International Journal of Computer Science & Emerging Technologies

ISSN 2044-6004 (Online)

Volume 2 Issue 2

April, 2011

www.ijcset.excelingtech.co.uk

ijcset@gmail.com

IJCSET BOARD MEMBERS

- **Nafiz Imtiaz Bin Hamid**, Bangladesh
- **Jasvir Singh**, India
- **Manas Ranjan Biswal**, India
- **Ratnadeep R. Deshmukh**, India
- **Sujni Paul**, India
- **Mousa Demba**, Saudi Arabia
- **Yasser Alginahi**, Saudi Arabia
- **Tarun Kumar**, India
- **Alessandro Agostini**, Saudi Arabia

TABLE OF CONTENTS

General Machine Learning Classifiers and Data Fusion Schemes for Efficient Speaker Recognition

Siwar Zribi Boujelbene1, Dorra Ben Ayed Mezghani[2] and Noureddine Ellouze [3]

[1]Faculty of Humanities and Social Sciences of Tunis – Tunisia,
[2]High Institute of Computer Science of Tunis– Tunisia,
[3] National School of Engineer of Tunis– Tunisia,
BP – 37 Campus Universitaire 1002 Tunis – Tunisia
siwarboujelbene@yahoo.fr, Dorra.mezghani@isi.rnu.tn, N.Ellouze@enit.rnu.tn

***Abstract*:** Data fusion methods can take advantage of the concepts of diversity and redundancy to improve system performance. Diversity can be used to improve system performance through the incorporation of different information. Similarly, redundancy can achieve the same goals through the re-use of data. These concepts have been thoroughly applied on pattern recognition problems. The basic idea is that if several classifiers can be constructed, whose errors are mutually uncorrelated, then performance advantages can be obtained through the propel classifiers fusion.

The contribution of this paper is to study the fusion of several machine learning classifiers and to analyze data fusion schemes for text independent speaker identification. Feature spaces are defined by combining the Mel-scale Filterbank Cepstrum Coefficients (MFCC) and delta coefficient. Each feature is modelled using the gaussian mixture model (GMM) that constructs a speakers' models dictionary used later as inputs for classification. Then, four popular supervised machine learning classifiers are considered, namely the multilayer perceptrons classifier (MLP), the support vector machines classifier (SVM), the decision tree (DT) classifier and the radial basis function networks classifier (RBF). The scores (outputs) of classifiers are considered according to different scenario. Results showed that the best performance had been achieved by fusing the SVM, the MLP and the DT classifiers that reported a speaker identification rate equal to 94.15 %.

***Keywords*:** SVM, MLP, DT, RBF, data fusion, speaker recognition**.**

1. Introduction

Speaker recognition refers to the concept of recognizing a speaker by his/her voice. There are two main tasks within speaker recognition task: speaker verification and speaker identification [23].

The objective of speaker verification is to verify the claimed identity of that speaker based on the voice samples of that speaker alone. Speaker identification deals with a situation where the person has to be identified as being one among a set of speakers by using his/her voice samples. The speaker identification problem may be subdivided into closed set and open-set [4]. If the target speaker is assumed to be one of the registered speakers, the recognition task is a closed-set problem. If there is a possibility that the target speaker is none of the registered speakers, the task is called an open-set problem. In general, the open-set problem is much more challenging. In the closed-set task, the system makes a forced decision simply by choosing the best matching speaker from the speaker database. However, in the case of open-set identification, the system must have a predefined tolerance level so that the similarity degree between the unknown speaker and the best matching speaker is within this tolerance. Another distinguishing aspect of speaker recognition systems is that they can either be text-dependent or text-independent depending on the application. In the text-dependent case, the input sentence or phrase is fixed for each speaker, whereas in the text-independent case, there is no restriction on the sentence or phrase to be spoken.

As any speech recognition system, speaker recognition system consists on two stages; namely, feature extraction and classification (see figure 1).

Feature extraction consists on obtaining the characteristic patterns of the signal of a speaker. It can be considered as a data reduction process that attempts to capture the essential characteristics of the speaker with a small data rate. The feature extractor converts the digital speech signal into a sequence of numerical descriptors, called feature vectors. Features provide a more stable, robust, and compact representation than the raw input signal. Classifier uses these features as inputs.

State of the art speaker recognition systems are based on a cepstral feature extraction follow by a GMM [24] or a hybrid GMM/SVM classifier [15], [17], [13]. Nowadays, in classification stage, a new approach consists in fusing different systems is increasingly used. This technique can be divided in tree main categories: systems based on feature's diversity [8], [12], systems based on a classifier's diversity [31] and systems based on data fusion diversity [10]. Indeed, searchers are looking for the best set of feature, the best set of classifier and/or the best set of data fusion schemes.

In this paper, our study deals with the two last categories. It consists, and after extracting the feature vectors, on using four popular supervised machine learning classifiers for text independent closed-set speaker identification. This includes the MLP, the SVM, the DT and the RBF classifiers. These classifiers are evaluated according to different scenarios. In addition, several data fusion schemes are considered namely the majority voting, the mean rule and the product rule.

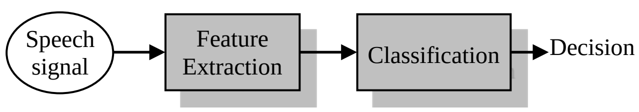


Figure 1. General diagram of speaker recognition system

This paper is organized as follows. Section 2 provides a description of the feature extraction and presents the features that we consider in this study. Afterward, in section 3, we introduce the machine learning theory for speaker recognition. Then, in section 4, we present briefly the MLP, the SVM, the DT and the RBF classifiers. This is followed in section 5 by a description of methods adopted for fusing the scores of different classifiers. Finally the experiments we made and the obtained results are drawn in section 6.

2. Feature Extraction

Feature extraction is a fundamental speech recognition stage. For speaker recognition, we are interested in the features that correlate with the physiological and behavioural characteristics of the speaker. These characteristics exist both in the supra-segmental features (voice source characteristics) of speech and in the spectral envelope (vocal tract characteristics). Although, it's impossible to separate these kinds of characteristics, signal measurements such as short term and long term spectra, and overall energy are easy to extract. These measurements provide the means for effectively discriminating among speakers [19].

2.1 Critera for feature selection

Features are numerical measurement used in computations that try to discriminate between classes. The selection of features depends largely on the application. For speaker recognition, Rose suggests that the optimal features should have the following properties [37]: easy to measure, high inter-speaker variation, low intra-speaker variation, robust against disguise and mimicry, robust against distortion and noise and maximally independent of the other features.
It is unlikely that a single feature would fulfil all these requirements. Fortunately, due to the complexity of speech signals, a vast number of complementary features can be combined to improve accuracy. In literature, a large number of features have been proposed for speaker recognition.

2.2 Spectral features

Feature extraction is usually computed by temporal methods like the Linear Predictive Coding (LPC) or frequencial methods like the Mel Frequency Cepstral coefficient (MFCC) or both methods like Perceptual Linear Coding (PLP). A nice property of spectral methods is that logarithmic scales (either amplitude or frequency), which mimic the functional properties of human ear, improve recognition rates [2]. In [27] Vergin et al. suggest that MFCC has been widely accepted as a features inputs for a typical speaker recognition system because it is less vulnerable to noise perturbation, gives little session variability and is easy to extract than others.
In a speech signal, the various type of information can change rapidly through time. For this reason, the signal $s[K]$ is divided into frames as in (1):

$$f[n] = \{s[nW + k] : k = 0,...W - 1\}, \quad (1)$$

each consisting of W samples. W must be large enough to include sufficient information, but it must also be small enough to ensure that the assumption of stationarity is reasonable. Frames normally overlap with their starting points following each other by L samples (L<W), because the signal does in fact change during the length of one frame..
For MFCC feature extraction, each feature vector is extracted from a frame. The frame is passed through a Humming filter and converted to the frequency domain using the discrete Fourier transform (DFT) [35]. Mel-scale frequency is related to linear frequency by the followed formula:

$$Mel(f) = 1127 \ln\left(1 + \frac{f}{700}\right) \quad (2)$$

The frequency range in Mel-scale is divided into a number of equal-sized bands. In linear frequency, triangle filters are positioned so that the width of each filter is equal to two bands in the Mel scale. Two successive filters also overlap each other by one of these Mel-scale bands. The value for energy in each band after filtering is called a Mel filter bank coefficient.
Cepstral analysis involves working with the spectrum of the spectrum. More specifically, the inverse Fourier transform is applied to the log-spectrum of the signal. Mel filter bank coefficients mi comes directly from the signal spectrum. They can be transformed into Mel-frequency cepstral coefficients (MFCCs) c_i by using the discrete cosine transform:

$$c_i = \sum_{j=1}^{B} m_j \cos\left(\frac{\pi i}{B}(j - 0.5)\right) \quad (3)$$

B is the number of Mel filter bank coefficients. The resulting MFCCs for each frame are grouped into a D-dimensional feature vector x.

2.3 Dynamic feature

While speaking, the articulators make gradual movements from a configuration to another one, and these movements are reflected in the spectrum. The rate of these spectral depends on the speaking style and speech context. Some of these dynamic spectral coefficients are clearly indicators of the speaker itself. According to Soong et al., delta features are the most widely used method for estimating feature dynamics [26]. Two kinds of delta features can be employed: first derivatives (delta) and second derivatives (delta-delta). These coefficients give an estimate of the time derivative of the features they are applied to.
The delta features coefficients are obtained by simply calculating the difference between two successive feature vectors. The resultant vector is appended to the second feature vector, making the procedure causal (i.e. only history is taken into account). The new feature vector then has double the dimension of the original.

3. Machine Learning

Machine learning is one of the hottest research areas. It has been widely adopted in real-world applications, including speech recognition, handwritten character recognition, image classification and bioinformatics.

3.1 Learning paradigms

In the literature, there are three main categories of machine learning classifiers, namely, those that are trained with supervised training classifiers, those that are trained with unsupervised training classifiers and those that are trained with hybrid training classifiers.

In supervised classifiers category, the data provided to the model training classifiers contains class information via a label [16]. However, unsupervised classifiers category does not require any information regarding class membership for the training data. Whereas, in hybrid classifiers category, training classifier combines both supervised and unsupervised learning. Part of the solutions (network weights, architecture, or computer programs) is determined through supervised training classifiers, while the others are obtained through unsupervised training classifiers.

In term of speaker recognition, unsupervised training classifier uses only data from a specific speaker to create a model. Whereas supervised training classifier assigns different labels to different speakers in a population. Examples of supervised training classifiers include multilayer perceptrons, support vector machines, decision trees and radial basis function networks. Supervised training classifiers have an advantage over unsupervised training classifiers in that they can better capture the differences between a particular speaker and other speakers in the population [10]. However, the amount of training data and hence, the computational effort in deriving a speaker model can be more than that for unsupervised training classifiers.

