpha \in R^{\tau(n,\delta_m)}$, $x^{\{\delta_V\}} \in R^{\varsigma(n,\delta_V)}$, $x^{\{\delta_s\}} \in R^{\varsigma(n,\delta_s)}$, $K \in R^{\varsigma(n,\delta_s)(\varsigma(n,\delta_V)+1)\times\varsigma(n,\delta_m)}$ and the quantities $\varsigma(n, \delta_m)$ and $\tau(n, \delta_m)$ are given by

$$\begin{aligned} \varsigma(n, \delta_m) &= \frac{(n + \delta_m)!}{n!\delta_m!} - 1 \\ \tau(n, \delta_m) &= \frac{1}{2}\varsigma(n, \delta_m)(\varsigma(n, \delta_m) + 1) - \varsigma(n, 2\delta_m) + n. \end{aligned}$$

4 Main Results

In this section, we present the main results of the paper. For this, we go back to the **RTM** introduced in [12], for estimating the Region of Asymptotic Stability (RAS) through reversing the system trajectory flow. Our main aim is to enlarge the RAS resulting form the method described in [3]. The idea consists in determining the maximal RAS of the system obtained via the implementation of the **RTM**. This allows determining an analytical expression of the maximal **DA** included in the RAS. Such a result is not given by the **RTM**, which finally gives an interesting graphical representation of the RAS. We want to compute an analytical expression of the RAS in terms of a Lyapunov function. We consider, for this purpose, a quadratic Lyapunov function of the form:

$$V(x) = x^T P x, \qquad \text{where } P = P^T > 0 \tag{17}$$

For the sake of simplicity, we assume that $P \in R^2$, with $P = \begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix}$, so that :
$V(x) = p_1 x_1^2 + 2p_2 x_1 x_2 + p_3 x_2^2$. The results are generalizable for matrices of larger sizes.

By using such a Lyapunov function we can express the RAS in terms of an ellipsoid in (x_1, x_2) plane which will be contained in the RAS obtained by the trajectory reversing method. The idea consists in estimating the parameters $p_i, (i = 1, 2, 3)$ and $u_i, (i = 1, 2, 3, 4, 5)$ via a genetic algorithm. By combining this algorithm with an LMI optimization we can obtain the largest ellipsoid contained in the RAS. The candidate solutions $p_i, (i = 1, 2, 3)$ and $u_i, (i = 1, 2, 3, 4, 5)$ are chosen arbitrarily and can be regarded as individuals. Each variable can be considered as a gene and the different steps of the Genetic Algorithm can be expressed as follows :

1. *Initial population* : Each individual in the population is represented by a chromosome that is coded in binary form.

2. *Selection* : In the selection process, strings must be of suitable fitness to be selected as a factual member in creating a new population.

3. *Replication* : In this process, the best strings which have a greater probability than others will be member of the next generation.

4. *Crossover* : Two random chromosomes are selected in recreation process. They will be exposed to the crossover rate and they will swap from the crossover points. The experienced boundary for crossover rate is from 0.6 to 0.9.

5. *Mutation* : We underline mutation relevance in just one way : for each bit we generate a random number and if it is less than the specified mutation probability, we flip the bit. If it is "one" we change it to "zero" and vice versa. The values, which are the boundaries for mutation rate, are between 0.1 and 0.3. In order to estimate, the parameter, by means of a genetic algorithm, we must encode the vector into binary code in order to let the parameters with genetic algorithm easier.

Theorem 2. *In this work, a **GA** is used to estimate the parameters $p_i, (i = 1, 2, 3)$ and $u_i, (i = 1, ..., 5)$ where the next constraint is satisfied by :*

$$p_1 > 0, p_3 > 0 \quad \text{and} \quad p_1.p_3 > (p_2)^2. \tag{18}$$

Theorem 3. *Each iteration of the **GA** optimization routine, results in new parameters values $p_i, (i = 1, 2, 3)$, $u_i, (i = 1, ..., 5)$ and $\hat{c}^*$ which is a solution of the LMI corresponding to this set of parameters. The genetic algorithm leads to the best set of parameters $p_i, (i = 1, 2, 3)$ and $u_i, (i = 1, ..., 5)$ which are used in the computation of the best solution $\hat{c}^*$ (defined by (13)) which is the solution of the LMI (14).*

- The set of variables $\{p_i, i = 1, 2, 3\}$ and $\{u_i, i = 1, ..., 5\}$ are encoded into the gene candidate.

- The fitness value is given by the maximum c^* for which there exists a feasible solution of the LMI optimization.

- The global optimization of the variables $p_i, (i = 1, 2, 3)$ and $u_i, (i = 1, ..., 5)$ is performed by the genetic operators (selection, recombination and mutation).

The fitness function is evaluated in two steps as follows :

First Step	**Second Step:**
Set $p_i, (i = 1, ..., 3)$ *as gene candidates*	*Set* $p_i, (i = 1, ..., 3)$ *as gene candidates*
while	*Set* $u_i, (i = 1, ..., 5)$ *as gene candidates*
$p_1 > 0, p_3 > 0$ and $p_1.p_3 > (p_2)^2$	*while :* $p_1 > 0, p_3 > 0$ and $p_1.p_3 > (p_2)^2$
Apply the LMI Optimization	*Apply the LMI Optimization*
If $c_j > c_i,\ j > i; i, j = 1, ..., N$	*If* $c_j > c_i,\ j > i; i, j = 1, ..., N$
Write p_{j1}, p_{j2}, p_{j3}	*Write* $p_{j1}, p_{j2}, p_{j3}, u_{j1}, u_{j2}, u_{j3}, u_{j4}, u_{j5}$
end	*end*
end	*end*
return $\hat{c}^* = c_j$ *as fitness value.*	*return* $\hat{c}^* = c_j$ *as fitness value.*

The following flowchart, in figure 1, presents the main steps needed to implement the synthesized algorithm, according to which, we can maximize the region of asymptotic stability. The obtained solution is specified by the definition of a maximal quadratic Lyapunov function.

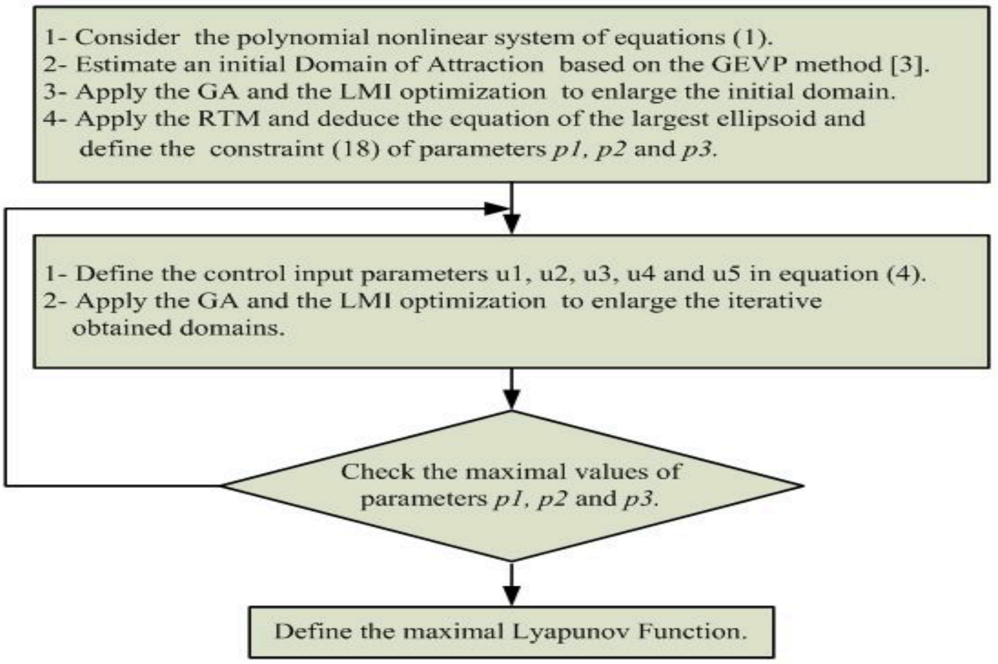


Figure 1: Flowchart of Advanced LMI Optimization Algorithm for Maximizing the DA

5 Illustrative Example

Consider the following polynomial nonlinear system [3]:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -x_1 - x_2 + x_2^2 + x_1^2 x_2 + u \end{cases} \tag{19}$$

Consider a controller that is linear in $y_1 = x_1$; that is, $u = u_1 x_1$, with : $\mathcal{U} = \{U = u_1 : -2 \leq u_1 \leq 2\}$. In order to find the shape of the **DA**, we employ a Lyapunov function of the form :

$$V(x) = p_1 x_1^2 + 2p_2 x_1 x_2 + p_3 x_2^2 \tag{20}$$

Since the degree δ_L of $\dot{V}(x, U)$ is 4, we can select $\delta_s = 1$ which implies $\delta_m = 2$. Vectors $x^{\{\delta_v\}}, x^{\{\delta_s\}}$ and $x^{\{\delta_m\}}$ are selected as: $x^{\{\delta_V\}} = x^{\{\delta_s\}} = (x_1, x_2)^T$ and $x^{\{\delta_m\}} = \left(x_1, x_2, x_1^2, x_1 x_2, x_2^2\right)^T$ which implies that:

$$D_f(\alpha, p_{1,2,3}) = \begin{pmatrix} -2p_2 & (p_1 - p_2 - p_3) & 0 & \alpha_1 & \alpha_2 + p_2 \\ (p_1 - p_2 - p_3) & (2p_2 - 2p_3) & -\alpha_1 & -\alpha_2 & p_3 \\ 0 & -\alpha_1 & 0 & p_2 & \alpha_3 \\ \alpha_1 & -\alpha_2 & p_2 & -2(\alpha_3 - p_3) & 0 \\ \alpha_2 + p_2 & p_3 & \alpha_3 & 0 & 0 \end{pmatrix}$$

$$D_g(p_{1,2,3}) = \begin{pmatrix} 2p_2 u_1 & p_3 u_1 & 0 & 0 & 0 \\ p_3 u_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}; S = \begin{pmatrix} s_1 & s_2 \\ s_2 & s_3 \end{pmatrix}$$

$$W(S, p_{1,2,3}) = \begin{pmatrix} s_1 & s_2 & 0 & 0 & 0 \\ s_2 & s_3 & 0 & 0 & 0 \\ 0 & 0 & \mu p_1 s_1 & \mu\alpha & 0 \\ 0 & 0 & \mu\alpha & \mu\alpha & \mu\gamma \\ 0 & 0 & 0 & \mu\gamma & \mu p_3 s_3 \end{pmatrix}; \quad W_2(S, p_{1,2,3}) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & p_1 s_1 & \alpha & 0 \\ 0 & 0 & \alpha & \beta & \gamma \\ 0 & 0 & 0 & \gamma & p_3 s_3 \end{pmatrix}$$

with $\alpha = p_1 s_2 + p_2 s_1, \beta = p_1 s_3 + 4p_2 s_2 + p_3 s_1$ and $\gamma = p_2 s_3 + p_3 s_2$.

We propose to encode the parameters p_i into 7 bits code : $p_i = (p_{0i}, p_{1i}, p_{2i}, p_{3i}, p_{4i}, p_{5i}, p_{6i})$, and put them into a chromosome as follows :

$$\begin{cases} Pop\_size = 50; & Crossover\_rate = 0.65; \\ Mutation = 0.1; & Max\_generations = 100. \end{cases} \tag{21}$$

Note that we consider the LMI representation as a fitness function for the genetic program. Then, we obtain: $p_1 = 3.5; \quad p_2 = 1.5; \quad p_3 = 5.5$.

This result justifies a domain of attraction that is larger than that obtained through the direct method [3]. The result of this study is shown in Fig. 2 where $\mu = \frac{1}{100}$. In this figure, solid line $\dot{V}(x, U) = 0$, and $V(x) =$ 2.5244 obtained by the GA method. Dashed line the $\dot{V}(x, U) = 0$ and $V(x) =$ 1.2324 obtained by the method developed in [3].

When we apply the RTM described in ([12]), the result given in Fig. 3 represents the region of asymptotic stability of the system under consideration. Hence, the equation of the largest **DA** is given by

$$V(x) = 9x_1^2 + 4.4x_1 x_2 + 15.5x_2^2 = 9.5.$$

When implementing our proposed method based on the LMI and the GA, we obtain the following result, with input $u = 0.4864x_1$:

$$V(x) = 9.5x_1^2 + 4x_1 x_2 + 15.5x_2^2 = 8.5066$$

The result of this study is depicted in Fig. 4 which shows that the resulting domain of attraction is larger than the initial one given by [3]. In this figure, solid line indicates the $\dot{V}(x,U) = 0$ and $V(x) = 8.5066$ obtained via the proposed method with GA, the dashed line indicates the $\dot{V}(x,U) = 0$ and $V(x) = 1.2324$ result in [3].

Now, we investigate the maximization of the **DA** by estimating parameters p_1, p_2 and p_3 of the Lyapunov function and u_1, u_2, u_3, u_4 and u_5 of the control input. Consequently, we have

$$u = u_1x_1 + u_2x_2 + u_3x_1^2 + u_4x_1x_2 + u_5x_2^2 \tag{22}$$

with $\mathcal{U} = \{U = [u_1, u_2, \ldots, u_5] \; : -2 \leq u_i \leq 2, i = 1, ..., 5\}$
We propose to encode the parameters p_i and u_i into 7 bits code

$$\begin{aligned} p_i &= (p_{0i}, p_{1i}, p_{2i}, p_{3i}, p_{4i}, p_{5i}, p_{6i}) \\ and\ u_i &= (u_{0i}, u_{1i}, u_{2i}, u_{3i}, u_{4i}, u_{5i}, u_{6i}) \end{aligned}$$

and put them into a chromosome as described in equation (21). This implies that $D_f(\alpha, p_{1,2,3})$, S, $W(S, p_{1,2,3})$ and $W_2(S, p_{1,2,3})$ remains unchanged, while $D_g(u_{1,2,3,4,5}, p_{1,2,3})$ is defined as :

$$D_g(u_{1,2,3,4,5}, p_{1,2,3}) = \begin{pmatrix} 2p_2u_1 & (p_3u_1 + p_2u_2) & p_2u_3 & (p_3u_3 + p_2u_4) & 0 \\ (p_3u_1 + p_2u_2) & 2p_3u_2 & 0 & (p_2u_5 + p_3u_4) & p_3u_5 \\ p_2u_3 & 0 & 0 & 0 & 0 \\ (p_3u_3 + p_2u_4) & (p_2u_5 + p_3u_4) & 0 & 0 & 0 \\ 0 & p_3u_5 & 0 & 0 & 0 \end{pmatrix}$$

The result of this study yields

$$V(x) = 9.5x_1^2 + 3.9x_1x_2 + 15.5x_2^2 = 38.6183$$

which means that the input is written as

$$u = 0.6348x_1 - 1.9996x_2 + 0.0006x_1^2 - 0.0015x_1x_2 - 1.0044x_2^2$$

In Figure 5, we represent the different **DA**(s) by continuous lines (which allow elliptic forms). The dash-dotted curves represent the constraints. Clearly, the largest **(DA)** is obtained when we optimize both parameters of the LF and those of the control input. In this figure, the DA (blue line) illustrates the results obtained through the optimization of parameters $p_1, p_2, p_3, u_1, u_2, u_3, u_4$ and u_5 by combining GA and LMI. DA with red line : obtained results while optimizing parameters p_1, p_2, p_3, u_1 by combining the GA and LMI [2]. DA with the black line : obtained results by optimizing parameters p_1, p_2, p_3, u_1, with LMI [3].This result demonstrates the consistency of the proposed method.

6 Conclusion

In this paper, the problem of enlarging the attraction domain for nonlinear controlled systems is investigated. We were particularly interested in the class of nonlinear polynomial systems which represents a large class of physical nonlinear dynamics that can be approximated by polynomials using Taylor series expansions. We rely on an optimization approach based on Linear Matrix Inequality (LMI) to compute an initial region of attraction. The main contribution consists in the determination of an explicit **DA** by using a parameterized Lyapunov function. The parameters of both the Lyapunov function and nonlinear control input are computed by combining the Genetic Algorithm and an LMI approach. The implementation of the reverse trajectory method leads to represent the largest elliptic shape of the RA. An illustrative example has demonstrated the efficiency of the established results.

Acknowledgements

This work has been supported, for M. Djemaï, by ARI-Projets and by International Campus on Safety and Intermodality in Transportation, the European Community, the Delegation Regionale a la Recherche et a la Technologie, the Ministere de l'Enseignement supérieur et de la Recherche, the Region Nord Pas de Calais and the Centre National de la Recherche Scientifique.

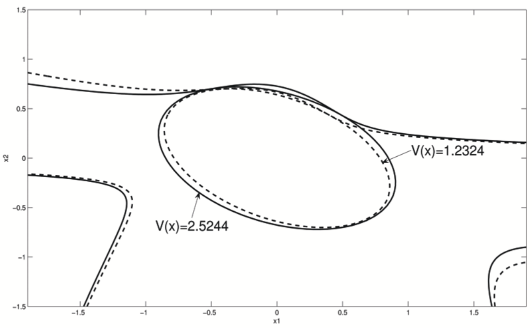


Figure 2: $\dot{V}(x, U) = 0$, and $V(x)$.

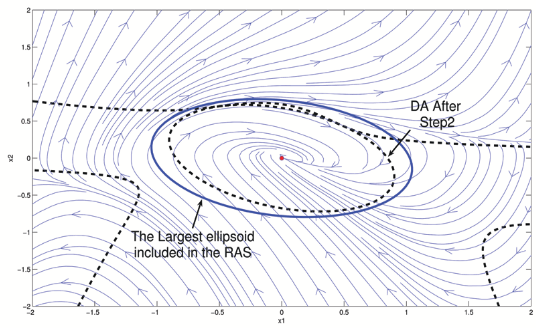


Figure 3: Largest Ellipsoid Included in RAS

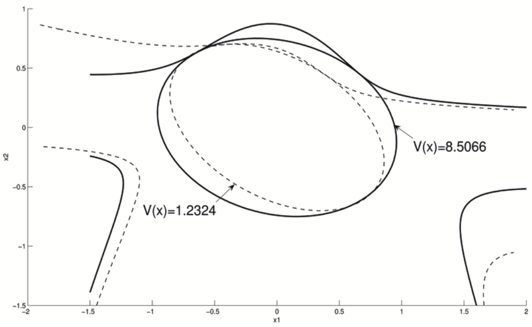


Figure 4: $\dot{V}(x, U) = 0$ and $V(x)$.

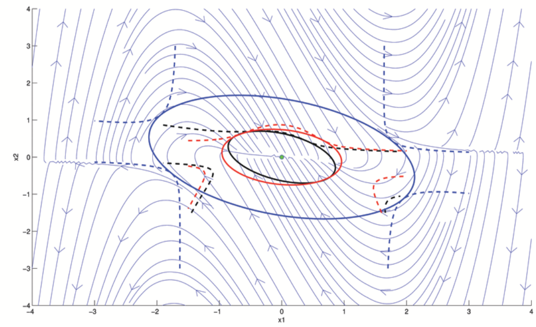


Figure 5: DA by combining GA and LMI.

Bibliography

[1] D. Rosinova and V. Vesely,Robust PID decentralized controller design using LMI, *INT J COMPUT COMMUN*, ISSN 1841-9836, 2(2):195-204, 2007.

[2] F. Hamidi, H. Jerbi, W. Aggoune, M. Djemaï and M. Naceur Abdkrim, Enlarging region of attraction via LMI-based approach and Genetic Algorithm, *Int. Conf. on Communications, Computing and Control Applications*, Hammamet Tunisia 2011.

[3] G. Chesi, Computing output feedback controllers to enlarge the domain of attraction in polynomial systems, *IEEE Trans. on Automatic Control*, 49: 846-1850, 2004.

[4] G. Chesi, A. Garulli, A. Tesi and A. Vicino, Characterizing the solution set of polynomial systems in terms of homogeneous forms: an LMI approach, *Int. J. of Nonlinear and Robust Control*, 13(13): 1239-1257, 2003.

[5] G. Chesi, A. Garulli, A. Tesi and A. Vicino, Solving quadratic distance problems: an LMI approach, *IEEE Transactions on Automatic Control*, 13: 200-212, 2003.

[6] G. Chesi, A. Garulli, A. Tesi and A. Vicino, Homogeneous Lyapunov function for systems with structured uncertainties, *Automatica*, 39(6): 1027-1035, 2003.

[7] H.K. Khalil, *Nonlinear systems*,Third Edition, Upper Saddle River, NJ :Prentice Hall, 2001.

[8] M. Loccufier and E. Noldus, A new trajectory Reversing Method for Estimating Stability Regions of autonomous Nonlinear Systems, *Nonlinear dynamics*, 21: 265-288, 2000.

[9] M. Loccufier and E. Noldus, On the estimation of asymptotic stability regions for autonomous nonlinear systems, *IMA J. of Mathematic Control an Information*, 12: 91-109, 1995.

[10] M.M. Belhaouane, R. Mtar, H. Belkhiria Ayadi and N. Benhadj Braiek, An LMI Technique for the Global Stabilization of Nonlinear Polynomial Systems, *INT J COMPUT COMMUN*, 4(4):335-348, 2009.

[11] O. Hachido, and B. Tibken, Estimating domains of attractions of a class of nonlinear dynamical systems with LMI methods based on the theory of moments, *Proc.of the 41st IEEE Conference on Decision and Control*, Las Vegas, Nevada, 3150-3155, 2002.

[12] R. Genesio, M. Tartaglia and A. Vicino, On the estimation of asymptotic stability regions: state of the art and new proposals, *IEEE Trans. on Automatic Control*, 30(8): 747-755, 1985.

[13] Y. Fujisaki, and R. Sakuwa. Estimation of asymptotic stability regions via homogeneous polynomial Lyapunov functions, *International Journal of Control*, 79(6): 617-623, 2006.

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):548-559, August, 2013.

Multithreaded Application for Real-Time Visualization of ECG Signal Waveforms and their Spectrums

S.S. Ilić, A.Č. Žorić, P. Spalević, Lj. Lazić

Siniša Ilić, Aleksandar Č. Žorić, Petar Spalević
University of Pristina in Kosovska Mitrovica
Faculty of Technical Sciences
Computer Sciences and Informatics Department
Serbia, 38220 Kosovska Mitrovica, Kneza Miloša 7
sinisa.ilic@pr.ac.rs

Ljubomir Lazić
State University of Novi Pazar
Serbia, 36300 Novi Pazar, Vuka Karadžića bb
llazic@np.ac.rs

Abstract: By using concept of virtual instrumentation, signals from human body can be digitized and transferred to computer for further processing. Software in a computer enables use of modern tools for digital signal processing that can be improved easily with emergence of new knowledge and with increasing of computer performances. Presenting the ECG signal in both: Time and Time-Frequency domains enables to cardiologist to obtain more reliable diagnosis.
In order to present simultaneously the waveform and spectrogram of ECG signal in the real time we use Fast Fourier and Discrete Wavelet transform in the multithreading environment of a standard personal computer. The synchronization of accessing the signal data by threads according to the principle one thread a time is performed by controlling the state of event type variables.
Keywords: Biomedical signal processing, Fast Fourier transforms, Multithreading, Real time systems, Wavelet transforms

1 Introduction

The standard data acquisition devices are powerful, relatively expensive and designed to perform one or more specific tasks defined by the vendor. However, the user generally cannot extend or customize them. Virtual instruments, by virtue of being PC-based, inherently take advantage of the benefits from the latest technology incorporated into off-the-shelf PCs.

Use of PC in so called virtual or personal instrumentation [1], [2] development enables realization of a new generation of superior devices. Virtual instrument consists of two sub-devices:

1. Acquisition hardware device that collects data and transform it in the form suitable for reading by PC and
2. Computer with software (user interface) device that reads digital data, process it and presents results.

Once the data has appeared at the input of computer in digital form, it can be stored and used for different types of processing as many times as needed by using different or same software. As the central role in the virtual instrumentation concept is played by the software [1], [3], the costs of development, maintenance and reconfiguration of instruments are reduced significantly. These devices can be connected to the internet, they offer easy modifications according to the users needs, it is possible to store large amount of data at a cheap price, and these data can be retrieved quickly.

By using of PC-based ECG device that is built upon the principles of virtual instrumentation, we can monitor changes of bio-potentials at the skin of the human body caused by heart activity ECG. Some of the tools for processing of ECG signal that can be implemented in this device are: segmentation (recognition and separation of ECG signal parts) [4], [5], measuring of heart rate variability [6], compression of signal for storage purposes [7], etc.

The analysis of ECG signal in Time-Frequency domain can help cardiologist to have a better look to the signal from a different angle. The presence of some frequency components in appropriate segments of ECG signals can help cardiologists in decision making [8] - [10]. The mathematical tools that can be used for this purpose are: Short Time Fourier Transform (STFT), Discrete Wavelet Transform (DWT) and Continual Wavelet Transform (CWT). It would be very useful for the users to monitor ECG signal in Time-Frequency domain in real time.

The need for real-time processing of medical images becomes higher and some implementations are presented in the following papers [11] [13]. In order to enable processing in real-time, faster algorithms [14], multithreading [15], modularity [16] and other improvements are needed.

We have come to idea to plot in real time simultaneously ECG signal in both Time and Time-Frequency domain using STFT and DWT. The idea is realized by development of multithreading application in software part of our virtual instrument [18].

2 The PC-ECG Device

The PC-based ECG device is described in [9], [17], but the most important features will be presented in this paper too. The main user requirements, suggested by cardiologists, were the following functional and control system performances:

- selection of one of twelve leads for real time monitoring and recording by the descriptive and intuitive GUI,
- recording of waveforms for each of twelve leads during a time interval of six seconds,
- connection to the database of patients,
- filtering of patient body signal from the public power network frequency,
- possibility to select, measure the selection time duration and zoom the selection,
- printed report generation (waveforms of 12 leads and cardiologist diagnosis),
- easily portable and no additional power supply (plug into PC and use),
- 100 % patient's body insulation.

2.1 Hardware part of the Device

The hardware part of the system is designed by using Microchip microcontroller PIC16F876 and USB communication port of a PC is selected as interface for bilateral communication between acquisition hardware and a PC. The communication between acquisition hardware and a PC can be obtained: 1) by using standard USB cable; in this case PC is power supplier of the acquisition hardware and 2) by using MaxStream XBee RF module; in this case the device is powered by battery and communication is wireless the USB RF dongle using PIC18F2550 MCU and XBee RF module is attached to a PC and it communicates with acquisition device equipped with the same XBee RF module [18]. The selection of communication type depends on distance between a patient and a PC if the distance is short, the USB cable will be used and if the distance is long the wireless communication will be selected. The XBee RF modules operate within the ZigBee protocol [18] and support the unique needs of low-cost, low-power wireless sensor networks.

The ECG signal is sampled with sampling rate of 960Hz. We used two bytes for transmission of each sample value; hence the transmission speed was set to 19600 Baud/s. The program of microcontroller selects the appropriate ECG channel (ECG lead) of multiplexer and performs A/D conversion or stops the program execution, depending on the code received from PC.

2.2 Graphical User Interface Design

The design of software part of the instrument is based on the traditional style of ECG paper plots with additional overlay displays and other features such as: zoom, filter, database support, etc. based on proposals of cardiologists.

The device can be used just to monitor signal waveforms (without saving the data) or it can be used to store signal at the buffer memory of PC. A user can select to preview and save the signal of any of 12 ECG leads. The application screen is divided to main and auxiliary window. When the Save button is clicked on the toolbar, the waveform of recorded signal is shown at the auxiliary window while the plotting of running signal is continued at the main window. A user can decide to overwrite already recorded signal by the signal he/she monitored at later time. After signal of all or some ECG leads are recorded at the buffers of PC (the signal of one lead is saved in one buffer), a user can save the samples from the buffers to the database with the: general data about patient, patient visit with medical anamnesis, explanation on interpretation of signals anomaly and selection of one of predefined diagnosis. All 12 leads with patient data, medical anamnesis, diagnosis and treatment can be printed in two-sided A4 paper report and given to the patient.

Software enables defining and marking the segments of ECG signal by a user (for example QT-interval), measuring the segment time duration (for example 0.412 seconds) and saving the boundaries of those segments along with signal to the database. The cardiologist can analyze the signal again, after it is recorded considering all interpretations he or somebody else put in database (for example: in the lead V1 ST depression up to 3mm, in the lead D3 small elevation, QT segment too wide etc.). By selection of diagnosis from the finite list of diagnoses for each patient visit, it becomes easier for cardiologist to search for appropriate signal waveforms, interpretations of waveforms and patients in database. Database also helps him to compare the waveforms and interpretations of ECG signal for the same patient taken in different patient visits and to make appropriate decisions on treatment and disease progress. The screenshot of application with three parts the main waveform (at the bottom), the main spectrogram/scalogram (at the central part for spectrum and on the right part for color legend), and auxiliary window (at the top-right) is presented in Figure 1.

3 Time-Frequency presentations of ECG signal in PC-ECG device

There are two modes of plotting ECG in Time-Frequency domain in the application: standard and real-time mode. The standard (off-line) mode enables plotting of spectrogram (in the case of STFT) or scalogram (in the case of wavelet analysis) after selecting the waveform of the lead already recorded in database or in memory buffer. In this case the waveform of 6 seconds duration is processed at once; the Time-Frequency coefficients are calculated and forwarded to plot function. The calculation and plotting of the coefficients is not time limited in this mode.

In the real time (on-line) mode samples of the signal appears continuously at the input port of a PC. It is not necessary to plot sample by sample in real-time, it is enough to plot group of samples one by one every 40 milliseconds (25 times in a second). The drawing of ECG

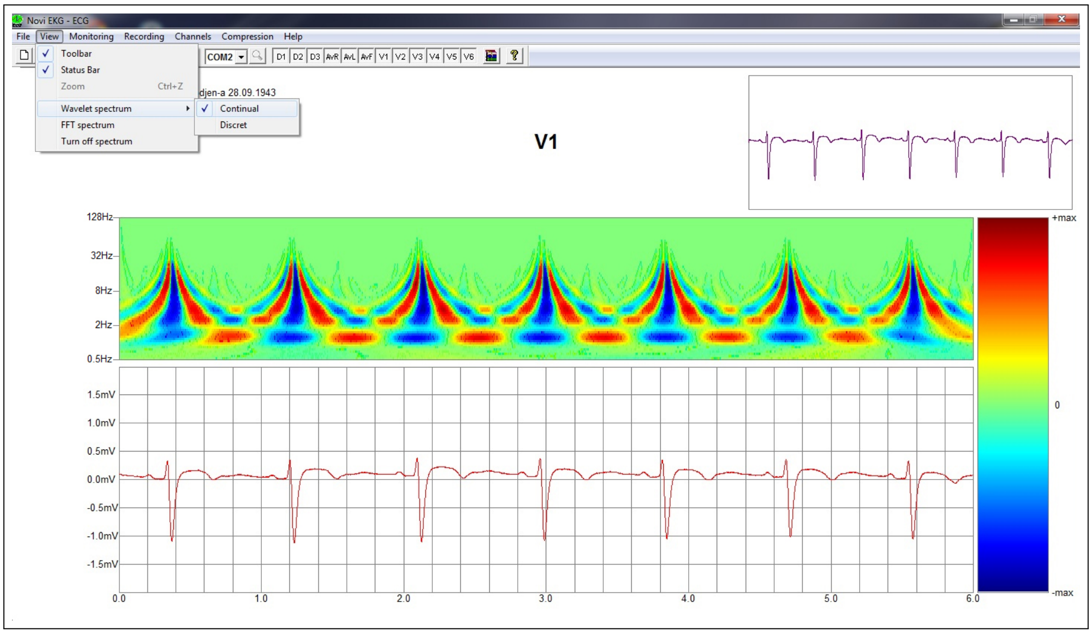


Figure 1: The Screenshot of PC-ECG Application

waveform, calculation of coefficients and drawing the Time-Frequency presentation of samples in 40 milliseconds time-frame must be performed between two calls of *OnTimer* function.

As the Time-Frequency plot consist of three axes: time, frequency and component value, we have decided to present the third axis by color. The color palette has 240 colors: from nuances of dark blue toward nuances of light blue, green, yellow, light red and dark red (see Figure 1). Before plotting of spectrogram (scalogram) the range of all calculated values of signal's coefficients in the standard mode is normalized by the maximum absolute value and proportionally divided to integer values in the range 0-239. In the real-time mode calculation of coefficients is performed on portions of the signal. The maximum of coefficients' values differs from portion to portion. This way of normalization might produce visually same spectrums for portions of signals with different amplitudes. Hence, the values of calculated coefficients are normalized by predefined maximum value (the maximum of maximum values found in standard mode spectrograms).

The STFT spectrum is calculated by using FFT function on overlapped windows. The window length (in samples), length of shifting and windowing function can be selected on appropriate application dialog (see Figure 2a).

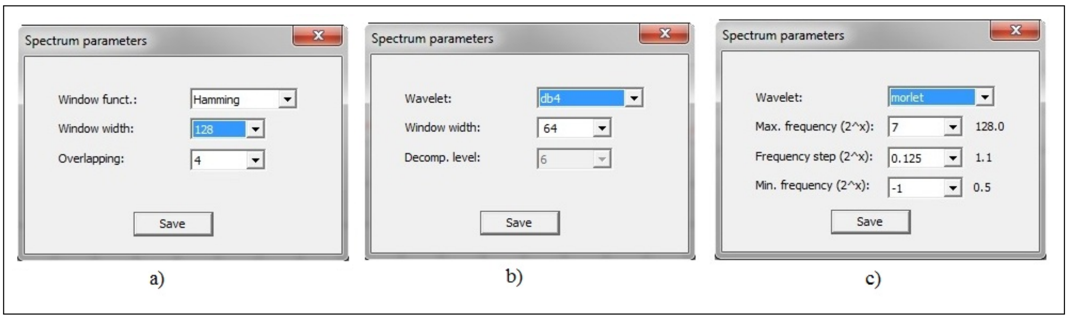


Figure 2: The Application dialogs for selecting parameters for calculating spectrograms

The samples of the window are multiplied with coefficients of windowing function, the FFT is then applied on the resulting set and spectral amplitudes are obtained. The window is then shifted by chosen number of samples and the process is repeated. If N is the total number of samples in recorded signal, W is the number of samples in window and O is number of samples for which the windows are shifted, then the following is valid: $N = W + k \times O$, where $k+1$ is the total number of windows to be calculated. From W samples of one window $W/2+1$ amplitudes of frequency components are calculated and presented over the y axis, so the total number of spectral coefficients of the signal is $(W/2+1) \times (k+1)$. In the standard mode all coefficients are calculated first, then normalized and plotted. In the real-time mode it is necessary to calculate, normalize and plot coefficients of n/O overlapping windows in less than 40 milliseconds, where n is the number of samples captured in the 40 ms time frame. The application function that calculates STFT spectrum in real time takes the following input parameters: 1) the window length W, 2) the shifting size O, 3) sample set size n, 4) pointer on last n samples ($*ns$) and 5) pointer on W samples captured before last n samples ($*ws$) on the input port of a PC. The function starts with making the window from $W-O$ last samples of $*ws$ and O starting samples of $*ns$ and calls the FFT function by forwarding samples of the created window. Then the new window is created by $W - 2 \times O$ last samples from $*ws$ and $2 \times O$ starting samples from $*ns$ and forwarded to FFT function. The process is repeated until the last shifting window of length W is created and its spectral coefficients calculated. The coefficients are then normalized and plotted at appropriate portion of spectrum part of the application screen.

The DWT spectrum is calculated by using convolution function. The wavelet function and the window length (W in samples) can be selected on appropriate application dialog (see Figure 2b). After selecting the wavelet function name, the wavelet and scaling function coefficients (of length h) are retrieved from the table of wavelets. In the standard mode signal (of length N) is extended on both sides because of border distortion by symmetric boundary value replications of length $h-1$ each and filtered with appropriate wavelet (high pass filter) and scaling (low pass filter) function. The filtering is performed by convolution function. As the number of samples in the recorded signal is large, the FFT convolution [20] is applied instead of classical convolution because of speed improvements. Two filtered signals are now obtained - approximation A_1 (low pass) and details D_1 (high pass). In each signal, N central coefficients are selected and from them odd coefficients are kept and even discarded thus resulting in two groups of $N/2$ coefficients. Detail coefficients D_1 are saved at buffer and the process of obtaining the new approximation A_2 and detail D_2 coefficients from approximation coefficients A_1 is repeated again by using the same procedure, but now beginning with $N/2$ coefficients of A_1 instead of N samples of signal. The process is repeated $L = \log 2W$ times and the number of obtained coefficients is halved in each iteration (level). The DWT spectrum consists of coefficients of details D_1, D_2, $\ldots$, D_L where levels represent frequency ranges.

In the real-time (on-line) mode the coefficients are calculated by processing DWT window by window (of length W) from FIFO buffer where ECG samples are captured. The border distortion is eliminating by keeping samples and coefficients of A_i in L buffers of length $h-1$ from previous windows. The coefficients of A_1 and D_1 are calculated from W samples of current window and $h-1$ samples of previous window, coefficients of A_2 and D_2 are calculated from $W/2$ coefficients of just calculated coefficients of A_1 and last $h-1$ coefficients of A_1 from previous window etc. The number of samples in window W is small and the standard convolution is used. The coefficients of details that are calculated in time of 40 ms are normalized and plotted at appropriate portion of spectrum part of the application screen.

The calculation of CWT is for now possible only in standard off line mode. From the samples of signal, CWT is obtained by filtering signal with wavelet functions of different scales, where scale represents measure of stretching of wavelet function in time.

The pseudo frequency components are obtained from scales by formula:

$$f_q = \frac{f_c}{sc \times \Delta} \quad (1)$$

where f_c is central frequency of chosen wavelet (for example f_c=0.8125Hz for Morlet wavelet), sc represents scale and Δ is sampling period of the signal [21]. Usually the CWT of ECG signal is presented in so called semi-log axes - the frequency axis is logarithmic and time axis is linear (see Figure 1). If the spectrogram (scalogram) is to be presented in logarithmic axis from frequency 2^{-1} to 2^7 Hz (scales from 1560 to 6.1 with use of $\Delta = 1/960$ s) in s (for example 65) scales, s filters has to be derived from chosen wavelet function. The orders of these filters for Morlet wavelet (of order 16) are from 98 (for frequency 2^7) to 24961 (for frequency 2^{-1}) and hence the complexity of calculating CWT is large. The number of filters determines y axis resolution - the larger number the better resolution. Final number of calculated coefficients is $s \times N$ where N is the total number of samples of recorded signal. The application dialog for selecting frequency range and resolution is presented in Figure 2c.

4 Multi-threading

The multi-threading in PC-based ECG device is performed by using three threads: the main window standard thread, the worker thread and the GDI - graphic thread.

The main window standard thread starts with starting of application execution. This thread can work stand alone. In the thread a user can open the database, search the particular patient by name, date and diagnosis, open the ECG waveforms of patient found, measure the time width of particular segments, and analyze the waveform already stored in both: time domain and time-frequency domain using FFT, DWT and CWT in standard mode. A cardiologist can analyze recorded standard ECG waveforms and change diagnosis and add some comments. The printing function is also controlled by this thread.