3.2 Classifiers structure for speaker recognition

The classifier consists of the speaker modelling, the pattern matching and the decision logic. Its operation constitutes two important steps (train and test). In the training step, feature vectors are used to obtain the M speaker models. For each speaker, a different model is obtained from his/her speech. In the testing step, feature vectors are first computed from an unknown speaker. For speaker identification, the feature vectors are compared with each of the M speaker models presented by the speakers' model dictionary in order to have the scores file: Score(1) to Score(M). These scores are used to bring up a decision. In a closed set scenario, the best score, Score(i), identifies the unknown speaker as speaker i. This means that speaker model i most likely generated the feature vectors. In an open set scenario, Score(i) is further compared against a threshold to decide if there is an adequate match between the unknown speaker and the best model i. If the match is deemed to be adequate, the speaker is identified. Otherwise, it is decided that no speaker model represents the unknown speaker. For speaker verification, the unknown speaker claims a certain identity j. Only Score(j) is calculated and compared against a threshold to verify or reject the claimed identity.

4. Supervised Machine Learning Classifiers for Speaker Recognition

Currently, several supervised training classifiers have been investigated for speaker recognition. These include multilayer perceptrons [22], [14], support vector machines [25], [28], [15], [29], [21], [32], decision trees [11], [30] and radial basis function networks [7]. Such classifiers are able to generate a model that can distinguish one speaker among M classes of speakers. In fact, during training, the supervision is affected to a label that is associated to each feature vector. This label determines the class membership of that vector (speaker to which it belongs). This partitioning of training data is illustrated in figure 2.

4.1 Multilayer perceptron

The multilayer perceptron (MLP) is a popular form of neural network that has been considered forvarious speech recognition [34]. Perceptrons use the basic architecture illustrated in figure 3.

The network functions through combining the various features vectors with some set of weights. This sum is then used as input for a single neuron's activation function. The output of the activation function is then taken to be the output of the network. Perceptrons with multiple outputs are composed of several independent perceptron networks each determining the value of a single output. The weights for MLPs are trained with the backpropagation algorithm such that they can associate a high output response with particular input patterns [38].

For speaker recognition, test vectors, from training data, should have a "one" response for that speaker's MLP for a specific speaker, whereas from different speakers, test vectors should have a "zero" response [14]. For speaker identification, all test vectors are applied to each MLP and the outputs of each vector are accumulated. The speaker is identified as a corresponding to the MLP with the maximum accumulated output. For speaker verification and in order to be verified, all test vectors are applied to the model of the speaker. The output is accumulated and then normalized. If the normalized output exceeds a threshold, the speaker is verified, else rejected.

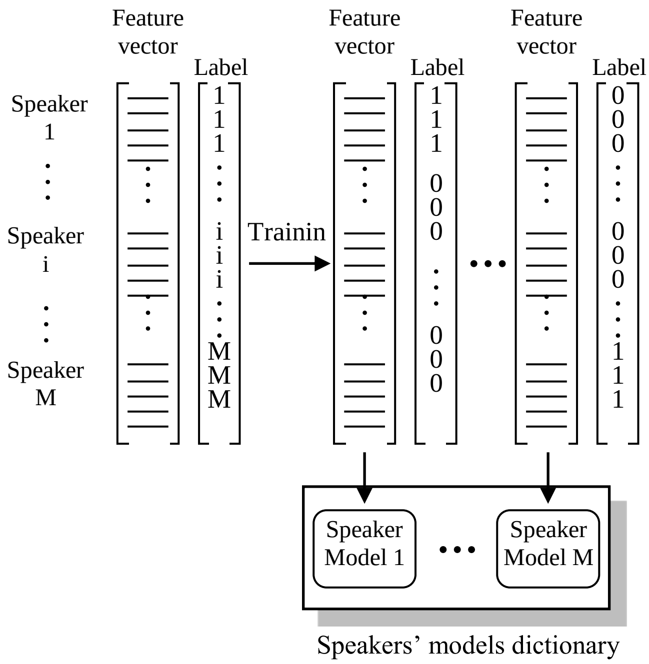


Figure 2. Supervised training data partitioning

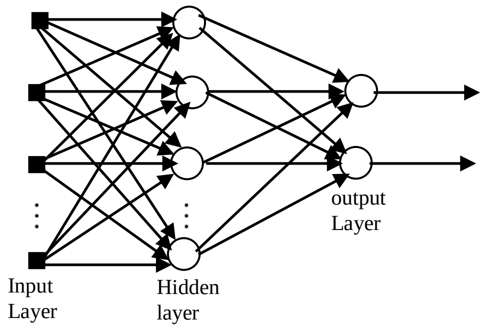


Figure 3. Multi layer perceptron architecture with one hidden layer

4.2 Decision tree

A decision tree (DT) describes a collection of rules, organized in a hierarchical fashion, that implement a decision structure. It consists of leaves and nodes. A leaf records an answer (often called a class) and a node specifies some test conditions to be carried out on a single feature value of an instance, with one branch and sub-tree for each possible result of the test [36]. For a given feature vector, a decision is made by starting from the root of a tree and moving through the tree determined by the outcome of a condition test at each node until a leaf is encountered [36]. The process of building a decision tree is a recursive partitioning of a training set.

For speaker recognition, the feature vectors are obtained from the training data for all speakers. Then, the data is labeled and a binary decision tree is trained for each speaker. The leaves of the binary decision tree identify the class label as follow: a one corresponds to the speaker and a zero corresponds to "not the speaker". For speaker identification, all feature vectors are applied to each decision tree for the test utterance. The labels are scored and the speaker having the maximum accumulated score is selected.

4.3 Support Vector Machines (SVM)

Support vector machines are, originally, introduced by Vapnik [39]. In a support vector machine, input vectors are mapped into a very high-dimension feature space through a non-linear mapping. Then a linear classification decision surface is constructed in the high-dimension feature space. This linear decision surface can take a non-linear form when it is mapped back into the original feature space.

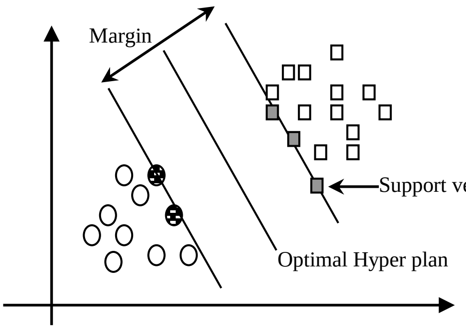


Figure 4. For SVM, a good separation is achieved by the hyperplane that has the largest distance to the neighboring data of two classes

Special properties of the decision surface ensure good generalisation ability of this learning machine [1]. While learning from data, SVM performs structural risk minimization (SRM) unlike the classical adaptation methods that minimize training error in a specific norm and maximize the geometric margin. Hence they are also known as maximum margin classifiers. The following figure presents the maximum-margin hyperplane and margins for a SVM trained with samples from two classes (linear separable case). Samples on the margin are called the support vectors. A critical aspect of using SVMs successfully is the design of the inner product, the kernel, induced by the high dimensional mapping.

For speaker recognition, the first approach in using SVM classifiers was implemented by Schmidt in [25]. Another approach became recently more popular, consists of using a combination of GMMs and SVMs. In [17], [18], [21], [29], [31], [13] several types of combination were proposed.

4.4 Radial basis functions

A Radial basis function (RBF) networks are embedded into a two layer feed forward neural network (see figure 5) [9]. Such a network is characterized by a set of inputs, a set of outputs and in between the inputs and the outputs there is a layer of processing units called hidden units. Each of them implements a radial basis function. The output units implement a weighted sum of hidden unit outputs. According to Park and al. [9], the RBF network classifier consists on: first specifying the hidden unit activation function, the number of processing unit, a criteria for modelling a given task and a training algorithm for finding the parameters of the network. Second and after having a set of input-output training data, RBF networks consists on clustering the training data into M clusters. The centroids of the set of clusters are used within the kernel functions, which are typically Gaussian kernels or sigmoids. The outputs of the kernel functions are used for training a single layer perceptron [7].

5. Data Fusion Schemes

The ultimate goal of designing pattern recognition systems is to achieve the best possible classification performance for one task. This led, traditionally, to the development of different classification methods.

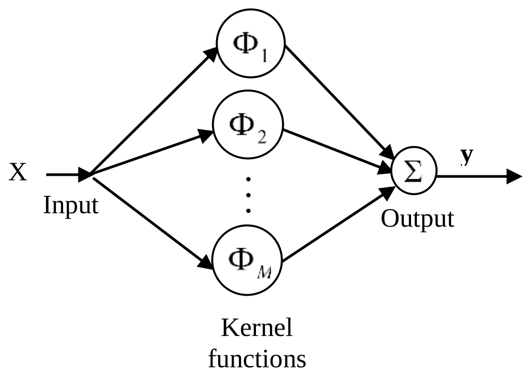


Figure 5. Radial basis function network

The results of an experimental assessment of the different classifiers would then be the basis for choosing one of the classifiers as a final solution to the problem. In [20], Kittler et al., suggested that in such studies, it had been observed that although one of the classifiers would yield the best performance, the sets of patterns misclassified by the different classifiers would not necessarily overlap. This suggested that various classifier designs potentially offered complementary information about the patterns to be classified which could be harnessed to improve the performance of the selected classifier.

These observations motivated the relatively recent interest in classifiers fusion. Multiple expert fusion aims to make use of many different designs to improve the classification performance. Over the last few years a myriad of methods for fusing the outputs of multiple classifiers have been proposed [5]. Several fusion schemes have been devised and it has been experimentally demonstrated that some of them consistently outperform a single best classifier [20]. However, there is presently inadequate understanding: why some fusion schemes are better than others, in what circumstances and if the way of fusion influences on the results.

In [3] and [6], Kittler developed a common theoretical framework for fusing several classifiers which use different pattern representations. They presented a number of possible fusion schemes, namely product, sum, min, max, and majority vote rules, and compared their performance empirically using two different pattern recognition problems.

Within the context of speaker recognition, data fusion comprises the fusion of scores from different classifiers trained for a speaker. These classifiers may be trained by considering different features data or different classifiers. It is desired that the errors of one classifier are corrected by the others and vice versa. Indeed, if all classifiers are in agreement upon an error (all classifiers make the same mistake), so no combination will rectify the error. However, as long as there is some degree of uncorrelation among the errors, performance can be improved with the proper combination [10].

6. Experiments and Results

The work presented here belongs to the category of combining the benefits based on classifiers' diversity and data fusion schemes diversity for text independent closed-set speaker identification (see figure 6). Different scenarios are implemented as follows:

- In the first scenario, the MLP, the SVM, the DT and the RBF classifiers were firstly evaluated individually.
- In the second scenario, the outputs of the MLP, the SVM and the DT classifiers are fused in different ways and with several data fusion schemes.
- In the third scenario, and in order to study the effect of the RBF classifier on such fusions, the output of the RBF is fused with the outputs of the MLP, the SVM and the DT fusions in different ways and with several data fusion schemes.

6.1. Common experimental setup

As a common experimental setup, feature extraction is the fundamental step that deals with the discriminative features used by the next stage of classification. Feature space forms the input to the classifier that recognizes the pattern. Features are extracted from the DR1 dialect (New England region) of TIMIT corpus through MATLAB Toolbox. They are extracted from the speech signal every 10 ms using a 25 ms window. In [31], Zribi Boujelbene and al. evaluated different combined features by using the MFCC, delta, delta-delta and energy coefficients and suggested that the combination of MFCC and delta yield significantly better than all other combination of latest coefficient for speaker identification task. Thus, we used for this study the combined MFCC and delta features. So, each feature vector contains 24 coefficients characterized by the middle frame of every utterance followed by the label.

As the second common experimental setup, the modelling step is assured by the gaussian mixture model and estimated through the EM algorithm (Expectation Maximization) that maximize the Likelihood criterion (ML). The aim of ML estimation is to find the model parameters, which maximizes the likelihood of the GMM given the training data. So, each speaker is modelled and referred by his/her model. Accordingly, we have a dictionary of speakers' models that constructs the inputs for different machine learning classifiers.

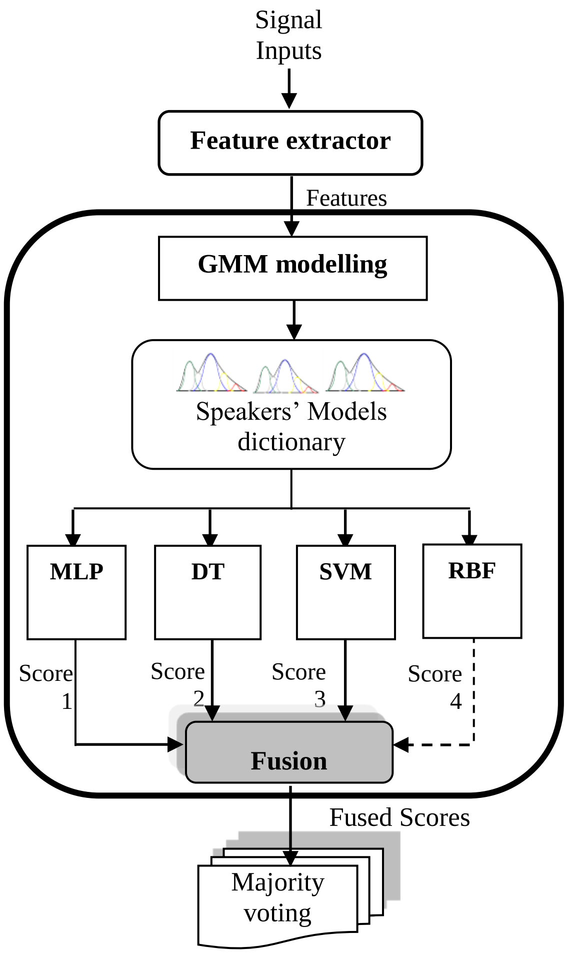


Figure 6. The main structure of different scenarios of fusions for efficient speaker recognition

6.2. Validation approach

One important aspect of recognition performance is the generalization error. The generalization performance of a learning classifier relates to its prediction capability on the independent test data. In [12], [33] Zribi Boujelbene and al., found that the cross validation approach using ten-fold is the most powerful approach for estimating the error of generalization for speech recognition.

All experiments were carried out using a ten-fold cross validation approach. In fact, data were randomly partitioned into ten equally sized where 90% were used for training and the remaining 10% were used for testing. This technique was repeated for ten times, each time with a different test data. Training and testing data were normalized, as data normalization is required for some kernels due to restricted domain, and may be advantageous for unrestricted kernels.

6.3. Results and discussion

In the first scenario, table I shows the performance of the individual classifiers.

In the second scenario, we focus on studying the performance of fusing the MLP, the SVM and the DT classifiers in different ways of fusion and different data fusion schemes. In table II, we are limited to present six ways of fusions namely Fusion1, Fusion2, Fusion3, Fusion4, Fusion5 and Fusion6.

In the third scenario, we focus on evaluating the effect of fusing the output of the RBF classifier with the outputs of the SVM, the DT and the MLP fusions. In table III, we are limited to study only two ways of fusions namely Fusion7 and Fusion8 characterised by the fusion of the output of the RBF classifier with Fusion3 and Fusion4.

Table I shows that the identification rate accuracy is varied between 12.76% and 93.88%. As the experimental results suggest, the RBF classifier presents the highest accuracy when compared to all other classifiers. Results show also that the MLP classifier performs slightly worse than the SVM classifier. We note that the RBF classifier performs well in higher dimensional spaces and it has the advantage over other classifiers, such DT, MLP and SVM.

Table II shows that the identification rate accuracy is varied between 90.42% and 94.15%. It shows also that fusions of the MLP, the SVM and the DT classifiers yield significantly better than the individual classifiers. This can be improved by adjusting the classifier weights on a speaker by speaker basis as opposed to using the same weights across all speakers. We note that the fusions way beginning by the SVM classifier (Fusion3 and Fusion4) have the same behavior whatever the data fusion schemes. That can be explained by the fact that SVM perform well in higher dimensional spaces. Moreover, SVM has the advantage over MLP and DT, that their training always reaches a global minimum.

It can be seen also, from table II, that the product rule outperformed others fusion schemes, what is a theoretical assumption apparently stronger than others rule. In fact, the identification rate accuracy is equal to 94.15% whatever the way of fusion. We conclude that the product rule is the most resilient to estimate the identification rate, which almost certainly explains its superior performance.

Table III shows that the identification rate accuracy is varied between 93.62% and 94.15%. It shows also that the majority voting scheme outperformed others fusion schemes. It can be seen, from table III, that the fusions classifiers have the same behaviour whatever the way of fusion. By using the RBF classifier, we waited for improving the fusions of the SVM, the DT and the MLP classifiers, but the experimental studies show that this scenario couldn’t influence on our results.

Table 1. Identification rate accuracy for individual classifiers

Individual classifiers	**Accuracy (%)**
MLP	**57.71**
SVM	**60.64**
DT	**12.76**
RBF	**93.88**

Table 2. Identification rate accuracy using the fusion of the MLP, SVM and DT classifiers

Fusions classifiers		**Data fusion schemes**		
Label	Way	Majority voting	Mean rule	Product rule
Fusion$_1$	**MLP** - SVM - DT	90.95	90.42	94.15
Fusion$_2$	**MLP** – DT - SVM	92.82	94.15	94.15
Fusion$_3$	**SVM** – MLP - DT	94.15	94.15	94.15
Fusion$_4$	**SVM** – DT - MLP	94.15	94.15	94.15
Fusion$_5$	**DT** – MLP - SVM	92.82	92.82	94.15
Fusion$_6$	**DT –** SVM **-** MLP	92.82	94.15	94.15

7. Conclusion

In this paper, we focus on combining the benefits based on classifiers’ diversity and data fusion schemes diversity for text independent closed-set speaker identification. In fact, our motivation consists on evaluating the fusion of the multilayer perceptrons, the support vector machines, the decision trees and the radial basis function networks classifiers and analysing the use of the majority voting, the mean rule and the product rule fusion schemes.

Table 3. Identification rate accuracy using the fusion of the MLP, SVM, DT and RBF classifiers

Fusions classifiers		**Data fusion schemes**		
Label	Way	Majority voting	Mean rule	Product rule
Fusion$_7$	**RBF** - SVM - DT - MLP	94.15	93.88	93.62
Fusion$_8$	SVM - DT - MLP - **RBF**	94.15	93.88	93.62

For this, feature spaces are defined by combining the Mel-scale Filterbank Cepstrum Coefficients (MFCC) and delta coefficient. Each feature is modelled using the gaussian mixture model that constructs a speakers' models dictionary presenting inputs for classification. Afterward, three scenarios were proposed. The classifiers are firstly used individually. Then a comparative study based on different ways of fusing the MLP, the SVM and the DT classifiers is provided. After that, the RBF classifier is added to the set of fused classifier in order to improve our results. A comparison study is also done between data fusion schemes according to different scenarios.

Experiments show that fusions of the MLP, the SVM and the DT classifiers yield significantly better than the individual classifier. They show also that the best performance is that obtained by Fusion3 and Fusion4 ways characterised by them stability for all data fusion schemes. They show also that the introduction of the RBF classifier in our fusions do not improve our results.

As a previous research works, we will try to combine multiple modalities such as face, voice, and fingerprint.

References

[1] Cortes and Vapnik V., "Support vector networks," Machine Learning, vol. 20, pp. 273–297, 1995.

[2] Marcos Faundez-Zanuy, Mohamed Chetouani, "Nonlinear predictive models: Overview and possibilities in speaker recognition," Progress in Nonlinear Speech Processing Springer Verlag, publisher, pp. 170-189, 2007.

[3] J. Kittler, A. Hojjatoleslami and T. Windeatt, "Strategies for combining classifiers employing shared and distinct pattern representations," Pattern Recognition Letters, Vol. 18, No. 11-13, pp. 1373-1377, November 1997.

[4] S. Furui. "Recent advances in speaker recognition" Pattern Recognition Letters, Vol.18, No 9,pp.859–872, 1997.

[5] Kittler, J., Hancock, E.R., "Combining Evidence in Probabilistic Relaxation," International Journal of Pattern Recognition and Artificial Intelligence, Vol. 3, pp. 29-51, 1989.

[6] Kittler, J., "Combining Classifiers: A Theoretical Framework," Pattern Analysis and Applications, pp. 18-27, 1998.

[7] Han Lian, Zheng Wang, Jianjun Wang, Liming Zhang, "Speaker Identification Using Reduced RBF Networks Array," Springer Berlin / Heidelberg Editor, vol. 3173, pp. 924-929, 2004.

[8] Monte-Moreno, E. and Chetouani, M. and Faundez-Zanuy, M. and Sole-Casals, J., "Maximum Likelihood Linear Programming Data Fusion for Speaker Recognition," Speech Communication . Vol. 51, No. 9, pp 820-830, 2009.

[9] Park, J., Sandberg, J. W., "Universal approximation using radial basis functions network", Neural Computation, Vol. 3, pp. 246-257, 1991.

[10] Ramachandran, R., Farrell, K., Ramachandran, R., and Mammone, R., "Speaker recognition - general classifier approaches and data fusion methods," Pattern Recognition, Vol. 35, pp. 2801–2821, 2002.

[11] Yamagishi Jun'ichi, Masuko Takashi, Tokuda Keiichi, Kobayashi Takao, "Speaker adaptation using context clustering decision tree for HMM-based speech synthesis," Journal IEIC Technical Report , Vol. 103, No. 264, pp. 31-36, 2003.

[12] S. Zribi Boujelbene, D. Ben Ayed Mezghani, and N. Ellouze, "Improved Feature data for Robust Speaker Identification using hybrid Gaussian Mixture Models - Sequential Minimal Optimization System," The International Review on Computers and Software (IRECOS), Vol. 4.3, ISSN: 1828-6003, pp. 344-350, May 2009.

[13] S. Zribi Boujelbene, D. Ben Ayed Mezghani, and N. Ellouze, "Robust Text Independent Speaker Identification Using Hybrid GMM-SVM System", Journal of Convergence Information Technology – JDCTA, Vol. 3.2, ISSN: 1975-9339, pp. 103-110, June 2009.

[14] Bennani Y., P. Gallinari, "A connectionist approach for speaker identification," IEEE International Conference on Acoustic, Speech and Signal Processing, Albuquerque, New Mexico, pp. 265–268, April 1990.