When a user wants to monitor ECG signal of a patient in real-time, by pressing appropriate button in application, the working thread and timer are started (by function *OnTimerStart*) and the main thread is then directed to execute function *OnTimer* periodically.

The communication between the acquisition module and a PC is available only by using the working thread. This thread represents listener because it listens for data appearance on appropriate serial communication port of PC that are sent by the acquisition module. When data appear on the port, the working thread copy them to the transmission buffer which task is to copy data from one thread to the other. The transmission buffer holds the data that can be accessed only one thread a time. As the size of the buffer is initialized at the program start, the varying size of currently transmitted data between threads is controlled by unsigned integer variable that shows how many bytes are copied to the buffer.

The main thread periodically (on timer ticks) checks for the new contents of the transmission buffer, copies data to the local buffer, resets the variable that shows how many bytes are copied from the buffer to zero and forwards data from local buffer to the drawing function. The drawing function then draws the portion of the signal based on just received data on the main screen using appropriate sizing of signal values. At this point, it is important to say that not the whole screen is re-drawn after reception of new data, but only the portion of it that corresponds to the new data size. As the time axis of the main screen is predefined to be 6 seconds, and the sampling frequency of input data is known, it is easy to calculate the window region that should be refreshed (MFC function *invalidaterect()*) upon reception of the new data. The timer frequency is set to be 25 ticks per second that results in about 40 samples per tick. If the ECG signal in those samples is "polluted" by public power network frequency of 50Hz, the band pass IIR filter will eliminate this noise and then the samples of "clean" signal are forwarded to the drawing function. The speed of modern computers allows this filter to be implemented in real-time without jeopardizing continuity of signal drawing on the main screen.

Unfortunately, the time-frequency transform and drawing spectrograms (scalograms) cannot be always executed in real time by processing samples of signal in the main thread after drawing the signal in time domain. By introducing GDI thread that works independently of the main thread, it is possible to copy new samples from the main to the GDI thread and to let the GDI thread calculate spectrum portion of the signal and present the values of its coefficients to the spectrum screen simultaneously with the main screen drawing of the signal waveform performed by main thread. The diagram of running multi threads in our PC-based ECG device is presented in Figures 3 and 4.

4.1 Threads synchronization

To synchronize work of three threads in parallel, we used six "Event" variables. This type of variable can have two states - "Set" and "Reset". Appropriate built-in functions: *WaitForSingleObject* and *WaitForMultipleObjects* that take parameters of "Event" type, enable checking of event states. The both functions can control the workflow of execution of the program lines within the thread based on the state of event variable they control. The input parameters of the functions are: the name(s) of event(s) they control and the time the function has to wait for event to change state from Reset to Set. If the input event variable changes the state to "Set" within the waiting time interval, the program executes one block of program lines - at the "detected Set event" output of the function, and if state of event variable remained "Reset" within the time interval, the program executes another block of program lines - at the "time out" output.

The behavior of these functions is symbolically presented by yellow decision modules in Figures 3 and 4, where "Yes" output directs the program flow to the first block of lines and "No" output directs to the second.

Two event variables - *MainWEvent* and*RdrEvent* are used to synchronize copying data between the working and the main thread. *MainWEvent* is controlled by the main thread (only main thread can change its state) and *RdrEvent* is controlled by the working thread. The working thread is started by command *OnTimerStart* in the main thread and *MainWEvent* is initiated to the "Set" state. It enables the working thread to copy data from serial port to transmission buffer_1 before the main thread starts checking the content of transmission buffer_1 (then the *MainWEvent* is put in "Reset" state). When data are received at the serial port in working thread, this thread resets the *RdrEvent* (by this action it forbids the main thread to access transmission buffer_1) and checks the state of *MainWEvent.* If the state of this event is "Set", data from local buffer (of this thread) are copied to transmission buffer_1. As the state of *RdrEvent* is "Reset" during copying, the main thread in Timer function is stopped for time period of 1ms. If the copying is finished before the waiting time expires, the reader thread change the state of *RdrEvent* to "Set", and enables main thread to copy data from transmission buffer_1 to its local buffer (see Figures 3 and 4).

It may happen that both variables - *MainWEvent* and *RdrEvent* are simultaneously in "Reset" state and both wait for each other to change the state. The working thread is set to wait for the state of *MainWEvent* infinitely, and the execution of this thread literally stops when *MainWEvent* is in "Reset" state, but the main thread is set to wait for the change of *RdrEvent* state just 1 ms. When 1 ms passes, the main thread resolves the conflict by changing the state of *MainWEvent* and thus enabling the working thread to continue.

The situation when both variables are simultaneously in "Set" state is impossible, because before checking the state of variable that is controlled by another thread, each thread resets the

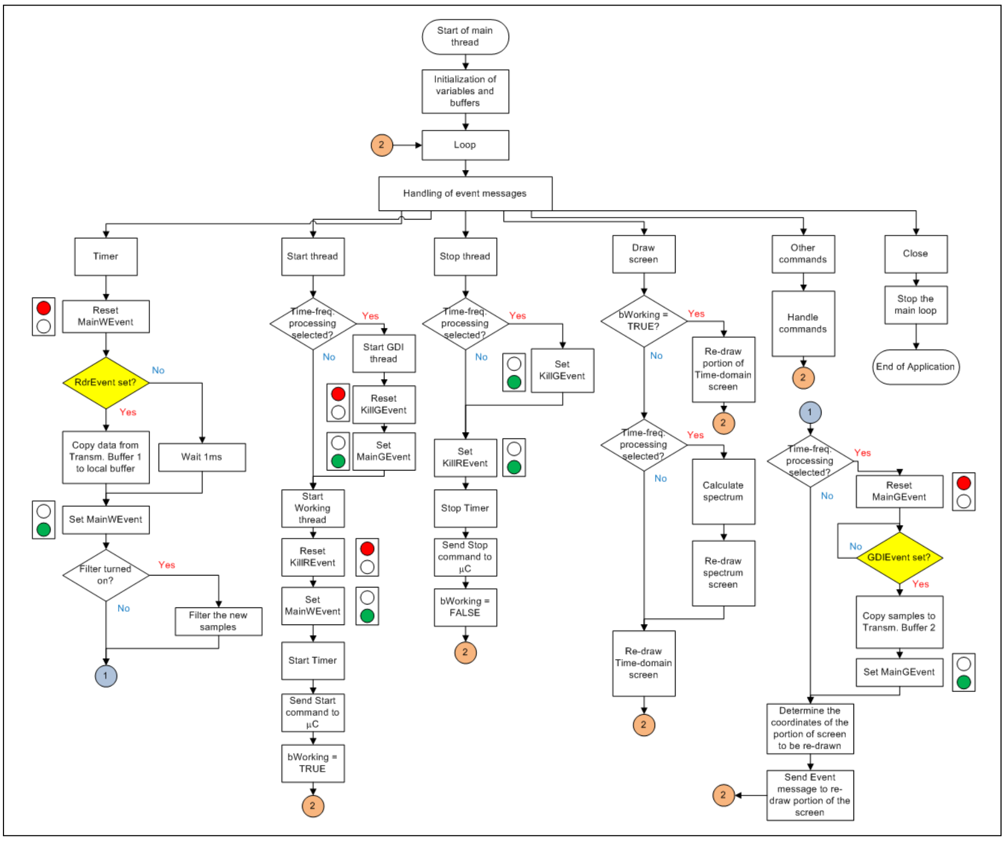


Figure 3: The work-flow diagram of the main thread

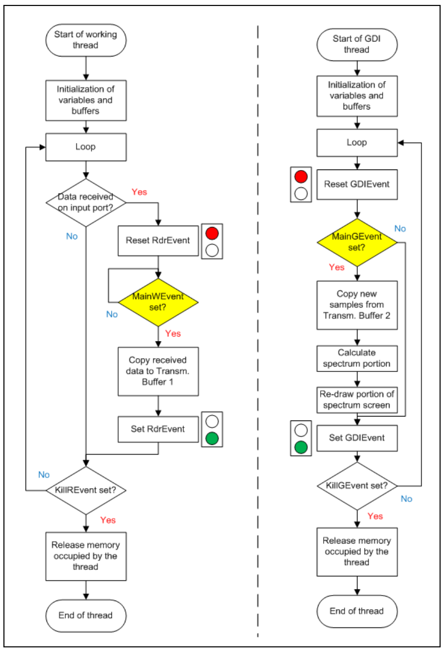


Figure 4: The work-flow diagram of the working (left) and graphical (right) threads

own event variable. Setting and resetting of event variables are presented in Figures 3 and 4 by small semaphores (traffic lights) with red and green lights.

The synchronization between the main and the graphical thread is organized in the same way as the synchronization between the main and the working threads. Here we have another two event variables - *GDIEvent* (controlled by the GDI thread) and *MainGEvent* (controlled by the Main thread). The transmission of data between these threads is performed through transmission buffer_2.

As can be seen from the Figures 3 and 4, the task of the working thread is to receive data from serial port and to copy it to the main thread. The main thread can draw the waveform portion of input samples to "Time-domain" screen in real-time and, if time-frequency analysis is selected, forward the samples to the graphical thread to perform calculation of signal spectrum and present spectrogram (scalogram) on separate screen window.

The role of the event variable *KillREvent* is to control running of program loop in working thread. The state of this variable is controlled by the main thread. When the working thread starts, the initial value of *KillREvent* is "Reset" and it enables the program lines in this thread to be executed in the loop. But when the state of the event has been changed, the loop in the working thread is finished and this thread is closed. The situation of closing the graphical thread is the same as for closing the working thread, but the variable *KillGEvent* is used.

5 Experimental Results and Discussion

In order to test efficiency of multi-threading, we recorded the waveform samples of ECG signal of 72 seconds duration (equal to 69120 samples). Then we tested the response of the software by sending recorded samples from another computer in four different baud-rates: 19200, 38400, 57600 and 115200 bits/second. The software is modified in a way that user can choose to see scalogram that will be 1) drawn in the same thread where the ECG waveform is drawn (we call this method "single") or 2) drawn in another (GDI) thread (we call this method "multi").

We measured time needed to main thread executes *OnTimer* function where:

- data are copied from working thread,
- filtering of the received samples is performed (using IIR filter) [17, 18],
- connection to the database of patients,
- the coordinates of ECG waveform are calculated and drawn,
- samples are sent to GDI thread (in the case of multi-threads) or wavelet transform is calculated and coefficients are drawn in separate portion of the screen (in the case of single thread).

We used the following C++ instructions for calculation of time of execution of *OnTimer* function:

```
void CECGView::OnTimer(UINT nIDEvent){
  LARGE_INTEGER ntime1,ntime2;
  LARGE_INTEGER freq;
  QueryPerformanceFrequency(&freq);
  QueryPerformanceCounter(&ntime1);
  . . .
  Set of commands in OnTimer function
  . . .
  QueryPerformanceCounter(&ntime2);
  ntime = (ntime2.QuadPart-ntime1.QuadPart)/(freq.QuadPart/1000);
}
```

In this way we were able to measure time between beginning and ending of function *OnTimer* with accuracy of 1ms. The wavelet we used in our experiments was Daubechies 4 with 8 coefficients and the window width was set to 32 samples.

The experiments were performed on 8 different computers with different processors, different graphical cards and different services running in background. The purpose of making this experiment was not to compare the speed of these computers, but just to compare the speed of running program in single and multiple threads in different baud rates. For each computer 8 different experiments were performed by changing baud-rate and number of threads. The number of samples received during one call of *OnTimer* function, duration of execution of instructions within the function and time step between two successive calls of the function for baud rate of 19200 bauds/second in the single thread is given in the Table 1.

Table 1: The measurements of number of samples received, duration of *OnTimer* function and time step between two succesive calls of *OnTimer* function in single thread

Number of samples received	Duration of OnTimer function	Time step (ms)
48	0	47
48	1	47
48	0	47
32	0	47
48	1	47
48	0	47
...	...	...

By averaging number of samples received in one call of *OnTimer* function we obtained the following values: 44.5, 88.5, 134 and 267 samples for baud-rates of 19200, 38400, 57600 and 115200 bits/second respectively. The average time between two successive calls of the function was 46.8ms for modern PC and 40ms for old computer with Intel Celeron working on frequency of 800 MHz (the timer clock was set to trigger function in 40ms for all computers). The average times of execution of *OnTimer* function in milliseconds are given in the Table 2.

Table 2: Average execution time in [ms] of *OnTimer* function in different computers

Computer	**Threads**	**19200**	**38400**	**57600**	**115200**
Intel I7 2600K	Multi	0.000	0.000	0.000	0.000
	Single	0.000	0.000	0.022	1.012
Compaq 8710p	Multi	0.007	0.008	0.008	0.009
	Single	1.172	2.640	3.932	7.210
Vostro 1510	Multi	0.007	0.007	0.013	0.012
	Single	1.340	3.109	2.314	4.837
Celeron 800MHz	Multi	0.000	0.000	0.000	0.016
	Single	1.422	2.357	3.749	8.112
ProBook 6460b	Multi	0.001	0.001	0.003	0.035
	Single	0.503	1.707	6.172	11.102
AMD AI 3000+	Multi	0.000	0.000	0.000	0.062
	Single	0.391	1.076	2.094	4.517
AMD AII 5000+	Multi	0.005	0.006	0.115	0.115
	Single	0.077	1.515	2.430	4.617
Intel Atom N450	Multi	0.109	0.124	0.147	0.168
	Single	4.148	9.048	13.674	29.955

When the software was executed on Intel Atom netbook computer, with single thread and baud-rate of 115200 bauds/second, after several seconds of running, the execution time of *OnTimer* function rose to 94ms and running of the software was blocked.

It is obvious from the Table 2 that calculating coordinates for drawing waveform, performing filtering of ECG samples and passing samples to GDI thread takes less time than with addition of calculating spectral coefficients and drawing them in the same thread. In the case of slower computers the next call of *OnTimer* function can happen before the previous running of the same function is finished. In this case both instances of the function will need access to the same buffer. Then running of the software is blocked.

6 Conclusion

The PC based ECG device is a solution for real time complex signal processing in analysis of ECG signal. It provides presentation of ECG signal in Time and Time-Frequency domain including filtering of signal and database connection.

We have designed source and processing modules to enable parallel processing of incoming data by using multi-threading technology on a PC. The capturing of incoming data, displaying data in Time domain and calculation of spectral coefficients and its presentations in Time-Frequency domain through spectrograms (scalograms) is realized in three threads respectively: working, main and graphical. The calculation of spectral coefficients is performed using three Time-Frequency transformations: Short Time Fourier Transform, Discrete Wavelet Transform and Continuous Wavelet Transform.

The experiments performed prove that using of multi-threads obviously enable larger number of calculations in shorter time compared to the same number of calculations using single thread.

The proposed multithread concept enables the real time display of signals spectrum and waveform, encourages and provides ample evidence to pursue multi-thread architecture as a serious candidate for real-time signal processing.

Acknowledgement

This work has been done within the project - 'Optimal Software Quality Management Framework', supported in part by the Ministry of Science and Technological Development of the Republic of Serbia under Project No.TR-35026.

Bibliography

[1] Ferrero, A. ; Software for Personal Instruments, *IEEE Trans. On Instrum. And Measurement*, 39(6): 860-863, 1996.

[2] National Instruments, Instrumentation: reference and catalogue, Austin, Texas, USA, 1996.

[3] House, R. ; Choosing the Right Software for Data Acquisition, *IEEE Spectrum*, March, 23-40, 1995.

[4] Kunzmann, U. et al ; Parameter extraction of ECG signals in real time, *Biomedizinische Technik*, Vol. 47, 2002.

[5] Marchesi, C. et al ; ECG processing algorithms for portable monitoring units, *The Internet Journal of Medical Technology*, 1(2), 2004.

[6] Tarvainen, M. et al ; Time varying analysis of heart rate variability signals with Kalman smoother algorithm, *Physiological Measurement*, 27(3): 225-239, 2006.

[7] Ilić, S. ; Comparison of Compression Ratios for ECG Signals by Using Three Time-Frequency Transformations, *Facta Univ. Ser.: Elec. Energ.*, 20(2): 223-232, 2007.

[8] Ilić, S. ; Detection of the Left Bundle Branch Block in Continuous Wavelet Transform of ECG Signal, *Electronics And Electrical Engineering*, 2(74): 33-36, 2007.

[9] Ilić, S. et al ; Improvement in decision-making in emergency medicine by using PC-based system for electrocardiography, *Tehnika*, 3, 2011.

[10] Addison, P. et al; Evaluating arrhythmias in ECG signals using wavelet transforms, *IEEE Engineering in Medicine and Biology*, Vol. September/October, 104-109, 2000.

[11] Ustun, T. et al; Real-time processing for Fourier domain optical coherence tomography using a field programmable gate array, *Review of Scientific Instruments*, 79, 2008.

[12] Gross, S. et al; RealTimeFrame A Real Time Processing Framework for Medical Video Sequences, *Acta Polytechnica*, 48(3): 15-19, 2008.

[13] Eom, T. et al; Real time display Fourier-domain OCT using multi-thread parallel computing with data vectorization, *Optical Coherence Tomography and Coherence Domain Optical Methods in Biomedicine*, http://dx.doi.org/10.1117/12.874476 , 2011.

[14] Pan, Y. et al; Continuous Wavelet Transform On Reconfigurable Meshes, *Proc. of the 15th International Parallel and Distributed Processing Symposium (IPDPS 01)*, 1158-1163, 2001.

[15] Thulasiraman, P. et al; A fine-grain load-adaptive algorithm of the 2D discrete wavelet transform for multithreaded architectures *Journal of Parallel and Distributed Computing*, 64: 68-78, 2004.

[16] Diduch, L. et al; A Framework for Modular Signal Processing Systems with High-Performance Requirements *IEEE Int. Conf. on Multimedia & Expo (ICME)*, Beijing, China, 1159-1162, 2007.

[17] Žorić, A. et al; PC-based system for electrocardiography and data acquisition, *Scientific Research and Essays*, 7(4): 468-476, 2005.

[18] Žorić, A. et al; Wireless electrocardiography system, *Proc. of 7th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services TELSIKS*, Niš, Serbia and Montenegro, 7(4): 468-476, 2012.

[19] Dunbar, M.; Plug-and-Play Sensors in Wireless Networks *IEEE Instrumentation and Measurement Magazine*, 4(1): 19-23, 2001.

[20] Smith, W.; *The Scientist and Engineers Guide to Digital Signal Processing*, California Technical Publishing, 1997.

[21] Misiti, M. et al; Wavelet Toolbox For Use with MATLAB,
http://www.mathworks.com/help/pdf_doc/wavelet/wavelet_ug.pdf

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):560-570, August, 2013.

A Model for Collaborative Filtering Recommendation in E-Commerce Environment

Y. Jing, H. Liu

Yuanchun Jing & **Honglu Liu**
School of Traffic and Transportation
Beijing Jiaotong University
PO Box 100044, Beijing, China
12121021@bjtu.edu.cn, hlliu@bjtu.edu.cn

Abstract: In modern business environment, product life cycle gets shorter and the customer's buying preference changes over time. Time plays a more and more important role in collaborative filtering. However, there is a gap in one class collaborative filtering (OCCF). On the basis of collecting different real-time information, this paper proposes an optimization model for e-retailers. Through comparing different methods with different weights, results show that real-time dependent in OCCF performs better in improving the quality of recommendation. The model is effective in cross-selling e-commerce, personalized, targeted recommendation sales.
Keywords: Integration of real-time information, one class collaborative filtering, e-commerce.

1 Introduction

With the rapid development of Internet and the increase of the amount of data on the Internet, Retail businesses face many challenges, such as assessment of customer interests to products, as well as an accurate assessment of products at the market's perspective in modern life. These challenges in the transaction process, often translate into how to find the best match between the customer and the product. In recent years, many companies began to get involved in e-commerce mode, making great progress. According to the "2012 China online retail market data monitoring report" released by the third-party e-commerce research institutes, China Electronic Commerce Research Center (100EC.cn): China online retail market transactions reached 1.3205 trillion RMB, attaining a year-on-year growth of 64.7% by December, 2012; this report also predicts: online retail market turnover of 2013 is expected to reach 1.8155 trillion RMB. As of 2012, China's online retail market transactions have been accounted for 6.3% of the same period's total retail sales of social consumer goods, while the proportion data was only 4.4% in 2011, which means that e-commerce has begun to change the retail landscape.

Despite a huge opportunity of e-commerce development, retailers have to solve complex problems which at a higher level. i.e., in recent years, because of the increase of customers and product, the retailers need to develop a systematic recommendation list. The list may be a list of items for the target customer, or a customer list for the target item. Collaborative filtering (CF) is the method that an associated list generated by the information from customer clearly expressed their interests or extracted from the potential customer behavior information. One of the most common research problems is the Netflix movie recommendation in this area.

Through huge amount of datasets, like purchases transaction datasets, news recommendation records of recommended sites, the problem to be solved can be converted to one class collaborative filtering (OCCF) problems. One class collaborative filtering is a problem with fewer researches, the effectiveness based on one class collaborative filtering depends on the consideration and processing of non-labeled or missing dataset. i.e., its basic idea is to construct a negative dataset (negative sample). As for the one class collaborative filtering problems which have weight distribution application for the matrix factorization technique, recent studies are often based on frequency information, such as customer number, product number, product popularity.

In spite of extensive literature researches we found little study in which similar real-time information is combined, undeniable, real-time plays an important role in the OCCF. Therefore, in this paper, we built a recommended model which can make full use of the real-time information about product and customer in the market. Through the integration of real-time information including: the information of product launched into the market which is related to product life cycle, the information about product itself, and customer recently visiting information related to the customer relationship management, this model can be divided into two aspects: Real-time customer-oriented weight and Real-time product-oriented weight, to improve the quality of one class collaborative filtering (OCCF) for electronic retailers and meet their demand of the online recommendation.

The rest of this paper is organized as follows: Section 2 contents relevant concepts; in Section 3, we proposed a real-time dependent model, added the weight to the model and compared seven different methods with different weight; Section 4, we use the MovieLens dataset to test the model and get results; in Section 5, conclusion and the future work.

2 Relevant concepts

Nowadays, customers are presented with a majority of options for products and services. On the other hand, a vendor in the e-retailer can reach lots of customers what means that there is no limiting factor between the e-retailers and global customers. So, in today's e-retails' competition, the winner may be the one, which can offer a highly level of personalization to an individual customer. For the personalization needs recommendation from the e-retailers, a good recommendation system is the key factor for them to solve the problem. The recommendation algorithm most widely used in the recommendation system is collaborative filtering, namely, CF.

2.1 Collaborative Filtering (CF)

The kernel of collaborative filtering problem is analysis of user interests, and finding some similar (interest) users for the specified user in the user group. After combining these similar users' evaluations to products and developing a recommendation system with this information, the e-commerce site can make prediction to the specified user preference.

The earliest defined "collaborative filtering" are Goldberg and his partners [1]. They developed the first recommendation system, Tapestry, mainly for coordination and filtering newsgroups streaming media files. Another system to study news item filtration, GroupLens, developed by Resnick, etc., similar to Tapestry. GroupLens is mainly for scoring, to test the level of users' interests on the news item. Based on a similar approach, GroupLens project team then got down to MovieLens project, from the filtration problem of news items transferred to movie recommendation problem.

In recent years, collaborative filtering algorithm has been widely studied at home and abroad. Such as, Li G. and Li L. (2012) [2], Shani G., D. Heckerman, and R. I. Brafman (2005) [3], Banati H. and Mehta S. (2010) [4]. On the basis of different technologies, CF system is mainly divided into two categories, one is based on storage, and the other is model. Pearson correlation approximation method can be applied to the collaborative filtering based on storage technology, this kind of CF can also make use of the item-to-item's Top-N recommendation method, which is widely used in e-commerce, like www.Amazon.com, www.jd.com, etc. that recommended for users of all kinds of goods they may like, such as books, audio-visual products, electrical appliances, clothing.

Model-based CF approach makes up for many deficiencies of the one based on the storage. This method usually takes steps: firstly, find the clear or potentially preferences complex patterns between users and items. Then, collect historical data, and provide recommendation model suggests, such as Bayesian model, relies on the network based on the model, clustering model, and the model based on the MDP.

2.2 One Class Collaborative Filtering (OCCF)

According to different data, collaborative Filtering processes can be divided into two categories: one category is the clear preference data processing, for example the score; the other is implicit data's processing, for instance page hits or not. The later always exists in the real world application environments widely, such as whether the user bought a product, whether the user clicked a web page or not. Because of no requirement that provide a clear score, the users, obtain this data much easier than the ratings data. Besides that, in the implicit datasets, only active factors can be clearly distinguished, while the negative cases are uncertain. So this kind of problem is called one class collaborative filtering (OCCF) problem. One class collaborative filtering's task is to rank recommendation sets according to the user's preference levels by analyzing the implicit information with the particular user's favorite. Despite easy acquisition for such data, it is difficult to explain. For example, as for the data of a user clicks the website, among all data, the data which consists of the user clicks the pages can be interpreted as active factors; the rest of the data is the mix of negative data and missing data. How to explain such combined data, and how to process this data after explanation, are the main difficulties of one class collaborative filtering's current studies.

Compare to the CF, the researches about one class collaborative filtering are fewer. We learned something from those few literatures, for example, Yang S. and Xue W. [4]. As for one class collaborative filtering problems, due to the active dataset rarely (sparsity), while the other two types of datasets, negative and missing datasets, are very confusing, it has many difficulties to future research.

Negative and unobserved data processing is the key issue of one class collaborative filtering. For this issue, a large quantity of research focuses on the technology of weight distribution, because the weighted low rank approximation can improve the quality of unobserved data recommendations. In this aspect, the typical studies contain: Pan, etc. proposed a matrix factorization model based on different weight distribution of potential factors [5]. Sindhwani used the formula simplifies weight distribution scheme, proposed another optimization variables, providing an unobserved data processing measure [6].

2.3 Real-time research

Concerning CF problems, most of models regard users' behaviors as static, for example, a customer who buys item A will buy item B. However, product has its product life cycle, and the customer's preferences will change over time, therefore, to regard these problems as static has certain deficiency. In 2009, Koren proposed a real-time changing model which is based on the evaluation of Netflix recommendation in the report of improving the performance [7]. Recent some related studies began to consider the real-time information, such as, the research of Lu, Agarwal, and Dhillon (2009) [8]; Xiang L. and Yang Q. (2009) [9]; Xiong L., X. Chen, T. Huang, J. Schneider, J. G. Carbonell (2010) [10]. But their studies focus on the CF, few mentioned the OCCF. Even few reports establish the problem that real-time information applied to the system of OCCF.

In one class collaborative filtering field, previous studies have only considered frequency-based information, there is few research which designs real-time information to expand collaborative filtering problem. So, this paper, on the basis of exploration the information of product launched into market and the customers' recent access information, considered the product life cycle (PLC) and Customer Relationship Management (CRM) real-time information Our model shows that predictions accuracy can be improved by OCCF recommendation model of e-commerce based on real-time information. Real-time information integration has significant significance for business processes such as cross-selling, up-selling and others' accurate recommendations.

3 Real-time dependent model

3.1 The notation and formula

To facilitate the study and discuss, firstly, we presume some items as follows.

X: Binary representation of actual transaction matrix, 1 means purchase; 0 is no purchase

U: User feature matrix with latent features of customers

P: Product feature matrix with latent features of products

Y: Recommendation resultant matrix

W: Weight matrix: $\in [0, 1]$

We are given that there are m customers, n products, then, X matrix has m rows and n columns. If i^{-th} row and j^{-th} column in the X matrix has a value of 1, it means that customer i purchased a product j. If it is 0, it means that this particular customer-product has no purchase record. As for the actual transaction, X matrix is very large, and there is a certain degree of sparsity, X manifests itself in a display that less 1 more 0.

In the matrix, r is the rank; it needs to analyze the number of potential characteristics. Presume U is on behalf of user feature matrix, so, $U = [u_1, u_2, \ldots, u_m]^T$ is a $m \times r$ matrix. i^{-th} in the matrix U means a customer, u_i means customer i in r-dimensional customer feature space. In the same way, supposes that P is in the name of product feature matrix, thus, $P = [p_1, p_2, \ldots, p_n]$ is a $r \times n$ matrix. j^{-th} in the matrix P stands for a product, p_j means product j in r-dimensional product feature space. Typically, there is $m, n \gg r$ and $m \times n \gg (m + n) \times r$. Joint matrix is expressed by $u_i^T p_j$ which represent the joint relationship between customers and products. Let's assume $Y = UP$, therefore, the problem can be converted into the following optimization problem:

$$\mathop{\arg\min}_{U \geqslant 0, P \geqslant 0} \sum L(X, Y) \tag{1}$$

In the formula, L is the square error function or other loss functions:

Square error:

$$L(X, Y) = \|X - Y\|^2 = \sum_{i=1, j=1}^{m,n} (X_{i,j} - Y_{i,j})^2 \tag{2}$$

Relative entropy loss:

$$L(X, Y) = D(X \| Y) = \sum_{i=1, j=1}^{m,n} \left(X_{i,j} \log \frac{X_{i,j}}{Y_{i,j}} - X_{i,j} + Y_{i,j} \right) \tag{3}$$

In order to prevent over-fitting, parameter λ is introduced in the formula (1), to revise the optimization model:

$$\mathop{\arg\min}_{U \geqslant 0, P \geqslant 0} \lambda \left(\|U\|_F^2 + \|P\|_F^2 \right) + \sum L(X, Y) \tag{4}$$

Among it, $\|U\|_F^2$ and $\|P\|_F^2$ are the U and P matrices' F norm.

Collaborative filtering system is primarily intended to provide customers with similar product recommendations. The above-mentioned optimal formula major considers the situation that just value of 1. Thus, In terms of the one class collaborative filtering, this formula is not perfect. In one class collaborative filtering, describing the potential customers' preferences to a variety of products or different importance degrees of different data sets through simply the weights setting.

Therefore, adds the weight to original optimization model:

$$\arg\min_{U\geqslant 0,P\geqslant 0} \lambda\left(\|U\|_F^2+\|P\|_F^2\right)+\sum WL\left(X,Y\right) \tag{5}$$

Srebro and Jakkola applied the low-rank approximation based on weights in collaborative filtering model with two extremes weights [11]: active factors has a weight of 1, other factors have weight of 0. Supposes X^1, $X^1=\{(i,j):X_{i,j}=1\}$ which contains (i,j) pairs corresponding all 1of a matrix X. X^0, $X^0=\{(i,j):X_{i,j}=0\}$ containing (i,j) pairs corresponding all 0 (no purchase) in X matrix.

According to the above weight setting scheme, the formula can be further amended to this:

$$\arg\min_{U\geqslant 0,P\geqslant 0} \lambda\left(\|U\|_F^2+\|P\|_F^2\right)+\sum_{i,j\in X^1} W_{i,j}L\left(X_{i,j},u_i^Tp_j\right) \tag{6}$$

Among it,

$$W_{i,j}=\begin{bmatrix} 1 & \forall\,(i,j)\in X^1 \\ 0 & \forall\,(i,j)\in X^0 \end{bmatrix}$$

However, the model ignores the non-buyers groups. In order to comply with the requirement of one class collaborative filtering that settings different weight for different types of factors, the above formula (6) further revised as follows:

$$\arg\min_{U\geqslant 0,P\geqslant 0} \lambda\left(\|U\|_F^2+\|P\|_F^2\right)+\sum_{i,j\in X^0} W_{i,j}L\left(0,u_i^Tp_j\right)+\sum_{i,j\in X^1} W_{i,j}L\left(1,u_i^Tp_j\right) \tag{7}$$

The formula covering all types of customers by setting the weighting values $W_{i,j}$: good customers, potential customers and non-customers. After setting the corresponding weights, the equation (7) can be rewritten as the following equation, similar to the model provided by Sindhwani:

$$\arg\min_{U\geqslant 0,P\geqslant 0} \lambda\left(\|U\|_F^2+\|P\|_F^2\right)+\|\Omega\otimes(X-UP)\|_F^2 \tag{8}$$

$\Omega_{i,j}=\sqrt{W_{i,j}}$, $\otimes$ means for computing according to element, it is to make the two different corresponding elements within matrices multiplied.

Finally, according to the steps described by Lee and Seung [12], final optimization scheme is made by multiplication replaced:

$$P=P\otimes\frac{U^T\left(\Omega\otimes X\right)}{U^T\left(\Omega\otimes(UP)\right)+\lambda P} \tag{9}$$

$$U=U\otimes\frac{\left(\Omega\otimes X\right)P^T}{\left(\Omega\otimes(UP)\right)P^T+\lambda U} \tag{10}$$

3.2 Weight setting scheme

Time weight applied to the collaborative filtering is more and more popular, we can find it from the previous studies. Such as, the research of Lan W. and Zhengjun Z. (2007) [13]; HuaizhenY., Xiaoqi C. and Meilian. L. (2009) [14]; Donghui L., Dewei P. and Hui, Zh. (2012) [15].

In our research, we compared the real-time weight with basic methods.

1) Basic methods

The basic method used by initializing values and other relevant information can be summarized as follows:

(1) 0 weight: weight matrix is a determine matrix similar to the transaction matrix, 1 is the weight of all transactions entered, 0 is the weight of all non-transaction entries.

(2) Full weight: whole weight matrix is 1. Error function in the formula not only needs to calculate the maximum error of transaction entry, but also need to calculate the maximum error of non-transactional input.

(3) Uniform weight: unlike the two extremes of weight setting, the method is to set a smaller weight $\delta \in (0,1)$ for all non-transaction input.

(4) The customer oriented weight: the weight of non-transaction setting is proportional to customer transactions. Associated customers weight is calculated as follows:

$$\delta_{i,j} \propto \frac{\sum_j^n X_{i,j}}{\left[\max\left(\sum_j^n X_{i,j}\right)\right]}$$

$\delta \in (0,1)$, If a customer has many products' historical purchase records, namely, a lot has been tagged data, that non-trading or unlabeled data, is likely to be treated as a customer non-purchases situation.

(5) The product oriented weight: similar to the method (4), weight setting in proportion to the number of traded products. Calculated as follows:

$$\delta_{i,j} \propto \left\{1, \frac{\sum_i^m X_{i,j}}{\left[\max\left(\sum_i^m X_{i,j}\right)\right]}\right\}$$

After obtained the results, using subtraction allows $\delta \in (0,1)$. The main principle is that if the transactions number of a product is small, and most of situations that the non-trading or missing data will be treated as the case of non-purchases.

2) Real-time weight

(1) Real-time customer-oriented weight: In addition to transaction data set, it also involves the customer's recent visiting. Record the history recently visiting vector as ζ^{CR}, which stands for that customers' recently visited records.

ζ^{CR}, It is the vector that customer recently visited distance; it can be expressed as follows:

$\Delta^{CR} = \tau_c - \zeta^{CR}$ Among it, τ_c is the current or assessment phase.

$$\delta_i^{CR} \propto \left[1 - \frac{\Delta_i^{CR}}{\max\left(\Delta^{CR}\right)}\right] \text{So}, \delta \in (0,1)$$

In the formula, δ_i^{CR} is customer i recently visited value based on the real time and the weight value.

The weight of each customer in real time is δ, $\delta \in (0,1)$. This means that if a customer has recently visited the e-commerce site, then compared to those who have not recently visited the site of all customers, he is more likely to become frequent customer.

(2) Real-time product-oriented weight: Reference different time information that products launched into market, set up a vector ζ^{PL}.

ζ^{PL} It is the vector which stands for the time information that products launched into market. It can be expressed as:

$\Delta^{PL} = \tau_c - \zeta^{PL}$. In the formula, τ_c is the current or assessment phase.

$$\delta_j^{PL} \propto \left[\frac{\Delta_j^{PL}}{\max\left(\Delta^{PL}\right)} \right] \text{So}, \delta \in (0,1)$$

δ_j^{PL}, it is the weight setting that product j based on the real time launched into the market.

If a product exists long time in the market, it will attract many customers. Because of innovation emerging, imperfect goods will have a longer product life cycle. In any retail establishment, the records that products launched into the market exist in the internal management organization, they are the information which can be collected.

In summary, different methods of setting the weights can be shown in the following table:

Table 1: Different weight-setting methods

	Weight($W_{i,j}$)	
methods	Transaction(1)	No-transaction(0)
0 weight	1	0
Full weight	1	1
Uniform weight	1	$\delta, \delta \in (0,1)$
The customer oriented weight	1	$\propto \sum_j X_{i,j}$
The product oriented weight	1	$\propto \left(m - \sum_j X_{i,j}\right)$
Real-time customer-oriented weight	1	$\propto \delta_i^{CR}$
Real-time product-oriented weight	1	$\propto \delta_j^{PL}$

4 Experiment and Results

The algorithm of OCCF based on real-time information model can be showed:
Input: customer-product matrix X, the rank r.
Output: the approximation matrix Y of X.

1) Initialize P with a random number less than 1;

2) Repeatedly using the formula (9), (10). update the U and P until the AUC value calculated has convergence;

3) $Y = UP$ return Y;

4) Using matrix Y make a recommend list.

This paper selected the area under the ROC curve (AUC) as a quality measure to compare measuring recommendation quality of different methods.

$$AUC = \frac{1}{|U|} \sum_u \frac{1}{|E(u)|} \sum_{(i,j) \in E(u)} \rho(\hat{x}_{ui} > \hat{x}_{uj})$$

ρ, is an indicator function, it can be showed like:

$$\rho(\alpha) = \begin{bmatrix} 1 & \text{if } \alpha \text{ is true} \\ 0 & \text{else} \end{bmatrix}$$

$E(u)$, is the target pair of evaluation.

$$E(u) = \{(i,j) | (u,j) \in S_{test} \wedge (u,j) \notin (S_{test} \vee S_{train})\}$$

For all evaluations, we use the frequently-used datasets, the MovieLens dataset that GroupLens research group provided. The dataset contains 943 users and 1682 films; each user has at least 20 film scores. Thus, the total of assessment records is 100,000. Despite huge number, this data set is very sparse, sparsity is 6.305%, which is only 6.305% of the items are rated. The simulation environment includes: the PC with windows 7, Intel Core Duo processor and 4G RAM.

When we make the $r = 3$, Figure 1 is the typical results for all data sets, with each method running.

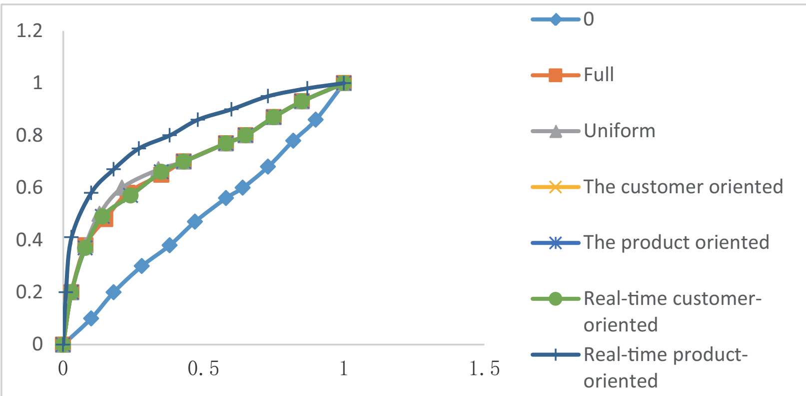


Figure 1: An AUC comparison that all methods in a single run.