[15] Fine, S., Navrátil, J., Gopinath, R.A., A hybrid GMM/SVM approach to speaker identification," the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'01), vol.1, pp.417-420, 2001.

[16] C. Ji and S. Ma, "Performance and efficiency: Recent advances supervised in learning," Proc. IEEE, vol. 87, pp. 1519–1535, Sept. 1999.

[17] Kharroubi, J., Petrovska-Delacretaz, D., Chollet, G., "Combining GMMs with support vector machines for textindependent speaker verification," in: Eurospeech, pp. 1757–1760, 2001.

[18] Kharroubi, J., Petrovska-Delacretaz, D., Chollet, G., "Text-independent speaker verification using support vector machines," in: Proceedings of Speaker Odyssey, pp. 51–54, 2001.

[19] Tomi Kinnunen, "spectral features for automatic text-independent speaker recognition", thesis, University of Joensuu, Department of computer science, Finland, December 2003.

[20] Kittler, J., Hatef, M., Duin, R.P.W., Matas, J., "On Combining Classifiers," the IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 20, pp. 226-239, 1998.

[21] P. Moreno and P. Ho, "A new approach to speaker identification and verification using probabilistic distance kernels," European Conference Speech Communication and Technology (EUROSPEECH 2003), Geneva, Switzerland, pp. 2965-2968, 2003.

[22] J. Oglesby, J.S. Mason, "Optimization of neural models for speaker identification," IEEE International Conference on Acoustic, Speech and Signal Processing, Albuquerque, New Mexico, pp. 261–264, April 1990.

[23] Reynolds, D, " An overview of automatic speaker recognition technology," the International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2002), Orlando, Florida, USA, pp. 4072–4075, 2002.

[24] [D. Reynolds and R. Rose, "Robust text independent speaker identification using gaussian mixture models," IEEE Trans. Speech and Audio Process., Vol. 3, No. 1, pp. 72-83, Janvier 1995.

[25] M. Schmidt, "Identifying speaker with support vector networks," In Interface '96 Proceedings, Sydney, 1996.

[26] Soong, F., and Rosenberg, A., “On the use of instantaneous and transitional spectral information in speaker recognition,” IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. 36, No. 6, 871–879, 1988.
[27] R. Vergin, B. O’ Shaughnessy and A. Farhat, “Generalized Mel frequency cepstral coefficients for large-vocabulary speaker idenpendent continuous-speech recognition,” IEEE Trans. On ASSP, Vol. 7, No. 5, pp. 525-532, Sept. 1999.
[28] Wan, V., Renals, S., “SVMSVM: support vector machine speaker verification methodology,” the International Conference on Acoustics Speech and Signal Processing, pp. 221–224, 2003.
[29] Wan, V., Campbell, W.M., “Support vector machines for verification and identification,” Neural Networks for Signal Processing X, Proceedings of the 2000 IEEE Signal Processing Workshop, pp. 775–784, 2000.
[30] Wang, W., Lv, P., Zhao, Q.W., Yan, Y.H., “A Decision-Tree-Based Online Speaker Clustering Source Lecture Notes,” In Computer Science; Vol. 4477. the 3rd Iberian conference on Pattern Recognition and Image Analysis, Girona, Spain, pp. 555 – 562, 2007.
[31] S. Zribi Boujelbene, D. Ben Ayed Mezghani and N. Ellouze, “Applications of Combining Classifiers for Text-Independent Speaker Identification,” the 16th IEEE International Conference on Electronics, Circuits and Systems ICECS 09, pp. 723-726, Hammamet-Tunisia, December 2009.
[32] S. Zribi Boujelbene, D. Ben Ayed Mezghani, and N. Ellouze, “Support Vector Machines approaches and its application to speaker identification”, IEEE International Conference on Digital Eco-Systems and Technologies DEST-09, Turkey, pp. 662-667, Jun 2009.
[33] Zribi Boujelbene, D. Ben Ayed Mezghani and N. Ellouze, “Vowel Phoneme Classification Using SMO Algorithm for Training Support Vector Machines”, the IEEE International Conference on Information and Communication Technologies: from Theory to Applications ICTTA-08, Syria, pp. 1-5, 2008.
[34] D.P. Morgan, C.L. Scofield, “Neural Networks and Speech Processing,” Kluwer Academic Publishers, Dordrecht, 1991.
[35] Poakis, J.G. and Manolakis, D.G, “Digital signal processing: principles, algorithms and applications”. 3rd edn. Prentice-Hall, 1996.
[36] Quinlan J. R., “C.4.5: programs for machine learning,” Morgan kaufmann, 1993.
[37] Rose, P. “Forensic Speaker Identification,” Taylor & Francis, London, 2002..
[38] D.E. Rumelhart, J.L. McClelland, “Parallel Distributed Processing,” MIT Cambridge Press, Cambridge, MA, 1986.
[39] V. Vapnik, “The nature of statistical learning theory,” Springer, N.Y., 1995

Author Biographies

S. Zribi Boujelbene received a university diploma in computer science in 1999 from the High Institute of Management of Tunis (ISG-Tunisia), the MS degree in electrical engineering (signal processing) in 2004 from the National School of Engineer of Tunis (ENIT-Tunisia) and prepared a Ph. D. degree in ing (signal processing) from ENIT-

She is currently an associate professor in the computer science department at the Faculty of Humanities and Social Sciences of Tunis (FSHST-Tunisia). Her research interests include data mining approaches such as decision tree, support vector machines, neural networks, fuzzy logic, and pattern recognition such as speech recognition and speaker recognition.
E-mail: siwarboujelbene@yahoo.fr, zribi.siwar@planet.tn

D. Ben Ayed Mezghani received computer science engineering degree in 1995 from the National School of Computer Science (ENSI-Tunisia), the MS degree in electrical engineering (signal processing) in 1997 from the National School of Engineer of Tunis (ENIT-Tunisia), the Ph. D. degree in electrical engineering (signal processing) in 2003 from (ENIT-Tunisia).

She is currently an associate professor in the computer science department at the High Institute of Computer Science of Tunis (ISI-Tunisia). Her research interests include fuzzy logic, support vector machines, artificial intelligence, pattern recognition, speech recognition and speaker identification.
E-mail:Dorra.mezghani@isi.rnu.tn, DorraInsat@yahoo.fr

N. Ellouze received a Ph.D. degree in 1977 from l’Institut National Polytechnique at Paul Sabatier University (Toulouse-France), and Electronic Engineer Diploma from ENSEEIHT in 1968 at the same University.
In 1978, Dr. Ellouze joined the Department of Electrical Engineering at the National School of Engineer of Tunis (ENIT–Tunisia), as assistant professor in statistic, electronic, signal processing and computer architecture. In 1990, he became Professor in signal processing; digital signal processing and stochastic process. He has also served as director of electrical department at ENIT from 1978 to 1983. General Manager and President of the Research Institute on Informatics and Telecommunication IRSIT from 1987-1990, and President of the Institut in 1990-1994. He is now Director of Signal Processing Research Laboratory LSTS at ENIT, and is in charge of Control and Signal Processing Master degree at ENIT.
Pr Ellouze is IEEE fellow since 1987; he directed multiple Masters and Thesis and published over 200 scientific papers both in journals and proceedings. He is chief editor of the scientific journal Annales Maghrébines de l’Ingénieur. His research interest include neural networks and fuzzy classification, pattern recognition, signal processing and image processing applied in biomedical, multimedia, and man machine communication.
E-mail:N.Ellouze@enit.rnu.tn

Self-Healing Strategy for Dynamic Security Assessment and Power System Restoration

T.A. Ramesh Kumaar [1] and Dr.I.A.Chidambaram[2]

[1]Annamalai University, Department of Electrical Engineering
[2]Department of Electrical Engineering, Annamalai University,
[1]Assistant Professor, [2]Professor
tarpagutharivu@gmail.com, driacdm@yahoo.com

Abstract: *This paper presents a self-healing strategy based methodology for power system dynamic security assessment and to restore the power system to its normal operating conditions. A Self-Healing approach is trained to determine the performance indices, which allows the classification of the operative system state security. The load flow inputs are compared by signals then indicates the disturbance severity and its ability to re- establish an acceptable equilibrium point. In this self-healing approach, the system is adoptively divided into islands with consideration of quick restoration. Then load shedding scheme is tested for IEEE 14-bus system. The dynamic security assessment and Islanding results are provided to highlight the overall accuracy and suitability of the approach.*

Keywords: *Self-Healing, Islanding, Dynamic Security Assessment, Load shedding, Power System Restoration.*

1. Introduction

Security assessment is an important studies carried out in an energy management system to determine the security and stability of the system under unfrozen contingencies. In an analysis performed to determine whether, and what extent the power system is safe without violating from serious interference to its operation. A system said to be secure if there is no violation of operating limit of power system components. Security assessment can be classified into three types namely, steady state, Dynamic state and Transient state respectively. All the three need to be performed on-line, with dynamic security being more complex and difficult than steady state and transient security [1-4] The former characterize the steady state behavior of the system under an outage, while transient security deals with transient disturbance upto 2sec. whereas dynamic security assessment deals with long-term behavior of the system of few seconds to minutes that is from the time system is transient secure to the time of system reaches steady state[5] Which first classifies on operating state as steady state secure or steady state insecure. The second stage classifies the steady state secure states as transiently secure or transiently insecure. The third stage classifies the steady state security and transient secure state into dynamically secure or insecure. Generally any deregulation problem introduces several new economic objectives for operation. As power transactions increase, weak connections, unexpected events, and hidden failures in protection systems, human errors and other reasons may cause the system to lose balance and even lead to catastrophic failures. The complexity of the power system network is increasing. Dynamic Security Assessment (DSA) generally needs detailed modeling of power system. That is detailed model of synchronous machine, exciter, stabilizing networks, governor's and turbines are required for DSA [13].Generator ranking usually done for rescheduling to improve the system security [2].

A system is said to transiently stable if the clearing time of fault is less than the critical clearing time. The critical clearing time is a complex function of pre-fault conditions, its location, types and protective relaying strategy. The likelihood of blackouts has been increasing because of various physical and economic factors. These include, the demand for larger power transfers over longer distances, insufficient investment in the transmission system, exacerbated by continued load growth, huge swings in power flow patterns from one day to the next that render classic off-line planning studies ineffective. These factors result in larger operational footprints and greater demands on the operator to deal with smaller error margins and shorter decision times. These circumstances have created a less reliable operating environment by pushing power systems close to their physical limits. Such an environment requires more intensive on-line analyses to better -coordinate controls. Wide-Area Monitoring and control tools, e.g., Phasor Measurement Units (PMU) and Flexible AC Transmission System (FACTS) devices, and distributed generation and storage devices are the primary technologies used to address such problems. A self-healing concept is expected to respond to threats, material failures, and other destabilizing influences by preventing the spread of disturbances [5]. This requires the following capabilities, timely recognition of impending problems redeployment of resources to minimize adverse impacts, a fast and coordinated response for evolving disturbances, minimization of loss of service under any circumstances, minimization of time to reconfigure and restore service

This paper presents a logic of designing a self-healing strategy after large disturbances. When a power system is subjected to large disturbances, such as simultaneous loss in power generation by several generating units or major failure in power transfer through transmission lines, and the vulnerability analysis indicates that the system is approaching a catastrophic failure, control actions need to be taken to limit the extent of the disturbance. In this approach, the system is separated into smaller islands at a slightly reduced capacity. The basis for forming the islands is to minimize the generation-load imbalance in each island, thereby facilitating the restoration process. Then, by exploring with a sophisticatedly designed load shedding scheme based on the rate of frequency decline, the extent of the disruption is limited and are able to restore the system rapidly. The method has two stages. In the first stage a load shedding is adopted. The second stage includes a restoration process [6]. Load restoration depends on the droop characteristic of the generators and maximum power capability of the generators.

2. Dynamic Security Assessment

Most commonly, contingencies result in relay operations that are designed to protect the system from faults or abnormal conditions. Typical relay operations result in the loss of a line, transformer, generator, or major load. When changes occur, the various components of the power system respond and hopefully reach a new equilibrium condition that is acceptable according to some criteria. Mathematical analysis of these responses and new equilibrium condition is called Security Analysis. Due to the nature of the disturbance and the set up of the power system network, there are two main elements to power system security assessment, static security assessment and dynamic security assessment [7]. Static security assessment is usually performed prior to dynamic security assessment. If the analysis evaluates only the expected post disturbance equilibrium condition (steady-state operating point), this is called Static Security Assessment (SSA). Static Security Assessment also applies the assumption that the transition to new operating conditions has taken place. Its main objective is to ensure that the operating conditions are met in the new operation conditions. Static Security Assessment basically ignores the dynamics of the system and synchronization of the power system network during the process of transiting into post fault condition state remains unknown. This level of assessment will only be able to give a rough estimation of the post contingency stability. It also made a dangerous assumption that the system remains stable in the event of fault.

If the analysis evaluates the transient performance of the system as it progresses after the disturbance, this is called **Dynamic Security Assessment** (DSA). Dynamic Security Assessment is an evaluation of the ability of a certain power system to withstand a defined set of contingencies and to survive the transition to an acceptable steady-state condition. DSA is required due to the constant variation of loads and change in the behaviour of the power system. Gradual changes such as load variations over the day are normal and can be anticipated to some extent. In the event of unexpected loss of a generating plant due to equipment failure, there will be a large impact on both the user and the supplier. These disruptive changes will cause the system variables such as frequency and voltage to oscillate regardless of how small the disturbance is. If the system is secured, these oscillations will decay and be damped out eventually. Otherwise, the oscillation of the frequency and voltage will grow to the extent of shutting down the generator.

Very early power systems were often separate and isolated regions of generators and loads. As systems became larger and more interconnected, the possibility of disturbances propagating long distances increased. The concept of the preventive (normal), emergency, and restorative operating states and their associated controls were introduced for reliable operation of power system (Fig. 1) [2].

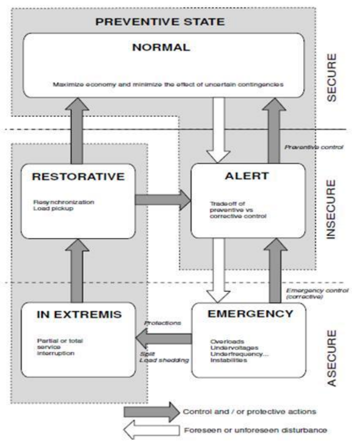


Figure 1: **The operating states and transitions for power systems**

a) Off-Line DSA

In off-line DSA analysis, detailed time-domain stability analysis is performed for all credible contingencies and a variety of operating conditions. In most cases, this off-line analysis is used to determine limits of power transfers across important system interfaces. These limits then are used in an operating environment that is hopefully not significantly different from those conditions considered. Since the analysis is performed off-line, there is not a severe restriction on computation time and therefore detailed analysis can be done for a wide range of conditions and contingencies. These studies include numerical integration of the models for a certain proposed power transfer condition and for a list of contingencies typically defined by a faulted location and specified fault-clearing time (based on known relay settings).The trajectories of the simulation are analyzed to find if voltage transients are acceptable, and to verify whether the transient stability is maintained during the specified fault-clearing time.

If the results for one level of power transfer are acceptable for all credible contingencies, the level of proposed power transfer is increased and the analysis is repeated. This process continues until the level of power transfer reaches a point where the system cannot survive all of the credible contingencies. The maximum allowable transfer level is then fixed at the last acceptable level, or reduced by some small amount to provide a margin that would account for changes in conditions when the actual limit is in force.

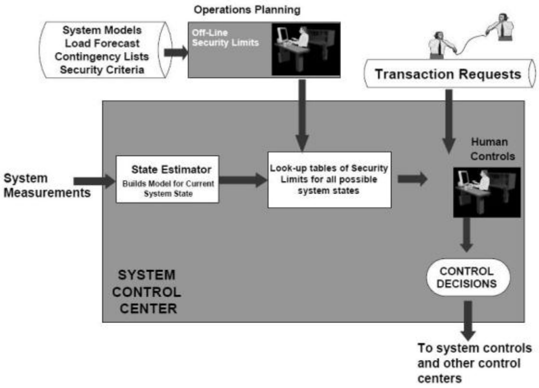


***Figure 2:* Traditional Off-line Security Assessment**

b) On-Line DSA

On-line DSA is used to supplement (or update) off-line DSA to consider current operating conditions. A basic on-line DSA framework includes essentially two steps. The first involves a rapid screening process to limit the number of contingencies that must be evaluated in detail. This rapid screening process might consist of some direct method that avoids long numerical integration times. In addition to giving fast stability evaluation, these methods inherently include a mechanism for assessing the severity of a contingency. That is, if a system is determined to be stable, the direct methods also provide an indication of ''how stable'' the system is. This indication usually takes the form of an 'energy margin'. On-line dynamic security assessment (DSA) is a promising solution.

Network topology and operating conditions are modeled in real-time as seen by the operator in the control center. System security and limits are calculated using this data Results more accurately characterize the existing system conditions than the off-line planning cases allow the operator to evaluate the impact of certain operating decisions with changing operating conditions[8]. Enhances the decision making capability of an operator and significantly reduces the risk of cascading blackouts by more accurately evaluating limits.

With the increase in transactions on the bulk power system there is a critical need to determine transient security in an on-line setting and also perform preventive or corrective control if the analysis indicates that the system is insecure.

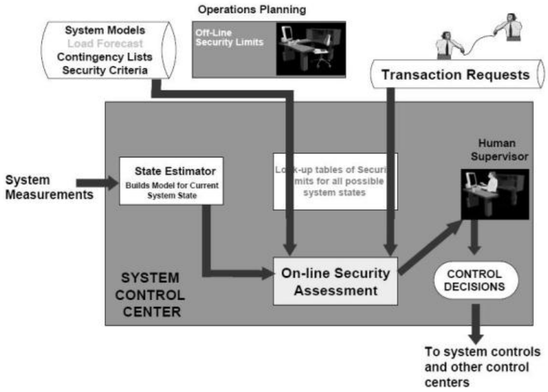


***Figure 3:* Traditional On-line Security Assessment**

In recent years, the industry has seen the development of large generation projects at concentrated areas of available fuel supplies .The stability properties of the system have been drastically altered, while the new ''nonutility'' plants are not cognizant of their impact on system stability. In this environment, stability issues may and will affect available transfer capability. Stability problems may not happen frequently, but their impact, when they do happen, can be enormous. Most of the time, off-line studies are performed to determine conservative limits.

3. Mathematical Model of Dynamic Security Assessment

The dynamic behavior of multi-machine power system is described by the detailed modeling of all the elements of the power system [8].

$$M\frac{d^2\delta}{dt^2} + D\frac{d\delta}{dt} + P_{ei} = P_{mi} \quad (1)$$

$$\frac{d\delta_0}{dt} = \omega_0 \quad (2)$$

$$P_{ei} = E_i \sum_{i=1}^{N_j} E_j \left[G_{ij} \cos(\delta_0 - \delta_1) + B_{ij} \sin(\delta_i - \bar{\delta j})\right] \quad (3)$$

Where i= 1,2,…………….. N_g, D_i are the inertia and damping constant of the Ith generator; P_{mi} mechanical input to the i_{th} generator; E_j is the EMF behind X'_{di} of the i_{th} generator; $G_{ij,}$ B_{ij} are the real and imaginary parts of the admittance matrix of the of the redused system, X'_{dt} is the transient reactance of the i_{th} generator; N_g is the number of synchronous generator in the system. Eq (3) provide's the general relation for dynamic behavior of a multi-machine power system used for both dynamic and transint security assessment. Modification of Eq(3) for dynamic security assessment could result as

$$\frac{Md^2\delta}{dt^2} = P_{mi} - P_{ei} = P_m - P_{\max,i} \quad (4)$$

4. Islanding Control

In this process the time-scale method is employed for determining the groups of the generators with coherency. This method considers the structured characteristic of the various generators and finds the strong and weak couplings. Through the selection of the time-scale option, the coherent group of generators can be obtained on any power systems and an automatic islanding program is also developed.

a. Islanding

In determining the islands, the inherent structured characteristic of the system should be considered. In addition, the choice of these islands should not be disturbance dependant. The time-scale method employed is on the application of the singular protection method in power systems [7, 9]. This method assumes the state variable of the n th order septum and are divided into x slow states, y and (n-x) fast stated f, in which the x slowest states represent x groups with coherency. The use provides the estimate for the number of groups. However, the automatic islanding program take into account the mismatch between generation, load and availability of tie lines to form islands and approximately combined groups when islands cannot be formed. For a time-scale method both linearized and non-linear power system models can be used. The basic classical second-order electromechanical model of an n-machine power system was adopted [8].

$$\delta_j = \Omega (N_j - 1) \qquad (5)$$

$$2H_j N_j = -A_j (N_j - 1) + (P_{mj} - P_{ej}) \quad j = 1,2,\ldots.n \qquad (6)$$

Where,

N_j= speed of machine j , in p. u.

Ω = base frequency, in radians per second

δ_j = rotor angle of machine j in radians.

P_{mj} = mechanical input power of machine j , in p. u.

P_{ej}= electrical input power of machine j , in p. u.

H_j = inertia constant of machine j , in per seconds.

D_j = damping constant of machine j , in per seconds.

An automatic islanding program based on description method [2] was developed in order to search for the optimum cut sets after the grouping of information is done. The optimum cut set is obtained considering the least generation-load imbalance. The approach begins with the characterization of the network structure or connectivity using the adjacent link table data structure [7], then through a series of reduction processes, the program forms a small network and performs an exhaustive search on its to get all the possible cut sets.

With the information of the coherent groups of generators and the exact locations of where to form the islands, is deployed to form the islands [5,10]. The relay used for islanding requires the local measurements and makes tripping decision using settings based on various offline contingency simulations. It shows much better performance than the conventional out of step relay, which is actually the impedance relay. Besides the impedance, the new relay uses the information of the rate of change of the impedance or resistance and gets better results in practice. Different switching lines make sure different corrective control actions are taken based on the level of the seriousness of the disturbance. When a fault trajectory enters into the range defined by the switching lines, the tripping action will takes place.