Obviously 0 weight is the worst, its' the ROC curve almost shows the diagonal. In all other ways, the degree of improvement in performance is similar, we can learn from the shape of the ROC. In the new methods, the real-time product-oriented weight method's ROC curve is the best of all. Compare with other methods, the real-time customer-oriented weight method has some performance improvement.

Then we repeated 10 times by changing the rank and got the average value of AUC. Like table 2 and Fig 2 shows:

From the figure and table, we found that 0 weight methods performed badly, while the Real-time customer oriented method and Real-time product oriented method had good effect for sake of the recommend.

Table 2: The average AUC of different rank and methods

		rank					
		1	3	6	9	12	15
methods	0	0.51	0.501	0.505	0.497	0.494	0.49
	Full	0.71	0.699	0.697	0.693	0.689	0.687
	Uniform	0.699	0.712	0.71	0.708	0.705	0.702
	The customer oriented	0.692	0.701	0.698	0.696	0.692	0.693
	the product oriented	0.692	0.702	0.699	0.698	0.695	0.694
	Real-time customer-oriented	0.697	0.718	0.714	0.71	0.709	0.704
	Real-time product-oriented	0.779	0.78	0.778	0.78	0.781	0.779

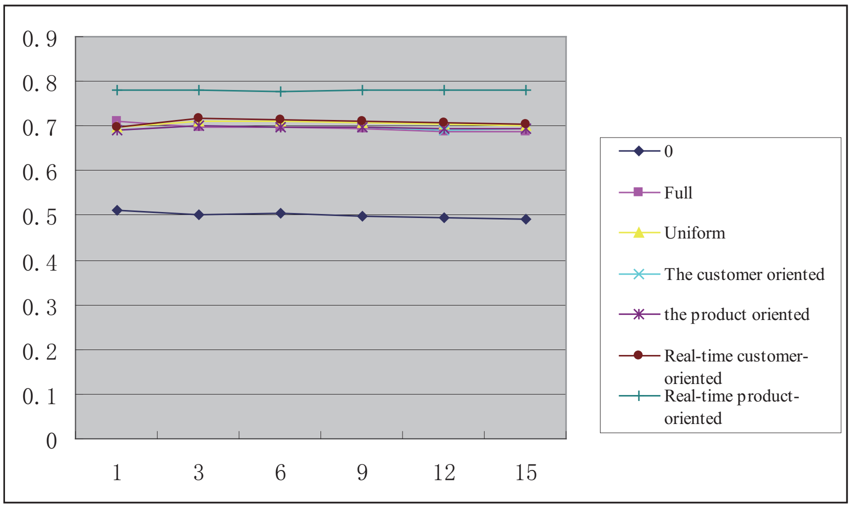


Figure 2: The average AUC of different ranks and methods.

5 Conclusion

On the basis of previous studies, this paper presents an integrated approach, through different weights to the one class collaborative filtering problem obtained the use of real-time information. The integration of real-time information including: the information of product launched into the market which is related to product life cycle, the information about product itself, and customer recently visiting information related to the customer relationship management. But this datasets used aren't from e-commerce websites, and it can't completely replace business process and actions. So how to apply the model to e-commerce needs to practice in the future. The next work is to test this model by changing the number of customers, products and transactions.

Now real-time information in the field of collaborative filtering experiments is limited, confined to movie ratings recommendations problem. One class collaborative filtering method which is based on e-commerce transaction record set, involved the customer's history access to sites and product information that launched into market, and got a good result in this paper. This solution to one class collaborative filtering problems will benefit the field of cross-selling e-commerce, personalized, targeted recommendations sales.

Bibliography

[1] Goldberg D., D. Nichols, B. M. Oki, and Terry D., Using collaborative filtering to weave an information tapestry, *Communications of ACM*, 35 (12): 61–70,1992.

[2] Li G. and Li L., Based on matrix decomposition algorithm for single-class collaborative filtering recommendation, *Application Research of Computers*, 29(5): 1662-1665, 2012.

[3] Shani G., D. Heckerman, and R. I. Brafman, An MDP-based recommender system, *Journal of Machine Learning Research*, 6: 1265–1295,2005.

[4] Banati H. and Mehta S., A Multi-Perspective Evaluation of ma and ga for collaborative Filtering Recommender System, *International J of Computer Science & Information Technology* 2(5): 103-22, 2010.

[5] Yang S. and Xue W., Classification based on a single collaborative filtering recommendation algorithm, *Computer Engineering* ,19(37): 59-61,2011.

[6] Pan R. and M. Scholz., Mind the gaps: Weighting the unknown in largescale one-class collaborative, *In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 667-676, 2009.

[7] Sindhwani V., S.S. Bucak, J. Hu, and A. Mojsilovic, A family of nonnegative

matrix factorizations for one-class collaborative filtering problems, *In Proc. of 3rd ACM RecSys Workshop on Recommendation-based Industrial Applications*, 2009.

[8] Koren Y., Collaborative filtering with temporal dynamics, *In Proc. of 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 447-456, 2009.

[9] Lu Z., D. Agarwal, and I. S. Dhillon, A spatio-temporal approach to collaborative filtering, *In Proc. of 3rd ACM RecSys Workshop on Recommendation-based Industrial Applications*.13-20, 2009.

[10] Xiang L. and Yang Q., Time-dependent models in collaborative filtering based recommender system, *In Proceedings of2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, (1), 450-457, 2009.

[11] Xiong L., X. Chen, T. Huang, J. Schneider, J. G. Carbonell, Temporal

collaborative filtering with Bayesian probabilistic tensor factorization, *In Proceedings of SIAM International Conference on Data Mining*, 2010.

[12] Srebro N. and Jakkola T., Weighted low rank approximations, Proceedings of 20th International Conference on Machine Learning, 720-727, 2003.

[13] Daniel D. Lee and H. Sebastian Seung, Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755): 788-791, 1999.

[14] Lan W. and Zhengjun Z., Collaborative filtering algorithm based on time weight, *Journal of computer application*, 27(9): 2302-2303, 2007.

[15] Huaizhen Y. Xiaoqi C. and Meilian L., Research on the Personalized Recommendation Algorithm Based on Time Weight, *Computer engineering& science*, 31(6): 126-128, 2009.

[16] Donghui L , Dewei P. and Hui Zh., Collaborative Filtering Algorithm Based on Time Weight and User's Feature, *Journal of wuhan university of technology*, 34(5): 144-148, 2012.

[17] Smaranda Cosma, Mădălina Văleanu, Dan Cosma, Dana Vasilescu, Grigor Moldovan, Efficient Data Organisation in Distributed Computer Systems using Data Warehouse, *INT J COMPUT COMMUN*, 8(3): 367-373, 2013.

[18] Kabir Golam and Hasin M. Ahsan Akhtar, Evaluation of customer oriented success factors in mobile commerce using fuzzy AHP, *Journal of Industrial Engineering and Management*, 4(2): 361-386, 2011.

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):571-577, August, 2013.

PyBNEq - A Tool for Computing Bayes-Nash Equilibria

I. Joldeş, B. Pârv, I. Parpucea, V. Lupşe

Iulian Joldeş, Bazil Pârv
Babeş-Bolyai University
Department of Computer Science
Romania, 400084 Cluj-Napoca, 1 M. Kogălniceanu Str.
E-mail: joldesiulian@yahoo.com, bparv@cs.ubbcluj.ro

Ilie Parpucea
Babeş-Bolyai University
Department of Mathematics and Statistics
Romania, 400591 Cluj-Napoca, 58-60 Teodor Mihali Str.
E-mail: ilie.parpucea@gmail.com

Vasile Lupşe
Technical University Cluj-Napoca
North Center Baia Mare
Romania, 430083 Baia Mare, 62/A Dr. Victor Babeş Str.
E-mail: vasilelupse@ubm.ro

Abstract:
This paper describes PyBNEq - a tool for computing Bayes-Nash equilibria for games of incomplete information. It is implemented in Python and has a graphical user interface, allowing the user to load/save/edit game data, and to find Bayes-Nash equilibria. Currently, PyBNEq implements Porter-Nudelman-Shoham algorithm for 2-player games and can be considered as a decision support system for solving games of incomplete information.
Keywords: Bayes-Nash equilibrium, decision support systems, game with incomplete information.

1 Introduction

Many real-world problems, including ones which contain germs of a crisis, are modeled as games of incomplete information. Examples include market competition, currency attacks, bank runs, liquidity crises, as well as military conflicts.

This paper describes a tool for computing Bayes-Nash equilibrium for games with incomplete information. It is structured in 5 sections, as follows. After this introductory section, the section 2 introduces the Bayesian games and a short discussion related to the computation of the Bayes-Nash equilibria. Section 3 describes the PyBNEq tool, especially its graphical user interface, while section 4 presents some case studies we used to test our tool. Finally, the last section compares our application with the existing ones and presents the future developments.

2 Theoretical background

In what follows, we consider the definition of a *Bayesian game* given in [1]. Such a game is defined as a tuple (N, A, Θ, Ω, u), where:

- $N = \{1, ..., n\}$ is the set of agents or players;
- $A = (A_1, A_2, \cdots, A_n)$ is the set of agents' actions, A_i being the set of actions available to agent i;

- $\Theta = (\Theta_1, \Theta_2, \cdots, \Theta_n)$, with $\Theta_i = (\theta_{i,1}, \theta_{i,2}, \cdots, \theta_{i,m_i})$ being the set of types for agent i and $\theta_{i,j}$ the j-th type of agent i;
- $\Omega = (\Omega_1, \Omega_2, \cdots, \Omega_n)$, with $\Omega_i = (\omega_{i,1}, \omega_{i,2}, \cdots, \omega_{i,m_i})$ being the set of probabilities assigned to the types of agent i and $\omega_{i,j}$ the probability assigned to $\theta_{i,j}$;
- $u = (u_1, u_2, \cdots, u_n)$ is the set of utility (payoff) functions, with $u_i = (u_{i,1}, u_{i,2}, \cdots, u_{i,m_i})$ being the set of utility functions of agent i and $u_{i,j}$ the utility function of $\theta_{i,j}$ whose arguments are the joint action $\mathbf{a} = (a_1, a_2, \cdots, a_n)$ and the other agents' types $\theta_{-i} = (\theta_1, \cdots, \theta_{i-1}, \theta_{i+1}, \cdots, \theta_n)$.

Alternative definitions can be found for example in [3,8,9]. The solution of a Bayesian game is based on the concept of *Bayes-Nash equilibrium*. It is known that the computation of this equilibrium is NP-complete, and the usual approach uses two steps [12]:

1. reduce the Bayesian game to a complete-information game, *and*
2. compute the Nash equilibrium for the complete-information game obtained in Step 1.

Step 1 above can be addressed in several ways. The paper by Ceppi et al. [1] contains a detailed discussion regarding the ways of performing reduction, based especially on the material in [5]. Our approach is based on the use of sequence form, due to the smaller payoff matrixes than the ones in normal form.

Step 2 means computing of the Nash equilibria for two-player complete-information games. Ceppi et al. paper [1] analyze three algorithms for such games in strategic form: Lemke-Howson (LH), Porter-Nudelman-Shoham (PNS), and Sandholm-Gilpin-Conitzer (SGC), and propose an extension of PNS to Bayesian games (B-PNS), which is implemented in our tool.

The B-PNS algorithm has the following generic steps:

1. enumerate all the possible joint supports;
2. select a support for each possible type of agent 1;
3. prune the spaces of actions of the agent 2's types by strict conditional dominance;
4. check the strict conditional dominance on agent 1's support;
5. select a support for each possible type of agent 2;
6. check the strict conditional dominance on agent 2's suport;
7. check the feasibility problem.

3 The PyBNEq tool

In order to implement the B-PNS algorithm, there were two interrelated design problems to solve: data representation and the individual algorithms for each step. A future paper will describe the design solutions in greater detail.

The implementation language is Python 2.7 [11], while data representation and numerical computation capabilities of `numpy` package [6] were extensively used. Also, the application makes use of `pycplex` library [13], a Python interface to the ILOG CPLEX Callable Library [2], for solving the feasibility problem (step 7 of the above-described B-PNS algorithm). Finally, the graphical user interface was produced by using the `wxPython` toolkit [14].

The application is designed with a friendly user interface, having the following components:

- *main window* contains a menu bar for creating, loading or saving a game and also for running it. When a game is loaded, this window becomes the current game window, where the game parameters are displayed (see Figure 1), and the Bayes-Nash echilibria can then be computed;

- *new game window*, allowing the user to define actions, types and probabilities for both players (see Figure 2);

- *payoffs window*, where the payoffs could be added or edited (Figure 3).

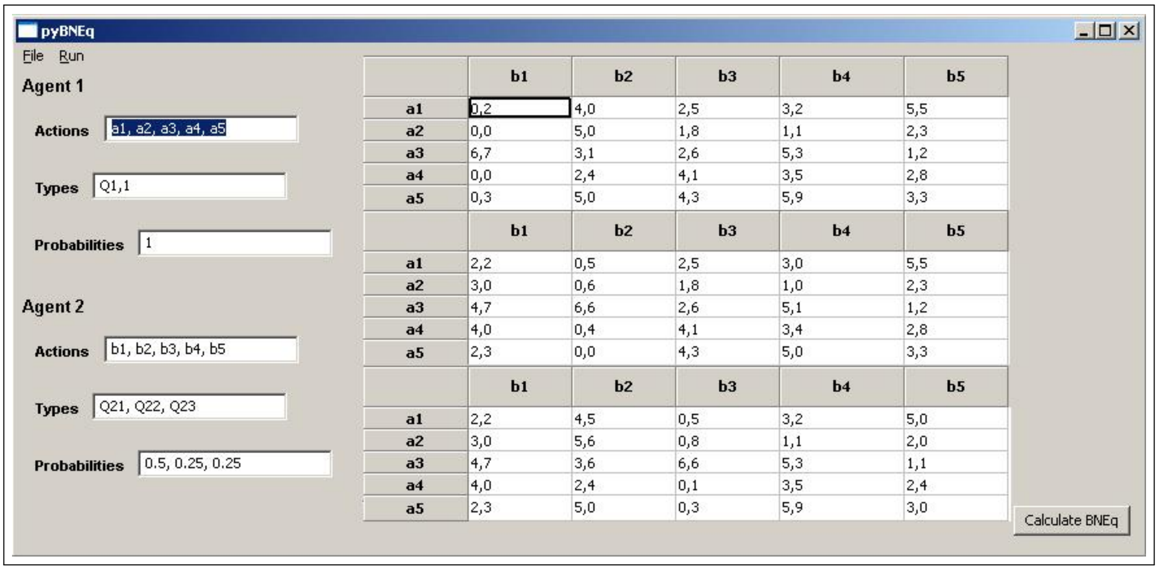


Figure 1: The main window of the application

The input data are stored in /data sub-directory in .dat files. The structure of input data file is:

Line1: first agent's index
Line2: first agent's actions
Line3: first agent's types
Line4: first agent's probabilities
Line5: second agent's index
Line6: second agent's actions
Line7: second agent's types
Line8: second agent's probabilities
Line9 and below: the payoffs of the two players

The result consists of the computed Bayes-Nash equilibria and it's saved in /output sub-directory, in an .out file. It represents the strategies profile.

4 Case studies

We tested our tool with three examples of Bayesian Games: market entry game (see [7]), gift game (see [4], and the example game from [1]. All examples consider two players, whose actions are denoted by a_i and b_j, respectively.

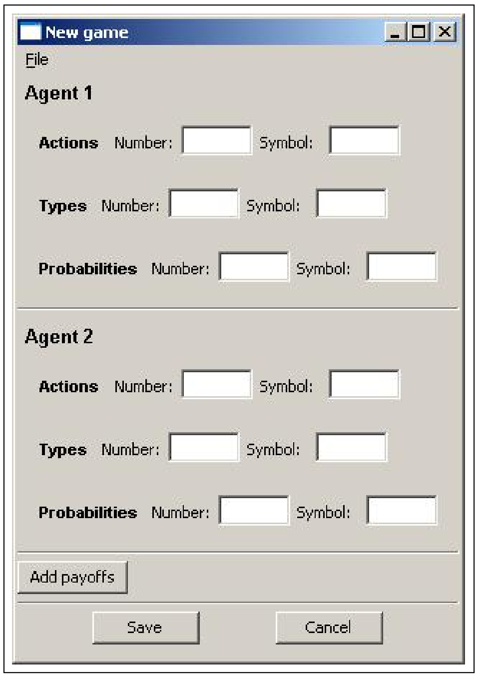


Figure 2: The Add new game window

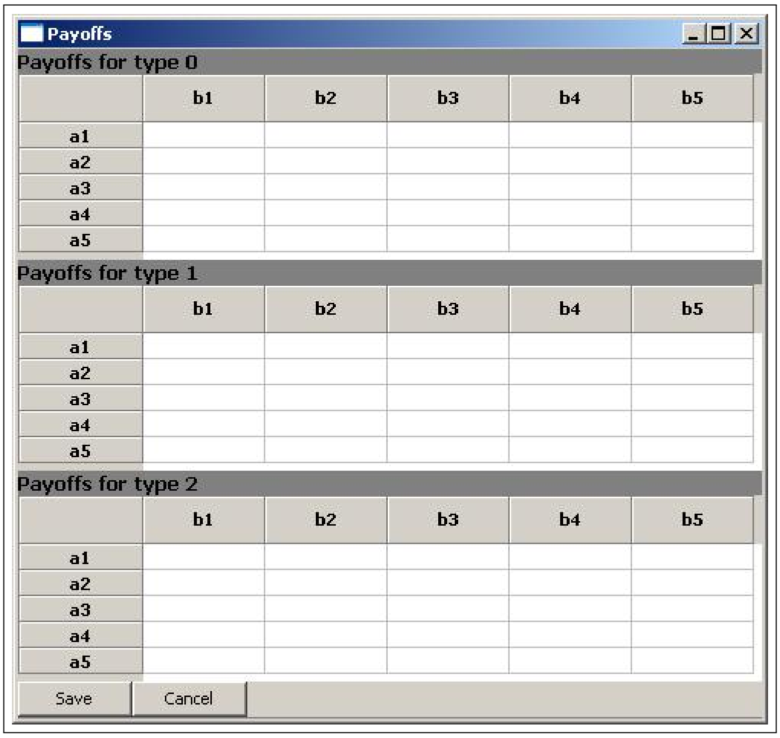


Figure 3: The payoffs window

4.1 Market entry game

Both players have a single type. The following payoff matrix represents the input data:

	b_1	b_2
a_1	$-\frac{1}{4}$, $-\frac{1}{4}$	$\frac{3}{4}$, 0
a_2	0, $\frac{1}{4}$	$\frac{1}{2}$, 0
a_3	$-\frac{1}{4}$, $\frac{1}{4}$	$\frac{1}{4}$, 0
a_4	0, $\frac{3}{4}$	0, 0

The application outputs three solutions: (a_1, b_2), (a_2, b_1), and (a_4, b_1).

4.2 Gift giving game

Player 1 has two types $(\theta_{1,1}, \theta_{1,2})$, and player 2 has one type $(\theta_{2,1})$. Type $\theta_{1,1}$ means $p > \frac{1}{2}$, while $\theta_{1,2}$ means $p < \frac{1}{2}$. The input data are represented by the following payoff matrix:

	b_1	b_2
a_1	0, 0	0, 0
a_2	1 - p, p - 1	p - 1, 0
a_3	p, p	-p, 0
a_4	1, 2 $\cdot$ p - 1	-1, 0

a) The type combination $(\theta_{1,1}, \theta_{2,1})$. Player 1 is more likely to be a friend ($p > \frac{1}{2}$); the application outputs two solutions: (a_1, b_2) and (a_4, b_1).

b) The type combination $(\theta_{1,2}, \theta_{2,1})$. Player 1 is more likely to be an enemy ($p < \frac{1}{2}$); the application outputs one solution: (a_1, b_2).

4.3 Third example: the game from [1]

Player 1 has one type $(\theta_{1,1})$ and player 2 has three types $(\theta_{2,1}, \theta_{2,2}, \theta_{2,3})$. There are three payoff matrices as input, corresponding to the combinations of players' types:

a) ($\theta_{1,1}$, $\theta_{2,1}$): $\omega_{2,1} = \frac{1}{2}$

	b_1	b_2	b_3	b_4	b_5
a_1	0, 2	4, 0	2, 5	3, 2	5, 5
a_2	0, 0	5, 0	1, 8	1, 1	2, 3
a_3	6, 7	3, 1	2, 6	5, 3	1, 2
a_4	0, 0	2, 4	4, 1	3, 5	2, 8
a_5	0, 3	5, 0	4, 3	5, 9	3, 3

The solutions are: (a_1, b_5), (a_2, b_1), and (a_3, b_4).

b) ($\theta_{1,1}$, $\theta_{2,2}$): $\omega_{2,2} = \frac{1}{4}$

	b_1	b_2	b_3	b_4	b_5
a_1	2, 2	0, 5	2, 5	3, 0	5, 5
a_2	3, 0	0, 6	1, 8	1, 0	2, 3
a_3	4, 7	6, 6	2, 6	5, 1	1, 2
a_4	4, 0	0, 4	4, 1	3, 4	2, 8
a_5	2, 3	0, 0	4, 3	5, 0	3, 3

The solutions are: (a_1, b_3), (a_2, b_1), (a_4, b_5), and (a_5, b_1).

c) $(\theta_{1,1},\ \theta_{2,3})$: $\omega_{2,3} = \frac{1}{4}$

	b_1	b_2	b_3	b_4	b_5
a_1	2, 2	4, 5	0, 5	3, 2	5, 0
a_2	3, 0	5, 6	0, 8	1, 1	2, 0
a_3	4, 7	3, 6	6, 6	5, 3	1, 1
a_4	4, 0	2, 4	0, 1	3, 5	2, 4
a_5	2, 3	5, 0	0, 3	5, 9	3, 0

The solutions are: (a_2, b_2), (a_3, b_2), (a_3, b_1), (a_4, b_5), and (a_5, b_4).

5 Conclusions and future work

In this paper we described a tool for computing Bayes-Nash equilibrium for two-player games of incomplete information.

The authors of [1] implemented a C version of the B-PNS algorithm, while we selected Python as the implementation language due to its benefits as rapid application development by using Component-based Software Development and the powerful libraries like numpy, pycplex, as well as the Graphical User Interface toolkit wxPython. The graphical user interface allows a very convenient way of playing with such games.

In the future we would like to extend the tool functionalities so that it can easely be integrated in more elaborate decision support systems, including custom-made GUIs for different classes of games. Also, the application programming interface will be improved and the tool will fully support games generated by the common generators like GAMUT.

Acknowledgements

This work was supported by the grant ID_2586, sponsored by NURC - Romanian National University Research Council (CNCSIS).

Bibliography

[1] Ceppi, S., N. Gatti, N. Basilico, Computing Bayes-Nash Equilibria through Support Enumeration Methods in Bayesian Two-Player Strategic-Form Games in WI-IAT, *Proc.of the 2009 IEEE/WIC/ACM Int. Joint Conference on Web Intelligence and Intelligent Agent Technology*, 2: 541-548, 2009, DOI: 10.1109/WI-IAT.2009.209.

[2] `cplex` reference, ftp://ftp.software.ibm.com/software/websphere/ilog/docs/optimization/cplex/refcallablelibrary.pdf

[3] Eichberger, I., *Game Theory for Economists*, Academic Press, 1993.

[4] Fehr, E. and K.M. Schmidt, Theories of Fairness and Reciprocity - Evidence and Economic Applications, in *M. Dewatripont, L.P. Hansen and S.J. Turnovsky (eds.) Advances in Economics and Econometrics, Econometric Society Monographs, 8th World Congress*, 1: 208-257, 2003.

[5] Koller, D., N. Megiddo, and B. von Stengel, Efficient computation of equilibria for extensive two-person games, Games and Economic Behavior, 14(2): 220-246, 1996.

[6] `numpy` home page, http://numpy.scipy.org/

[7] Ochs, J. Coordination problems. In J.H. Kagel and A.E. Roth (eds.) Handbook of Experimental Economics, Princeton University Press, 195-251, 1995.

[8] Parpucea, I., B. Pârv, and T. Socaciu, T. Modeling Uncertainty in a Decision Problem by Externalizing Information, *INT J COMPUT COMUN*, 6(2): 328-336, 2011.

[9] Pârv, B. and I. Parpucea, Bayes-Nash Equilibrium in the Presence of Information Sources: Computational Issues, Studia Univ. Babes-Bolyai, Informatica, LVI (2011), 3: 33-38, 2011.

[10] Porter, R., E. Nudelman, and Y. Shoham, Simple search methods for finding a Nash equilibrium, *Proc. of the AAAI Conference on Artificial Intelligence (AAAI)*, 664-669, 2004.

[11] Python home page, http://www.python.org/

[12] Shoham, Y. and K. Leyton-Brown, *Multiagent Systems: Algorithmic*, Game Theoretic and Logical Foundations. Cambridge, USA: Cambridge University Press, 2008.

[13] `pycplex` reference, http://www.cs.toronto.edu/ darius/software/pycplex/

[14] `wxpython` reference, http://wxpython.org/

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):578-587, August, 2013.

Information Security Engineering: a Framework for Research and Practices

M. Li, M. Tang

Meng'gang Li
China Center for Industrial Security Research,
Beijing Jiaotong University, China
morganli@vip.sina.com

Mincong Tang
International Center for Informatics Research,
Beijing Jiaotong University, China
mincong@bjtu.edu.cn

Abstract: Information security is not a new topic in academics and industry. However, through a comprehensive literature review, we found that most research in information security focus on technical perspectives including evaluation methods and mathematical approaches for securities, risk mitigation algorithms, with some research focus on economic perspective of information security and even a few talked about social engineering of information security. There is not a unique framework to integrate different types of research in information security. We believe that information security research apply the theories and methodologies in systems engineering to investigate the problems, that is, information security engineering. In this paper, we propose a conceptual framework of information security engineering. This framework explicitly illustrates the methodological system, content system, procedures and strategies for information security engineering research and practices.
Keywords: Computer Science, Information Security Engineering (ISE), Systems Engineering, Information Systems.

1 Introduction

Information security problem has been coming along with the development of our society. In the past two decades, information technologies (IT) and Internet have changed our world in an unbelievable speed. They bring to us many conveniences as well as risks and uncertainties. Thus information security problem becomes very prominent under such circumstance and it now plays a decisive position in national securities of a country. Information security mainly involves governments, enterprises, organizations, associations and individuals. It spreads with a very wide time domain and covers all fields of the society including politics, economy, culture, military etc. Compared with the securities of politics, economy and military, information security has the following features: 1) information security is the core of national security in information era; 2) the nature of information security is information resource security; 3) information security relies more and more on the securities of technological systems; 4) attack sources of information security are characterized as spectral and concealed.

Information security nowadays is no longer an isolated and scattered but a very complicated systems engineering problem. Thus we propose that it's necessary and urgent to conduct in-depth and comprehensive studies to examine information security from the viewpoint of engineering. We define this as information security engineering (ISE): ISE is a kind of systems engineering which is based on information technology and takes information security management as its approach and information security laws and policies as its assurance. As a subset of systems engineering, ISE is the embodiment of information system securities for systems security engineering, systems engineering and system acquisitions. ISE is a product of the integration of security engineering, information management and information systems development. The contents of ISE mainly include: connotations, contents and objectives of ISE; information security risk analysis and evaluation procedures, methods and tools; requirement analysis methods, security strategies, security architecture, security solutions, security implementation standards, security test and operations, emergency measures and methods, security education and training.

The thinking of engineering has been well applied into software development and security management, from which two subjects: software engineering and security engineering have been formed and developed. Though these two subjects are relatively completed, studies of information security have been affected by the knowledge structures of researchers, rapid development of our society and other factors, the findings of information security research are scattered and the theory lags behind the demand of applications. On the other hand, though the sense of security has been strengthen in the past 10 years, from installing anti-virus software to procuring security products, the understanding of security still stays at the very stage of 'treat only where the pain is'. The problem of information security cannot be solved by pure technology, nor can it be solved by putting security products together. It has to rely on complicated systems engineering—information security engineering (ISE). Therefore, there is a need to conduct in-depth research in information security from a systems engineering perspective.

We organize this paper in the following ways: in the next section we present a comprehensive literature review, which forms the foundations of ISE. Followed by is the processes of ISE in section 3. After that we propose a framework of ISE, the framework is used to suggest future trends within ISE research and practices in section 4.

2 Foundations of ISE

As developed from software engineering and security engineering, ISE follows the foundations of systems engineering but its context is limited to information security and information systems. Different from typical systems engineering, ISE research focuses on the development of methodology and architecture for information security. It intends to unite the diversities in information security research and makes it an interdisciplinary research without limiting the analysis to a particular discipline (i.e. computer sciences, information systems etc.). By comprehensively reviewing existing information security literature, we mainly found that four issues of information security have been well studied. These four issues are:

1) Security Management (SM): SM refers to the ways like information systems planning and evaluation, to maintain secure information systems within organizations. Backup, recovery, contingency management are included in SM.

2) Communication Security (CS): CS refers to the measures adopted to ensure secure communication achieved between people.

3) Access of Information and Systems (AIS): AIS refers to the measures that control people's access to information and systems.

4) Secure Information Systems Development (SISD): SISD refers to the methods, policies and procedures that lead a secured information system to be developed.

According to the differences in research approaches of the literatures we reviewed, we only incorporate those studies after 2000 for the purpose of most updated knowledge in these areas, we summarized the findings from existing literature as table 1.

Table 1: Summary of Existing Literature in Information Security

Issue	**Source**	**Propositions/Findings**	**Approach**
Security Management	Dhillon & Backhouse (2000)	Responsibility, integrity, trust and ethicality (RITE) principles hold the key for successfully managing information security In addition to confidentiality, integrity, and availability (CIA).	Conceptual
	Eloff & Eloff (2003)	To successfully secure the information and technology related assets of an organization, management should aim towards establishing an information security management system (ISMS)	Conceptual & Organizational
	Solms & Solms (2004)	10 essential aspects like not realizing that information security is a corporate governance responsibility and not realizing that information security is a business issue and not a technical issue and so on must be taken into account in an information security governance plan to make it a success	Conceptual
	Solms (2005)	A separate information security compliance management department is needed when talking about information security management	Conceptual
	Saint-Germain (2005)	For organizations to fall into several regulatory realms, they need to establish a comprehensive, flexible framework for implementing cost-effective compliance, deployed via a governing system that maintains security policies and controls	Organizational (Case study)
	Tsoumas & Gritzalis (2006)	The separation of security requirements from their technical implementations facilitates security management	Technological
	Chang & Lin (2007)	Organization culture will significantly influence the effectiveness of implementing information security management	Organizational (Survey)
	Ashenden (2008)	Human challenge of Information Security management has largely been neglected and people need to look at the skills needed to change organizational culture, the identity of the Information Security Manager and effective communication between Information Security Managers, end users and Senior Managers.	Conceptual
	Werlinger et al. (2009)	18 challenges like technical complexity, mobility and so on can affect IT security management within organizations	Organizational (Survey)
Communication Security	Chaddoud et al. (2001)	Baal protocol as a scalable solution to group key management problems and it can resolve the user's revocation problem	Technological
	Khadra et al. (2003)	Impulsive synchronization of two chaotic systems is very robust, this robustness is useful in designing chaos based cryptosystems, which is used to ensure secure communication	Technological

Issue	**Source**	**Propositions/Findings**	**Approach**
	Yang (2004)	Impulsive synchronization is more robust than continuous synchronization. Based on a combination of both conventional cryptographic method and impulsive synchronization of chaotic systems, a new chaotic secure communication scheme is proposed, which is to ensure secure communication	Technological
	Liang et al. (2008)	Secrecy capacity region of the Gaussian BCC complements the secrecy capacity region of the discrete memoryless BCC. This will enhance secure communication.	Technological
	Kiani-B et al. (2009)	The proposed encoding chaotic communication has achieved a satisfactory, typical secure communication scheme. Results show that security is enhanced based on spreading the signal in frequency and encrypting it in time domain in the proposed system.	Technological
Access of Information and Systems	Kagal et al. (2001)	Trust is added as a new dimension to pervasive computing, allowing greater flexibility in designing policies and providing more control over accessing services and information	Conceptual
	Bonatti, & Samarati (2002)	An approach for regulating service access and information disclosure on the Web is proposed, which consists of a uniform formal framework to formulate and reason about – both service access and information disclosure constraints. This approach ensures communicating users' requirement while disclosing no private information.	Technological
	Whitman (2003)	'Understanding the enemy' is believed to be an important component of information protection	Conceptual
	Gritzalis & Lambrinoudakis (2004)	A security architecture based on a role-based access scheme is proposed and found to be effective to identify different users' access to local sources and other sites	Conceptual
	Karyda et al. (2005)	Contextual factors including organizational culture, management support etc. for the application of IS security policies have been discussed, these factors are to be taken into consideration when implementing information security policies including access to information and systems.	Conceptual
	Cheng et al. (2007)	A new model for risk-based access control is proposed. This model is based on fuzzy multi–level security access control and found to be more effective than traditional Bell–Lapadula model.	Technological
Secure IS development	Jurjens (2001)	UML is used to express security requirements in system development.	Technological
	Georg et al. (2002)	An aspect-oriented approach to modeling is proposed to allows developers to encapsulate design concerns like security, availability of services, and timeliness so that they can be woven into a design in a systematic and consistent manner	Technological

Issue	Source	Propositions/Findings	Approach
	Jones & Rastogi (2004)	Security has to be "baked in" to the overall systems development life-cycle process.	Technological
	Villarroel et al. (2005)	Eleven secure systems design methodologies have been compared to see how they should be adopted in system development	Conceptual
	Mouratidis et al. (2005)	An approach considering security concerns as an integral part of the entire system development process is proposed to be necessary	Organizational (Case study)
	Mellado et al. (2007)	Security has to be dealt with at all stages of IS development, especially in the establishment of security requirements to achieve a robust IS.	Conceptual
	Cheng et al. (2008)	A new concept of security engineering environment (SEE) is proposed, SEE concept with high security requirements can provide a base for designers, developers, users, and maintainers with standard, formal, and consistent supports	Technological
	Mouratidis & Jurjens (2010)	two prominent approaches, a goal-oriented security requirements engineering approach called Secure Tropos and an MBSE approach called UMLsec have been integrated to help how elicited security requirements can be realized in the design stage and how the developed design can be verified against the security requirements of the system	Technological

Based on the discussion from existing literature, we believe that these four issues of information security above consist of the major contents of ISE (Figure 1) and form the foundations of ISE. Specifically, the four issues in ISE can be elaborated as followings.

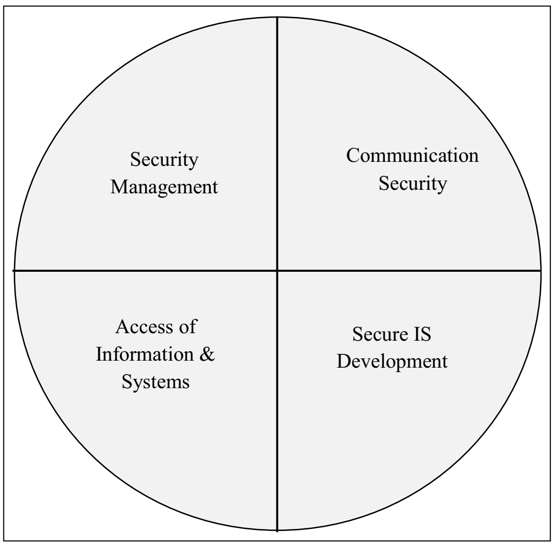


Figure 1: Fours issues in Information Security Studies

While the contents of information security mainly cover the four issues, ISE research issues will include but not limits to: philosophical foundations of information security, definitions of information security, mathematical foundations of information security, safety rheology and mutation laws, physiological and psychological issues in information security, ISE methods of analysis, forecasting and control, information security risk evaluation and information security management methods, human-computer environment analysis and design etc. These research issues interact with the four issues of information security.

3 Processes of ISE

As a subset of systems engineering, ISE mainly follows the processes of systems engineering, which is illustrated by figure 2.

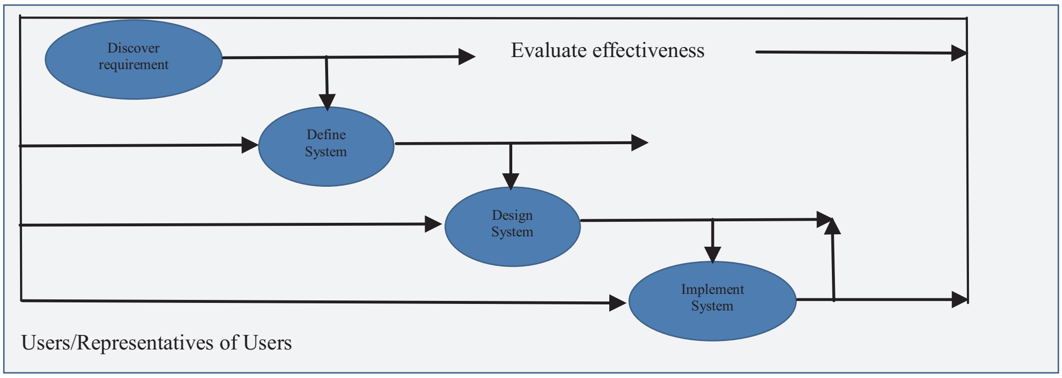


Figure 2: Processes of Systems Engineering

The processes described above are usually carried out in the following ways:

1) Discover tasks or requirements;
2) Define functionalities of system;
3) Design system;
4) Implement system;
5) Evaluate effectiveness

For ISE, the processes above are customized to specific information securities. The key is to fulfill the requirements of information protection by implementing systems engineering processes. ISE can facilitate the development of system products and process solutions to satisfy the users' requirements. Thus, the processes of ISE becomes:

1) Discover requirements of information security: ISE will first investigate users' requirement, policies, standards, vulnerabilities and threats regarding of information. Then ISE will mark the users of information and systems, their roles and responsibilities in information security.
2) Define information security system: users' requirements of information protection and description of information system environment are interpreted as the objectives and functionalities of information security system. In this stage, ISE will define what can be done by information protection system and the executions of information security system as well as the internal and external interface of information security system.
3) Design information security system: ISE will design the architecture of information security system and detail the design scheme of information security system.

4) Implement information security system: according to the requirements of information security, this stage aims to development, procure, integrate, test and verify the collections of configuration of information security system. Similar with the corresponding stages in systems engineering, ISE will conduct implementation and testing in this stage.

5) Evaluate effectiveness of information security system: ISE emphasizes the capabilities of providing confidentialities, integrities, availabilities and non-repudiation for information.

ISE processes emphasize marking, conceiving and controlling information security risks and optimize these risks to protect potential losses due to various possible threats and attacks.

4 A Framework for ISE Research and Practices

In Oxford Dictionary, framework is defined as a structure upon or into which contents can be put and further relates it to thoughts that are directed for a purpose. The ISE Framework proposed in this study provides academics and practitioners with an understanding of how to conduct an information security research and practices from an engineering point of view so as to align ISE theories with applications.

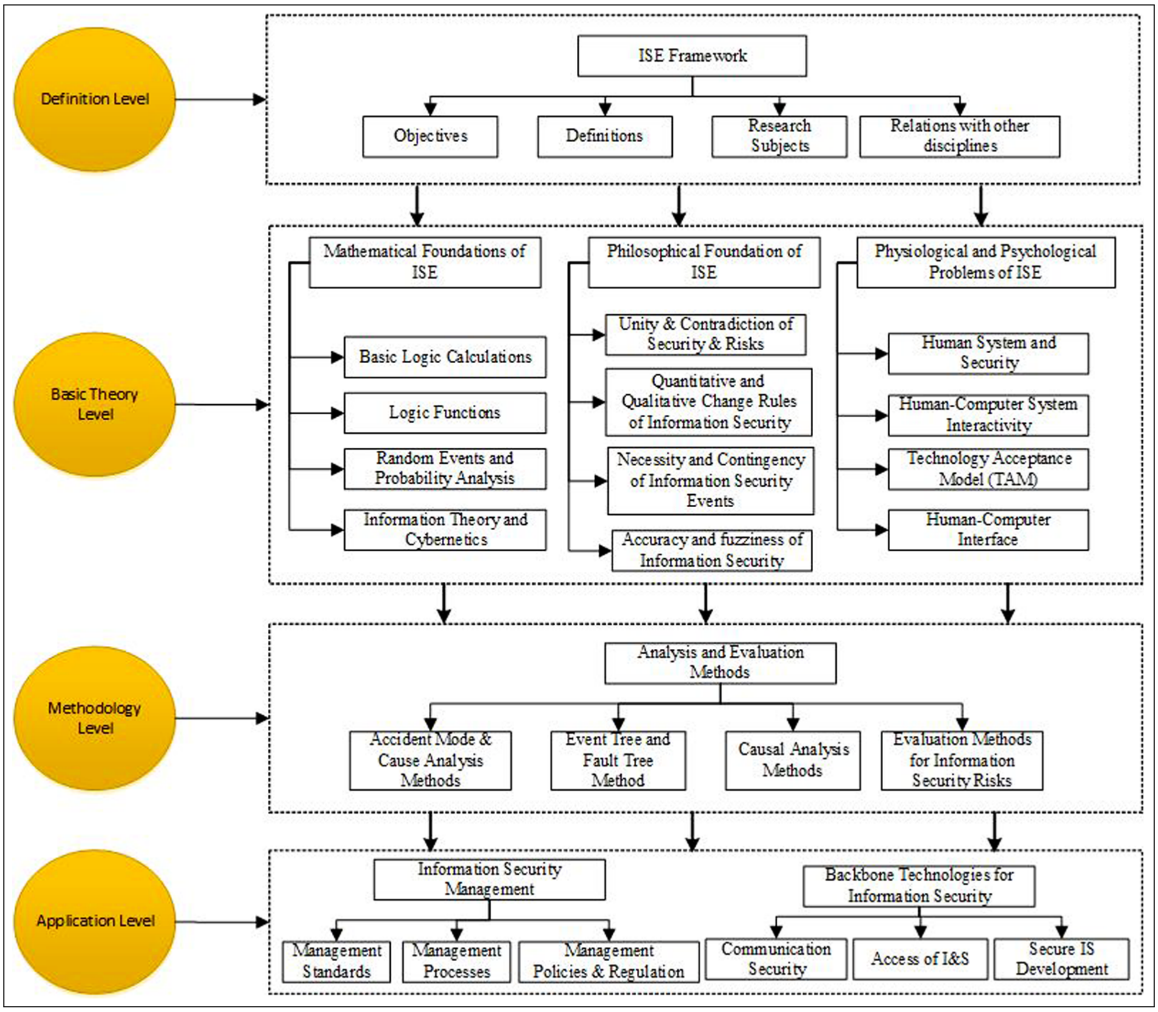


Figure 3: A Framework for Information Security Engineering Research and Practices

As illustrated in figure 3, there are four levels in the framework: definition, basic theory, methodology and application.

At the definition level, four elements are included. We propose that objectives, definitions, research subjects and relations with other disciplines belong to this level. For ISE research, these four elements must be clearly defined.

At the basic theory level, three elements are included. Philosophical foundations of ISE refer to those philosophical issues like unity and contradiction of security and risks, accuracy and fuzziness of information security etc., these provide the highest level of understanding of ISE. Mathematical foundations of ISE provide theoretical and analytical methods to solve the problems of ISE while physiological and psychological problems of ISE focus on the behavioral perspectives of ISE research.

At the methodology level, different techniques (methods) for analysis and evaluation are included. These do not only include analytical methods (i.e. event tree, fault tree etc.), but also include causal analysis methods which are widely used in behavioral research.

At the application level, two categories are proposed to provide guideline for practitioners in ISE, which include information security management and backbone technologies for information security.

This framework provides guidelines and directions for researchers in information security areas. First, it proposes three elements in the basic theory level. These elements cover a very large section of the existing literature in information security especially for the mathematical foundations. Second, from a conceptual approach point of view, the philosophical foundations cover many discussions in literature. Third, physiological and psychological problems, which are still gaps in literature of information security, they are areas calling for further studies. For practitioners, this framework provides insights for their practices in information security and management. Behaviorally, three managerial perspectives are proposed: standards, processes, policies and regulations. Technologically, three issues of information security have been proposed, they are regarded as backbone technologies through which the objectives of information security could be achieved from a technological view.

5 Concluding Remarks

Information security engineering is a comprehensive and cross-disciplined subject which covers the knowledge in mathematics, physics, telecommunication, computer sciences and management. It is not a simple combination of various tools of security technologies, nor does it equal to a series of managerial regulations and safety standards. It is a complicated system engineering. By conducting a comprehensive literature review, we have a body of knowledge on information security research, which forms the basics and contents of ISE. Establishing information security engineering as an independent subject is critical to assure national information securities and improve the levels of information security training. Moreover, it guides theoretical research and practical applications in this area and make it possible to integrate the critical security technologies and standards and subsequently create a unique and effective system for information securities.

Acknowledgement

This paper is part of the project ‘Research on China Industrial Security Index’ (No. B09C1100020) funded by the Ministry of Education, China. We are grateful for the suggestions and comments from two experts who reviewed this article for us.

Bibliography

[1] Ashenden, D., Information Security management: A human challenge?, *Information Security Technical Report*,13(4): 195-201, 2008

[2] Bonatti, P. & Samarati, P., A Uniform Framework for Regulating Service Access and Information Release on the Web, *Journal of Computer Security*, 10(3):241-271, 2002

[3] Chrisment, I. & Schaff, A., Dynamic Group Communication Security, *Proceedings of Sixth IEEE Symposium on Computers and Communications*, 49-56, 2001

[4] Chang, S. & Lin, C., Exploring Organizational Culture for Information Security Management, *Industrial Management & Data Systems*, 107(3):438-458, 2007

[5] Cheng, et al., Fuzzy Multi-Level Security: An Experiment on Quantified Risk-Adaptive Access Control, *IEEE Symposium on Security and Privacy*, 222-230, 2007

[6] Cheng et al., A Security Engineering Environment Based on ISO/IEC Standards: Providing Standard, Formal, and Consistent Supports for Design, Development, Operation, and Maintenance of Secure Information Systems, *International Conference on Information Security and Assurance*, 350-364, 2008

[7] Dhillon, G. & Backhouse, J., Information System Security Management in the New Millennium, *Communications of the ACM*, 43(7): 125-128, 2000

[8] Eloff, J. & Eloff, M., Information Security Management: a New Paradigm, *Proceedings of the 2003 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on Enablement Through Technology*, 130-136, 2003

[9] Georg, G., Ray, I. & France, R., Using Aspects to Design a Secure System, *Proceedings of the Eighth IEEE international Conference on Engineering of Complex Computer Systems*, 117-226, 2002

[10] Gritzalis D. & Lambrinoudakis, C., A Security Architecture for Interconnecting Health Information Systems, *International Journal of Medical Informatics*, 73(3):305-309, 2004

[11] Hong, K., Chi, Y., Chao, L. & Tang, J., An Integrated System Theory of Information Security Management, *Information Management & Computer Security*, 11(5): 243-248, 2003

[12] Jones, R. & Rastogi, A., Secure Coding: Building Security into the Software Development Life Cycle, *Information Systems Security*, 13(5): 29-39, 2004

[13] Jurjens, J., Towards Development of Secure Systems Using UMLsec, *Fundamental Approaches to Software Engineering Lecture Notes in Computer Science*, Vol. 2029: 187-200, 2001

[14] Kagal, L., Finin, T. & Joshi, A., Trust-Based Security in Pervasive Computing Environments, *Computer*, December: 151-157, 2001

[15] Karyda, M., Kiountouzis, E. & Kokolakis, S., Information Systems Security Policies: a Contextual Perspective, *Computers & Security*, 24(3):246-260, 2005

[16] Khadra, A., Liu, X. & Shen, X., Robust Impulsive Synchronization and Application to Communication Security, *Dynamics of Continuous, Discrete and Impulsive Systems Series B: Applications & Algorithms*, 10: 403-416, 2003

[17] Kiani-B, A., Fallahi, K., Pariz, N. & Leung, H., A Chaotic Secure Communication Scheme Using Fractional Chaotic Systems Based on an Extended Fractional Kalman Filter, *Communications in Nonlinear Science and Numerical Simulation,* 14(3), 863-879, 2009

[18] Liang,Y., Poor, H. & Shamai, S., Secure Communication Over Fading Channels, *IEEE Transactions on Information Theory*, 54(6):2470-2492, 2008

[19] Mellado, D., Fernández-Medina, E. & Piattini, M., A Common Criteria Based Security Requirements Engineering Process for the Development of Secure Information Systems, *Computer Standards & Interfaces*, 29(2): 244-253, 2007

[20] Mouratidis, H., Giorgini, P. & Manson, G., When Security Meets Software Engineering: A Case of Modelling Secure Information Systems, *Information Systems,* 30(8): 609-629, 2005

[21] Mouratidis, H. & Jurjens, J., From Goal-Driven Security Requirements Engineering to Secure Design, *International Journal of Intelligent Systems*, 25(8):813-840, 2010

[22] Saint-Germain, R., Information Security Management Best Practice Based on ISO/IEC 17799, *The Information Management Journal*, July/August: 60-66, 2005

[23] Siponen, M. & Oinas-Kukkonen, H., A Review of Information Security Issues and Respective Research Contributions, *The DATA BASE for Advances in Information Systems*, ISSN 1532-0936, 38(1): 60-80, 2007

[24] Solms, B., Information Security Governance–Compliance Management Vs. Operational Management, *Computers & Security*, 24(6): 443-447, 2005

[25] Solms, B. & Solms, R., The 10 Deadly Sins of Information Security Management, *Computers & Security*, 23(5): 371-376, 2004

[26] Tsoumas, B. & Gritzalis, D., Towards an Ontology-based Security Management, *Proceedings of the 20th International Conference on Advanced Information Networking and Applications*, 2006

[27] Werlinger, R., Hawkey, K. & Beznosov,K., An Integrated View of Human, Organizational, and Technological Challenges of IT Security Management, *Information Management & Computer Security*, 17(1): 4-19, 2009

[28] Villarroel, R., Fernández-Medina, E. & Piattini, M., Secure Information Systems Development: A Survey and Comparison, *Computers & Security*, 30(8):609-629, 2005

[29] Whitman, M., Enemy at the Gate: Threats to Information Security, *Communications of the ACM*, 46(8):91-95, 2003

[30] Yang, T., A Survey of Chaotic Secure Communication Systems, *International Journal of Computational Cognition*, 2(2): 81-130, 2004

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):588-593, August, 2013.

Smart and Safe Vehicle Monitoring with Fuzzy Integral and Haar-like Features

Se-bin Oh, Yeon-ho Chung

Se-bin Oh, Yeon-ho Chung
Dept. of Information and Communications
Engineering, Pukyong National University,
Daeyon 3 Dong, Nam Gu, Busan, Korea
osebin@pknu.ac.kr, yhchung@pknu.ac.kr

Abstract: An on-board Android-based smart and safe vehicle monitoring system is presented. The on-board monitoring system (OMS) performs important monitoring functions: Record, Report and Alert (RRA). The Record function records front images of a moving vehicle. During the recording, any accidents or other emergency conditions will be automatically reported via the Report function for an emergency rescue operation. For the detection of shocks or accidents, we use acceleration based shock sensors that utilize fuzzy integral algorithm. The OMS also focuses on drowsiness that is largely regarded as the main cause of most accidents. The Haar-like feature is used to detect any sign of drowsiness and the Alert function is performed to alert the driver. All the vehicle-borne information is stored at a remote server via wireless communication links for later use or post-processing. A test bed has been developed and verified thoroughly for its accurate operations. The proposed smart and safe vehicle monitoring system offers advanced safety features and is expected to substantially reduce fatigue related accidents.
Keywords: Fatigue, Haar-like, Monitoring, Vehicle Safety.

1 Introduction

Over the last half-decade, smartphones have evolved dramatically and would soon behave like laptops. Most mobile devices are nowadays integrated by a smartphone as convenient infrastructure and intelligent functionality has been built to further users ultimate convenience [1]. For example, transport machine industry, particularly vehicle industry, provides vehicle-borne information via a car navigation system. Even various driving environment information, not to mention information about driving itself, is provided to the driver. However, conventional devices or modules are focused on data storage and do not offer any drowsiness related safety features. Thus, technologies in mobile and vehicle telematics need to be developed further and also necessitate advanced safety-oriented modules. In particular, a fatigue related accident prevention feature must be sought together with basic monitoring functions, e.g. recording of vehicle data and images incorporated with GPS for track-down and investigation of any accidents occurred.

This paper addresses a smart and safe vehicle monitoring issue. The system is based on an Android mobile device, offering advanced monitoring capability of the recording of vehicle-borne data and image, GPS data and accelerator sensors for automated alarm function. Also, it offers an alertness function to prevent fatigue related accidents using the Haar-like algorithm [1, 2].

2 Design and Structure of OMS

The proposed system is composed of a microcontroller (MCU) module, an Android device and database server. Fig. 1 shows a block diagram of the system. Basically, the camera fitted on the mobile device will monitor and record front images. The accelerator sensor will detect shocks or any accidents occurred through the fuzzy integral algorithm [3]. These recorded images will be saved in the SD memory of the mobile device. When required, these will be stored in a remote server via wireless links from the SD memory. When an accident occurs or an emergency condition exists, a rescue message will automatically be sent to next of kin. The Android-based mobile development is based on Google Android 2.3 Operating System [4] and some Google libraries have been used to implement the proposed functions. The top menu consists of three submenus: Camera View, Map View and Accelerometer View.

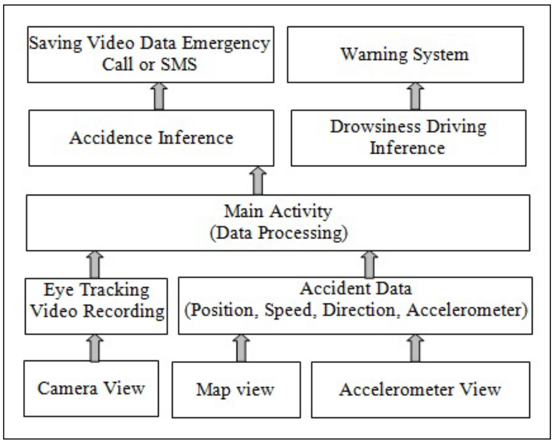


Figure 1: Block diagram of the proposed system

2.1 Camera view

Camera View is implemented with a periodical recording function. Various Google libraries are used. For example, recInstallize() is defined for image formatting, encoding modes and the length of recording, whereas fileOutStream() creates camera recorded image files and chkAccident() detects an occurrence of accident and recPrepare() sets up a saving route for the files. Once recorder.start() is invoked, the image recording will begin for the pre-defined time. When recorder.stop() is invoked, the files will be stored at the SD memory.

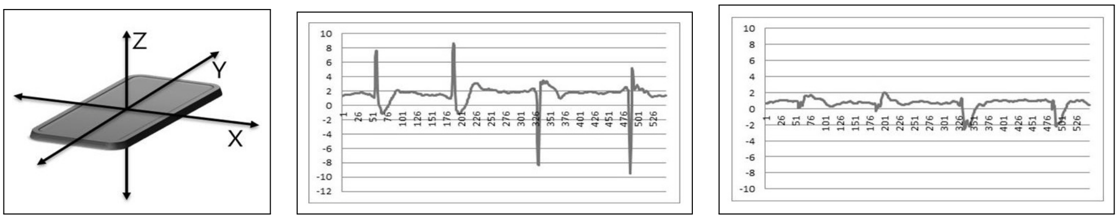
During this main activity, if the MCU and accelerator sensor receive any data for the possible detection of accidents or shocks, an alarm dialogue will be sent out and the associated file will be automatically saved with a file extension of Accident.To control an overflow issue, old files are sequentially deleted while any files with the extension of Accident remain stored.

2.2 Map view

To obtain location information, a few more libraries are utilized such as onCreate() and getBestProvider() for the best provider of the information (Wi-Fi, 3G or GPS satellite). As location changes, LocationListener() moves its position and updates its latitude and longitude. With the assumption that the power is supplied from the vehicle, setPowerRequirement() and setAccuracy() are set to ACCURACY-FINE and POWER-HIGH for a fine collection of the data. All collected data are stored as a text file and transmitted to a remote database server. The location of the vehicle will be displayed on Google Map online.

2.3 Accelerometer view

Sensors applied in this development are accelerator sensors, gyroscope and proximity sensors. For the detection of accidents (or shocks), accelerator sensors and GPS speed data as well as shock sensors located inside MCU are employed. Fig. 2(a) shows the directions of vehicle from the perspective of a mobile placed upright, where Y axis indicates left and right, and Z axis the front and rear, respectively. Figs. 2(b) and 2(c) show the response of the accelerator sensor when two shocks are applied in front and another two from the rear.



(a) (b) (c)

Figure 2: Accelerometer View: (a) Definition of vehicle direction (b) and (c) Values of the accelerometer sensor under shock conditions

Although this method of shock detection provides a degree of accuracy, it needs to enhance its accuracy further, particularly when the vehicle is attacked diagonally or at different angles other than basic four directions. This drawback would result in inaccurate detection of accident. In order to address this ambiguous situation, the study has employed the fuzzy logic and integral algorithm by treating this condition as fuzziness. Fig. 3 shows accelerator values and its scope ranging from S1 up to S8. Values of Y and Z axes and rule-base of accelerator are shown in Fig.4.

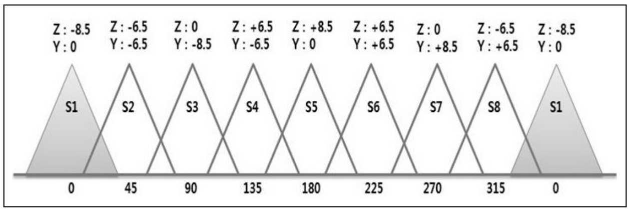
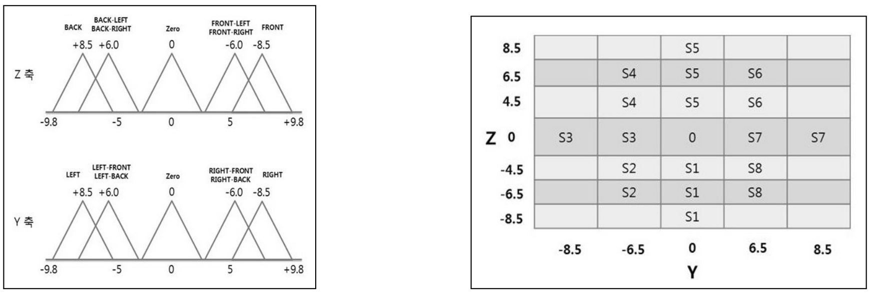


Figure 3: Block diagram of the proposed system



(a) (b)

Figure 4: (a) Values of Y and Z axes (b) Rule-base of accelerator

We first define variables for Y and Z axes and each accelerator value will undergo fuzzification and defuzzification. Then, by making use of the center of gravity, the angle of vehicle shock can be determined. Fig. 5 shows how to find the angle of shock using this method.

2.4 Methodology of drowsiness detection

In order to reduce or prevent accidents due to driving under drowsiness, a detection mechanism of drowsiness has been developed. An eye tracing algorithm known as Haar-like feature

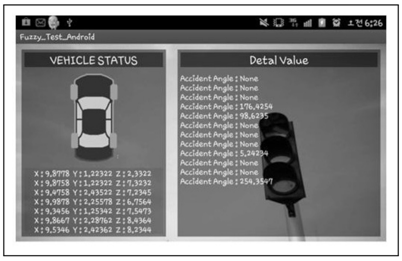


Figure 5: Method of finding the angle of shock

[5] has been used to determine if a driver is in a drowsy condition. The Haar-like features are digital image features used in object detection and recognition [2, 5]. The method first identifies face and eye features and adds pixel values within that area. Then, some weighting factors are multiplied to find a final value. For this reason, this feature is useful in moving pictures rather than still images.

To detect drowsiness, we have used eye closure. This method is well known and widely used for detecting drowsiness [6]. When the camera fitted shows an image of a driver, the detection algorithm starts to find face. Once the face is recognized, the detection for eyes begins to initiate the eye closure based drowsiness procedure. The method assumes that the drivers eyes are closed if eyes are not detected. In detecting the eye closure, blink count increments and timer starts to measure the time during which eyes are closed. Therefore, PERCLOS [6] is computed as follows:

$$PERCLOS = \frac{Eye\ closure\ time}{Measurement\ time} X100$$

According to the values of PERCLOS, we determine if the driver is under drowsiness. The detailed procedure of the method is shown in Fig. 6.

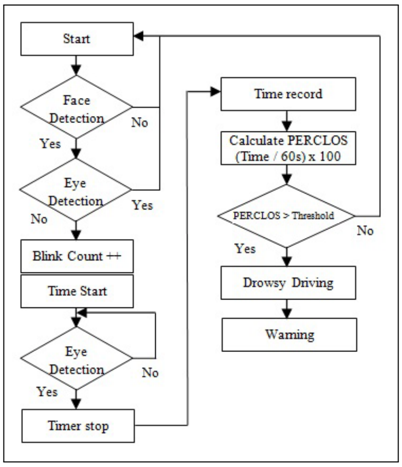


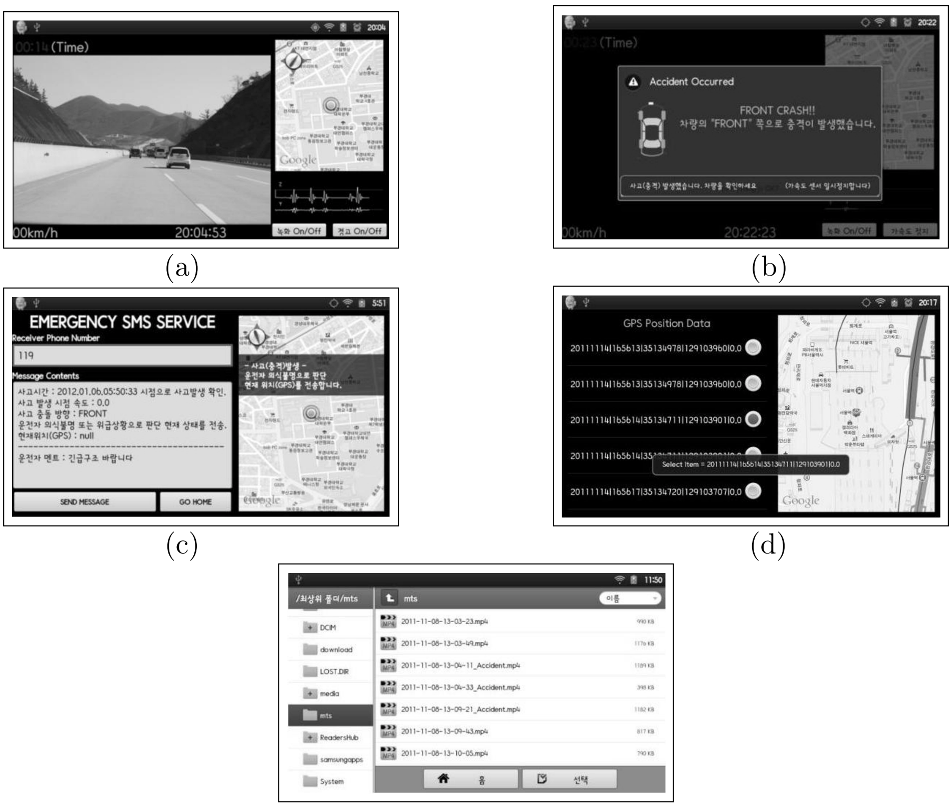
Figure 6: Flow chart of drowsiness detection

3 Verification of OMS

The OMS has been under rigorous verification for its all functions. As mentioned previously, we first evaluate the main menu composed of the front image, Google Map and accelerator sensor as shown in Fig.7 (a). When the vehicle is in motion, the accelerator sensor is in operation for detecting any shocks or accidents. If accidents or shocks occur, an alarm dialog window appears as shown in Fig.7 (b). In order to avoid any false alarm situations, the dialog window allows users to tap OK button to continue the driving, otherwise it regards as an emergency condition to initiate rescue operations, e.g. call initiation to next of kin as shown in Fig.7(c). In addition, the recorded image files before and after that moment of the emergency condition are stored with GPS location information into the SD card memory of the mobile device and transmitted to a remote database server via wireless links, e.g. Wi-Fi or 3G. These files can be viewed anytime as indicated clearly with an extension of Accident. Figs.7 (d) and 7(e) show the file storage and GPS data.

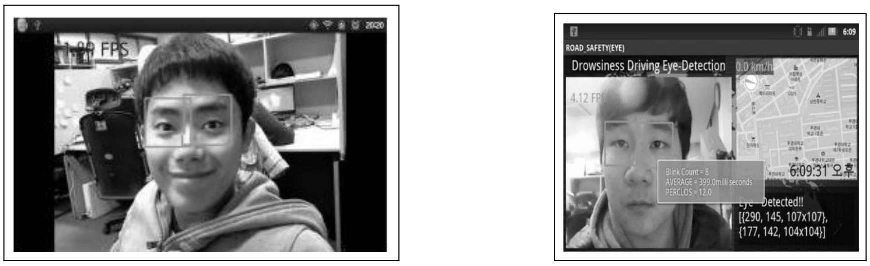
While the vehicle is in motion, the drivers eyes are continuously monitored. If the value of PERCLOS is higher than the predefined threshold value, the system is set to send out a warning message and an alarm sound also goes off to alert the driver for the prevention of any possible accidents. Fig.8 (a) shows the detection of face and eyes subsequently. The eye closure detection is then performed and PERCLOS is computed to determine if the driver is in a drowsy condition as shown in Fig.8 (b).

The verification process demonstrates that the OMS operates the proposed functions accurately as designed.



(a) (b) (c) (d) (e)

Figure 7: Verification of OMS functions



(a) (b)

Figure 8: (a) Detection of face (b) Eye detection for drowsiness

4 Conclusions

Fuzzy integral and image processing techniques integrated on a smartphone are employed to provide the enhanced safety of vehicle. In addition to providing monitoring functions such as the Record and Report functions, an advance warning system for fatigue related accidents is developed using the Haar-like feature. The test bed comprising hardware and associated software has been evaluated through its verification process. The system shows its accurate and correct operations on possible driving scenarios. Also, the drowsiness of a driver is successfully detected using PERCLOS. It is therefore expected that the developed vehicle monitoring system will play an essential role in providing enhanced safety feature to transportation machine to reduce vehicle-borne accidents significantly.

Bibliography

[1] P. Viola; M. Jones, (2001); Robust real-time object detection, *Second International Workshop On statistical and computational Theories of Vision Vancouver*, Canada.

[2] J. Zhou; L. Jiang; L.Shen, (2009);Haar-like features based eye detection algorithm and its implementation on TI TMS320DM6446 platform, *International Workshop on Imaging Systems and Techniques.*

[3] J.M. Mendel (1995); Fuzzy logic systems for engineering: a tutorial, *Proc. of IEEEl*, 83(3):345-377.

[4] http://www.developer.android.com

[5] Alexander Kuranov; Rainer Lienhart; Vadim Pisarevsky.(2002); *An Empirical Analysis of Boosting Algorithm for Rapid Objects with an Extended Set of Haar-like Features*,Intel Technical Report MRL-TR.

[6] U.Trustschel; Bo.Sirois; D.Sommer; M.Golz; D.Edwards (2011); PERCLOS: An alertness measure of the pastl, *Proc. of the Sixth International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Designl*, 172-179.

[7] J.Y.Hwang; S.I. Jeong; Y.H.Chung (2011); Development of vehicle motion monitoring module based on smartphonel, *Journal of Information and Communication Eng, KIICEl*, 15(9):1903-1909.

[8] http://www.atmel.com

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):594-607, August, 2013.

A Fuzzy Data Envelopment Analysis Approach based on Parametric Programming

S.H. Razavi, H. Amoozad, E.K. Zavadskas, S.S. Hashemi

Seyed Hossein Razavi Hajiagha
Institute for Trade Studies and Research,
No. 240, North Kargar St., Tehran, Iran,
s.hossein.r@gmail.com

Hannan Amoozad Mahdiraji
Kashan Branch, Islamic Azad University,
Kashan, Iran,
h.amoozad@ut.ac.ir

Edmundas Kazimieras Zavadskas*
3*Vilnius Gediminas Technical University,
Faculty of Civil Engineering, Sauletekio al. 11,
LT-10223Vilnius, Lithuania.
edmundas.zavadskas@vgtu.lt
*Corresponding author

Shide Sadat Hashemi
Kashan Branch, Islamic Azad University,
Kashan, Iran,
shide_hashemi@yahoo.com

Abstract: In this paper, a fuzzy version of original data envelopment models, CCR and BCC, is extended and its solution approach is developed. The basic idea of the proposed method is to transform the original DEA model to an equivalent linear parametric programming model, applying the notion of α-cuts. Then, a bi-objective model is constructed which its solution has determined the optimal range of decision making units efficiency. The proposed method can be used both for symmetric and asymmetric fuzzy numbers, while the feasibility of its solution for the original problem is guaranteed. The application of the proposed method is examined in two numerical examples and its results are compared with two current models of fuzzy DEA.
Keywords: Data envelopment analysis; Fuzzy numbers; α-cuts; Parametric programming.

1 Introduction

From the early stages of modernization, the limitation of resources was one of the major challenges in managerial decisions. This limitation motivated the managers to be sensitive about the utilization of acquired resources. The origin of efficiency problem can be referred to this sensitivity. Roughly speaking, efficiency tries to answer this question: how well an organization uses its resources to produce the desired outputs? Economists suggest the concept of production function as a tool to appraise efficiency and therefore, a majority of methods that are introduced for efficiency appraisal are based on the approximation of this function (see [15] for a definition of production function). Based on the concept of production function, the efficiency evaluation methods can be classified into two groups: (1) parametric methods which directly approximate the production function, like stochastic frontier analysis [20], and (2) non-parametric methods which indirectly approximate this function. Data envelopment analysis is one of the well known and widely accepted methods in non parametric class [6].

Farell (1957) in his paper introduced a method of efficiency evaluation which is known as the origin of DEA. According to the Pareto-Koopmans definition, he divided the efficiency of each unit into two technical and assignment components [12]. Later, in 1978, Charnes, Cooper, and Rhodes developed the DEA method based on the Farells model. The first DEA model was called CCR model due to its authors [4]. After 1978, the DEA method was widely known and accepted as a permanent paradigm in efficiency evaluation. For instance, Emrouznejad [11] and Cook and Seiford [7] surveyed more than thousands of papers and applications of DEA in different fields.
DEA is a set of linear programming based methods for evaluation the efficiency of a group of homogeneous units which use a set of inputs to produce a set of outputs. DEA considers the efficiency of each unit as the ratio between its weighted sums of outputs to the weighted sums of inputs. In contrast with the classical methods of constant weights, DEA allows each unit to take its variable weights such that its efficiency is maximized, while the efficiency of all units is constrained to be less than one. It can be concluded that the DEA weights are closely related to its inputs and outputs data, and a small swing in units data will have a great influence on the DEA results. The original DEA models were developed based on crisp and deterministic data and no deviation in data were allowed. However, in practical applications, this assumption is violated.
Scholars proposed some frameworks to deal with data uncertainty and non crispness. Some papers examined the efficiency problem under stochastic data [27, 28]. Bellman and Zadeh [2] in their highly cited paper introduced the concept of decision making in a fuzzy environment. While many inputs and outputs are stated by qualitative or lingual variables, applications of the fuzzy sets theory in DEA are proposed in literature.
The application of fuzzy sets theory in DEA can be traced to Sengupta [29]. Since that time, there are a continuously increasing interest on fuzzy DEA methods and applications. Hatami-Marbini et al. [14] classified the fuzzy DEA method into four primary categories which some instances are just introduced here: (1) the tolerance approach [29], (2) the α-level based approach [19, 26], (3) the fuzzy ranking approach [13], and (4) the possibility approach [21, 22]. Also, some approaches like Luban [23], Wang et al. [31], and Zerafat Angiz et al. [33] are known as other developments in fuzzy DEA.
In this paper, a method is proposed to solve the fuzzy DEA problems by transformation of fuzzy DEA problem to an equivalent interval problem, using the concept of α-cuts. The obtained interval problem is a parametric problem based on α which a solution method is proposed to solve it as a bi-objective problem with the concept of compromise programming. The Pareto efficiency of the transformed interval problem is also proved for the fuzzy DEA problem.
The rest of paper is organized as follows. Section 2 consist a brief overview on multiplier and envelopment form of DEA CCR and BCC models. The required fuzzy set definitions and operations are overviewed in section 3. The fuzzy DEA problem and its solution procedure are described in section 4. Two numerical examples are solved in section 5 and the proposed methods solutions are compared with some of the current fuzzy DEA methods. Finally, the paper is concluded in section 6.

2 Data envelopment analysis

Data Envelopment Analysis (DEA) measures the relative efficiency of a set of congruent decision making units (DMUs) that consume multiple inputs to produce multiple outputs. In fact, DEA is a multi-factor productivity analysis model for measuring the relative efficiencies of a homogenous set of DMUs. The efficiency score in the presence of multiple inputs and outputs is defined as the weighted sum of outputs to the weighted sum of inputs. Let, there are n DMUs which used m − dimensional input vector $x_j = \lfloor x_{1j}, ..., x_{mj} \rfloor$ to produce an s − dimensional output vector $y_j = \lfloor y_{1j}, ..., y_{sj} \rfloor$. Then, the relative efficiency of DMU_0, $0 \in \{1, 2, ..., n\}$ will be as follows:

$$E_0 = \sum_{r=1}^{s} u_r y_{r0} / \sum_{i=1}^{m} v_i x_{i0}. \tag{1}$$

The basic form of CCR model can be illustrated as follows:

$$\begin{gathered} Max \sum_{r=1}^{s} u_r y_{r0} / \sum_{i=1}^{m} v_i x_{i0} \\ \sum_{r=1}^{s} u_r y_{rj} / \sum_{i=1}^{m} v_i x_{ij} \leqslant 1,\ j = 1, 2, ..., n \\ u_r \geqslant 0,\ \ r = 1, 2, ..., s \\ v_i \geqslant 0,\ \ i = 1, 2, ..., m \end{gathered} \tag{2}$$

The model (2) is a fractional programming which is converted to a linear programming model by Charnes " Cooper [4] variable transformation as follows:

$$\begin{gathered} Max \sum_{r=1}^{s} u_r y_{r0} \\ \sum_{i=1}^{m} v_i x_{i0} = 1, \\ \sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} \leqslant 0,\ j = 1, 2, ..., n \\ u_r \geqslant 0,\ \ r = 1, 2, ..., s \\ v_i \geqslant 0,\ \ i = 1, 2, ..., m \end{gathered} \tag{3}$$

Model (3) is called the multiplier output oriented CCR model. The CCR models are extended under the constant return to scale assumption. In the case of variable return to scale, the BCC model [1] can be used. The BCC formulation is as follows:

$$\begin{gathered} Max \sum_{r=1}^{s} u_r y_{r0} \\ \sum_{i=1}^{m} v_i x_{i0} - v_0 = 1, \\ \sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} - v_0 \leqslant 0,\ j = 1, 2, ..., n \\ u_r \geqslant 0,\ \ r = 1, 2, ..., s \\ v_i \geqslant 0,\ \ i = 1, 2, ..., m \end{gathered} \tag{4}$$

Where, is the free in sign return to scale variable. A model is CCR or BCC technically efficient if the objective functions in models (3) or (4) are equal to one and its slack variables are zero. A comprehensive review on different DEA models and their economic interpretations are discussed in $[8, 25]$.