5. Load Shedding and Restoration

a. Load Shedding

Load shedding is an emergency control operation and various load shedding schemes have been used in the industry. Most of these are based on the frequency decline in the system. By considering only one factor, namely the frequency, in these schemes the results were less accurate. Any part of a power system will begin to deteriorate if there is an excess of load over available generation. The prime movers and their associated generators begin to slow down as they attempt to carry the excess load. Tie-lines to other parts of the system, or to other power systems across a power pool, attempt to supply the excess load. This combination of events can cause the tie-lines to open from overload or the various parts of the systems to separate due to power swings and resulting instability.

The result may be one or more electrically isolated islands in which load may exceed the available generation. Further, the drop in frequency may endanger generation itself. While a hydro-electric plant is relatively unaffected by even a ten percent reduction in frequency, a thermal generating plant is quite sensitive to even a five percent reduction. Power output of a thermal plant depends to a great extent on its motor driven auxiliaries such as boiler feed water pumps, coal pulverizing and feeding equipment, and draft fans. As system frequency decreases, the power output to the auxiliaries begins to fall off rapidly which in turn further reduces the energy input to the turbine generator.

The situation thus has a cascading effect with a loss of frequency leading to a loss of power which can cause the frequency to deteriorate further and the entire plant is soon in serious trouble. An additional major concern is the possible damage to the steam turbines due to prolonged operation at reduced frequency during this severe overload condition. To prevent the complete collapse of the island, under frequency relays are used to automatically drop load in accordance with a predetermined schedule to balance the load to the available generation in the affected area. Such action must be taken promptly and must be of sufficient magnitude to conserve essential load and enable the remainder of the system to recover from the under frequency condition. Also, by preventing a major shutdown, restoration of the entire system to normal operation is greatly facilitated and expedited. Where individual operating utility companies are interconnected, resulting in a power pool, it is essential that system planning and operating procedures be coordinated to provide a uniform automatic load shedding scheme.

Although the earlier schemes were considerably successful, they lacked efficiency. They shed excessive load which was undesirable as it caused inconvenience to the customers. Improvements on these traditional schemes led to the development of load shedding techniques based on the frequency as well as the rate of change of frequency. This led to better estimates of the load to be shed thereby improving accuracy. The measurement and recording equipments for analysis have undergone developments. Usually, Phasor Measurement Units, PMU are used for measuring the real time data. The load shedding is on a priority basis, which means shedding less important loads, while expensive industrial loads are still in service. Thus the economic aspect plays an important part in load shedding schemes.

Usually, a step wise approach is incorporated for any scheme. The total amount of load to be shed is divided in discrete steps which are shed as per the decline of frequency. For example, when the frequency decreases to the first pick up point a certain predefined percentage of the total load is shed. If there is a further decay in frequency and it reaches the second pickup point, another fixed percentage of the remaining load is shed. This process goes on further till the frequency increases above its lower limit. Increasing the number of steps reduces the transients in the systems. The amount of load to be shed in each step is an important factor for the efficiency of the scheme. By reducing the load in each step the possibility of over shedding is reduced.

b. Load Shedding Techniques

Different methods for load shedding have been developed by many researchers[11-17]. Currently there are various load shedding techniques used in the power industry worldwide. These conventional load shedding schemes are discussed first and latter includes a discussion on under frequency and under voltage load shedding techniques which are proposed by researchers and are yet to be incorporated by the power industry. Industry Techniques for Load Shedding, some of the conventional industry practices for load shedding [11] has definite load shedding requirements. The load serving members must install under frequency relays which trip around 56% of the total load in case of an automatic load shedding scheme. It has nine steps for load shedding. The pickup frequencies are 59.7 Hz for the first step and 59.1 Hz for the last step. The frequency steps, time and the amount of load to be shed is in the table 1. The steps from A to F follow the shedding of load as per a downfall in the frequency. The steps L, M and n are peculiar since they indicate load shedding during a frequency rise. The purpose of this is to avoid stagnation of frequency at a value lower than the nominal. Thus if the frequency rises to 59.4 Hz and continues to remain in the vicinity for more than 10 seconds, 5% of the remaining load is shed so that the frequency increases and reaches the required nominal value. The effectiveness of this scheme is tested every five years by the a specific Stability Working Group (SWG). Based on this scheme certain frequency targets are established. The frequency must remain above 57 Hz and should recover above 58 Hz in 12 seconds. In addition, the frequency must not exceed 61.8 Hz due to excessive load shedding.

Another scheme which can be implemented incorporates both automatic as well as manual load shedding. If the frequency goes lower than 59.5 Hz, the status of the generators is noted. If sufficient load has not been shed further steps of load shedding are undertaken. The immediate action on account of a decision to shed load is to inform the power sector participants regarding the suspension of the hour ahead or the day ahead markets due

***Table 1:* Steps performed for load shedding steps**

Under Frequency Load Shedding Step	Frequency (hertz)	Time Delay (seconds)	Amount of load to be shed (% of the total load)	Cumulative Amount of load (%)
A	59.7	0.28		9
B	59.4	0.28	7	16
C	59.1	0.28	7	23
D	58.8	0.28	6	29
E	58.5	0.28	5	34
F	58.2	0.28	7	41
L	59.4	10	5	46
M	59.7	12	5	51
N	59.1	8	5	56

Another stepwise load shedding procedure can also be adopted. In this, the generator protection is also considered when establishing the frequency set points and the amount of load to be shed at each step. The generator protection relay is set to trip the generators after the last load shedding step. The scheme has the following requirements. They have three basic load shedding steps as shown in table 2. The number of load shedding steps can be more than three provided the above schedule is maintained. This scheme is a distributed scheme as it sheds loads from distributed locations as opposed to centralized schemes. The loads tripped by this scheme are manually restored.

***Table 2:* Alternative approach to perform load shedding**

Amount of load to be shed (% of total load)	Frequency set points (Hertz)
10%	59.3
20%	58.9
30%	58.5

Time delay settings are applied to the under frequency relays with a delay of 0.1 seconds. These relays are required to maintain ± 0.2 Hz stability in set point and ± 0.1 seconds in time delay. The styles and manufacturing of these relays is required to be identical to obtain approximately similar response rates. An Under frequency load shedding database can also be maintained with the informations related to the load shed at each step, the total number of steps and records every load shedding event.

Under voltage load shedding scheme[15] have also been developed to protect their system against fast and slow voltage instability. The scheme has been designed for two voltage instability scenarios. The first one is associated with the transient instability of the induction motors within the first 0-20 seconds. The second one is up to several minutes. This collapse may be caused due to the distribution regulators trying to restore voltages at the unit substation loads. According to the topology of the system the Imported Contingency Load Shedding Scheme (ICLSS) has been developed. This scheme uses distribution SCADA computers and consists of PLCs. The Albuquerque area system has been used for testing this method. Thirteen load shedding steps were required to correct the frequency deviation.

Another load shedding scheme based on under frequency relays can be performed in three steps. In case the frequency decline cannot be curbed in three steps, additional shedding steps are carried out. Other actions may include opening lines, creating islands. These actions are carried out once the frequency drops below 58.7 Hz. The scheme is inherently automatic but in case it fails to achieve successful frequency restoration, manual load shedding is incorporated.

Besides the above various load shedding schemes, certain techniques are also employed [12] having three steps in load shedding. In the first step, up to 10% of the load but no more than 15% is required to be shed. In the second step up to 20% of the load but no more than 25% is required to be shed. The third step requires up to 30% but not more than 45% of the existing load to be shed. This scheme is based on the decline of frequency and sheds load as the frequency decreases below its nominal value. The proportion of the load selected for shedding is based on the average of three months of load data and is annually updated. The first three steps of load shedding are set up at three manned substation or substations with remote supervisory control. The amount of load seems to be lesser when the load to be shed is evenly distributed over the system. A new eleven step scheme has been recently suggested. An automatic under frequency load shedding scheme is used [12] to minimize the load to be shed based on the severity of load unbalance and the availability of spinning reserves. It is based on the declining average system frequency.

A similar scheme is incorporated which has established a five stage load shedding scheme with the first pick up frequency of 49.5 Hz (on a 50 Hz system) and the pickup frequency of the last stage is 47.7 Hz. An efficient under frequency load shedding scheme [14] is reviewed with the operating guidelines for every five years. The total load it sheds is up to 25% of the system load. Similar to the basic under frequency scheme it constitutes of three steps. It's pickup frequency for step one is 59.3 Hz as shown in table 3.

Table 3: **Frequency threshold with Load Shedding Scheme**

Frequency Threshold	Load Relief
59.3 Hz	5% of the System Load (Total 5%)
58.9 Hz	An additional 10% of the System Load (Total 15%)
58.5 Hz	An additional 10% of the System Load (Total 25%)

An intelligent adaptive load shedding scheme proposed [16] divides the system into small islands when a catastrophic disturbance strikes it. Further, an adaptive load shedding scheme is applied to it based on the rate of change of frequency decline. Under frequency load shedding mainly sets up relays to detect frequency changes in the system. As soon as the frequency drops below a certain value a certain amount of load drops, if the frequency drops further, again a certain amount of load is dropped. This goes on for a couple of steps. The amount of load to be shed and the location of the load to be shed are predetermined.

Terzija [13] had developed under frequency load shedding in two stages. During the first stage the frequency and rate of frequency changes of the system are estimated by non-recursive Newton-type algorithm. In the second algorithm, the magnitude of the disturbance is estimated using the simple generator swing equation.

In another approach Thalassinakis et al [17] have developed a method which uses the Monte Carlo simulation approach for the settings of load shedding under frequency relays and selection of appropriate spinning reserve for an autonomous power system. The settings of the under frequency relays are based on the four parameters; the under frequency level, rate of change of frequency, the time delay and the amount of load to be shed. Three sets of system indices are defined. These sets are for the purpose of comparisons between load shedding strategies. The three aspects of the power systems that were developed in the simulation were, Operation of the power system as performed by the control centre. Primary regulation of the generating units may not be sufficient to control the various parameters after the failure of a generating unit has occurred. Secondary regulation utilizes the spinning reserves. Three different cases of comparing the spinning reserves with the load mismatch are considered. One, when the spinning reserve is sufficient or greater. Thus the load can be restored immediately. Second, when the spinning reserve is slightly insufficient and the rapid generating units will require a certain amount of time to be started. Thus it will be 10-20 minutes before the load can be completely restored. Third, the spinning reserves are insufficient and there are not enough rapid generating units thus implying that the load will not be restored for a considerably long period of time.

c. Load Restoration

Power system restoration is a unique event. However, there are certain goals, steps and aspects that are common to all restoration procedures. Fig. 4 shows all the aspects of power system restoration. They involve almost all aspects of power system operation. If a load shedding program has been successfully implemented, the system frequency will stabilize and then recover to 50 Hz. This recovery is assisted by governor action on available spinning reserve generation, or by the addition of other generation to the system. The recovery of system frequency to normal is likely to be quite slow and may extend over a period of several minutes. When 50 Hz operation has been restored to an island, then interconnecting tie lines with other systems or portions of systems can be synchronized and closed in. As the system frequency approaches the normal 50 Hz, a frequency relay can be used to automatically begin the restoration of the load that has been shed.