3 Fuzzy sets theory

Fuzzy sets are introduced by Zadeh [32] as a generalization of classic sets. Suppose that U is a universe. A fuzzy set $\tilde{A}$ in U is defined as $\tilde{A} = \left\{ \left(x, \mu_{\tilde{A}}(x) \right) | x \in U \right\}$, where $\mu_{\tilde{A}}(x)$ is called the membership function of $\tilde{A}$. If $\mu_{\tilde{A}}(x) : U \to [0,1]$, then $\tilde{A}$ is called a normal fuzzy set. Jain [18] and Dubois and Prade [10] initially defined the concept of fuzzy numbers. A fuzzy number is a normal and convex fuzzy set $\tilde{A}$ in universe U. The most common form of fuzzy numbers in practical problems, especially in decision making related problems are trapezoidal and triangular fuzzy numbers. A trapezoidal fuzzy number can be shown as the quadruple $\tilde{A} = (l, m_1, m_2, r)$, where $l \leqslant m_1 \leqslant m_2 \leqslant r$are real numbers. A trapezoidal fuzzy number $\tilde{A}$ is characterized by its membership function as follows:

$$\mu_{\tilde{A}}(x) = \begin{cases} 0, \;\; x \leqslant l \\ \dfrac{x-l}{m_1 - l}, \;\; l \leqslant x \leqslant m_1 \\ 1, \;\; m_1 \leqslant x \leqslant m_2 \\ .\dfrac{r-x}{r-m_2}, \;\; m_2 \leqslant x \leqslant r \\ 0, \;\; x \geqslant r \end{cases} \tag{5}$$

Triangular fuzzy number is a specific form of trapezoidal fuzzy numbers where $m_1 = m_2$. A required concept of fuzzy numbers in this paper is the concept of α-cuts. For a fuzzy set $\tilde{A}$, its α - cut is defined as $\tilde{A} = \left\{ x \in U | \mu_{\tilde{A}}(x) \geqslant \alpha \right\}$. The α - cuts can be shown as crisp intervals which are called α - level interval:

$$\left(\tilde{A}\right)_\alpha = \left[\left(\tilde{A}\right)_\alpha^l, \left(\tilde{A}\right)_\alpha^u \right] = \left[\begin{array}{l} \min\limits_x \left\{ x \in U | \mu_{\tilde{A}}(x) \geqslant \alpha \right\}, \\ \max\limits_x \left\{ x \in U | \mu_{\tilde{A}}(x) \geqslant \alpha \right\} \end{array} \right] \tag{6}$$

For a trapezoidal fuzzy number $\tilde{A} = (l, m_1, m_2, r)$, its α - level interval is determined as follows:

$$\left(\tilde{A}\right)_\alpha = [m_1\alpha + l(1-\alpha), m_2\alpha + r(1-\alpha)] \tag{7}$$

The arithmetic operations can be defined on fuzzy numbers [34]. An alternative way of fuzzy arithmetic can be defined based on interval arithmetic of α-level intervals. The interval arithmetic is described in Moore et al. [24]. If $\tilde{A}$ and $\tilde{B}$ be two fuzzy numbers with α-level intervals $\left(\tilde{A}\right)_\alpha = \left\lfloor \left(\tilde{A}\right)_\alpha^l, \left(\tilde{A}\right)_\alpha^u \right\rfloor$ and $\left(\tilde{B}\right)_\alpha = \left\lfloor \left(\tilde{B}\right)_\alpha^l, \left(\tilde{B}\right)_\alpha^u \right\rfloor$, then it follows that [3, 30]:

$$\left(\tilde{A}\right)_\alpha + \left(\tilde{B}\right)_\alpha = \left\lfloor \left(\tilde{A}\right)_\alpha^l + \left(\tilde{B}\right)_\alpha^l, \left(\tilde{A}\right)_\alpha^u + \left(\tilde{B}\right)_\alpha^u \right\rfloor \tag{8}$$

$$\left(\tilde{A}\right)_\alpha - \left(\tilde{B}\right)_\alpha = \left\lfloor \left(\tilde{A}\right)_\alpha^l - \left(\tilde{B}\right)_\alpha^u, \left(\tilde{A}\right)_\alpha^u - \left(\tilde{B}\right)_\alpha^l \right\rfloor \tag{9}$$

$$\left(\tilde{A}\right)_\alpha \left(\tilde{B}\right)_\alpha = \left[\begin{array}{l} \min \left\{ \left(\tilde{A}\right)_\alpha^l \left(\tilde{B}\right)_\alpha^l, \left(\tilde{A}\right)_\alpha^l \left(\tilde{B}\right)_\alpha^u, \left(\tilde{A}\right)_\alpha^u \left(\tilde{B}\right)_\alpha^l, \left(\tilde{A}\right)_\alpha^u \left(\tilde{B}\right)_\alpha^u, \right\}, \\ \max \left\{ \left(\tilde{A}\right)_\alpha^l \left(\tilde{B}\right)_\alpha^l, \left(\tilde{A}\right)_\alpha^l \left(\tilde{B}\right)_\alpha^u, \left(\tilde{A}\right)_\alpha^u \left(\tilde{B}\right)_\alpha^l, \left(\tilde{A}\right)_\alpha^u \left(\tilde{B}\right)_\alpha^u, \right\} \end{array} \right] \tag{10}$$

$$1/\left(\tilde{B}\right)_\alpha = \left\lfloor 1/\left(\tilde{B}\right)_\alpha^l, 1/\left(\tilde{B}\right)_\alpha^u \right\rfloor \tag{11}$$

$$\left(\tilde{A}\right)_\alpha \div \left(\tilde{B}\right)_\alpha = \left(\tilde{A}\right)_\alpha \times \frac{1}{\left(\tilde{B}\right)_\alpha} \tag{12}$$

For an interval number $\left(\tilde{A}\right)_\alpha = \left\lfloor \left(\tilde{A}\right)_\alpha^l, \left(\tilde{A}\right)_\alpha^u \right\rfloor$, its center is defined as follows:

$$\left[\left(\tilde{A}\right)_\alpha\right]_c = \frac{\left(\tilde{A}\right)_\alpha^l + \left(\tilde{A}\right)_\alpha^u}{2} \tag{13}$$

4 Fuzzy Data Envelopment Analysis

A fuzzy DEA problem can be stated as follows: consider a group of n decision making units. Each $DMU_j,\ j = 1, 2, ..., n$ used a set of fuzzy inputs $\tilde{X}_i = (\tilde{x}_{i1}, \tilde{x}_{i2}, ..., \tilde{x}_{im})$ to produce a set of fuzzy outputs $\tilde{Y}_i = (\tilde{y}_{i1}, \tilde{y}_{i2}, ..., \tilde{y}_{is})$,where components of $\tilde{X}_i$ and $\tilde{Y}_i$ are fuzzy numbers. Then, the fuzzy CCR model can be formulated as follows:

$$\begin{gathered} Max \sum_{r=1}^{s} u_r \tilde{y}_{r0} \\ \sum_{i=1}^{m} v_i \tilde{x}_{i0} \cong 1 \\ \sum_{r=1}^{s} u_r \tilde{y}_{rj} - \sum_{i=1}^{m} v_i \tilde{x}_{ij} \tilde{\leqslant} 0,\ j = 1, 2, ..., n \\ u_r \geqslant 0,\ r = 1, 2, ..., s \\ v_i \geqslant 0,\ i = 1, 2, ..., m \end{gathered} \tag{14}$$

Where $\cong$ and $\tilde{\leqslant}$ are fuzzy equality and inequality, means "approximately equal to" and "approximately smaller than". The model in Eq. (14) is an input oriented fuzzy CCR model (I F-CCR). In a specific α-level, $\alpha \in [0, 1]$, the α-level efficiency of DMU_0, E_α, can be achieved by solving the following model:

$$\begin{gathered} E_\alpha = Max \sum_{r=1}^{s} u_r (\tilde{y}_{r0})_\alpha \\ \sum_{i=1}^{m} v_i (\tilde{x}_{i0})_\alpha = 1 \\ \sum_{r=1}^{s} u_r (\tilde{y}_{rj})_\alpha - \sum_{i=1}^{m} v_i (\tilde{x}_{ij})_\alpha \leqslant 0,\ j = 1, 2, ..., n \\ u_r \geqslant 0,\ r = 1, 2, ..., s \\ v_i \geqslant 0,\ i = 1, 2, ..., m \end{gathered} \tag{15}$$

While $(\tilde{y}_{rj})_\alpha \subseteq \tilde{y}_{rj}$ and $(\tilde{x}_{i0})_\alpha \subseteq \tilde{x}_{i0}$, then for each $\alpha \in [0,1]$ the feasible space of model (15) is a subset of feasible space of model (14) and therefore, each feasible solution of (15) is a feasible solution of (14). Therefore, it can be stated that:

Lemma1. *If S_a is the feasible space of model (15) and S_0 is the feasible space of model (14), then* $\forall\alpha \in [0,1],\ S_\alpha \subseteq S_0$.

Substituting the α-level intervals in model (15), the model (16) will be obtained as follows:

$$E_\alpha = Max \sum_{r=1}^{s} u_r \left[(\tilde{y}_{r0})^l_\alpha, (\tilde{y}_{r0})^u_\alpha\right]$$

$$\sum_{i=1}^{m} v_i \left[(\tilde{x}_{i0})^l_\alpha, (\tilde{x}_{i0})^u_\alpha\right] = 1 \tag{16}$$

$$\sum_{r=1}^{s} u_r \left[(\tilde{y}_{rj})^l_\alpha, (\tilde{y}_{rj})^u_\alpha\right] - \sum_{i=1}^{m} v_i \left[(\tilde{x}_{ij})^l_\alpha, (\tilde{x}_{ij})^u_\alpha\right] \leqslant 0,\ j = 1, 2, ..., n$$

$$u_r \geqslant 0,\ \ r = 1, 2, ..., s$$

$$v_i \geqslant 0,\ \ i = 1, 2, ..., m$$

The model (16) is an interval linear programming model and can be solved by interval linear programming techniques. To solve the model (16), first let define some ordering relations between interval numbers. The objective function of model (17) is as the following form:

$$E_\alpha = Max \left[\sum_{r=1}^{s} u_r (\tilde{y}_{r0})^l_\alpha, \sum_{r=1}^{s} u_r (\tilde{y}_{r0})^u_\alpha\right] \tag{17}$$

According to Ishibuichi and Tanaka [17] and Das et al. [9], the E_α will be maximized if and only if its lower bound and center are maximized, i.e.:

$$E_\alpha = Max \begin{bmatrix} \sum\limits_{r=1}^{s} u_r (\tilde{y}_{r0})^l_\alpha, \\ \sum\limits_{r=1}^{s} u_r \left((\tilde{y}_{r0})^l_\alpha + (\tilde{y}_{r0})^u_\alpha / 2\right) \end{bmatrix} \tag{18}$$

The first constraint of model (16) is justified as follows:

$$\sum_{i=1}^{m} v_i \left[\frac{(\tilde{x}_{i0})^l_\alpha + (\tilde{x}_{i0})^u_\alpha}{2}\right] = 1 \tag{19}$$

Because this constraint required that $\sum\limits_{i=1}^{m} v_i (\tilde{x}_{i0})^l_\alpha = 1$, and $\sum\limits_{i=1}^{m} v_i (\tilde{x}_{i0})^u_\alpha = 1$. Adding these equations and dividing by 2, the Eq. (19) is obtained.

Also, the second set of constraints is modified as follows to a linear set of constraints:

$$\begin{bmatrix} \sum\limits_{r=1}^{s} u_r (\tilde{y}_{rj})^l_\alpha - \sum\limits_{i=1}^{m} v_i (\tilde{x}_{ij})^u_\alpha, \\ \sum\limits_{r=1}^{s} u_r (\tilde{y}_{rj})^u_\alpha - \sum\limits_{i=1}^{m} v_i (\tilde{x}_{ij})^l_\alpha \end{bmatrix} \leqslant 0,\ \ j = 1, 2, ..., n \tag{20}$$

The constraints of the form Eq. (20) are handled as follows:

$$\sum_{r=1}^{s} u_r(\tilde{y}_{rj})_{\alpha}^{u} - \sum_{i=1}^{m} v_i(\tilde{x}_{ij})_{\alpha}^{l} \leqslant 0,\ j = 1, 2, ..., n$$

$$\sum_{r=1}^{s} u_r \left(\frac{(\tilde{y}_{rj})_{\alpha}^{l} + (\tilde{y}_{rj})_{\alpha}^{u}}{2}\right) - \sum_{i=1}^{m} v_i \left(\frac{(\tilde{x}_{ij})_{\alpha}^{l} + (\tilde{x}_{ij})_{\alpha}^{u}}{2}\right) \leqslant 0,\ \ j = 1, 2, ..., n \tag{21}$$

Integrating the Eqs. (18) – (21), the following multi objective model is constructed as an equivalent model for input oriented fuzzy CCR model (14).

$$E_{\alpha} = Max \sum_{r=1}^{s} u_r(\tilde{y}_{r0})_{\alpha}^{l}$$

$$Max \sum_{r=1}^{s} u_r \left((\tilde{y}_{r0})_{\alpha}^{l} + (\tilde{y}_{r0})_{\alpha}^{u}/2\right)$$

Subject to

$$\sum_{i=1}^{m} v_i \left[\frac{(\tilde{x}_{i0})_{\alpha}^{l} + (\tilde{x}_{i0})_{\alpha}^{u}}{2}\right] = 1$$

$$\sum_{r=1}^{s} u_r(\tilde{y}_{rj})_{\alpha}^{u} - \sum_{i=1}^{m} v_i(\tilde{x}_{ij})_{\alpha}^{l} \leqslant 0,\ j = 1, 2, ..., n \tag{22}$$

$$\sum_{r=1}^{s} u_r \left(\frac{(\tilde{y}_{rj})_{\alpha}^{l} + (\tilde{y}_{rj})_{\alpha}^{u}}{2}\right) - \sum_{i=1}^{m} v_i \left(\frac{(\tilde{x}_{ij})_{\alpha}^{l} + (\tilde{x}_{ij})_{\alpha}^{u}}{2}\right) \leqslant 0,\ \ j = 1, 2, ..., n$$

$$u_r \geqslant 0,\ \ r = 1, 2, ..., s$$

$$v_i \geqslant 0,\ \ i = 1, 2, ..., m$$

The Eq. (22) is a bi-objective parametric model. This model is then decomposed to two distinct model to (1) maximize the E_{α}^{l}, and (2) maximize the E_{α}^{c}. . If PPS is the feasible space of model (22), these two models are formulated as follows:

$$MaxE_{\alpha}^{l} = \sum_{r=1}^{s} u_r(\tilde{y}_{r0})_{\alpha}^{l} \quad (u, v) \in PPS$$

$$MinE_{\alpha}^{c} = \sum_{r=1}^{s} u_r \left((\tilde{y}_{r0})_{\alpha}^{l} + (\tilde{y}_{r0})_{\alpha}^{u}/2\right) \quad (u, v) \in PPS \tag{23}$$

The models (22) and (23) are linear programming problems which can be solved easily by available packages. The solutions of these models determine the optimal range of DMUs efficiency, i.e. $[E_{\alpha}^{l*}, E_{\alpha}^{U*}]$, which in turn is the efficiency α-cut. In fact, while the outputs of the DMUs are determined by fuzzy numbers, their efficiency scores will be also fuzzy numbers with unknown membership function. The result of model (22) for a specific value of α is the efficiency membership function's α-cut.

The following lemma shows an important relation between models (21) and (14).

Lemma2. *If (u, v) is a feasible solution of model (21), it will be a feasible solution of the original fuzzy DEA model (14).*

The proof is easily obtained from lemma1.

In the case of variable return to scale, the fuzzy BCC model can be formulated as follows:

$$E_\alpha = Max \sum_{r=1}^{s} u_r (\tilde{y}_{r0})_\alpha^l$$

$$Max \sum_{r=1}^{s} u_r \left((\tilde{y}_{r0})_\alpha^l + (\tilde{y}_{r0})_\alpha^u / 2 \right)$$

Subject to

$$\sum_{i=1}^{m} v_i \left[\frac{(\tilde{x}_{i0})_\alpha^l + (\tilde{x}_{i0})_\alpha^u}{2} \right] - v_0 = 1$$

$$\sum_{r=1}^{s} u_r (\tilde{y}_{rj})_\alpha^u - \sum_{i=1}^{m} v_i (\tilde{x}_{ij})_\alpha^l - v_0 \leqslant 0,\ j = 1, 2, ..., n \tag{24}$$

$$\sum_{r=1}^{s} u_r \left(\frac{(\tilde{y}_{rj})_\alpha^l + (\tilde{y}_{rj})_\alpha^u}{2} \right) - \sum_{i=1}^{m} v_i \left(\frac{(\tilde{x}_{ij})_\alpha^l + (\tilde{x}_{ij})_\alpha^u}{2} \right) - v_0 \leqslant 0,\ \ j = 1, 2, ..., n$$

$$u_r \geqslant 0,\ \ r = 1, 2, ..., s$$

$$v_i \geqslant 0,\ \ i = 1, 2, ..., m$$

$$v_0 : \text{unrestricted}$$

The model (24) can be solved with a similar approach to model (21). Analyzing the efficiency of DMUs with the proposed method, for a set of n different values of α, e.g. $\alpha_i,\ \ i = 1, 2, ..., n$ results in a set of efficiency α-cuts $[E_{\alpha_i}^{L*}, E_{\alpha_i}^{U*}]$.Therefore, it will be necessary to obtain an integrated efficiency score for DMUs to rank them. While the membership functions of efficiency are not determined, the conventional methods of fuzzy numbers aggregation which need membership functions cannot be used. Here, the Chen and Klein [5] method is proposed to rank the efficiency scores of DMUs based on their α-cuts, like Kao and Liu [19]. Chen and Klein [5] introduced the following index for fuzzy numbers:

$$I_j = \frac{\sum\limits_{i=0}^{n} \left((E_{\alpha_i}^{U*}) - c \right)}{\sum\limits_{i=0}^{n} \left((E_{\alpha_i}^{U*}) - c \right) - \sum\limits_{i=0}^{n} \left((E_{\alpha_i}^{L*}) - d \right)},\ \ n \to \infty \tag{25}$$

Where, $c = min_{i,j} \left\{ \left(E_{\alpha_i}^{L*} \right) \right\}$ and $d = max_{i,j} \left\{ \left(E_{\alpha_i}^{U*} \right) \right\}$. While n is grown, the methods validity is increased, however Chen and Klein believed that $n = 3$ or 4 is sufficient.

5 Numerical examples

In this section, two numerical examples are solved by the proposed method and the results are compared with previously presented methods.

Example1. Consider 4 DMUs with inputs and outputs which are presented in table1.

Table 1: Inputs and outputs of 4 DMUs

DMU	Input	α-cut	Output	α-cut
A	(11,12,14)	[11+α,14-2α]	10	[10,10]
B	30	[30,40]	(12,13,14,16)	[12+α,16-2α]
C	40	[40,40]	11	[11,11]
D	(45,47,52,55)	[45+2α,55-3α]	(12,15,19,22)	[12+3α,22-3α]

Kao and Liu [19] solved this problem in an 11 point scale for α based on BCC model. Table 2 shows the Kao and Liu's results.

Considering the DMU D, the model (22) is designed for this DMU as follows. First, two single objective models are solved:

$$
\begin{aligned}
&E_l^{\alpha}(D) : Max(12+3\alpha)u_1\\
&\text{Subject to}\\
&\left(\frac{100-\alpha}{2}\right)v_1+v_0=1\\
&10u_1-(11+\alpha)v_1-v_0\leqslant 0\\
&(16-2\alpha)u_1-30v_1-v_0\leqslant 0\\
&11u_1-40v_1\leqslant 0\\
&(22-3\alpha)u_1-(45+3\alpha)v_1-v_0\leqslant 0\\
&10u_1-\left(\frac{25-\alpha}{2}\right)v_1-v_0\leqslant 0\\
&\left(\frac{28-\alpha}{2}\right)u_1-30v_1-v_0\leqslant 0\\
&\frac{17u_1}{2}-\left(\frac{100-\alpha}{2}\right)v_1-v_0\leqslant 0\\
&u_1\geqslant 0,\ v_1\geqslant 0,\ v_0:\text{unrestricted}
\end{aligned}
$$

$$
\begin{aligned}
&E_l^{\alpha}(D) : Max\frac{17}{2}u_1\\
&\text{Subject to}\\
&\left(\frac{100-\alpha}{2}\right)v_1+v_0=1\\
&10u_1-(11+\alpha)v_1-v_0\leqslant 0\\
&(16-2\alpha)u_1-30v_1-v_0\leqslant 0\\
&11u_1-40v_1-v_0\leqslant 0\\
&(22-3\alpha)u_1-(45+3\alpha)v_1-v_0\leqslant 0\\
&10u_1-\left(\frac{25-\alpha}{2}\right)v_1-v_0\leqslant 0\\
&\left(\frac{28-\alpha}{2}\right)u_1-30v_1-v_0\leqslant 0
\end{aligned}
$$

$$\frac{17u_1}{2} - \left(\frac{100-\alpha}{2}\right) v_1 - v_0 \leqslant 0$$
$u_1 \geqslant 0,\ v_1 \geqslant 0,\ v_0$: unrestrict end

These models are designed and solved for different α-levels. The results are shown in table 2.

Let consider two sets of solutions with more details. Considering the DMUs A and C, which their outputs are stated by crisp numbers, it is intuitionally more acceptable that their efficiency in an input oriented model and in each specific level of α be a crisp number, as it is in table 2.

Now consider the DMU A. Solving its lower bound model with Kao and Liu [19], for $\alpha = 0$, the optimal solution is as follows:

Table 2: The α-cuts of the efficiency at eleven α-values based on the proposed method and Kao and Liu [19] method

A	$\left[(E_A)^l_\alpha, (E_A)^u_\alpha\right]$		$\left[(E_B)^l_\alpha, (E_B)^u_\alpha\right]$		$\left[(E_C)^l_\alpha, (E_C)^u_\alpha\right]$		$\left[(E_D)^l_\alpha, (E_D)^u_\alpha\right]$	
	Proposed	Kao and Liu	Proposed	Kao and Liu	Proposed	Kao and Liu	Proposed	Kao and Liu
0.0	0.9497	[1.0,1.0]	[0.71,0.95]	[0.71,1.0]	0.5436	[0.54,0.91]	[0.54,1]	[0.74,1.0]
0.1	0.9557	[1.0,1.0]	[0.73,0.95]	[0.73,1.0]	0.5523	[0.55,0.90]	[0.56,1]	[0.77,1.0]
0.2	0.9615	[1.0,1.0]	[0.75,0.95]	[0.75,1.0]	0.5612	[0.56,0.88]	[0.58,1]	[0.80,1.0]
0.3	0.9671	[1.0,1.0]	[0.76,0.95]	[0.76,1.0]	0.5703	[0.57,0.87]	[0.61,1]	[0.83,1.0]
0.4	0.9725	[1.0,1.0]	[0.78,0.95]	[0.78,1.0]	0.5795	[0.58,0.86]	[0.63,1]	[0.86,1.0]
0.5	0.9776	[1.0,1.0]	[0.79,0.95]	[0.79,1.0]	0.5890	[0.58,0.85]	[0.65,1]	[0.89,1.0]
0.6	0.9826	[1.0,1.0]	[0.81,0.95]	[0.81,1.0]	0.5987	[0.59,0.83]	[0.68,1]	[0.93,1.0]
0.7	0.9873	[1.0,1.0]	[0.83,0.95]	[0.83,1.0]	0.6086	[0.60,0.82]	[0.70,1]	[0.96,1.0]
0.8	0.9917	[1.0,1.0]	[0.85,0.95]	[0.85,1.0]	0.6187	[0.61,0.81]	[0.73,1]	[0.99,1.0]
0.9	0.9960	[1.0,1.0]	[0.87,0.95]	[0.87,1.0]	0.6290	[0.62,0.80]	[0.76,1]	[1.0,1.0]
1.0	1.0	[1.0,1.0]	[0.88,0.95]	[0.88,1.0]	0.6395	[0.63,0.79]	[0.78,1]	[1.0,1.0]

$\lfloor u_1^* = 0.1, v_1^* = 0.07142857, v_0^* = 0.511766 \rfloor$. Replacing this solution in the first constraint of model (14), the following equality is obtained: (1.29747587, 1.3689044, 1.51176158) = 1. However, it is clear that $1 \in (1.29747587, 1.3689044, 1.51176158)$ and therefore this constraint is violated. In fact, the Kao and Liu [19] method overestimated the DMUs efficiency. In this case, the proposed methods solution is:

$\lfloor u_1^* = 0.09497207, v_1^* = 0.03351955, v_0^* = 0.5810056 \rfloor$, and the considered constraint is as follows: (0.94972065,0.9832402,1.0502793)=1 which it is clear that $1 \in (0.94972065, 0.9832402, 1.0502793)$.

Example2. Saati-Mohtadi et al. [26] applied their model on an example with 10 DMUs which used two inputs in order to produce two outputs. Data are presented in table 3.

Table 4 presents the results of efficiency appraisal of DMUs based on the proposed method and Saati-Mohtadi et al. [26] method.

Applying Eq. (25) the following results are obtained:
$I_1 = 0.9407,\ I_2 = 0.9599,\ I_3 = 0.3704,$
$I_4 = 0.4629,\ I_5 = 0.5015,\ I_6 = 0.2942,\ I_7 = 0.1409,\ I_8 = 0.1439,\ I_9 = 0.0471,\ I_10 = 0.9243.$
Therefore, the DMUs are ranked based on their efficiencies as follows: $\tilde{E}_2 \succ \tilde{E}_4 \succ \tilde{E}_{10} \succ \tilde{E}_5 \succ \tilde{E}_4 \succ \tilde{E}_3 \succ \tilde{E}_6 \succ \tilde{E}_8 \succ \tilde{E}_7 \succ \tilde{E}_9$

Table 3: Data for 10 DMUs

DMUs	I_1	I_2	O_1	O_2
D_1	(6.0, 7.0, 8.0)	(29.0, 30.0, 32.0)	(35.5, 38.0, 41.0)	(409.0, 411.0, 416.0)
D_2	(5.5, 6.0, 6.5)	(33.0, 35.0, 36.5)	(39.0, 40.0, 43.0)	(478.0, 480.0, 484.0)
D_3	(7.5, 9.0, 10.5)	(43.0, 45.0, 48.0)	(32.0, 35.0, 38.0)	(297.0, 299.0, 301.0)
D_4	(7.0, 8.0, 10.0)	(37.5, 39.0, 42.0)	(28.0, 31.0, 31.0)	(347.0, 352.0, 360.0)
D_5	(9.0, 11.0, 12.0)	(43.0, 44.0, 45.0)	(33.0, 35.0, 38.0)	(406.0, 411.0, 415.5)
D_6	(10.0, 10.0, 14.0)	(53.0, 55.0, 57.5)	(36.0, 38.0, 40.0)	(282.0, 286.0, 289.0)
D_7	(10.0, 12.0, 14.0)	(107.0, 110.0, 113.0)	(34.5, 36.0, 38.0)	(396.0, 400.0, 405.0)
D_8	(9.0, 13.0, 16.0)	(95.0, 100.0, 101.0)	(37.0, 41.0, 46.0)	(387.0, 393.0, 402.0)
D_9	(12.0, 14.0, 15.0)	(120.0, 125.0, 131.0)	(24.0, 27.0, 28.0)	(400.0, 404.0, 406.0)
D_{10}	(5.0, 8.0, 10.0)	(35.0, 38.0, 39.0)	(48.0, 50.0, 51.0)	(470.0, 470.0, 470.0)

Table 4: DMUs efficiency scores in six α-values based on the proposed method (P.) and Saati-Mohtadi et al. [26]

DMUs	0.0		0.2		0.4		0.6	
	P	Saati et al.	P	Saati et al.	P	Saati et al.	P	Saati et al.
D_1	[0.91, 0.94]	1.00	[0.93, 0.96]	1.00	[0.95, 0.97]	1.00	[0.96, 0.98]	1.00
D_2	[0.94, 0.95]	1.00	[0.95, 0.96]	1.00	[0.96, 0.97]	1.00	[0.97, 0.98]	1.00
D_3	[0.48, 0.57]	0.84	[0.50, 0.57]	0.79	[0.52, 0.58]	0.75	[0.54, 0.58]	0.71
D_4	[0.60, 0.62]	0.76	[0.61, 0.63]	0.74	[0.62, 0.63]	0.71	[0.63, 0.64]	0.70
D_5	[0.62, 0.64]	0.78	[0.63, 0.65]	0.75	[0.64, 0.65]	0.73	[0.66, 0.66]	0.71
D_6	[0.44, 0.49]	0.69	[0.46, 0.50]	0.67	[0.47, 0.50]	0.65	[0.49, 0.51]	0.63
D_7	[0.35, 0.35]	0.63	[0.38, 0.39]	0.59	[0.39, 0.40]	0.55	[0.41, 0.42]	0.51
D_8	[0.32, 0.33]	0.85	[0.36, 0.37]	0.75	[0.38, 0.40]	0.66	[0.41, 0.4]	0.59
D_9	[0.31, 0.31]	0.46	[0.34, 0.34]	0.44	[0.34, 0.34]	0.42	[0.35, 0.35]	0.40
D_{10}	[0.89, 0.93]	1.00	[0.91, 0.94]	1.00	[0.93, 0.96]	1.00	[0.95, 0.97]	1.00

DMUs	0.8		1.0	
	P	Saati et al.	P	Saati et al.
D_1	[0.98, 0.99]	1.00	[1.0, 1.0]	1.00
D_2	[0.98, 0.99]	1.00	[1.0, 1.0]	1.00
D_3	[0.56, 0.58]	0.66	[0.61, 0.61]	0.61
D_4	[0.64, 0.65]	0.68	[0.65, 0.65]	0.66
D_5	[0.67, 0.67]	0.69	[0.68, 0.68]	0.68
D_6	[0.53, 0.54]	0.60	[0.58, 0.58]	0.58
D_7	[0.43, 0.44]	0.48	[0.45, 0.45]	0.45
D_8	[0.45, 0.46]	0.53	[0.47, 0.47]	0.47
D_9	[0.35, 0.35]	0.38	[0.36, 0.36]	0.36
D_{10}	[0.97, 0.98]	1.00	[1.0, 1.0]	1.00

Now, let compare the results of the proposed method with [26]. Table5 presents that D_1 attain to full efficiency in all α-levels. Suppose that $\alpha = 0$. If the problem is solved with Saati-Mohtadi et al. [26] method, its results will be $u_1^* = 0.02439024, u_2^* = 0, v_1^* = 0.1406398, v_2^* = 0.005384859$. Replacing this solution in the first constraint of model (14), the result will be as follows: (1, 1.146024, 1.297434) which the membership degree of 1 is zero and it violates the " $\cong$ " relation. However, when the model is solved with the proposed method, both in the multipliers in the lower limit model are $u_1^* = 0, u_2^* = 0.002235469, v_1^* = 0, v_2^* = 0.03278689$ and in the center model, the multipliers are $u_1^* = 0.005162761, u_2^* = 0.001776794, v_1^* = 0, v_2^* = 0.03278689$, where both cases, the first constraint of model (14) is become as (0.95082, 0.983607, 1.04918) which apparently satisfied the " $\cong$ " relation. Both examples show that the feasibility of the proposed method is guaranteed. In fact, in both methods the first constraint is transformed to $\sum\limits_{i=1}^{m} v_i \tilde{x}_{i0} \geq 1$ which consequently overestimated the efficiency of DMUs, while this is prevailed in the proposed method.

6 Conclusions

In this paper a model is proposed to solve data envelopment analysis models, when the inputs and outputs are determined ambiguously by fuzzy numbers. The method is developed based on the concept of α-cuts which transform the fuzzy problem to an equivalent parametric problem. Then, the parametric problem is solved based on different values of α. The proposed method is developed either for constant return to scale CCR model and variable return to scale BCC model. The application of the proposed method is also presented and compared with two existing methods.

One of the advantages of the proposed method is that it can be used for symmetric and asymmetric fuzzy numbers. Also, the proposed method provides frameworks to analysis the efficiency appraisal problems under CCR and BCC models. The major advantage of the proposed model is that its results guaranteed the feasibility of DEA results in original fuzzy model, while other methods don't have such property in some cases. Another feature of the proposed method is that it is a linear programming based parametric method which makes it easy to solve it with the present approaches and applications.

Bibliography

[1] Banker R.D., Charnes A., Cooper W.W., Some models for estimating technical and scale inefficiencies in data envelopment analysis, *MANAGE. SCI.* ISSN 0025-1909, 30 (9): 1078-1092, 1984.

[2] Bellman R.E., Zadeh L.A., Decision making in a fuzzy environment, *MANAGE. SCI.* ISSN 0025-1909, 17 (4): 141-164, 1970.

[3] Bojadziev G., Bojadziev M.,*Fuzzy sets, fuzzy logic, applications*, World Scientific Publishing, Singapore, 1996.

[4] Charnes A., Cooper W.W., Programming with linear fractional functions, *NAV. RES. LOG.* ISSN 0894-069X, 9 (3-4): 181-186, 1962.

[5] Charnes A., Cooper W.W., Rhodes E., Measuring Efficiency of Decision Making Units, *EUR. J. OPER. RES.* ISSN 0377-2217, 2: 429-444, 1978.

[6] Chen C.B., Klein C.M., A simple approach to ranking a group of aggregated fuzzy utilities. IEEE. T. SYST. MAN. CY B. ISSN 1083-4419, 27 (1): 26-35, 1997.

[7] Coelli T.J., Prasada Rao D.S., O'Donnell C.J., Battese G.E., *An Introduction to Efficiency and Productivity Analysis,* 2nd edition, Springer, New York, 2005.

[8] Cook W.D., Seiford L.M., Data envelopment analysis (DEA) " thirty years on. *EUR. J. OPER. RES.* ISSN 0377-2217, 192 (1): 1-17, 2009.

[9] Cooper W.W., Seiford L.M., Tone K., *Data Envelopment Analysis: A Comprehensive Text with Models, Applications, References, and DEA-Solver Software,* 2nd ed., Kluwer Academic Publishers, New York, 2006.

[10] Das S.K., Goswami A., Alam S.S., Multiple objective transportation problem with interval cost, source and destination parameters, *EUR. J. OPER. RES.* ISSN 0377-2217, 117 (1): 100-112, 1999.

[11] Dubois D., Prade H., Operations on fuzzy numbers. *INT. J. SYST. SCI.* ISSN 0020-7721, 9 (6): 613-626, 1978.

[12] Emrouznejad A., Parker B.R., Tavares G. Evaluation of research in efficiency and productivity: A survey and analysis of the first 30 years of scholarly literature in DEA, *SOCIO. ECON. PLAN. SCI.* ISSN 0038-121, 42 (3): 151-157, 2008.

[13] Fare R., Zelenyuk V., On Farrell's Decomposition and aggregation. *INT. J. BUS. ECON.* ISSN 1607-0704, 4 (2): 167-171, 2005.

[14] Guo P. Fuzzy data envelopment analysis and its application to location problems, *INFORM. SCIENCES.* ISSN 0020-0255, 179 (6): 820-829, 2009.

[15] Hatami-Marbini A., Emrouznejad A., Tavana M., A taxonomy and review of the fuzzy data envelopment analysis literature: two decades in the making, *EUR. J. OPER. RES.* ISSN 0377-2217, 214 (3): 457-472, 2011.

[16] Hirschey M., *Managerial Economics,* 12th ed., South Western Cengage Learning, Ohio, 2008.

[17] Ishibuchi H., Tanaka H., Multiobjective programming in optimization of the interval objective function, *EUR. J. OPER. RES.* ISSN 0377-2217, 48 (2): 219-225, 1990.

[18] Jain R., Decision-Making in the presence of fuzzy variables, *IEEE. T. SYST. MAN. CY B.* ISSN 1083-4419, 6 (10): 698-703, 1976.

[19] Kao C., Liu S.T., Fuzzy efficiency measures in data envelopment analysis, *FUZZY. SET. SYST.* ISSN 0165-0114, 113 (3): 427-437, 2000.

[20] Kumbhakar S.C., Knox Lovell C.A., *Stochastic Frontier Analysis*, Cambridge University Press, Cambridge, 2003.

[21] Lertworasirikul S., Fang S.C., Joines J.A., Nuttle H.L.W., Fuzzy data envelopment analysis (DEA): a possibility approach, *FUZZY. SET. SYST.*, ISSN 0165-0114, 139 (2): 379-394, 2003.

[22] Lertworasirikul S., Fang S.C., Nuttle H.L.W., Joines J.A., Fuzzy BCC model for data envelopment analysis, *FUZZY. OPTIM. DECIS. MA.* ISSN 1568-4539, 2 (4): 337-358, 2003.

[23] Luban F., Measuring efficiency of a hierarchical organization with fuzzy DEA method, *ECONOMIA. SERIA. MANAGEMENT.*, ISSN 1454-0320, 12 (1): 87-97, 2009.

[24] Moore R.E., Baker Kearfott R., Cloud M.J., *Introduction to interval analysis,* SIAM, Philadelphia, 2009.

[25] Ray S.C., *Data Envelopment Analysis: Theory and Techniques for Economics and Operations Research*, first ed., Cambridge University Press, New York, 2004.

[26] Saati-Mohtadi S., Memariani A., Jahanshahloo G.R., Efficiency analysis and ranking of DMUs with fuzzy data, *FUZZY. OPTIM. DECIS. MA.*, ISSN 1568-4539, 1 (3): 255-267, 2002.

[27] Sengupta J.K., Data envelopment analysis for efficiency measurement in the stochastic case, *COMPUT. OPER. RES.* ISSN 0305-0548, 14 (2): 117-129, 1987.

[28] Sengupta J.K., Measuring economic efficiency with stochastic input-output data, *INT. J. SYST. SCI.*, ISSN 0020-7721, 20 (2): 203-213, 1989.

[29] Sengupta J.K., A fuzzy systems approach in data envelopment analysis, *COMPUT. MATH. APPL.* ISSN 0898-1221, 24 (8-9): 259-266, 1992.

[30] Siler W., Buckley J.J., *Fuzzy expert systems and fuzzy reasoning*, John Wiley and Sons, New Jersey, 2005.

[31] Wang C.H., Chuang C.C., Tsai C.C., A fuzzy DEA-neural approach to measuring design service performance in PCM projects, *AUTOMAT. CONSTR.*, ISSN 0926-5805, 18 (5): 702-713, 2009.

[32] Zadeh L.A., Fuzzy Sets. *INFORMATION AND CONTROL.*, ISSN 0019-9958, 8 (3): 338-353, 1965.

[33] Zerafat Angiz M., Emrouznejad A., Mustafa A., Al-Eraqi A.S., Aggregating preference ranking with fuzzy data envelopment analysis, *KNOWL-BASED SYST.* ISSN 0950-7051, 23 (6): 512-519, 2010.

[34] Zimmermann H.J., *Fuzzy sets theory and its application*, forth ed., Kluwer Academic Publishers, Massachusetts, 2001.

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):608-621, August, 2013.

Toward a Holistic Delay Analysis of EtherCAT Synchronized Control Processes

M. Sung, I. Kim, T. Kim

Minyoung Sung, Ikhwan Kim, and Taehyoun Kim*
Dept. of Mechanical and Information Engineering,
University of Seoul, Seoul 130-743, Korea
{mysung, ihkim, thkim}@uos.ac.kr
*Corresponding author

Abstract: This paper analyzes the end-to-end delay of EtherCAT-based control processes that use the events of message frames and global clock for synchronized operation. With the end-to-end delay defined as the time interval between the start of a process cycle and the actual input or output, we develop a holistic delay model for control processes with EtherCAT, by taking into account the time for in-controller processing, message delivery, and slave-local handling. Based on the measurements from a real EtherCAT control system, we discuss the average and deviation of the process delay as we vary the number of slaves and process cycle time. The experiment results show that the output delays are mainly increased by the average controller delay, whereas the input delays are more affected by the deviation rather than the average of the controller delay. Our in-depth analysis on the controller reveals that the DMA (Direct Memory Access) overhead chiefly enlarges the controller delay for increasing number of slaves, while task release jitter is the main cause of the increased delay for longer cycle time. The presented delay model and evaluation results can be essentially used for the design of EtherCAT-based automation that requires highly synchronized operations, such as for coordinated motion and high-precision data sensing.
Keywords: real-time Ethernet, end-to-end delay, synchronized processes, automation system.

1 Introduction

Presently, automated control systems are experiencing a steady but fundamental change, i.e., the use of industrial Ethernet as a replacement for conventional fieldbuses [1, 2]. EtherCAT, in particular, is one of the industrial Ethernet standards (IEC 61784 and 61158-2), which is gaining increasing acceptance in precision automation [3–7]. It offers numerous attractive features such as short cycle time as low as dozens of microseconds [6], globally synchronized clock with deviation of the sub-microsecond range [8], and compatibility with TCP/IP. Because of these benefits, EtherCAT is currently being applied in various automation fields including factory automation, robotics, and production machinery [9–11].