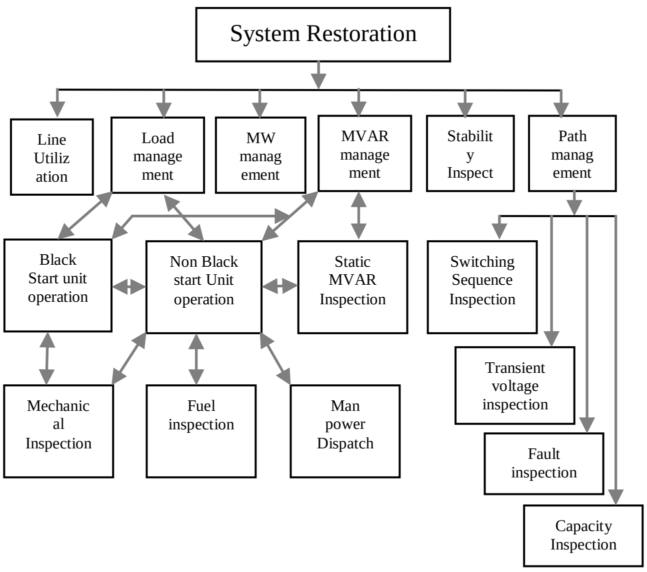


Figure 4 Power System Restoration Aspects

The amount of load that can be restored is determined by the ability of the system to serve it. The criterion is that the available generation must always exceed the amount of load being restored so that the system frequency will continue to recover towards 50 Hz. Any serious decrease in system frequency at this point could lead to undesirable load shedding repetition, which could start a system oscillation between shedding and restoration. This would be a highly undesirable condition. The availability of generation, either locally or through system interconnections, determines whether or not the shed load can be successfully restored. Therefore, a load restoration program usually incorporates time delay, which is related to the amount of time required to add generation or to close tie-lines during emergency conditions. Also, both the time delay and the restoration frequency set points should be staggered so that the entire load is not reconnected at the same time.

Reconnecting loads on a distributed basis also minimizes power swings across the system and thereby minimizes the possibility of initiating a new disturbance. In general, wide frequency fluctuations and the possibility of starting a load shedding/restoration oscillation can be greatly minimized if the amount of load restored per step is small and the spinning reserve generation available is adequate. There should also be adequate time delay provided between load restoration steps to allow the system to stabilize before an additional block of load is picked up. In general, wide frequency fluctuations and the possibility of starting a load shedding/restoration oscillation can be greatly minimized if the amount of load restored per step is small and the spinning reserve generation available is adequate. Horowitz et al [18] suggests spinning reserve availability at least three times the size of the load to be restored at any given step. There should also be adequate time delay provided between load restoration steps to allow the system to stabilize before an additional block of load is picked up.

6. Proposed Load Shedding and Restoration

a. Load Shedding Procedure

Consider a region consists of *m* generators, the electromechanical equation for the generators can be written as

$$\frac{J_k}{\omega_s}\frac{d\omega_k}{dt} = P_{m,k} - P_{e,k} - P_{d,k}(p.u.) \quad k = 1,\dots,m \tag{7}$$

The electrical power produced by the each generator can be found using the reduction matrix method. The complex produced power for each generator can be of

$$S_g = \quad Diag\{V_g \cdot\dot{}\, g^*\} = [S_1 \cdots S_k \cdots S_m] \tag{8}$$

Therefore, the electrical power needed, $P_{e,k}$, needed in Equation (7) is the real part of S_k as

$P_{e,k}$ = Real $\{S_{e,k}\}$

The damping power for Generator k can be obtained from

$$P_{D,k} = D_k\, \omega_k \qquad k = 1, \ldots, m \tag{9}$$

The produced electric powers of the generators are calculated from the power flow equation at each time step of the differential equations being solved. Also the differential equation corresponds to excitation can be expresses as

$$\frac{d_{E,K}}{dt} = \frac{K_{B,k}}{T_{E,k}}\left(V_{t,k} - V_{ref,k}\right) - \frac{E_k}{T_{E,k}} \quad k = 1, \ldots, m \tag{10}$$

When the region is connected, the input mechanical power of the generators is equal to power associated with the output signal of the governor. However, when the region becomes an island, i.e. is disconnected from the network, it goes into frequency control mode.

In this situation the input mechanical power of the Generator k,$P_{m,k,}$,is considered to be related to the power associated with the governor of Generator K, $P_{gov,k}$, by a first order differential equation as:

$$\frac{d_{m,k}}{dt} = \frac{P_{gov,k} - P_{m,k}}{T_{g,k}} \qquad k = 1, \ldots, m \tag{11}$$

For Generator K the output power associated with the output signal of the governor, $P_{gov,k}$, is related to the frequency, f_k , the output mechanical power associated with the output signal of the governor , $P_{gov,k}$, needs to be expressed in terms of frequency as

$$P_{gov,k} = \frac{R_k\, P_{\max k} + f_k}{R_k} \qquad k = 1, \ldots, m \tag{12}$$

Consider the case that the region is connected to the main network through a tie line. The region imports electrical power from the main network. If the circuit breaker of the tie-line is tripped, the region in the network then becomes separated from the network. After disconnecting the region from the main network, the region is referred to as the electrical island. As the load is typically less than the generation in the island at the time of separation, the frequency in the island starts to drop.

According to the shedding procedure the loads are shed until the frequency drop is stopped and starts to rise. The shedding procedure is based on a predefined shedding policy. Any policy for load shedding such as prioritized load can be considered in the shedding procedure. Considering the case that the total rated power of the generators in the island can meet the total load in the island, eventually the frequency of the island becomes stable and the region keeps its integrity by initially removing some of the loads in the region and gradually restoring them. Fig. 5 shows the flowchart of shedding process for shedable load i.

The frequencies f_{island} and f_{shedi} are the island frequency and shedding frequency of shedable load i. The ordering in load shedding (if any) might be coming from a shedding policy set by the utility in the island. When the frequency stabilized, i.e. there is no more reduction in the island frequency, then due to predefined droop characteristic of each generator, the produced power of the generators increase. Thus, the frequency rises.

b. Load Restoration Procedure

In the proposed algorithm for restoration process, the droop characteristics of the generators need to be adjusted so that the rated load is supplied at the frequency which is more than *50 Hz*.

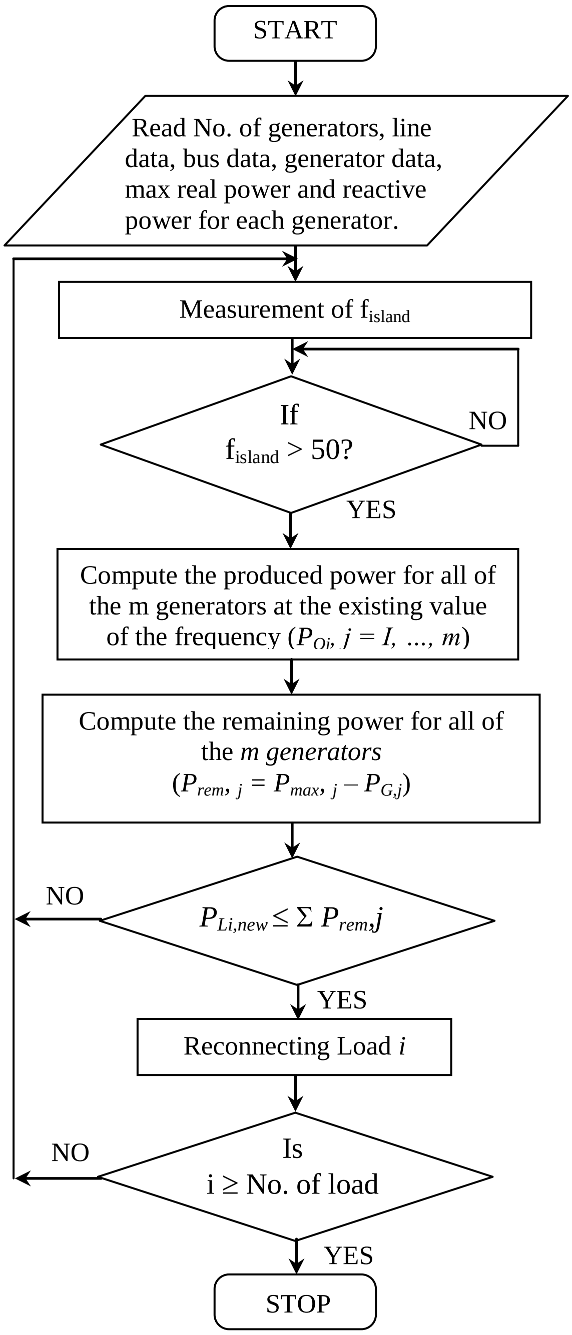


***Figure 6*: The Flowchart for the Restoration Procedure**

The restoration process comes to effect when the frequency goes more than *50 Hz.* In the simulation, at each time step of the restoration process, the produced power of each generator is computed according to the frequency of the island at that step using the droop characteristic of each generator. Then, for each generator the produced power is subtracted from the maximum power of the generator and the result is referred as the remaining power, $P_{rem,1}$, for that generator. The restoration of a specific load will be performed if the power of the of the incoming load, P_L , is less than or equal to the total remaining of the generators. In the real power system the droop characteristic of the generators are to be chosen to be similar or, alternatively, for each load the droop characteristic of all of the generators are provided. Therefore the loads can asses that which time suits it to reconnect to the load. Also the restoration policy may incorporate the prioritized loads in restoration.

7. Simulation Results

Case: 1

The load shedding scheme is tested on a 14-bus, 5-generator system. The following lines are tripped:

1. Bus 6–Bus 5;
2. Bus 9–Bus 4;
3. Bus 7–Bus 4.

After islanding, the system is divided into two islands shown as in Fig. 7. The bus 1, 2,3,4,5 forms the island which is generation rich.