An automation system typically requires highly synchronized operations of its components for coordinated actuation and sensing. For example, an industrial robot should be able to actuate its constituent motors in a synchronized manner so that the consequential motion accurately follows the planned trajectory. High-precision distributed measurement is another example, where synchronous operation is important for synchronized sensing and data freshness [12].

With an Ethernet-based control system, the design of such synchronized operations relies on a thorough analysis of the networked process delay, which includes the time for in-controller processing, message delivery, and local handling by each slave, i.e., the controlled device. Thus far, however, there have been few studies that have analyzed the end-to-end delay of EtherCAT automation systems. Although some recent works have addressed the performance of EtherCAT-based automation, they remain incomplete in that they either lack any consideration of the actual automation workload [13] or only deal with the network-level performance [5–7].

This paper evaluates the performance of EtherCAT-based synchronized processes in terms of the end-to-end delay. In EtherCAT, synchronized control schemes can be realized using events from the EtherCAT frames and synchronized clock. These two types of events relate directly to the precision of modern networked control processes: at each slave, the event of a frame reception begins a new cycle of computation, while the event from the synchronized clock can be utilized to latch input data and/or actuate the device in a globally synchronized manner. We formulate the end-to-end delay of the synchronized control schemes and investigate their performance characteristics in a comprehensive way. Aiming for providing an in-depth and practical insight, the performance evaluation has been conducted using a prototype EtherCAT controller that was constructed from open source software. Based on the measurement results using the controller, we discuss the delay performance as we vary the number of slaves and process cycle time.

The contributions of our work are two-folded. First, we present a delay analysis of the EtherCAT synchronized processes. Using a real automation testbed that operates a number of commercial motor drives, we evaluate the performance of EtherCAT control systems in terms of the average and deviation of the end-to-end process delay. To the best of our knowledge, our study is the first to analyze the end-to-end delay of EtherCAT control processes in a holistic way, considering not only the control network but also the actual in-controller automation workload. The experiment results confirm the importance of our in-depth analysis of the controller delay: the average end-to-end delays of output processes are mainly increased by the average controller delay while the input processes are directly affected by the deviation rather than the average of the controller delay. Second, we show the feasibility of using open source solutions to build up an automation controller. On a bare PC, we set up Xenomai-patched Linux [14] and IgH EtherCAT stack [15]. We used Beremiz [16] to generate automation workload, and we extended its communication interfaces so that it supports EtherCAT network. The experimental controller could successfully operate tens of drives in position or velocity mode with a cycle time of 0.5 ms. The largest actuation deviation among the drives was analyzed to be around 30 μs and 0.1 μs, respectively, in the frame-driven and clock-driven synchronization schemes.

The remainder of this paper is organized as follows. In Section 2, we review the background for EtherCAT-based synchronized processes. In Section 3, we present our end-to-end delay analysis and in Section 4 we describe the evaluation results. Section 5 concludes the paper.

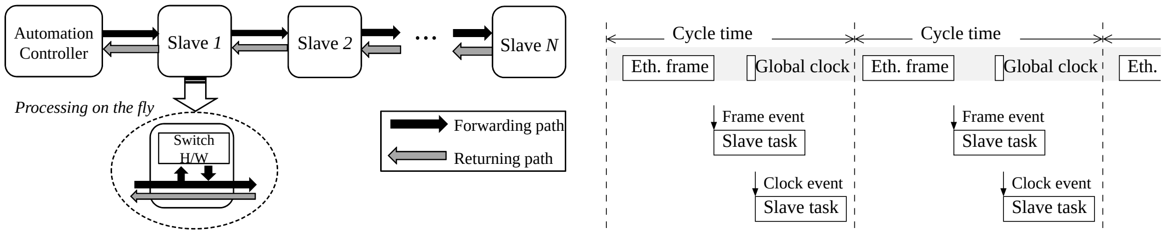
2 Background

2.1 EtherCAT and Synchronized Processes

Among the emerging real-time Ethernet profiles [1], we chose EtherCAT for our analysis because it has desirable features for realizing highly synchronized operations.

First, EtherCAT supports high-speed deterministic communication. Figure 1(a) shows a typical configuration of EtherCAT network, which is in the form of a line topology. Along the forwarding path, every slave in an EtherCAT network relays message frames between the input and output ports *on-the-fly* using switch hardware. When the message frame is relayed by each slave, the output or input process data in the frame is written to, or updated from, the corresponding part of the buffer in the slave. Once an EtherCAT frame arrives at the end of the network, it returns to the controller. EtherCAT allows only peer-to-peer connections between any two consecutive slaves, thereby eliminating the possibility of indeterminism arising from multi-party simultaneous access to the medium. The design of a synchronized process can be greatly benefited from the almost deterministic message delivery time.

Second, EtherCAT provides efficient clock synchronization, which is known as the Distributed Clock (DC) mechanism [8]. Basically, the DC-enabled slave closest to the EtherCAT controller



(a) EtherCAT system in a line topology

(b) Cyclic execution of slave tasks using frame events vs. clock events

Figure 1: EtherCAT-based control system.

acts as the timing reference for the entire EtherCAT network. In the setup phase, the controller collects local clock values from each DC-enabled slave. The controller calculates the offsets between the reference and local clock values, and then it writes the offsets to the slaves, with which they can compute the global time based on their local clocks. After this is done, the controller periodically distributes the value of the reference clock to all the slaves for the compensation of clock skew that is caused by local clock drift. The DC mechanism is rather simple, but it can accurately synchronize the clocks of distributed slaves with a skew of usually less than 1 μs.

EtherCAT provides two types of synchronized events to trigger a local task on each slave, i.e., the frame and clock events (see Fig. 1(b)). The cycle time in this paper is defined as the period with which the controller repeats the automation process. For each process cycle, a slave task can be designed to start its execution immediately after the arrival of an EtherCAT frame that is cyclically generated by the controller. However, such frame-driven cycle may have a relatively large deviation, e.g., a few to dozens of microseconds, due to the variable processing delays on the controller and the intermediate slave devices. Alternatively, a slave task can be invoked by the autonomous interrupt that is generated synchronous to the global clock, in which case the deviation can be reduced by up to a few nanoseconds.

2.2 IEC 61131-3 and Open Source Automation

Currently, the control logic of many industrial automation processes is usually implemented using PLC (Programmable Logic Controller). Owing to the proliferation of PLCs in various industries, it is becoming very important for programmers and field engineers with different domain backgrounds and skills to easily handle PLC-based systems. However, the lack of a consistent approach to PLC programming makes it difficult to integrate devices from different vendors to build a large and complex automation system. In order to solve this problem, the International Electrotechnical Commission (IEC) introduced IEC 61131-3, a standard for programming industrial PLC systems [17].

As IEC 61131-3 is gaining worldwide acceptance by the industry, there have been efforts in the open source community to provide an IEC 61131-3 compliant Integrated Development Environment (IDE). Beremiz [16] is a representative IDE, which is used and distributed freely under the GNU license. Beremiz has three major components: PLCOpen editor, MatIEC back-end compiler, and plugin extensions. The PLCOpen editor lets users write PLC programs using the languages defined in the IEC 61131-3 standard. The MatIEC compiler translates the textual form of automation programs into corresponding ANSI C codes.

The most attractive feature of Beremiz is the *plugin* extension. The plugin structure allows adding new functions to the IDE simply with the implementation of the corresponding class definitions. Because of its extensibility and open source policy, Beremiz has been utilized in many studies of automation systems [18–20]. For the same reason, our prototype controller employed Beremiz, which was extended to include the EtherCAT plugin.

2.3 Related Work

There have been a few studies on the evaluation of EtherCAT networks. Early works [5, 6] formulated the end-to-end delay of the EtherCAT network and analyzed the achievable minimum cycle time according to the slave number and packet sizes. On the basis of a theoretical performance model, Seno *et al.* presented the simulation model and the performance results in terms of two important performance indicators, i.e., minimum cycle time and jitter [7, 21]. Robert *et al.* also presented an in-depth analysis of minimum cycle time for Ethernet-based real-time protocols [22]. Although the previous studies provided the analysis models for analyzing important performance indicators, they did not address the device-level delay factors, only dealing with the network transmission times.

Recent studies have started to address the performance of automation controller or slave devices for networked control systems. Cereia *et al.* evaluated the performance of a Linux-based EtherCAT controller in terms of the cycle accuracy of periodic control tasks [13]. Kim *et al.* presented a statistical delay analysis of EtherCAT motor drives [4]. These studies, however, have limitations because they either lack any consideration of the actual workload [13] or ignore the controller latency [4]. Precise clock synchronization among devices is also crucial for coordinated actuation and sensing. A recent study by Cena *et al.* evaluated the performance of the DC mechanism using extensive measurements [8, 23].

3 EtherCAT Control Processes and Delay Analysis

In this section, we describe the EtherCAT control schemes for synchronized processes [24], and formulate their end-to-end delays. Aiming for an in-depth analysis, we explain the design of an open source EtherCAT controller, and present the delay model for the in-controller processing.

3.1 End-to-end Delay Model of Synchronized Processes

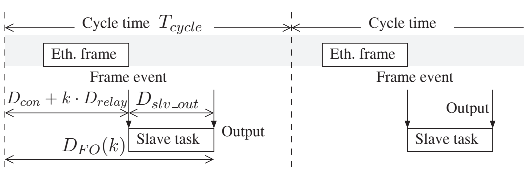
Although EtherCAT allows other topologies such as star and tree, we assume the line topology. It is because the line topology or daisy chain is the most preferred in industries, and our analysis can be easily extended to other types of topologies if a proper model for the frame delivery is given.

Figure 2(a) shows the frame-driven output (FO) scheme that uses the frame event only. We define the end-to-end delay for an output process as the time taken from the start of a process cycle to the corresponding output. Let $D_{FO}(k)$ denote the end-to-end delay on slave k ($1 \leq k \leq N$) in the FO scheme where k means that the slave is k-th nearest to the controller. Table 1 summarizes the notations used in this paper. Assuming homogeneous slaves, we can express $D_{FO}(k)$ as

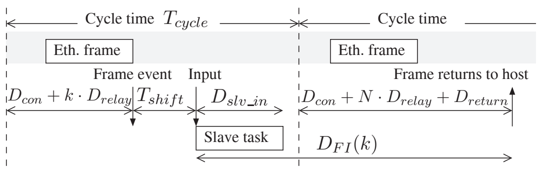
$$D_{FO}(k) = D_{con} + k \cdot D_{relay} + D_{slv\_out}, \tag{1}$$

where D_{con}, D_{relay}, and D_{slv_out} refer to the respective delays for in-controller processing, per-slave message relay, and slave-local output handling. The D_{slv_out} is defined as the time taken to compute the output signal, which is usually a constant value given by the slave device vendor. Note that D_{relay} includes the propagation time on the link between consecutive nodes (controller or slave) as well as the link-level handling time to relay frames.

In the frame-driven input (FI) scheme shown in Fig. 2(b), the frame event triggers the slave task that latches an input data and copies it to the memory in the EtherCAT switch hardware.

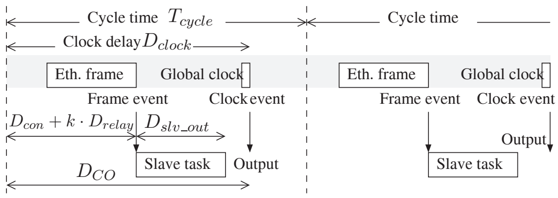


(a) Output process

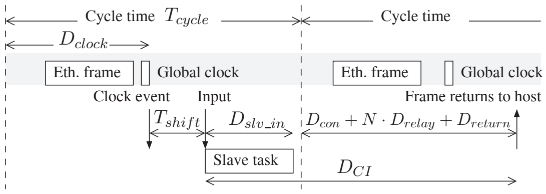


(b) Input process

Figure 2: Frame-driven synchronized processes.



(a) Output process



(b) Input process

Figure 3: Clock-driven synchronized processes.

The slave-local input delay, D_{slv_in} specifies the time required for the latch and copy operation. The input data is conveyed ultimately to the automation controller by the subsequent EtherCAT frame. The slave may allow the controller to adjust the *shift-time*, T_{shift}, by which it delays the input latch to improve the freshness of the sensed data. In the input process, we define the end-to-end delay as the time interval between the time when the input is latched and the time when the frame that contains the input data returns to the controller. The end-to-end delay in the FI configuration is then computed as

$$D_{FI}(k) \quad = \quad T_{cycle} - (D_{con} + k \cdot D_{relay} + T_{shift}) + D_{con} + N \cdot D_{relay} + D_{return}, \tag{2}$$

where T_{cycle} and D_{return} refer to the process cycle time and frame return delay, respectively. In order to ensure that the input data is ready by the arrival of the next frame, it is required that

$$\forall k, D_{con}^{max} + k \cdot D_{relay}^{max} + T_{shift} + D_{slv\_in}^{max} \leq T_{cycle} + D_{con}^{min} + k \cdot D_{relay}^{min}. \tag{3}$$

Therefore, T_{shift} should be determined such that

$$T_{shift} \quad \leq \quad T_{cycle} - (D_{con}^{max} - D_{con}^{min}) - N \cdot (D_{relay}^{max} - D_{relay}^{min}) - D_{slv\_in}^{max}. \tag{4}$$

Note that, hereinafter, X^{avg}, X^{min}, and X^{max} represent the average, minimum, and maximum value of variable X, respectively (e.g., D_{con}^{max} means the maximum of D_{con}).

In order to achieve a higher degree of synchronization, the clock events should be used. As shown in Fig. 3(a), the clock-driven output (CO) scheme uses the clock event to synchronously actuate the output after the computation has been completed. If we denote the delay of the global clock event by D_{clock}, the end-to-end delay in the CO scheme becomes

$$D_{CO}(k) = D_{CO} = D_{clock}, \tag{5}$$

where D_{clock} requires that

$$D_{clock}^{min} \geq D_{con}^{max} + N \cdot D_{relay}^{max} + D_{slv\_out}^{max}. \tag{6}$$

Table 1: Notations.

Notation	Description
N	Number of slaves.
$D_{FO}(k)$	End-to-end delay at slave k ($1 \leq k \leq N$) in the frame-driven output scheme.
$D_{FI}(k)$	End-to-end delay at slave k in the frame-driven input scheme.
D_{CO}	End-to-end delay in the clock-driven output scheme.
D_{CI}	End-to-end delay in the clock-driven input scheme.
D_{con}	In-controller processing time. The delay from the start of a process cycle to the release of EtherCAT frames.
D_{relay}	Per-node message relay time. It includes the time for frame relay by a slave and inter-node frame propagation.
D_{slv_out}	Slave output delay. The time taken to compute the output signal.
D_{slv_in}	Slave input delay. The time required for latching the input data and making it ready to be transferred.
D_{return}	Frame return delay. The time taken for a frame to return from the N-th slave to the controller.
D_{clock}	Global clock delay, the time interval from the start of a process cycle to the generation of the clock event.
D_{con_sw}	In-controller processing time to packetize the control data.
D_{con_dma}	DMA delay to copy data from the EtherCAT driver to the FIFO in the network interface.
T_{cycle}	Process cycle time - constant.
T_{shift}	Shift time by which the slave delays the input latch in order to improve the data freshness - constant.
J_{con_task}	Release jitter of control task.
X^{avg},X^{min}, X^{max}	Average, minimum, and maximum value of variable X: e.g., D_{con}^{max} means the maximum of D_{con}.

The clock-driven input (CI) configuration in Fig. 3(b) uses the clock event to synchronize the input latch. The end-to-end delay is expressed as

$$D_{CI} \quad = \quad T_{cycle} - (D_{clock} + T_{shift}) + D_{con} + N \cdot D_{relay} + D_{return}, \tag{7}$$

where D_{clock} and T_{shift} should be set such that

$$D_{clock}^{max} + T_{shift} + D_{slv\_in}^{max} \leq T_{cycle} + D_{con}^{min} + D_{relay}^{min}. \tag{8}$$

Note that the end-to-end delays in the clock-driven processes are independent of the slave position in the network.

3.2 Controller Delay Analysis

For our study, we constructed an EtherCAT controller using open source software. Our controller design uses Xenomai-patched Linux, IgH EtherCAT protocol stack, and Beremiz as the key software components. This facilitates a highly synchronized control process, whereby the Xenomai kernel significantly reduces the deviation of controller delay, while EtherCAT enhances the predictability of the message and clock events through its almost deterministic message relaying and precisely synchronized clock.

In Beremiz, we implemented the EtherCAT plugin and integrated the EtherCAT stack via the plugin interface. The plugin is mainly composed of two types of classes, each representing the profiles of the controller and slave devices. We also implemented a C wrapper API to the EtherCAT stack. The class definitions together with the EtherCAT API are used by the IEC compiler for the generation of runtime codes. When a new program has been configured to use EtherCAT, the plugin support module imports information on the slave profile and the EtherCAT API from the plugin definition.

The constructed build procedure facilitates the development of EtherCAT automation programs. During the build procedure, illustrated in Fig. 4, PLC codes are translated into C codes

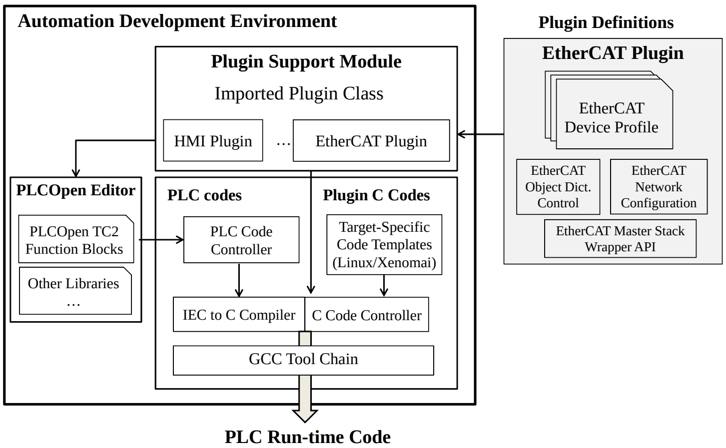


Figure 4: Build procedure for EtherCAT automation programs.

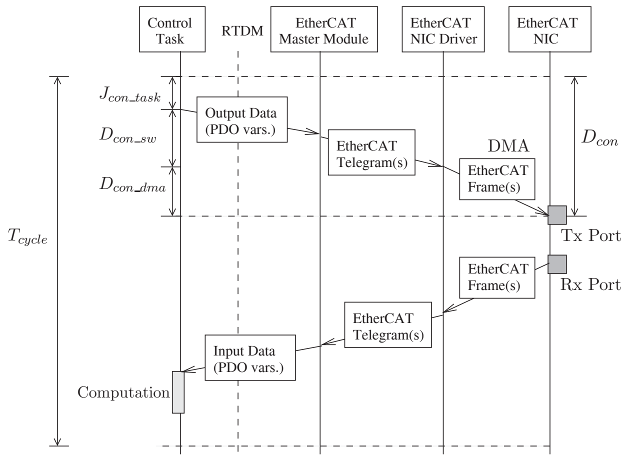


Figure 5: Execution sequence of the control task.

by the IEC compiler, and after being compiled and linked with target-specific stub modules, it is finally converted into an executable runtime. The stubs have interfaces to the time and task management functions of the target operating system, i.e., Xenomai Linux in our case. This build procedure may seem to be complicated, but most of the build steps are performed automatically and thus transparently to users.

In order to model the in-controller delay, we investigate the behavior of the automation runtime. Figure 5 shows the sequence diagram of the control task that executes the automation process. An automation program is typically realized as a periodic control task that repeats reading and writing variables in the slaves. The input and output variables of the task are mapped from the PDOs (Process Data Objects) that are defined in the slave profiles. For each process cycle, the task executes a sequence of transmission of output data, reception of input data, and computation. At the beginning of the cycle, the control task writes output data into the EtherCAT kernel module via the Xenomai interface for real-time I/O, called RTDM (Real-Time Driver Model). In turn, the EtherCAT module packetizes the control data into EtherCAT telegrams and copies them to the buffer in the EtherCAT NIC (Network Interface Card) driver. Finally, the NIC driver encapsulates the telegrams into Ethernet frames and copies them to the transmission FIFO in the NIC.

From the execution flow for frame transmission, we can define the in-controller processing delay D_{con} as the sum of J_{con_task}, D_{con_sw}, and D_{con_dma}. The J_{con_task}, D_{con_sw}, and D_{con_dma}, respectively, refer to the release jitter of the control task, the time taken for copying the data from the user space to the device driver buffer, and the time for sending out EtherCAT frames through the NIC via DMA (Direct Memory Access). Note that D_{con_sw} includes the packetization and queuing overheads as well.

4 Performance Results

For the experiments, we set up an EtherCAT automation system with our open source controller and collected the time taken for the components of the end-to-end delay. Based on the measurements, we discuss the characteristics of the EtherCAT process delays.

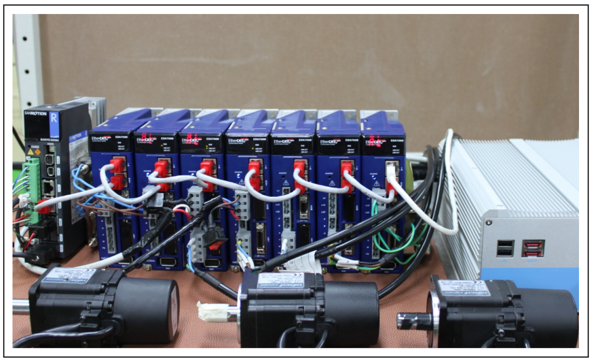


Figure 6: Experimental automation system.

Item	Description
Controller	
CPU	Intel Core 2 Duo E4500 running at 2.2 GHz (1 Core disabled)
Memory	3 GB DDR2 SDRAM
OS	Linux 2.6.37 with Xenomai 2.6.0
Network	Realtek RTL 8139D 100 Mbps Ethernet
Auto. S/W	Beremiz 1.0.3 with IgH EtherCAT Master 1.5.0
Slaves	
Product	Sanyo Denki AC Servo Drive
PDO	64 bytes for each slave (RxPDO 34 bytes, TxPDO 30 bytes)

Table 2: Specification of automation system.

4.1 Measurement of Delay Components

Table 2 summarizes the specification of our experiment system. The controller was connected to a group of commercial EtherCAT servo drives [25], each of which was configured to use an identical PDO set for communication (See Fig. 6). This set was 64 bytes in size and included all the necessary PDOs to command a drive in cyclic synchronous position or velocity mode [26].

Table 3: Measurement results for the controller delay.

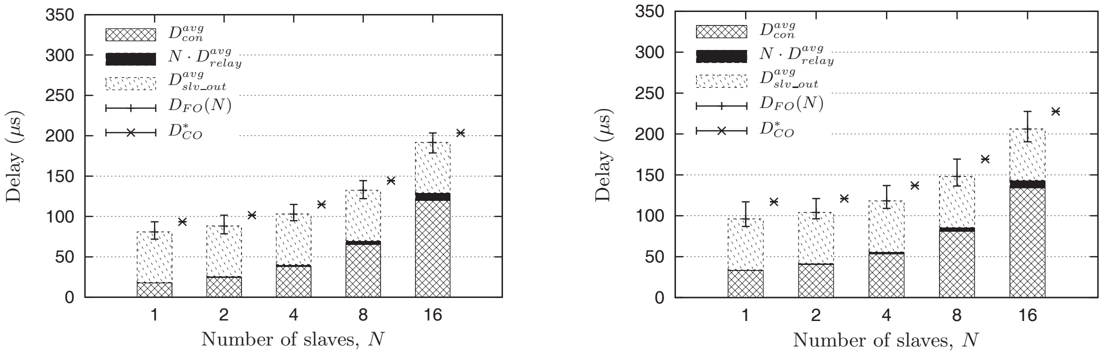
T_{cycle}	4 ms					2 ms					1 ms					0.5 ms				
N	1	2	4	8	16	1	2	4	8	16	1	2	4	8	16	1	2	4	8	16
D_{con} (μs)																				
avg	33.1	40.4	53.3	80.9	134.1	24.2	32.0	45.3	72.3	127.0	20.1	27.1	40.2	67.5	121.9	17.8	24.4	38.0	65.3	119.8
st.d	0.7	0.7	0.8	1.0	1.3	0.6	0.6	0.7	0.9	1.0	0.8	1.1	1.1	1.1	1.2	0.6	0.9	0.9	0.9	0.8
min	23.9	32.8	44.2	69.3	118.7	15.2	15.5	36.0	59.0	114.5	11.0	12.8	30.5	56.5	109.5	8.8	15.0	30.0	55.0	107.1
max	53.8	57.3	71.9	101.9	155.3	41.5	50.0	62.3	89.6	145.3	32.8	43.7	55.2	82.1	135.6	30.3	37.8	49.9	77.0	131.2
Low 95%	33.6	40.8	53.8	81.5	134.7	24.4	32.2	45.5	72.7	127.5	20.9	28.3	41.3	68.7	123.2	18.9	26.1	39.6	66.9	120.7
Low 99%	36.6	43.5	56.8	85.1	140.7	26.3	34.1	47.3	76.0	132.8	21.3	28.9	42.0	70.2	125.0	19.1	26.2	39.7	67.2	121.8
J_{con_task} (μs)																				
avg	16.4	16.5	16.1	16.4	14.9	7.5	8.1	8.0	7.7	8.0	3.9	4.0	3.8	3.8	3.8	2.0	1.9	2.0	2.0	2.0
st.d	0.5	0.6	0.6	0.8	0.9	0.4	0.5	0.5	0.6	0.7	0.3	0.4	0.4	0.5	0.6	0.3	0.3	0.3	0.5	0.6
min	6.6	6.7	6.7	2.7	-1.0	-3.2	-8.9	-3.2	-6.5	-6.0	-5.8	-9.9	-6.6	-10.1	-9.3	-8.7	-8.7	-8.2	-9.6	-10.5
max	26.1	27.7	28.4	28.8	31.0	17.8	22.2	21.1	19.5	21.5	13.0	17.5	15.1	17.3	16.5	12.5	12.6	13.2	13.0	13.1
D_{con_sw} (μs)																				
avg	3.2	3.3	3.3	3.5	4.2	3.1	3.3	3.3	3.5	4.0	3.0	3.2	3.3	3.4	3.9	3.0	3.1	3.2	3.3	3.6
st.d	0.3	0.3	0.4	0.5	0.8	0.3	0.3	0.3	0.5	0.8	0.2	0.2	0.3	0.4	0.4	0.2	0.2	0.2	0.3	0.3
min	3.0	3.2	3.2	3.3	3.7	3.0	3.1	3.1	3.2	3.7	2.9	3.0	3.1	3.2	3.6	2.8	2.9	3.0	3.1	3.5
max	14.2	11.5	12.0	18.0	17.4	11.3	11.2	13.5	16.6	17.0	10.0	12.8	13.0	14.8	16.2	12.6	12.7	11.6	12.9	13.3
D_{con_dma} (μs)																				
avg	13.5	20.6	33.9	61.0	115.0	13.5	20.6	34.0	61.0	115.0	13.1	20.0	33.1	60.3	114.2	12.9	19.5	32.9	60.0	114.1
st.d	0.1	0.1	0.1	0.3	0.3	0.1	0.1	0.1	0.3	0.4	0.5	0.8	0.8	0.7	0.8	0.4	0.7	0.7	0.6	0.5
min	13.2	14.9	30.1	57.2	110.8	10.9	17.2	30.3	56.8	110.4	7.0	16.3	28.0	56.0	109.6	9.3	15.7	29.2	55.9	109.6
max	18.1	25.6	37.5	66.5	121.4	17.3	24.5	37.6	67.7	120.7	20.2	27.3	40.4	68.0	122.6	12.6	25.8	39.6	67.8	121.4

The in-controller processing time, D_{con} was measured as the difference of TSC (Time Stamp Counter) values logged at the time instants when the control task was activated and when the EtherCAT frames had been transmitted. The time instant of frame transmission could be determined from the TX_OK interrupt generated by the NIC [27]. It should be noted that D_{con} includes the cycle time deviation, which corresponds to the difference of the measured time interval between two consecutive task activations from the intended cycle time. Table 3 summarizes the results. It shows the measured delays as we varied the cycle time, T_{cycle} and the number of slaves, N. The measurement was performed for 30 minutes for each T_{cycle}, of which collected data size amounts to 3,600,000 samples with T_{cycle}=0.5 ms, for instance.

The D_{relay} was measured using a Tektronix DPO3012 oscilloscope with a time resolution of 0.01 μs. We connected the probes to the frame interrupt pins of the EtherCAT switch hardware in two adjacent slaves, and we measured the time difference of their trigger events for the same EtherCAT frame. The measured D_{relay} turned out to be highly deterministic, having negligible deviation: The average of D_{relay} was 0.59 μs, and the maximum and minimum were 0.61 μs and 0.57 μs, respectively. We observed that T_{cycle} and N did not make any difference on D_{relay}. Since we can assume that the backward frame transmission exhibits similar timing behavior as the forward case, without loss of accuracy, we use D_{relay} for D_{return} as well. The D_{return} is hence calculated as $D_{return} = N \cdot D_{relay}$. The jitter of the global clock was measured by reading the *system time difference* register that is available in the EtherCAT switch hardware. This register maintains the difference between the local clock and the reference clock in a nanosecond resolution. The average, minimum, and maximum of the jitter were measured as -1 ns, -12 ns, and 7 ns, respectively. For D_{slv_in} and D_{slv_out}, we used the constant values provided by the manufacturer, which were 415 μs and 62.5 μs, respectively [25].

4.2 End-to-end Process Delay

Using the developed delay models together with the measurement results in the previous sections, we evaluate the end-to-end delays. Figure 7 shows the end-to-end output delays at slave N that experiences the largest delays. The graphs show the average, minimum, and maximum of the output delays along with their major delay components. It can be seen that the average



(a) T_{cycle}: 0.5 ms (b) T_{cycle}: 4 ms

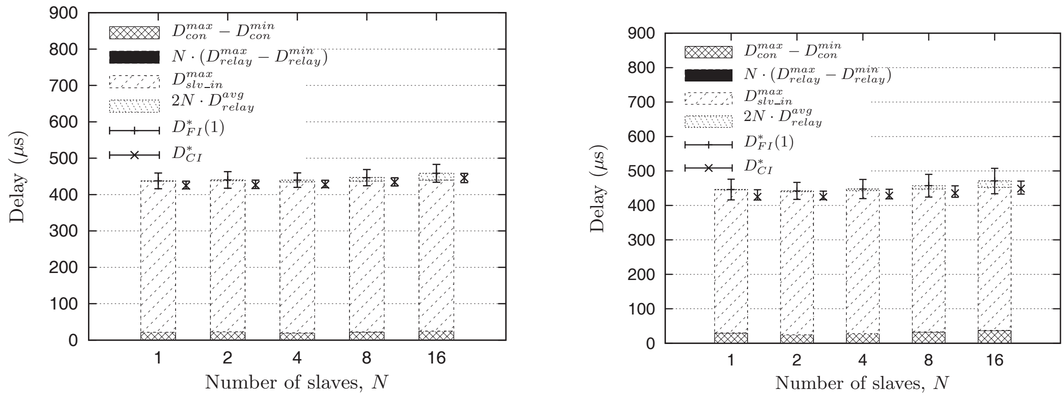
Figure 7: End-to-end output delays at slave N.

output delays increase linearly with the number of slaves. It is mainly due to the highly increased D_{con}. In contrast, the messaging delay, i.e., $N \cdot D_{relay}$ has a much lower effect on the end-to-end delay, although it grows slightly as N increases. As shown in Table 3, for a large N, the in-controller DMA time dominates D_{con} and its linearly increasing feature directly affected the overall end-to-end delays. In the figure, D_{CO}^{*} represents the minimum feasible delay in the clock-driven output scheme under the requirement of Eq. (6). We can observe that D_{CO}^{*} is affected mainly by the maximum of the delay components, i.e., D_{con}, $N \cdot D_{relay}$, and D_{slv_out}, out of which D_{con} again is the main cause of the increase in D_{CO}^{*} for growing N.

It can be seen that we have longer output delays for a larger cycle time. This is because D_{con} tends to increase as T_{cycle} gets larger. We also see that D_{con} has a larger span with a longer cycle. With D_{FO}, this resulted in a noticeably larger deviation while, with D_{CO}^{*}, it was the main reason of an increased average. This is shown clearly in Fig. 7(b) when compared with Fig. 7(a). It should be noted that D_{CO}^{*} has a negligible deviation, being less than 0.1 μs throughout the results, whereas the deviation of D_{FO} is far larger, reaching up to 30 μs and 50 μs with the cycle time of 0.5 ms and 4.0 ms, respectively. This reconfirms the strength of the clock-driven scheme, which in our platform enabled a highly synchronized actuation among servo drives, having only a few nanoseconds of standard deviation.

Based on the results, it could be estimated that the maximum N that is possible with the experimental system for a cycle time of 0.5 ms would be around 32. It is because $D_{FO}(N)$ is calculated to be 330 μs for N of 32, which leaves little time for the slave to prepare for the next cycle. This is confirmed by our experimental experience, where we failed to maintain the system in a stable state with 32 slaves. On the other hand, the experiment results with the prototype controller indicate that the open source software can be a viable automation solution. The controller could successfully operate tens of drives in position or velocity mode with a cycle time of 0.5 ms.

Figure 8 shows the end-to-end input delays on the first slave that has inevitably the largest delay among the slaves. The D_{FI}^{*}, the minimum possible input delay in the frame-driven scheme is obtained when we maximize T_{shift} using Eq. (4). Thus, from Eq. (2) we can determine the



(a) T_{cycle}: 0.5 ms

(b) T_{cycle}: 4 ms

Figure 8: End-to-end input delays at slave 1.

respective average, minimum, and maximum of $D_{FI}^{*}(1)$ as

$$
\begin{aligned}
D_{FI}^{*\,avg}(1) &= (D_{con}^{max} - D_{con}^{min}) + N \cdot (D_{relay}^{max} - D_{relay}^{min}) + D_{slv\_in}^{max} + (N-1) \cdot D_{relay}^{avg} + D_{return}^{avg} \\
&\simeq (D_{con}^{max} - D_{con}^{min}) + D_{slv\_in}^{max} + 2N \cdot D_{relay}^{avg}, \\
D_{FI}^{*\,min}(1) &= (N-1) \cdot D_{relay}^{max} + D_{slv\_in}^{max} + D_{return}^{min} \\
&\simeq D_{slv\_in}^{max} + 2N \cdot D_{relay}^{avg}, \\
D_{FI}^{*\,max}(1) &= 2 \cdot (D_{con}^{max} - D_{con}^{min}) + 2N \cdot D_{relay}^{max} - (N+1) \cdot D_{relay}^{min} + D_{slv\_in}^{max} + D_{return}^{max} \\
&\simeq 2 \cdot (D_{con}^{max} - D_{con}^{min}) + D_{slv\_in}^{max} + 2N \cdot D_{relay}^{avg}.
\end{aligned} \tag{9}
$$

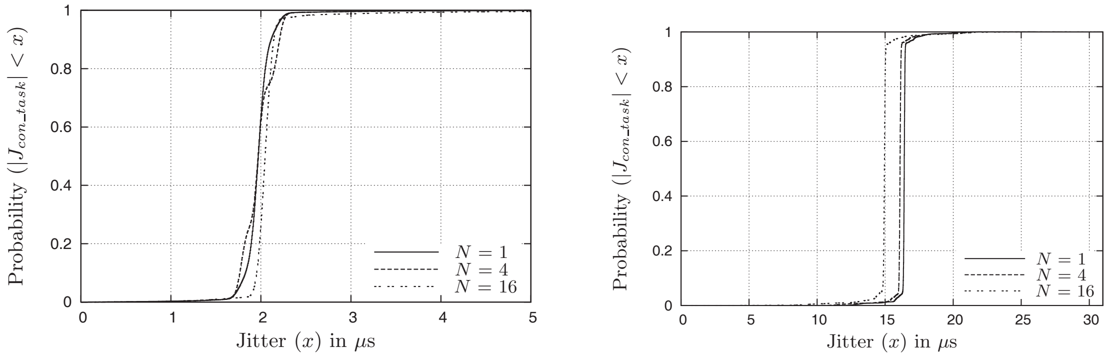
In the equations, we used the approximation of $D_{return} = N \cdot D_{relay}$ and $D_{relay}^{avg} = D_{relay}^{min} = D_{relay}^{max}$. The D_{CI}^{*} can be derived as follows. The jitter of the global clock is negligible, and hence we can write $D_{clock} + T_{shift} = D_{clock}^{max} + T_{shift}$, the maximum of which is given by Eq. (8). Thus, based on Eq. (7), D_{CI}^{*} becomes

$$
\begin{aligned}
D_{CI}^{*} &= D_{con} + N \cdot D_{relay} + D_{return} - D_{con}^{min} - D_{relay}^{min} + D_{slv\_in}^{max} \\
&\simeq D_{con} - D_{con}^{min} + D_{slv\_in}^{max} + 2N \cdot D_{relay}^{avg}.
\end{aligned} \tag{10}
$$

We see from Fig. 8 that the frame-driven end-to-end input delay is affected by the deviation of D_{con} but not by the average. This contrasts with the output cases where the mean of D_{con} has the biggest impact on the delays. The backward message relay in EtherCAT is highly deterministic and the slave-local input processing time is given as a constant; hence, it can be stated that it is essential to minimize the frame jitter caused by the controller in order to reduce D_{FI}.

It can be seen that D_{CI}^{*} outperforms D_{FI}^{*} in minimizing both the average and deviation of the delays, whereas in the output processes, the clock-driven scheme produced a larger average delay than the frame-driven. The use of clock events makes the delay dependent only on the variance of the frame that carries the input data, thereby disengaging it from the effect of the preceding frame. The frame-driven input, on the other hand, is affected by the variance of the frame twice (see Eq. (2)). In our testbed system, the D_{CI}^{*} had a deviation of -16.0 ∼ 37.9 μs while $D_{FI}^{*}(1)$ had -37.2 ∼ 73.9 μs.

Observe that the input delays are relatively less affected by the number of slaves when compared with the output delays. This is mainly due to the characteristics of D_{con}, whose deviation does not change much with varying N.



(a) T_{cycle}: 0.5 ms (b) T_{cycle}: 4 ms

Figure 9: Release jitter of control task (J_{con_task}).

As shown in the evaluation results for the end-to-end delays, the performance of the EtherCAT-based synchronized processes depends heavily on the delays by the controller. The measurement results in Table 3 show that the controller delay D_{con} tends to grow as we increase T_{cycle}. We see that this mainly comes from the increased J_{con_task}: the J_{con_task} gets larger in line with the increased T_{cycle}, whereas other in-controller delay components, D_{con_sw} and D_{con_dma} are relatively unaffected by T_{cycle}. In Fig. 9, we plot the distributions of $|J_{con\_task}|$, i.e., the absolute value of J_{con_task}. It is shown that 99% of $|J_{con\_task}|$ lies within 3.26 μs and 18.16 μs for T_{cycle} of 0.5 ms and 4 ms, respectively. It is considered that the cache pollution due to other background tasks and kernel activities, such as for debug and monitoring, is enlarging the release jitter of the real-time task.

On the other hand, the observed maximum D_{con} was quite large, ranging from 109.5% to 171.5% of the average depending on the cycle time. As can be seen in Table 3, however, the low 95% and 99% range values are very close to the average, being far lower than the maximum. It can be said that the frame release jitter of our system is statistically kept under a tolerable range although it remains to be improved in case of relatively large control cycle.

5 Conclusion and Future Work

In this paper, we have addressed the performance analysis of EtherCAT control systems in term of the end-to-end delay. On the basis of two types of cyclic events, i.e., the frame and clock events, we explain the control schemes that enable synchronized input and output in EtherCAT, and we formulate their end-to-end delays with the time for in-controller processing, message delivery, and slave-local handling. Using the developed delay model, we discuss the characteristics of EtherCAT synchronized processes for varying number of slaves and process cycle time. For an in-depth and practical evaluation, the analyses have been conducted using a real EtherCAT controller that was constructed from open source software.

The experiment results show that the controller has a crucial effect on the precision of the networked control processes. We observed that the average end-to-end delay of output processes is mainly increased by the average controller delay while the input processes are directly affected by the deviation of the controller delay. It is shown that, with a larger number of slaves, the in-controller DMA time primarily contributes to the increased average controller delay, whereas, for a longer cycle time, the task release jitter tends to get larger and increases the controller delay. It should be also noted that the use of a global clock event significantly reduces the deviation of the end-to-end delay for both input and output processes.

Aiming for a holistic delay analysis, we have dealt with the controller delay, mainly discussing its average and deviation. In our future research, we will extend our analysis by studying the probability model. Together with our already developed delay model for motor drives [4], it will enable a delay-guaranteed synchronized control system, which is planned to be used for our development of a sub-micron-level motion stage.

Acknowledgement

This research was supported in part by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (No. 2013R1A1A2004984) and in part by the National Research Foundation of Korea(NRF) grant funded by the Korea government (No. 2012015266).

Bibliography

[1] M. Felser, Real Time Ethernet: Standardization and implementations, *Proc. of Int. Symposium on Industrial Electronics*, 3766-3771, Jul. 2010.

[2] F. Benzi, G. S. Buja, and M. Felser, Communication architectures for electrical drives, *IEEE Trans. Industrial Informatics*, 1(1): 47-53, Feb. 2005.

[3] *EtherCAT Technology Group, EtherCAT: Ethernet for control automation technology.* [Online]. Available: http://www.ethercat.org.

[4] K. Kim, M. Sung, and H.-W. Jin, Design and Implementation of a Delay-Guaranteed Motor Drive for Precision Motion Control, *IEEE Trans. Industrial Informatics*, 8(2): 351-365, May 2012.

[5] J. Jasperneite, M. Schumacher, and K. Weber, Limits of increasing the performance of Industrial Ethernet protocols, *Proc. 12th IEEE Int. Conf. on Emerging Technologies and Factory Automation*, 17-24, Sep. 2007.

[6] G. Prytz, A performance analysis of EtherCAT and PROFINET IRT, *Proc. 13th IEEE Int. Conf.on Emerging Technologies and Factory Automation*, 408-415, Sep. 2008.

[7] L. Seno, S. Vitturi, and C. Zunino, Real Time Ethernet networks evaluation using performance indicators, *Proc. 14th IEEE Int. Conf. on Emerging Technologies and Factory Automation*, 1-8, Sept. 2009.

[8] G. Cena, I. C. Bertolotti, S. Scanzio, A. Valenzano, and C. Zunino, Evaluation of EtherCAT Distributed Clock Performance, *IEEE Trans. Ind. Informatics*, 8(1): 20-29, v Feb. 2012.

[9] R. Ramesh, S. Jyothirmai, and K. Lavanya, Intelligent automation of design and manufacturing in machine tools using an open architecture motion controller, *Journal of Manufacturing Systems*, 32(1): 248-259, Jan. 2013.

[10] J. Schacht, J. Sachtleben, H. Jensen, U. Stutz, and M. Wiese, Piezo-valve controller for the gas inlet system of the fusion experiment Wendelstein 7-X, *Fusion Engineering and Design*, 87(12): 1961-1966, Dec. 2012.

[11] S. G. Robertz, R. Henriksson, K. Nilsson, A. Blomdell, and I. Tarasov, Using Real-Time Java for Industrial Robot Control, *Proc. of 5th Int. Workshop on Java Technologies for Real-Time and Embedded Systems (JRTES)*, 104-110, Sep. 2007.

[12] P. Ferrari, A. Flammini, D. Marioli, and A. Taroni, A Distributed Instrument for Performance Analysis of Real-Time Ethernet Networks, *IEEE Trans. Industrial Informatics*, 4(1): 16-25, vol. 4, Feb. 2008.

[13] M. Cereia, I. C. Bertolotti, and S. Scanzio, Performance of a Real-Time EtherCAT Master Under Linux, *IEEE Trans. Ind. Informatics*, 7(4): 679-687, Nov. 2011.

[14] Xenomai. [Online]. Available: http://www.xenomai.org.

[15] IgH EtherCAT Master for Linux. [Online]. Available: http://www.etherlab.org.

[16] E. Tisserant, L. Bessard, and M. de Sousa, An Open Source IEC 61131-3 Integrated Development Environment, *Proc. 5th IEEE Int. Conf. on Industrial Informatics*, 183-187, Jun. 2007.

[17] *IEC 61131-3: Programmable Controllers - Part 3: Programming Languages 2^{nd} Ed.*, IEC Std., 2005.

[18] C. Gerber, S. Preusse, and H.-M. Hanisch, A Complete Framework for Controller Verification in Manufacturing, *Proc. 15th IEEE Int. Conf. on Emerging Technologies and Factory Automation*, 1-9, Sep. 2010.

[19] O. Ljungkrantz, K. Akesson, M. Fabian, and Y. Chengyin, Formal Specification and Verification of Industrial Control Logic Components, *IEEE Trans. Automation Science and Engineering*, 7(3): 538-548, Jul. 2010.

[20] J. Farines, M. H. De Queiroz, V. G. da Rocha, A.M.M. Carpes, F. Vernadat, and X. Cregut, A model-driven engineering approach to formal verification of PLC programs, *Proc. 16th IEEE Int. Conf. on Emerging Technologies and Factory Automation*, 1-8, Sep. 2011.

[21] L. Seno and C. Zunino, A simulation approach to a Real-Time Ethernet protocol: EtherCAT, *Proc. 13th IEEE Int. Conf. on Emerging Technologies and Factory Automation*, 440-443, Sep. 2008.

[22] J. Robert, J.-P. Georges, E. Rondeau, and T. Divoux, Minimum Cycle Time Analysis of Ethernet-Based Real-Time Protocols, *INT J COMPUT COMMUN*, ISSN 1841-9836, 7(4): 743-757, Nov. 2012.

[23] G. Cena, I. C. Bertolotti, S. Scanzio, A. Valenzano, and C. Zunino, On the accuracy of the distributed clock mechanism in EtherCAT, *Proc. 8th IEEE Int. Workshop on Factory Communication Systems*, 43-52, May 2010.

[24] *ETG.1020: EtherCAT Protocol Enhancements v.1.0.0*, EtherCAT Technology Group Std., Aug. 2011.

[25] *AC Servo Systems with EtherCAT Interface for Rotary/Linear Motor: Instruction Manual*, Sanyo Denki, Inc., Mar. 2011.

[26] *ETG.6010: Implementation Directive for CiA 402 Drive Profile v.1.0.0*, EtherCAT Technology Group Std., Feb. 2012.

[27] *Single-Chip Muilti-Function 10/100Mbps Ethernet Controller with Power Management Datasheet Rev. 1.2*, RealTek, Inc., 2005.

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):622-634, August, 2013.

Using Augmented Reality in Remote Laboratories

H. Vargas, G. Farias, J. Sanchez, S. Dormido, F. Esquembre

Hector Vargas* and Gonzalo Farias
School of Electrical Engineering
Pontificia Universidad Católica de Valparaíso
Avda. Brasil 2147, Valparaíso, Chile
{hector.vargas, gonzalo.farias}@ucv.cl
*Corresponding author

Jose Sanchez and Sebastian Dormido
Department of Computer Science and Automatic Control
Spanish University for Distance Education
c/ Juan del Rosal 28040, Madrid, Spain
{jsanchez, sdormido}@dia.uned.es

Francisco Esquembre
Department of Mathematics
University of Murcia
Campus de Espinardo 30100, Murcia, Spain
fem@um.es

Abstract:
This paper introduces the concept of "augmented reality" as a novel way to enhance visualization in remote laboratories for engineering education. In a typical remote experimentation session, students get access to a remote real plant located at the laboratory to carry out their assignments. Usually, the graphical user interface allows users to watch the equipment by video stream in live. However, in some cases, visual feedback by video stream could be enhanced by means of augmented reality techniques, which mix together in one image, the video stream and computer generated data. Such mixture produces an added value to remote experimentation, increasing the sense of presence and reality, and helping to understand much better the concepts under study. In this work, a Java-based approach to be used in the remote experimentation context for pedagogical purposes is presented. Firstly, a pure Java example is given to readers (including the source code) and then, a more sophisticated example using a Java-based open source tool known as *Easy Java Simulations* is introduced. This latter option takes advantage of a new developed component, called camimage, which is an easy-to-use visual element that allows authors to capture video stream from IP cameras in order to mix real images with computer generated graphics.

Keywords: Augmented reality, Virtual-labs, Remote-labs, Engineering education.

1 Introduction

Augmented reality (AR for short) is a field of computer science which deals with the combination of real world and computer generated data. Virtual objects (computer generated graphics) can be overlaid with real world images in order to make them coexist dynamically in the same space. This basic idea is, nowadays, being widely applied in many areas such as [1] and [2]: TV marketing, mobile phones, video games, medicine, industry, among others [3–5].

In [1], author defines as AR-enabled applications when they fullfil with the following properties:

a) Combines real and virtual objects in a real environment.

b) Runs interactively, and in real time.

c) Registers (aligns) real and virtual objects with each other.

This last feature is the more complex one but, fortunately, it is not so important in the remote experimentation context because of the fact that in most cases, object and camera have static locations. Thus, remote laboratories should take advantage of the use of augmented reality techniques in order to give end users (mainly students) a major feeling of physical presence in the laboratory when working with the didactic equipment (also called plant) from distance.

Although the use of augmented reality in the field of engineering education could be quite convenient, there is no so much work done in this area so far, especially from an authoring point of view. Authors (mainly instructors) can try to combine video stream with computer generated images in their remote laboratories, however this process can require a specialized knowledge and advanced computer programming skills. In this context, the goal of this work is to provide a simple and easy-to-use approach to add augmented reality to remote laboratories.

The approach proposed uses *Easy Java Simulations* (EJS) to create the AR-enabled applications [6]. EJS is a freeware simulation tool developed by one of the authors of this article which has been awarded by the Science Prize for Online Resources in Education (SPORE) in november 2011 [7]. EJS can be used to build the graphical user interface of a remote laboratory. The augmented reality is added by selecting some interactive visual elements provided by EJS in combination with the element **camimage**, which is able to capture video stream from many different IP cameras by means the Java library **webcam.jar**. The development and use of the element *camimage* and the package *webcam.jar* is described in detail in this paper.

The article is organized as follows: Section 2 introduces the AR in the remote experimentation. In Section 3 a Java-based approach to add AR by using the Java library webcam.jar is described. Section 4 shows the use of *Easy Java Simulations* to create AR-enabled applications by means the element camimage. Some examples of remote laboratories are also shown in Section 4. Finally some conclusions and future work are discussed.

2 Augmented Reality in Remote Experimentation

The advances of the information and communication technologies has provided great new opportunities for education [8–10]. Networks, computer graphics and interactivity are just some great examples of them applied to engineering education.

In control education the impact of these technologies is even more significant. Experimentation in traditional laboratories is essential for students, who need to understand the fundamental concepts from both perspectives: theoretical and practical. However, the high costs associated with equipment, space, and maintenance staff, impose certain constraints on resources. A great effort of researching has focused on ways to overcome these limitations. Two of the most important results of such work are virtual and remote laboratories.

Virtual laboratories, which have become common place in the past few years, usually consist of a computer-based model of a real plant. This kind of simulations requires normally high levels of interactivity and visualization, since they are usually used to teach, in a human-friendly way, many key concepts in a particular discipline [11]. See some examples of virtual laboratories in [12] and [13].

The remote operation of real equipment is commonly referred to as remote laboratory and it allows students to manipulate physical plants (see Figure 1), located at the university, from their home computers. This kind of experimentation reduces the time and location constraints of traditional laboratories [14]. See some examples of remote laboratories in [13, 15].

Figure 1: Heat Flow apparatus of Quanser.

Both virtual and remote laboratories provide, thanks to remote access, great opportunities for teachers to support the continuous process required for a life-long learning, as mentioned before.

Although the importance of virtual and remote laboratories for engineering education, the use of these tools as learning objects is not so usual. The main reason could be that the development of interactive applications is a difficult task from a computer programming point of view. Instructors, who are commonly not programming experts, can encounter problems trying to add user interaction or advanced visualization to existing virtual and remote laboratories. This is further complicated by the presence of different computer languages, programming techniques, network protocols, etc.

Despite the aforementioned difficulties, it is possible to find many examples of educational institutions that include virtual and remote laboratories as teaching tools in their current engineering and sciences curricula. Most of these examples offer the possibility of working on either a simulated version of a physical process or a real device located at the universities laboratories. However, such platforms have certain limitations [16] that should also be enhanced to take into account the advantages of augmented reality instead of showing only visual feedback by video stream, specifically:

- These developments are mainly focused on solving the technical issues related to the building of web-based lab solutions and not providing specific software tools designed to meet these goals. In general, they do not take into account the programming issues that hinder control engineering teaching staff when designing and developing virtual and remote laboratories [17–21]. Remote laboratory environments such as [17, 18] and [20] only provides raw video stream as feedback for learners.

- Most of these environments do not consider the social context of interaction and collaboration among students (and between teachers and students) in traditional hands-on laboratories [22, 23]. Such environments use augmented reality with their remote laboratories, but they lack of easy-to-use authoring tools to create new AR-enabled applications.

- Unlike the Cyberlab and eLabs FEUP [24], most of the cases are isolated experiences coming from university engineering departments that offer only a limited set of experiments (only those existing in their own labs). There is any kind of augmented reality added to these laboratories, only video stream is available on them.

This article proposes to take advantage of augmented reality in remote experimentation by mixing video and simulation all in one, in the same way that an AR-enabled application does it. Figure 2 show this idea.

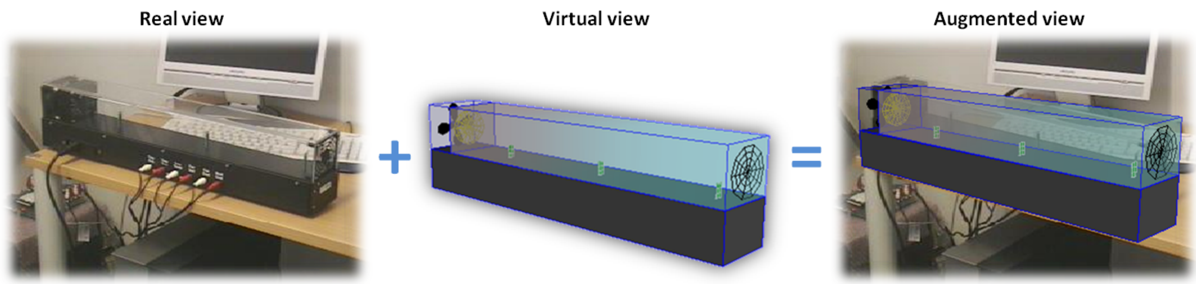


Figure 2: Heat Flow with augmented reality.

Based on previous description, it is possible to realize that, in the remote experimentation context, both real world information (remote laboratories) and computer generated information (virtual laboratories) could be used together.

The mixing between real-world and virtual objects at the same time (in one image) can be used in remote experimentation with pedagogical purposes under the following working schemes:

- *Virtual objects whose behaviour depend on simulated data.* The real-world visualization is overlaid with computer generated graphics based on the dynamics of a mathematical model of the plant.

- *Virtual objects whose behaviour depend on real data.* The real-world images are mixed with computer generated graphics by using real measurements of the remote target plant.

The first scheme can be used to compare the system's response of a mathematical model against the behaviour of a real equipment located at the laboratory. The second approach mixes real-world images into computer generated graphics in order to add extra visual information to the video combination. Both schemes are introduced here in order to be used when creating remote experimentation in engineering education.

In the Sections 3 and 4 two approaches to add augmented reality to remote laboratories are shown. The first approach describes in detail the use of a Java package, called *webcam.jar*. Second approach introduces the EJS visual element *camimage*, which uses internally the library *webcam.jar*. This latter approach is highly recommended to be used by instructors in order to create remote laboratories with augmented reality features.

3 A Java-based approach

In the remote version of a Web-based laboratory, adding a visual feedback module is an essential element in any tele-operated environment since it allows the users to feel and be aware of the consequences of their actions during a remote working session. As a result of this, users are more motivated and confident in the use of the system.

On the other hand, AR usually considers capturing of video stream (real world information) in a local mode whereas in remote experimentation the video stream should be obtained from a remote source.

To fullfil with the requirement above mentioned, two alternatives could be tested. The first one considers the use of an IP camera with a built-in video server dedicated to capturing images and post-publishing them as video stream through its network interface while the second alternative considers using a conventional webcam with serial interface (USB o Firewire) directly connected to a server computer (where the server computer could be the same host that controls the real plant). The publishing of the video stream is not direct in this case since the image data must be acquired via webcam through its serial interface first and then socket techniques should be used to send these data across the network.

3.1 The Java library webcam.jar

In order to read the video stream published by the video server of an IP camera a Java library called *webcam.jar*[1] was developed. This API allows Java programmers to use the classes and methods of the library to get video images from any IP camera connected to the Internet.

The software provides a first level of interface with the camera (see listing 1), allowing to write high level software hiding the low level programming details such as the structure of the received packets, the decoding of the HTTP headers, or the sockets implementation.

Listing 1: Main Java class of webcam.jar.

```
package webcam;

// Main Java Interface of the library webcam.jar
public interface IVideo{

  // Public methods to dialog with a video camera
  connect();
  disconnect();
  boolean isConnected():
  int getAvailableBuffer();
  Image getImage();

}
```

The main class of the API, called *Video.class*, implements the Java interface shown in previous listing. The constructor of this class allows to create an instance of a *Video* object with the following input arguments:

- A *String* indicating the URL where the camera supplies the video stream.
- A *boolean* indicating what the reading format either (MPEG[2] (true) or JPEG[3] (false)).
- An *integer* to add a delay (in milliseconds) in the reading of the images.

Once an object of Video class has been created, a *Java thread* is ready to be started. As mentioned above, *Video* class provides a set of public methods to dialog with the camera. These methods are described below:

`public connect();`

When this method is invoked on a *Video* object previously created, the `run()` method of a *Java thread* is executed. This method opens a socket connection to the remote IP camera and it starts reading of the video stream according to the chosen format.

`public disconnect();`

[1] Accesible from `http://www.profesores.pucv.cl/hvargas/augmented/augmentedreality.html`

[2] The Moving Picture Experts Group, commonly referred to as simply MPEG, is a working group of ISO/IEC charged with the development of video and audio encoding standards.

[3] In computing, JPEG is a commonly used method of compression for photographic images. The degree of compression can be adjusted, allowing a selectable tradeoff between storage size and image quality. JPEG typically achieves 10 to 1 compression ratio with little perceivable loss in the image quality.

This method stops reading the video stream and closes the socket objects created at low level. The `run()` method of the thread is over when this method is executed.

`public boolean isConnected();`

Enquires if the link to the camera is up or down. Commonly, this boolean indicator is used when the `getImage()` is invoked to get the images.

`public int getAvailableBuffer();`

Enquires if there are available images in the input buffer. The `getImage()` method should be invoked if we are in this case.

`public Image getImage();`

The images captured from the IP camera can be obtained from the input buffer by using this method. The method returns an object of the *java.awt.Image* class. Image objects are easily displayed on *Java swing* [4] *components.*

Summarizing, Java programmers can import this library to capture the video stream of any IP camera connected to the network. With regards to the present essay, the library was developed in such away that its inclusion in the development process of a new virtual and remote laboratory can be achieved with a minimum programming effort.

3.2 A simple pure Java example of AR

Listing 2 shows a very naive example but descriptive of the basic idea about augmented reality in the remote experimentation context. In this example, a small virtual object is moved on real world video images coming from a remote IP camera connected to Internet (see Figure 3).

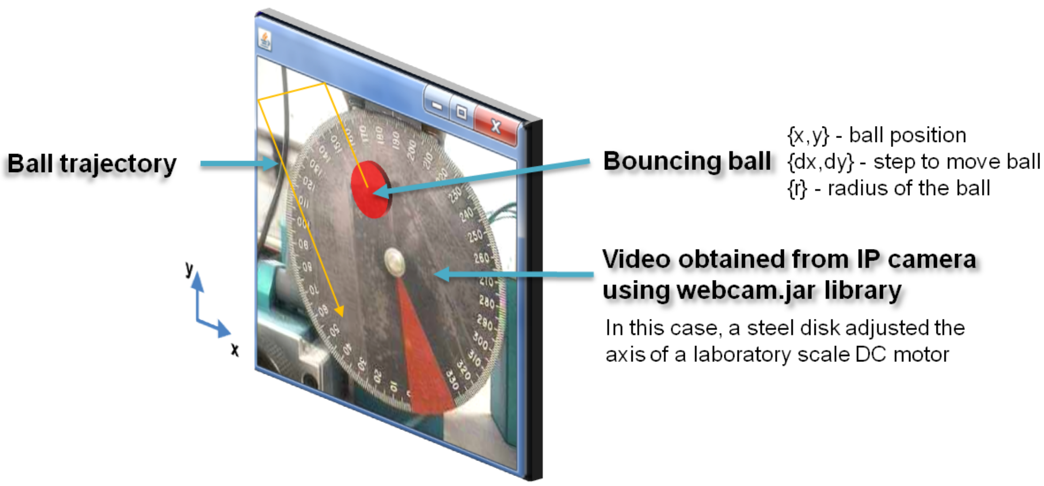


Figure 3: Simple Java-based AR application.

This example illustrates, in a very basic way, how to apply AR techniques in the remote experimentation context. However, as the reader can appreciate, drawing and animating virtual objects in Java is not a trivial task for non-expert programmers. Moreover, in most cases, augmented reality involves the creation of professional animations in which a set of virtual objects compose a more complex animated structure than a simple bouncing ball.

[4]Swing is a widget toolkit for Java. It is part of Sun Microsystems' Java Foundation Classes (JFC) - an API able to provide a graphical user interface (GUI) for Java programs. Swing was developed to provide a more sophisticated set of GUI components than the earlier Abstract Window Toolkit.

Listing 2: A pure Java example of AR.

```java
package arbasic;

import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import webcam.Video;

// A bouncing ball on remote video streaming
public class ARExample extends JFrame implements Runnable {

  int x = 150, y = 50, r = 25;
  int dx = 11, dy = 7;
  Thread animator;
  String URL = "http://IP_address/mjpg/video.cgi";
  Video cam = new Video(URL, true, 20);
  volatile boolean pleaseStop;

  //It implements Runnable interface
  public void run() {
    while (!pleaseStop) {
      animate();
      try {
        Thread.sleep(40);// Wait 40 milliseconds
      }catch (InterruptedException e) {
      } // Ignore interruptions
    }
  }
  // It renders both video images and ball
  public void paint(Graphics g) {
    Graphics2D g2d = (Graphics2D)g;
    g2d.drawImage(video.getImage(), null, 0, 20);
    g2d.setColor(Color.red);
    g2d.fillOval(x-r, y-r, r*2, r*2);
  }
  // It calcules new position of ball and repaints
  public void animate() {
    Rectangle bounds = getBounds();
    if ((x-r+dx < 0) || (x+r+dx > bounds.width))
      dx = -dx;
    if ((y-r+dy < 0) || (y+r+dy > bounds.height))
      dy = -dy;
    x += dx;
    y += dy;
    repaint();
  }
  // It starts animation and video
  public void start() {
    animator = new Thread(this);
    pleaseStop = false;
    animator.start();
    video.connect();
  }
  // Flag to stop the animator thread
  public void stop() {
    pleaseStop = true;
  }
  // Main method of the Java class
  public static void main(String args[]) {
    final ARExample frame = new ARExample();
    frame.start();// Executing run() method
    frame.addWindowListener(new WindowAdapter(){
      public void windowClosing(WindowEvent we){
        frame.stop();
        System.exit(0);
      }
    });
    frame.setSize(350, 263);
    frame.setResizable(false);
    frame.setVisible(true);
  }
}
```

In this context, next section describes a new approach that considers the use of the open source software tool called *Easy Java Simulations* (EJS). EJS helps to create more complex Java programs that exhibit a high degree of interactivity and graphical standards.

4 A novel approach using EJS

As shown in previous section, displaying a simple Java animation on real world images is not a straightforward task for non-expert programmers. In order to cope with this problem, the following subsections show how it is possible to create more sophisticated AR-enabled Java applications by using the open source tool *Easy Java Simulations*.

4.1 What is EJS?

EJS is an open-source, free of charge, authoring tool specialized in the creation of interactive simulations for (mainly, but not only) pedagogic uses. EJS is designed to make it easy for a non programming specialist to implement a scientific model into computer form and to design and build a graphical user interface (GUI) which meets both the visualization and interaction capabilities required for an effective computer-based instruction of a given scientific phenomenon.

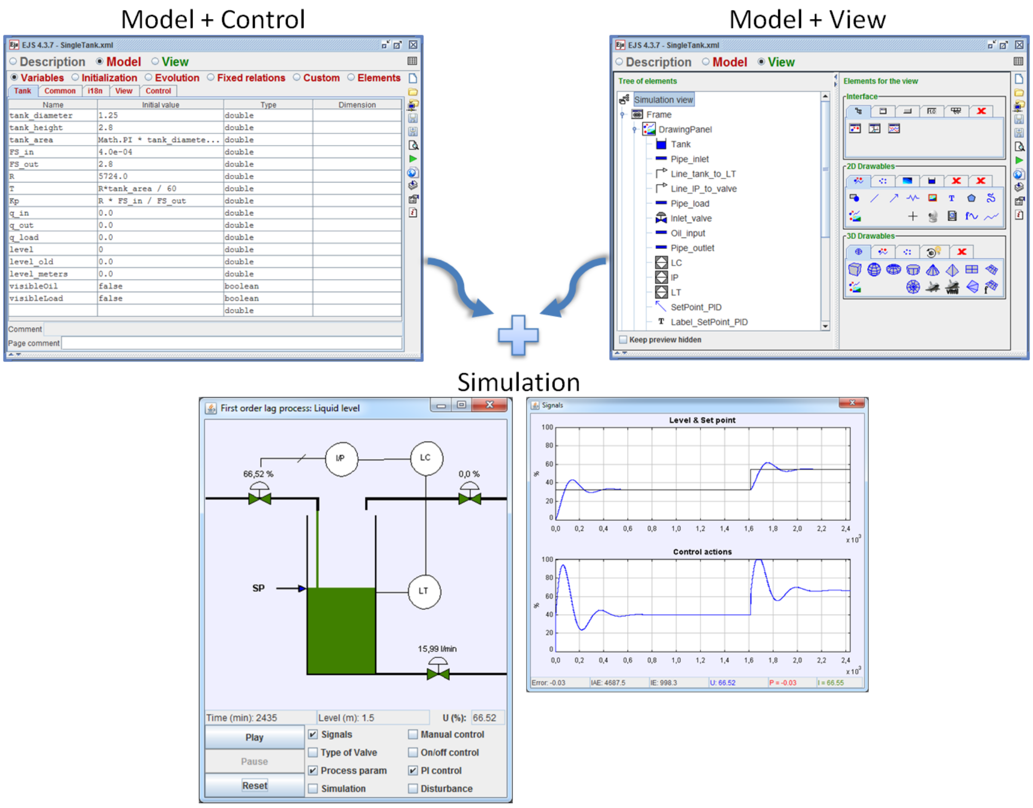


Figure 4: Creation process of an EJS application.

The architecture of EJS derives from the model-view-control (MVC) paradigm, whose philosophy is that interactive simulations must be composed of three parts (see Figure 4) :

- The *model*, which describes the process under study in terms of, 1) variables, which hold the different possible states of the process, and 2) relationships among these variables, expressed by computer algorithms.

- The *control*, which defines certain actions that a user can perform on the simulation.
- The *view*, which shows a graphical representation (either realistic or schematic) of the process states.

EJS makes things even simpler by eliminating the "control" element of the MVC paradigm and fuses one part in the *view* and the other one in the *model*. Thus, applications are created in two steps:

1. Defining the *model* to simulate by using the built-in simulation mechanism of EJS and,
2. Building the *view* showing the model state and answers to the changes made by users.

Figure 4 shows a simple virtual-lab created by EJS for teaching basic control concepts based on the well-known single-tank process. As appreciated, part of the control logic is programmed when defining the *model* and the another when creating the *view*.

EJS provides developers a set of predefined graphical elements to compose the graphical aspect of a simulation. In this context, EJS may accept new graphical elements to make easier the task of composing a *view*. Following this philosophy, the library **webcam.jar** was integrated into EJS as a new *view* element. The following subsection presents this new component and how it can be used by EJS developers.

4.2 Integration of webcam.jar into EJS

In order to simplify the use of the **webcam.jar** library, a new view graphical element of EJS called **camimage** was developed (see Figure 5). The novelty of this approach lies in that the element was implemented based on the characteristics of an existing graphical component of EJS, the **InteractiveImage**.

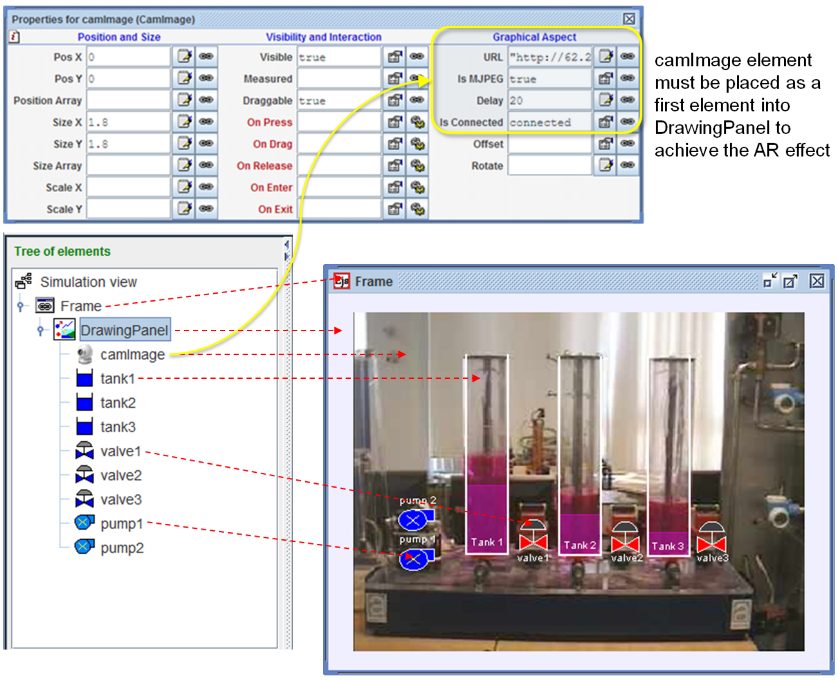


Figure 5: Simple illustration of the *augmented reality* concept for remote labs in EJS.

Thus, the `camimage` element inherits all methods and properties of this last object and, can therefore, be placed on an EJS container with coordinate axis such as the "DrawingPanels" or the "PlottingPanels" in order to render the images from the camera.

In addition to the properties inherited from the InteractiveImage element, four new parameters must be configured. These parameters are related to the IP camera used since a `Video` object (from the `webcam.jar` library) is instantiated when adding this object into an EJS view. The parameter **URL** has to be set with the URL of the IP camera video stream. The option **MJPEG** can be used to indicate whether the image format is MJPEG. The refresh rate of the images can be controlled by **DELAY**. Finally, the option **Connected** enables or finishes the connection with the IP camera.

Applying the augmented reality concept in EJS is now very simple. Figure 5 shows how to compose an EJS view with augmented reality features. This effect is achieved when adding a **camimage** element as a first object of a drawing container of EJS, since the captured images will always appear in the background, behind other virtual graphical elements that may form more advanced Java animations.

The direct application of the augmented reality concept and its benefits in remote experimentation will be illustrated by means of examples in the following subsections.

4.2.1 Example 1: An AR-enabled remote laboratory of a thermal process

Figure 5 showed how to apply AR techniques on a scene 2D using EJS. However, enhancing visualization in 3D scenes is also possible because of the fact that **camimage** element can be nested within a DrawingPanels3D container.

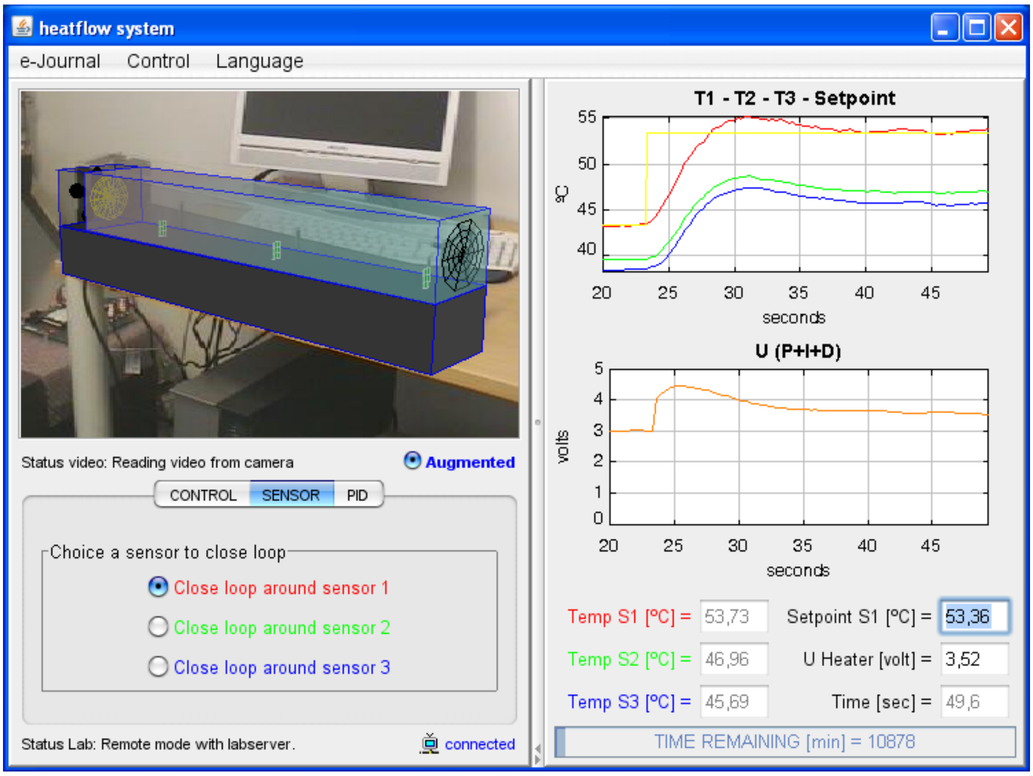


Figure 6: Working remotely with AR.

Figure 6 shows an example where the previously described idea is applied. This application is part of a set of virtual and remote laboratories that the Department of Computer Science and Automatic Control at the Spanish University for Distance Education (UNED) provides to students for remote experimentation, see other example in [25]. The heatflow system is based on a thermal process that allows students to conduct practical experiments on temperature control systems with transport delays. The interface is easily accessed by students through Internet by means of a simple web browser that loads an EJS-based Java applet.

Students use this tool to study basic control concepts such as *systems' identification* and *PID controllers design*. Firstly, they perform open-loop experiments in order to get data registers that allow them to identify a model of the process. In a second stage, students must design PID controllers to regulate temperature in each sensor based on the previously identified models.

As the reader can imagine, temperature changes are not visible to the human eye. To facilitate visualization of the heat-flow dynamics, the interface's left panel displays a 3D representation of the apparatus in which the inner air's color changes according to temperature data obtained from a remote server. This virtual representation is displayed onto video streaming captured from a remote IP camera. In this case, the augmented reality is a very useful technique that helps students to get a better insight about the physical phenomenon under study.

4.2.2 Example 2: An AR-enabled Web-based lab of a robot arm

A very nice AR-enabled remote lab developed with the approach previously described is shown in Figure 7. The system located at the University of Alicante (UA) was entirely developed for the teaching and learning of automation and robotics.

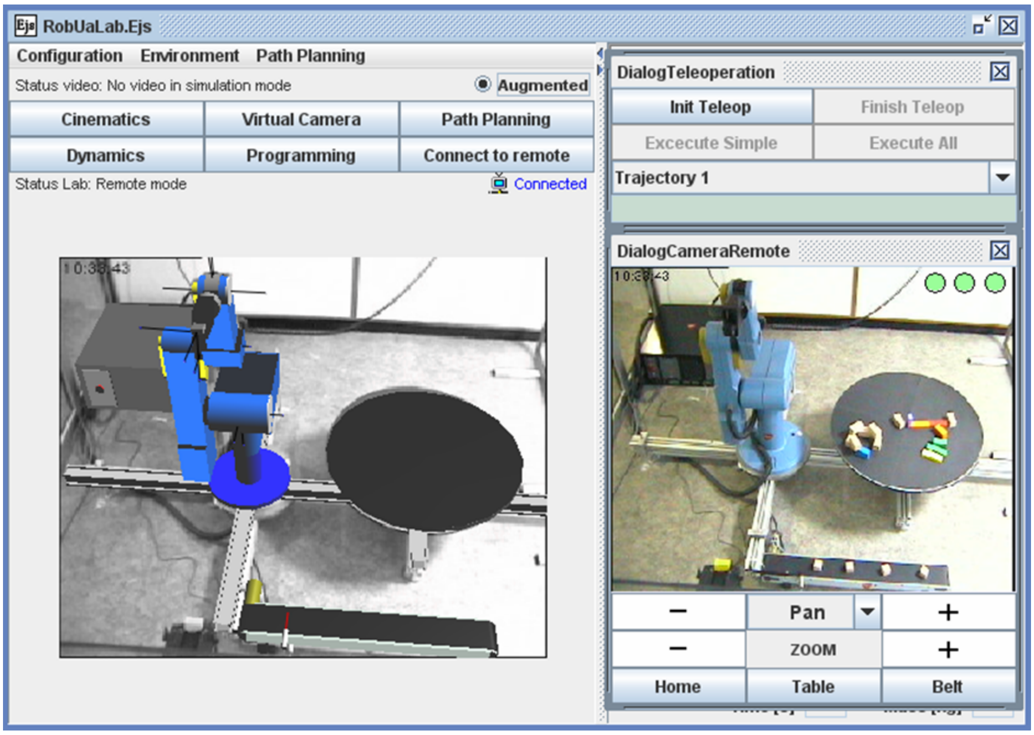


Figure 7: Working remotely with AR.

The main hardware components of the system are an industrial robot Scorbot ER-IX (Intelitek) with five degrees of freedom, a small warehouse for pieces, a conveyor belt, and a rolling table. Right side on Figure 7 shows the actual plant, which was built and assembled by a research group at UA. For further details about this laboratory see [26].

The video streaming captured from the remote IP camera is enriched by a virtual representation of the robot arm following the proposed approach. The dynamics of this virtual view can be updated by two ways: by using position data from the encoders of the actual robot arm, or by using position data calculated from the mathematical model of the system. Note that the location of the source of the comming position data is different in both cases. In the first one data arrive from the remote server, whereas in the second case data are computed locally by the EJS solver. In this example, AR allows to compare the simulated and real behaviour of the system by observing a motion gap between the virtual and real arm.

5 Conclusion and Future Works

The article introduces the using of the augmented reality in remote experimentation with pedagogical purposes. Augmented reality mixes real world images with computer generated data in order to give a major feeling of physical presence to students. This work focuses on providing to instructors a new approach to create application with augmented reality capabilities.

The approach described uses *Easy Java Simulations* in combination with a novel visual element called **camimage** to build the AR-enabled applications. The **camimage** element uses, in turn, the Java package **webcam.jar** to manipulate easily different types of IP cameras. Some examples of the use of the approach proposed are described. Future work will take into account the register issue, which could be quite useful when either the camera or the physical device are moving, such as the case of a remote laboratory of mobile robots.

Bibliography

[1] Azuma, R. (1997); A Survey of Augmented Reality, *Presence: Teleoperators and Virtual Environments*, 6(4): 355-385.

[2] Azuma, R. et al. (2001); Recent advances in augmented reality, *IEEE Computer Graphics and Applications*, ISSN 0272-1716, 21(6): 34-47.

[3] Feiner, S. et al. (1997); A touring machine: Prototyping 3D mobile augmented reality systems for exploring the urban environment, *Personal and Ubiquitous Computing*, ISSN 1617-4909, 21(6): 208-217.

[4] Kalkofen, D. et al. (2007); Interactive Focus and Context Visualization for Augmented Reality, *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, ISBN 978-1-4244-1749-0, pp. 1-10.

[5] Klopfer, E. and Squire, K. (2008); Environmental Detectives the development of an augmented reality platform for environmental simulations, *Educational Technology Research and Development*, ISSN 1042-1629, 56: 203-228.

[6] Esquembre, F. (2004); Easy Java Simulations: a software tool to create scientific simulations in Java, *Computer Physics Communications*, ISSN 0010-4655, 156(2): 199-204.

[7] *Ejs Java Simulations* home page, `http://www.um.es/fem/EjsWiki/`.

[8] Dormido, S. (2002); Control learning: Present and future, *IFAC Annual Control Reviews*, 28: 115-136.

[9] Dormido, S. et al. (2005); The role of interactivity in control learning, *The International Journal of Engineering Education*, 21(6): 1122-1133.

[10] Sánchez, J. et al. (2005); The learning of control concepts using interactive tools, *Computer Applications in Engineering Education*, 13(1): 84-98.

[11] Farias, G. (2010); Adding Interactive Human Interface to Engineering Software, *Ph.D. Thesis from Department of Computer Science and Automatic Control.*

[12] Farias, G. et al. (2010); Java Simulations of Embedded Control Systems, *Sensors*, 9(10): 8585-8603.

[13] Vargas, H. et al. (2009); Web-enabled Remote Scientific Environments, *Computing in Science and Engineering*, 11(3): 36-46.

[14] Vargas, H. (2010); An Integral Web-based Environment for Control Engineering Education, *Ph.D. Thesis from Department of Computer Science and Automatic Control.*

[15] Farias, G. et al. (2010); Developing Networked Control Labs: A Matlab and Easy Java Simulations Approach, *IEEE Transactions on Industrial Electronics*, 57(10): 3266-3275.

[16] Aliane, N. (2010); Limitations of Remote Laboratories in Control Engineering Education, *International Journal Online Engineering*, 6(1): 31-33.

[17] Casini, M. (2003); The Automatic Control Telelab: A User-Friendly Interface for Distance Learning, *IEEE Transactions on Education*, 46(2): 252-257.

[18] Casini, M. (2007); Web-Based Control and Robotics Education, *Publisher Springer*, pp. 127-151.

[19] Martín, C. (2007); Web-Based Control and Robotics Education, *Publisher Springer*, pp. 103-125.

[20] Internet Remote Experimentation from NUS (2011); `http://vlab.ee.nus.edu.sg/vlab`.

[21] Dormido, S. el al. (2005); Adding Interactivity to Existing Simulink Models Using Easy Java Simulations, *Proc. 44th IEEE Conference on Decision and Control and the European Control*, Sevilla, Spain.

[22] Nguyen, A.V. (2006); Iterative Design and Evaluation of a Web-Based Experimentation Environment, *User-Centered Design of Online Learning Communities*, pp. 286-313.

[23] Nguyen, A.V. (2007); Activity Theoretical Analysis and Design Model for Web-Based Experimentation, *Proc. Int'l Conf. Human-Computer Interaction.*

[24] Restivo M.T. and Silva M.G. (2009); Portuguese Universities Sharing Remote Laboratories, *Int'l J. Online Eng.*, 5: 16-19.

[25] Dormido, R. et al. (2008); Development of a Collaborative Web-Based Control Laboratory for Automation Technicians: The Three-Tank System, *IEEE Transactions on Education*, 51(1): 35-44.

[26] Vargas, H. et al. (2011); A Network of Automatic Control Web-based Laboratories, *IEEE Transactions on Learning Technologies*, 4(3): 197-208.

INT J COMPUT COMMUN, ISSN 1841-9836
8(4):635-646, August, 2013.

Knapsack-model-based Schemes and WLB Algorithm for the Capacity and Efficiency Management of Web Cache

A.B. Xiang, L. Xu, B.Z. Niu

Anbo Xiang
Development Research Center of the State Council, Beijing, 100010, China
abxiang@drc.gov.cn

Liang Xu
Department of Logistics Management, School of Business Administration,
Southwestern University of Finance and Economics, Chengdu, 611130, China
arecxuliang1112@gmail.com

Baozhuang Niu*
Lingnan College, Sun Yat-sen University, Guangzhou, 510275, China
niubzh@mail.sysu.edu.cn
*Corresponding author

Abstract: Web cache refers to the temporary storage of web files/documents. In reality, a set of caches can be grouped into a cluster to improve the server system's performance. In this paper, to achieve the overall cluster efficiency, we propose a weighted load balancing (WLB) routing algorithm by considering both the cache capability and the content property to determine how to direct an arrival request to the right node. Based on Knapsack models, we characterize three new placement/replacement schemes for Web contents caching and then conduct the comparison based on WLB algorithm. We also compare WLB algorithm with two other widely used algorithms: Pure load balancing (PLB) algorithm and Round-Robin (RR) algorithm. Extensive simulation results show that the WLB algorithm works well under the examined cluster content placement/replacement schemes. It generally results in shorter response time and higher cache hit ratio, especially when the cache cluster capacity is scarce.
Keywords: web cache placement/replacement scheme, Knapsack model, load balancing and routing algorithm, performance analysis.

1 Introduction

In recent years, with the rapid growth of the size of contents delivered and the number of internet users, the World Wide Web (WWW) is better known as "World Wide Wait" [1]. Although various information technologies are being developed rapidly, which makes the internet becomes faster and faster, according to [2]'s observation, "the trend of increasing traffic on the Internet is likely to continue". It is reported that if a web page can not be loaded within eight seconds, then users are likely to give up or load the link in a second browser [3]. As a result, a loss of revenue will be suffered, which is called the *Zona effect* [1]. For example, the possible revenue loss by increasing a millisecond to execute automated trades, according to [3], can be as high as \$100 million. Therefore, there exist plenty of incentives to reduce the web response time when the server capacity is constrained (It may cost a lot to enlarge the server capacity.). In practice, an effective approach to alleviate the user-waiting problem and to optimize the Web resource utilization is *caching*, in which the Web content is generated once and kept for a period of time for the future use. When a user requests a previously requested content, the content can be accessed directly from the cache. Recently, content caching has been a very active research area.

Generally speaking, a good caching system must address the following issues:

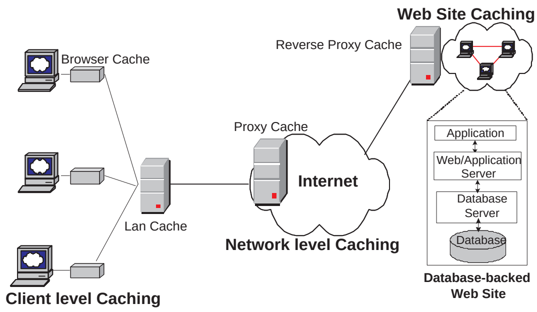


Figure 1: Overview of Caching

(1) What to cache. Cacheable elements include DNS mapping, connection, and content.

(2) Where to cache. A cache usually can reside in one of the three locations: (a) client level, at the client browser and LAN, (b) network level, at a proxy somewhere in between the client and the Web site, or (c) at the Web site itself.

(3) When and how to cache. When are elements are placed in cache and when are elements are evicted from cache.

Industrial experts have already found clear answers/rules to address the first two issues [4], however, more analysis and/or solving methods are needed to address the third issue [5]. Therefore, in this paper, we mainly focus on the third issue for the Web server with content caching cluster at the web site, i.e., reverse proxy cache. Reverse proxy cache, which is a web accelerator, can reduce the workload of a busy Web application server that provides both static and dynamic contents. The static contents can be cached on the reverse proxy while the Web application server is freed up to better handle the dynamic contents.

Towards the cache placement/replacement schemes, in practice, the Least-recently-used (LRU) and least-frequently-used (LFU) are commonly used due to their simplicity. However, with no or limited considerations of content properties or request patterns, these schemes could not provide the best performance. Many other schemes have been proposed, which take into account the retrieval latency of contents, their sizes, the popularity of references and temporal locality of requested contents, to solve the problem in a systematic manner (e.g., [6]).

From a theoretic perspective, the focused problem in this paper is closely related to the *Knapsack problem* that maximizes the benefit under a constraint capacity. Different approaches have been proposed based on the Knapsack model in caching or similar applications, for example, [7] and [8]. Most of these works utilize the cost-benefit optimizing idea of Knapsack model to develop cache management schemes.

It is worth noting that, unlike hard disk storage space, cache's internal main memory is still a limited resource that must be managed wisely [9]. Accordingly, there is another trend in caching technology which provides larger cache capacity (hereafter, cache capacity refers to its internal main memory) to support more concurrent users: cluster-based design for web application server (e.g., [10], [11], [12]). With this design, a set of caches are grouped into a loosely coupled cluster to solve some common issues such as capacity, availability, and performance. When clustered, cache's capacity increases linearly and the number of cacheable missed requests sent to the Web application server is reduced accordingly. The impact of an individual cache node failure in the cluster on the site availability and performance is also reduced considerably. This Web cache cluster architecture is represented in Figure 2, in which a load balancer utilizes the inherent locality of the requests and an adaptive scheme to tune the load allocated to each node in the cluster based on that node's capability. Some leading companies also propose similar commercial

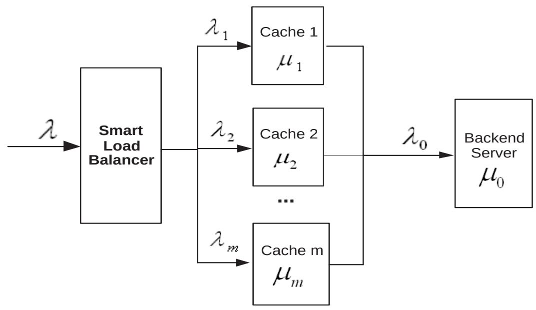


Figure 2: Model of Web Cache Cluster

solutions, such as Microsoft Internet Security and Acceleration Server with Cache Array Routing Protocol, IBM Web Traffic Express proxy servers with Network Dispatcher, Oracle9iAS Web Cache Cluster, Cisco Cache/Content Engine, and so on.

In short, our study was motivated by problems arising in practice and trends in cluster caching. Most of the previous works required or assumed that all nodes in the cluster are symmetric with an equal chance to serve any incoming request. We would like to relax these assumptions and propose a more general solution. Also we want to consider the nonhomogeneity of the content properties together with the request pattern so as to maximize the utilization of the valuable cache capacity.

We first study how to efficiently place/replace cache contents by accurately monitoring the properties of contents, i.e., whether to cache a content, and if we do that, which cluster node to place it so that the system performance is optimized? Furthermore, based on the prior knowledge about the requests arrival pattern, contents properties, and the cache contents placement/replacement scheme, we develop a routing algorithm to determine how to direct an arrival request by efficiently coordinating the cluster nodes.

The remainder of the paper is organized as follows. In the next section, we introduce our Web cache cluster model. The content placment/replacement schemes are discussed based on the Knapsack model in section 3. In Section 4, we propose a weighted load balancing algorithm, by which we can design a smart load-balancer to determine how to direct an arrival request to a cluster node. Extensive simulations have been carried out to evaluate the performance of the algorithms. These simulation results are presented and discussed in section 5. Section 6 summarizes the paper.

2 Model Description

As a reverse proxy, our cache cluster is dedicated to a single or a set of original Web server systems. The model consists of a set of cacheable contents $N = \{1, 2, ..., n\}$, and a set of caches $M = \{1, 2, ..., m\}$, as illustrated in Figure 2. For convenience, we introduce some notation below. More detailed descriptions will be presented when they are used.

C_j capacity of cache j, $j \in M$,

λ total system requests arrival rate,

μ_0 service rate of back-end server,

λ_0 requests arrival rate of the back-end server,

μ_j service rate of the cache j, $j \in M$,

λ_j requests arrival rate of the cache j, we have $\lambda = \sum_{j \in M} \lambda_j$,

O_j set of all contents currently in the cache j, $j \in M$,

w forwarding cost per unit of time in the cache cluster,

R_i retrieval cost of the requested content i from the back-end server,

S_i size of the content i, $i \in N$,

p_i "profit" from serving the content i, $i \in N$,

$P_n(i)$ probability that an arriving request is made for the content i, $i \in N$, given that there are n contents in N.

We formulate the cache cluster content placement/replacement problem as a Multiple-Knapsack problem (MKP) with an objective of maximizing the total "profit" by placing the most "valuable" contents in the available caches:

$$Maximize \sum_{i=1}^{n} \sum_{j=1}^{m} p_i x_{ij} \ , \tag{1}$$

subject to,

$$\sum_{i=1}^{n} S_i x_{ij} \leq C_j \ , \quad j \in M, i \in N, \tag{2}$$

where

$$x_{ij} = \begin{cases} 1, & \text{object } i \text{ is placed in cache } j. \\ 0, & \text{otherwise.} \end{cases}$$

3 Content Placement/Replacement Schemes

The cache cluster content placement/replacement scheme specifies the contents in caches to achieve the specific performance objective. At each decision epoch, the following actions are carried out:

1. initiation, preparing the initial cacheable contents,
2. update the cache contents when cache hit ratio is lower than a preset threshold value,
3. or update the cache contents when a hit miss occurs.

We can achieve different specific objective by specifying appropriate p_i in the equation (1). For example,

1. when $p_i = P_n(i)$, the objective is to maximize the cache hit ratio,
2. when $p_i = P_n(i)S_i$, the objective is to maximize the byte hit ratio,
3. when $p_i = P_n(i)R_i$, the objective is to minimize the retrieval cost.

The Knapsack problem is known to be NP-hard. However, there exist fast heuristics with good performance records. In the following subsections, we discuss how different cache management methodologies are applied in different cases.

3.1 As many contents in cache as possible (AMAP)

In this case, a cache cluster is regarded as a pure Multiple-Knapsack. We follow the principle of placing *as many* contents in caches *as possible* to effectively utilize the capacity of the whole cluster. Consequently we need to add one more constraint in addition to (2):

$$\sum_{j=1}^{m} x_{ij} \leq 1, \quad i \in N. \tag{3}$$

This constraint guarantees that no content would be cached redundantly. It is a reasonable consideration when the contents space is huge while the cluster capacity is limited.

[13] present an approximate dynamic programming (ADP) approach for the multidimensional Knapsack problem that produces near optimal solutions efficiently. We apply their adaptive fixing heuristic to solve our cluster content placement/replacement problem. Our computational evidence suggests that the ADP-based heuristic is an attractive methodology that usually generates good quality solutions in reasonable time.

3.2 Popular contents replication and hit ratio threshold (Threshold value)

In this case, we aim at achieving a high *availability* of the most popular contents on condition that a certain level of hit ratio is guaranteed.

Some prior knowledge of the Web request pattern can facilitate Web cache resource planning and cache hierarchy design, and help us to predict the most popular contents. It has been shown that the Web page request follows a Zipf-like distribution (see [14]). Following this result, we rank all the pages in order of their popularity where page i is the i'th most popular page. Suppose that the number of contents in the system is n, the probability that an arriving request is made for page i is approximated by

$$P_n(i) = \frac{\Omega}{i^{\alpha}}\ , \tag{4}$$

where $\Omega = \left(\sum_{i=1}^{n} \frac{1}{i^{\alpha}} \right)^{-1}$ and α is the Zipf parameter determined by the system property.

If the top k most popular contents can guarantee a certain level of hit ratio, we only place these top k most popular contents in caches. If there are extra free space, we duplicate these contents in caches sequentially until there are no more free space in the cluster. Let P_{th} be a threshold value for hit ratio, then k is determined by the formula below:

$$k = \arg \inf_{l \in N} \textstyle\sum_{i=1}^{l} P_n(i) \geq P_{th}.$$

With this replication caching scheme, the more popular contents are cached redundantly and higher hit ratio can be achieved at any single node. At the same time, a high availability of popular contents can still be maintained when any individual node fails. Due to the replication of popular contents, the tradeoff is less effective usage of the combined capacity of the whole cluster.

3.3 Web contents space partitioning (Partition)

In a cache cluster, cache members may not be identical and have different capabilities in dealing with the arrival requests. Capability (capacity, processing power, bandwidth, etc.) represents a member's potential contribution to the cluster.

In this case, we follow [10] to partition Web contents space based on members' capability, as shown in Figure 3. Each cache is assigned to a certain part of the Web contents. Consequently,

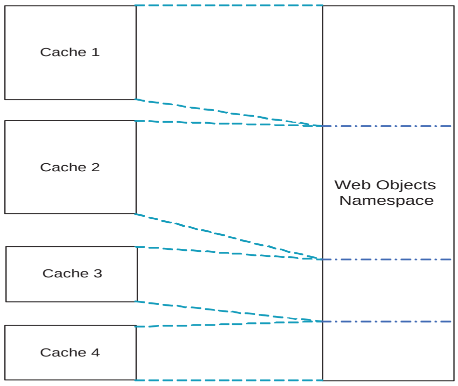


Figure 3: Partitioning Web Objects Namespace based on Caches' Capability

the MKP problem (1) reduces to multiple single-Knapsack problems with cache j being in charge of the contents subset N_j, $N_j \subseteq N$.

A well-known Knapsack problem solution method is the Greedy heuristic, which selects objects for inclusion in the knapsack using the "density" of object as the criterion to be greedy upon. Let $d_{i_j} = p_{i_j}/S_{i_j}$ be the density of content i_j, $i_j \in N_j$. We set $x_{i_j} = 0$ for all i_j that satisfies $S_{i_j} > C_j$, and then arrange the remaining contents in decreasing order of density from top to bottom. Starting from the top, the Greedy heuristic sets $x_{i_j} = 1$ as it goes down until the cache capacity C_j is reached. At each stage, if the next content cannot be included in the cache because its size exceeds the remaining capacity, x_{i_j} is set to 0 for that content and the process continues with the contents below it. The process terminates either when the cache capacity is used up (in this case, $x_{i_j} = 0$ for all contents below the current one), or when all the contents have been examined in this way.

In the simulation section, we will compare the performance of these three schemes: placing as many contents in caches as possible, threshold value for hit ratio and replication for popular contents, and partitioning of Web contents based on caches' capabilities under different parameter settings.

4 Weighted Load Balancing (WLB) Routing Algorithm

Cache clustering introduces a new problem: When a browser requests a particular content, to which cache in the cache cluster should the request be directed? How do we match an incoming request with the cache best able to respond?

Content-aware routing working at application layer certainly can increase the cache hit ratio. But inspecting every incoming HTTP request would increase the system delay, so that the load-balancer may easily become a bottleneck itself and slow down the entire system. Thus, the load-balancer should be kept as "light-weight" and simple as possible to avoid introducing new vulnerabilities into the system (system is only as secure as its weakest component). So in designing our load-balancer, we should consider the system response time and the cache hit ration simultaneously. We want it to work at transport layer and focus on forwarding the data at maximal speed without inspecting every incoming HTTP request, while still guarantees an acceptable cache hit performance.

4.1 The dynamic routing model

When a new request arrives and is assigned to cache j, cache j will be associated with an expected cost function $F(j)$ for handling the request(s). We want to find a suboptimal and easy implemented routing strategy to determine which cache to serve the new arrival request so that the cluster-wide total expected cost is minimized.

In general, serving a request directly from Web cache cluster, i.e., a cache hit, is significantly faster than forwarding the request to the back-end server for generation. In the cache cluster, we also assume that the network bandwidth between peer caches is large and network latency is low, and thus retrieving a cached content from a peer cache is also significantly faster than getting the same content from the back-end server. We denote as w the forwarding cost per unit time in the cache cluster.

There are three possible ways for an arriving request to be served when routed to a cache: (1) the cache contains the requested content and serves the request immediately; (2) the cache does not contain the requested content, but another cache in the cluster contains the content and serves the request; (3) the content is not cached and the request is eventually routed to the back-end server. For the cases (1) and (3), the cost associated with routing the request to any cache is the same. Thus, the cost difference only lies in the forwarding cost in the cluster when the requested content is not cached in the assigned cache but is cached in the other node(s) in the cluster.

From a request's "viewpoint", the routing decision could be determined by the expected cost of sending the request to a cache. i.e. we should direct each arriving request to the queue with the minimum expected waiting cost. Therefore we need to find $l = arg \min\limits_{j \in M} F(j)$. Such a policy minimizes each arriving request's individual expected waiting cost as well as the long-run system waiting cost [15].

4.2 WLB routing algorithm

We assume that the request inter-arrival times and service times are all exponential. Let λ_j and μ_j denote the request arrival rate and the service rate at cache j, respectively. Let O_j denote the set of all contents currently in the cache j and k_j denote the number of outstanding requests waiting at the cache j. $p(i \notin O_h, i \in \bigcup\limits_{v=1}^{m} O_v)$ denotes the probability that the request content i is not in the cache h but is in the cluster.

Based on the above notation, assumptions and analysis, we need to compare two arbitrary decisions h and l. We have

$$F(h) - F(l) = \frac{k_h w}{\mu_h - \lambda_h} \sum_{i=1}^{n} p_n(i) p(i \notin O_h, i \in \bigcup_{v=1}^{m} O_v) - \frac{k_l w}{\mu_l - \lambda_l} \sum_{i=1}^{n} p_n(i) p(i \notin O_l, i \in \bigcup_{v=1}^{m} O_v) ,$$

where

$$p(i \notin O_h, i \in \bigcup_{v=1}^{m} O_v) = \frac{\sum\limits_{j=1}^{m} \sum\limits_{e \in O_j} p_n(e) - \sum\limits_{e \in O_h} p_n(e)}{\sum\limits_{j=1}^{m} \sum\limits_{e \in O_j} p_n(e)} .$$

The two terms in the right-hand side of the equation represent the forwarding costs when the requested content is not cached in the assigned cache h (l) but is cached in the other node(s) in the cluster, respectively. The router should direct the arriving request to cache l if $F(h) - F(l) > 0$, or to cache h otherwise. Thus, we have the following heuristic routing policy:

Rule: when a request arrives, the router directs it to the server l if

$$l = \arg\min_j \Big\{ \frac{k_j}{\mu_j - \lambda_j} P[\ the\ requested\ file\ is\ in\ another\ cache\ instead\ of\ cache\ j\ ] \Big\}.$$

The essence of this WLB algorithm is that the queue length k_j is weighted by the probability that the cache does not have the target content and the service capability. The router directs each arrival request to the cache with the lightest *effective* workload. This WLB routing algorithm makes use of the information related to the cache (it's service capability and workload) and the property of the content (the probability that a request can be satisfied in one cache). Moreover, this algorithm can be easily extended to implement Web server cluster. Since each Web server can satisfy all of the requests, it is natural for us to utilize a pure load balancing policy to guarantee the Web server cluster's performance.

5 Performance Analysis

In this section we compare the performances of different caching schemes. With the WLB arrival request routing algorithm, we first compare performance of LRU scheme with our proposed three new content placement/replacement schemes based on the Knapsack model:

1. placing *as many* contents in caches *as possible* (AMAP scheme, see Section 4.1),
2. placing only contents with popularity values higher than a *threshold value* in caches (Threshold Value scheme, see Section 4.2),
3. partitioning of contents based on caches' capabilities (Partition Scheme, see Section 4.3).

Secondly, we compare the performance of the WLB algorithm with the following two routing algorithms under the *threshold value* content placement/replacement scheme:

1. *Pure Load Balancing* (PLB) Algorithm. With this algorithm, a request is directed to the server with the shortest queue length;
2. *Round-Robin* (RR) Algorithm. A request is directed to the server next to the server which received the previous request.

Using simulation, the average response time (ART), cache hit ratio (CHR), and cache cluster hit ratio (CCHR), the ratio between the total requests and the requests are not served rightly by the assigned caches but by other cache in the cluster, of these algorithms have been compared over following parameters setting:

1. *size rate*, the ratio of cache cluster size vs contents total size;
2. ρ, the ratio of request arrival rate vs the cache cluster service rate. This factor indicates the level of system's workload;
3. *cache types*, we assumed that the caches can be all identical, or be divided into two groups with different capabilities (in terms of service rate, capacity, etc), or all be different in capability.

5.1 Model-driven simulation

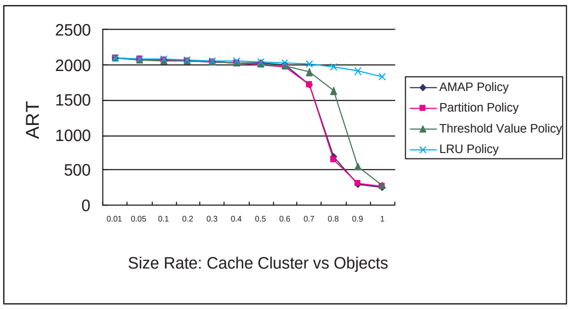
The primary motivation for performing model-driven simulation is to understand the effect of different schemes on cache cluster content management. In the model-driven simulation experiment, the arrival requests follow a poisson stream with rate λ. The target content of the request has Zipf-like frequency distribution with the Zipf parameter α. Generally, without specifying otherwise, the default settings are $\lambda = 0.3$, $\mu_0 = 0.05$, $\mu_j = 0.045$, $m = 10$ (i.e. there are 10 caches in the cluster). Hence the $\rho = 0.6$ (this means the system is of moderate level of workload). The *size rate* equals 0.5, the Zipf parameter $\alpha = 0.8$. We use 10^3 contents of sizes uniformly distributed between 1 and 1000.

The ART, CHR, and CCHR are explored under different parameters combinations. The results are presented in four sets of figures below. We now discuss the numerical results in detail.

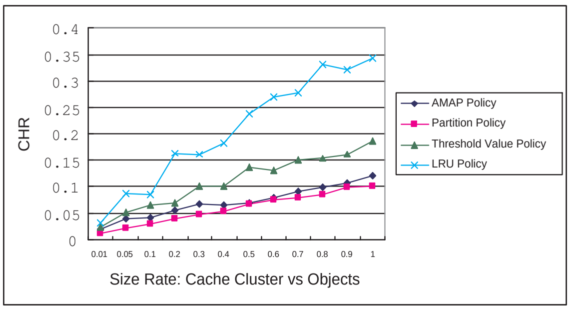
Observation 1: *Increasing cache cluster's capacity can improve the system performances.* We observe the impact of varying the caches' size on the system performances. From Figure 4 and Figure 5, ART is decreasing in cache cluster's size, CHR and CCHR are increasing in cache cluster's size.

Observation 2: *The threshold value scheme is more efficient when the cache cluster capacity is scarce. The partition scheme is more efficient when cache cluster capacity is sufficient.* We compare the performances of different content placement/replacement schemes under different situations aim to find which scheme is more appropriate for a certain situation. From Figure 4(a), it is noticed that when the cache cluster size is small (size rate < 0.6), the *threshold value* scheme achieves shorter ART than other content placement/replacement schemes.

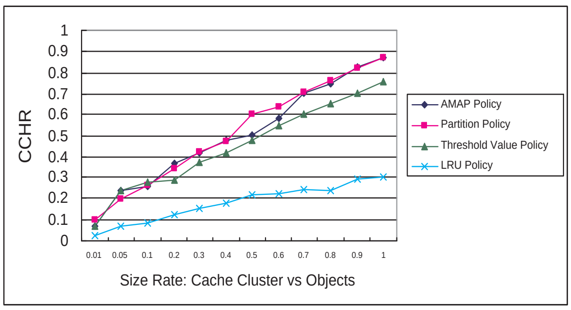
Observation 3: *Better performance can be achieved by considering both node's capability and content's property.* With the *threshold value* content placement/replacement scheme, we compare the performances of the WLB, PLB, and RR routing algorithms under different situations by varying the setting of *size rate*, ρ, and *cache cluster types*. The simulation results (Figures 5) suggest that the WLB has shorter ART, higher CHR and CCHR than PLB and RR algorithm in most of the situations.



(a) Average Response Time vs Size Rate



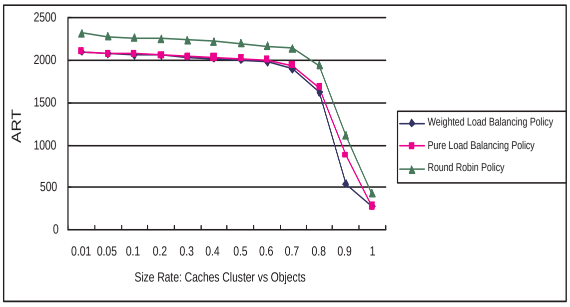
(b) Cache Hit Ratio vs Size Rate



(c) Cache Cluster Hit Ratio vs Size Rate

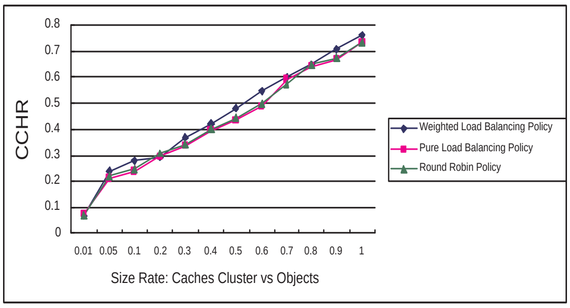
Figure 4: Performances of content placement/replacement schemes with WLB algorithm

Of all the cases simulated, the average improvement in the ART achieved by the WLB is 2.56% compared with the PLB algorithm, and 13.96% compared with the RR algorithm.



(a) Average Response Time vs Size Rate

(b) Cache Hit Ratio vs Size Rate



(c) Cache Cluster Hit Ratio vs Size Rate

Figure 5: Performances of different routing algorithms with the threshold value scheme

5.2 Trace-driven simulation

We now present results from trace-driven simulations using real Web server's traces, the access log data of a Squid proxy server system which was in operation at HKUST in September 2012. The trace length is 409623 user accesses. We ran the trace-driven simulation for different values of the cache cluster capacity by varying the size rate.

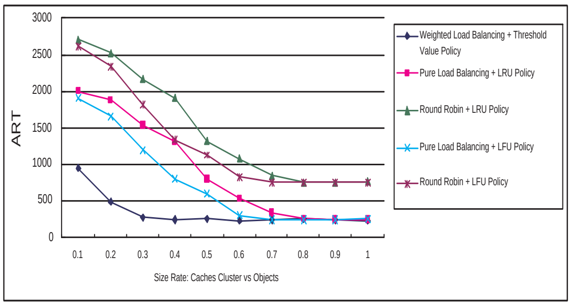
We first compare the performance of the following cache cluster management policies:

1. WLB routing algorithm with threshold value content placement/replacement scheme,
2. PLB routing algorithm with LRU content placement/replacement scheme,
3. RR routing algorithm with the LRU Value content placement/replacement scheme,
4. PLB routing algorithm with LFU content placement/replacement scheme,
5. RR routing algorithm with the LFU content placement/replacement scheme.

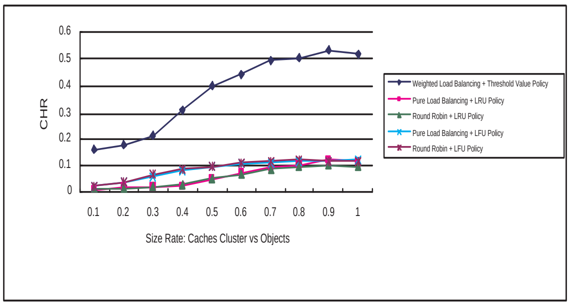
From Figure 6, we find that WLB algorithm with threshold value content placement/replacement scheme achieves better performance than other combination policies significantly, especially when the size rate is smaller. This observation validates the advantage of our Knapsack-based cache cluster management scheme.

6 Conclusion

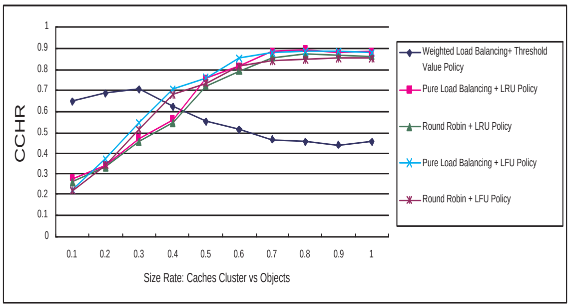
In this paper, based on the Knapsack model, we first propose three efficient placement/replacement schemes for content caching, and then develop a WLB routing algorithm working at transport layer, which considers both of the node's capability and content's property, to determine how to direct an arrival request to the right node by efficiently coordinating the cluster nodes.



(a) Average Response Time vs Size Rate



(b) Cache Hit Ratio vs Size Rate



(c) Cache Cluster Hit Ratio vs Size Rate

Figure 6: Performances of cache cluster management policies under Trace-driven Simulation

Extensive simulation results show that the WLB algorithm with the content replacement scheme leads to satisfactory quality of service (shorter response time and higher cache hit ratio). The simulation results also indicate that content placement/replacement scheme dominates the system performance, and modeling the cache cluster into a multiple Knapsack problem is a good way for us to study the content placement/replacement scheme. Furthermore, the simulation results point out which content placement/replacement scheme is more appropriate under different situations.

In the future study, we plan to integrate the “admission control" and “request priority" mechanisms into our Web cache cluster model with the objective to maximize the system reward. We also expect to investigate the distributed data caching infrastructure over Internet and the distributed resources optimization problems. Moreover, to consider how to organize the cluster (e.g., finding out the optimal number of cache nodes in the cluster) will bring some insight in cache cluster management.

Acknowledgement

This project is supported in part by National Natural Science Foundation of China under Grant 71201175 and National Natural Science Foundation of China under Grant 71201127 and Guangdong Natural Science Foundation under Grant S2012040008081 and S2011040001069.

Bibliography

[1] Datta, A. et al (2003); World Wide Wait: A Study of Internet Scalability and Cache-based Approaches to Alleviate It, *Management Science*, ISSN 0025-1909, 49: 1425-1444.

[2] Kumar, C. ; and Norris, J. (2008); A New Approach for A Proxy-level Web Caching Mechanism, *Decision Support Systems*, ISSN 0167-9236, 46: 52-60.

[3] http://www.economist.com/node/15523761.

[4] http://plone.org/documentation/kb/optimizing-plone/what-to-cache.

[5] Xiang, A. (2006); Essays on information service systemes. PhD Dissertation. Hong Kong University of Science and Technology.

[6] Bahat, O.; Makowski, A. M.; (2003) Optimal Replacement Policies for Non-uniform Cache Objects with Optional Eviction. *IEEE INFOCOM 2003*, San Francisco, California, April.

[7] Wang, B. et al (2002); Proxy-based Distribution of Streaming Video over Unicast/Multicast Connections. *Proceedings of IEEE Infocom*, New York, June.

[8] Otoo, E. et al (2002); Disk Cache Replacement Algorithms for Storage Resource Managers in Data Grids. *The 15th Annual Supercomputer Conference*, Baltimore, Maryland, November.

[9] Rabinovich, M.; Spatscheck, O. (2002); *Web Caching and Replication*. Addison-Wesley, Boston, MA, US.

[10] Feenan,J. et al (2002); Clustering Web Accelerators. *IEEE WECWIS Proceedings*, San Diego, June.

[11] Zhang, X., et al (1999); HACC: An Architecture for Cluster-based Web Servers. *3rd USENIX Windows NT Symposium*, Seattle, Washington, July.

[12] Clevenot, F., et al (2005); Stochastic Fluid Models for Cache Cluster. ISSN 0166-5316, *Performance Evaluation*, 59: 1-18.

[13] Bertsimas, D.; Demir, R. (2002); An Approximate Dynamic Programming Approach to Multidimensional Knapsack Problems. ISSN 0025-1909, *Management Science*, 48: 550-565.

[14] Breslau, L., et al (1999); Web Caching and Zipf-like Distributions: Evidence and Implications. ISSN 0743-166X, *IEEE INFOCOM*, 1, 126-134.

[15] Liu, L. et al (2005); Weighted Load Balancing for Web Server Clusters with Caching. Working paper. Hong Kong University of Science and Technology.

Author index

Abdelkrim M.N., 538
Acarman T., 514
Aggoune W., 538
Amoozad H., 594

Chikamai K., 499
Chung Yeon-ho, 588
Costin H., 525

Djemaï M., 538
Dormido S., 622

Ercan M., 514
Esquembre F., 622

Farias G., 622

Geman O., 525

Hamidi F., 538
Hashemi S.S., 594

Ilić S.S., 548

Jerbi H., 538
Jing Y., 560
Joldeş I., 571

Kim I., 608
Kim T., 608

Lazić L., 548
Li M., 578
Liu H., 560
Lupşe V., 571

Niu B.Z., 635

Oh Se-bin, 588

Pârv B., 571
Parpucea I., 571

Razavi S.H., 594

Sanchez J., 622
Spalević P., 548
Sung M., 608

Tang M., 578
Tapamo J-R., 499

Vargas H., 622
Viriri S., 499

Xiang A.B., 635
Xu L., 635

Zavadskas E.K., 594
Zorić A.Č., 548