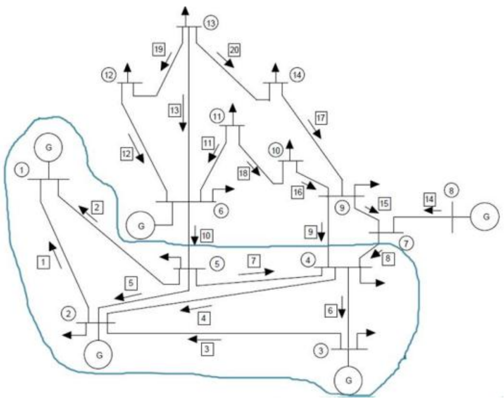


Figure 7 Two islands for IEEE 14 bus system

Table 4: Load flow of IEEE 14 Bus system

No of Iteration 116

Bus No.	Voltage Magnitude	Voltage Angle	Real Power	Reactive Power
1	1	0	2.3515	0.2169
2	0.9601	-5.3281	0.1830	-0.5510
3	0.9681	-14.9735	-0.9420	0.4000
4	0.9501	-11.7966	-0.4780	0
5	0.9525	-10.0902	-0.0760	-0.0160
6	1.0168	-16.5429	-0.1120	0.2400
7	0.9871	-15.1529	0	0
8	1.0282	-15.1529	0	0.2400
9	0.9709	-16.9200	-0.2950	-0.1660
10	0.9710	-17.1672	-0.0900	-0.0580
11	0.9898	-16.9818	-0.0350	-0.0180
12	0.9988	-17.4876	-0.0610	-0.0160
13	0.9915	-17.5197	-0.1350	-0.0580
14	0.9605	-18.3375	-0.1490	-0.0500

Frequency of the system 49.9842

Table 5: Load flow for power System Island consists of 3 Generator buses and 2 Load buses

No of Iteration: 31

Bus No.	Voltage Magnitude	Voltage Angle	Real Power	Reactive Power
1	1.0000	0	1.3849	-0.1006
2	0.9810	-3.2879	0.1830	-0.5510
3	0.9972	-11.0980	-0.9420	0.4000
4	0.9868	-6.9533	-0.4780	0
5	0.9891	-5.5758	-0.0760	-0.0160

Governor Output power at the base frequency: 2.5061, 2.5006

Governor Output power after Islanding: 2.5295, 2.5240

Table 6: Generation at the buses after Islanding:

Bus No	P Generated (p.u)
1	2.3200
2	0.4000
3	0
4	0
5	0

Table 7: Load at the Buses after Islanding

Bus No	P Load (p.u)
1	0
2	0.2170
3	0.9420
4	0.4780
5	0.0760

Frequency of the system after Islanding: 50.4675

Governor Output power after Load Restoration: 2.5060, 2.5005

Table 8: Generation at the buses after Load Restoration:

Bus No	P Generated (p.u)
1	1.4840
2	0.2300
3	0
4	0
5	0

Frequency of the system after Load Restoration = 49.9978Hz

Case: 2

The bus 2,3,4,5 forms the island which is Load rich.

Table 9: Load flow for power System Island consists of 2 Generator buses and 2 Load buses.

No of Iteration: 20

Bus No	Voltage magnitude	Voltage angle	Real Power	Reactive Power
1	1.0000	0	1.5512	-0.5165
2	1.0130	-8.3663	-0.9420	0.4000
3	0.9986	-5.0495	-0.4780	0
4	0.9983	-4.2343	-0.0760	-0.0160

Governor Output power at the base frequency: 2.5006, 2.5016.

Governor Output power after Islanding: 2.4718, 2.4918.

Table 10: Generation at the buses after Islanding:

Bus No	P Generated (p.u)
2	0.4000
3	0
4	0
5	0

Table 11: Load at the buses after Islanding:

Bus No	P Load (p.u)
2	0.2170
3	0.9420
4	0.4780
5	0.0760

Frequency of the system after Islanding: 49.4244Hz

Governor Output power after Load Restoration: 2.5008, 2.5108

Table 12: Load at the buses after Load Restoration:

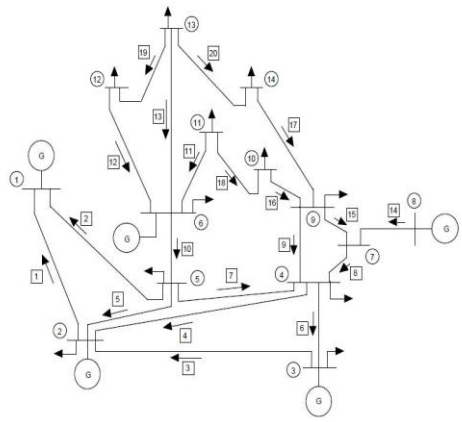
Bus No	P Load (p.u)
2	0.0564
3	0.2449
4	0.1243
5	0.0198

Frequency of the system after Load Restoration: 50.0034Hz

8. Conclusion

In this paper, a self-healing scheme for large disturbances with concentration on a new load shedding scheme is described. A method is presented for survival of an electrical island when it becomes separated from the main network. The method has two stages. In the first stage a load shedding is adopted. The second stage includes restoration process. Results indicate that using this method successfully prevents the island from blackout and restore the system when it becomes separated from the network. The scheme is tested on the 14-bus sample system and shows very good performance.

9. Appendix



- Transmission Line #'s

- Bus #'s

Figure 8: IEEE 14-bus test system one line diagram

Table 13: Generator Data

Generator Bus No.	1	2	3	4	5
MVA	615	60	60	25	25
x_l (p.u.)	0.2396	0.00	0.00	0.134	0.134
r_a (p.u.)	0.00	0.0031	0.0031	0.0014	0.0041
x_d (p.u.)	0.8979	1.05	1.05	1.25	1.25
x'_d (p.u.)	0.2995	0.1850	0.1850	0.232	0.232
x''_d (p.u.)	0.23	0.13	0.13	0.12	0.12
T'_{do}	7.4	6.1	6.1	4.75	4.75
T''_{do}	0.03	0.04	0.04	0.06	0.06
x_q (p.u.)	0.646	0.98	0.98	1.22	1.22
x'_q (p.u.)	0.646	0.36	0.36	0.715	0.715
X''_q (p.u.)	0.4	0.13	0.13	0.12	0.12
T'_{qo}	0.00	0.3	0.3	1.5	1.5
T''_{qo}	0.033	0.099	0.099	0.21	0.21
H	5.148	6.54	6.54	5.06	5.06
D	2	2	2	2	2

Table 14: Bus Data

Bus No.	P Generated (p.u.)	Q Generated (p.u.)	P Load (p.u.)	Q Load (p.u.)	Bus Type*	Q Generated max.(p.u.)	Q Generated min.(p.u.)
1.	2.32	0.00	0.00	0.00	2	10.0	-10.0
2.	0.4	-0.424	0.2170	0.1270	1	0.5	-0.4
3.	0.00	0.00	0.9420	0.1900	2	0.4	0.00
4.	0.00	0.00	0.4780	0.00	3	0.00	0.00
5.	0.00	0.00	0.760	0.0160	3	0.00	0.00
6.	0.00	0.00	0.1120	0.0750	2	0.24	-0.06
7.	0.00	0.00	0.00	0.00	3	0.00	0.00
8.	0.00	0.00	0.00	0.00	2	0.24	-0.06
9.	0.00	0.00	0.2950	0.1660	3	0.00	0.00
10.	0.00	0.00	0.0900	0.0580	3	0.00	0.00
11.	0.00	0.00	0.0350	0.0180	3	0.00	0.00
12.	0.00	0.00	0.610	0.0160	3	0.00	0.00
13.	0.00	0.00	0.1350	0.0580	3	0.00	0.00
14.	0.00	0.00	0.1490	0.0500	3	0.00	0.00

Table 15: Line Data

From Bus	To Bus	Resistance (p.u.)	Reactance (p.u.)	Line charging (p.u.)	Tap ratio
1	2	0.01938	0.05917	0.0528	1
1	5	0.5403	0.22304	0.0492	1
2	3	0.04699	0.19797	0.0438	1
2	4	0.05811	0.17632	0.0374	1
2	5	0.5695	0.17388	0.034	1
3	4	0.6701	0.17103	0.0346	1
4	5	0.01335	0.4211	0.0128	1
4	7	0.00	0.20912	0.00	0.978
4	9	0.00	0.55618	0.00	0.969
5	6	0.00	0.25202	0.00	0.932
6	11	0.099498	0.1989	0.00	1
6	12	0.12291	0.25581	0.00	1
6	13	0.06615	0.13027	0.00	1
7	8	0.00	0.17615	0.00	1
7	9	0.00	0.11001	0.00	1
9	10	0.3181	0.08450	0.00	1
9	14	0.12711	0.27038	0.00	1
10	11	0.08205	0.19207	0.00	1
12	13	0.22092	0.19988	0.00	1
13	14	0.17093	0.34802	0.00	1

References

[1] H.You, V.Vittal, Z.Yang, "Self-healing in power systems: an approach using islanding and rate of frequency decline based load shedding“, *IEEE Transaction on Power System*, Vol.18, No.1, 2003, pp.174 -181.

[2] P.Kundur, K.Morrison, L.Wang, “Power System Security Assessment”, *IEEE Power & Energy Management*, Vol.2, No.5, 2004, pp. 30-39.

[3] D.H.Kov, A.Bose, “A generation rescheduling method of dynamic security enhancement”, *IEEE Transactions on Power Systems*, Vol 10, No.1, 1995, pp. 68-76.

[4] J.R.Winkelman, J.H.Chlow, B.C.Bowler, B.Avramovic, P.V.Kokotovic, “An analysis of inter-area dynamics of multi-machine systems”, *IEEE Transactions on Power Apparatus and Systems*, Vol PAS 100, 1981, pp.754-763.

[5] Banejad, Mahdi, Kashem, Mohammad, Ledwich, Gerard “Operation of Power System Islands”, *AUPEC 2005 CD Proceedings: Australian Universities Power Engineering Conference*, Australia, 2005.

[6] K.S.Swraup, R.Mastakar, K.V. Prasad Reddy, Decision tree for Steady state security assessment, in *IEEE TENCON Region 10 International Conference*, 1989, pp. 945-948.

[7] K.Demaree, T.Athay, K.W.Chang, Y.Mansour, E.Vaheedi, A.Y. Chang, B.R. Corns, B.W. Garrett, "An On-line dynamic security analysis system implementation", IEEE Transaction of Power Systems, Vol.9, No.4, 1994, pp.1716-1722.

[8] G.C.Ejebe, B.F. Wollenberg, "Automatic contingency selection", IEEE Transactions on Power Apparatus and Systems, Vol 98, No.1, 1979, pp.97-109.

[9] N.Balu, T.Betram, A. Bose, V.Brandwajn, G.Cauley, A.Fouad, L.Fink, M.G.Luby, B.F.Wollenberg, J.N.Wrubel, "On-line Power System Security Analysis", Proceedings on IEEE, Vol 80, No.2, 1992, pp.262-282.

[10] M.S.Tsai, "Development of islanding early warning mechanism for power systems" in *Power Eng. Soc. Summer Meeting*: IEEE, 2000, Vol.1, pp.22 – 26.

[11] *FRCC Automatic under frequency Load Shedding Program*, Florida Reliability Coordinating Council, FRCC Standards Handbook, July 2003.

[12] C. Concordia et al., "Load shedding on an isolated system," *IEEE Transactions on Power System*, vol. 10, No.3, 1995, pp.1467–1472.

[13] Vladimir.V.Terzija, "Adaptive Under frequency Load Shedding Based on the Magnitude of the Disturbance Estimation", *IEEE Transactions on Power Systems*, 2006, pp.1260-1266.

[14] A.A Mohd Zin, H.Mohd Hafiz and M.S. Aziz, "A review of Under Frequency Load Shedding Scheme on TNB System", *Power and Energy Conference*, 2004, Malaysia, pp. 170 - 174.

[15] Jett Mechenbier, Abraham Ellis, Richard Curtner, Satish Ramade, "Design of under voltage load shedding scheme", *IEEE Power Engineering Society General meeting*, 2004, pp. 631-639.

[16] "ERCOT Under frequency Load Shedding Assessment and Review", *Power Engineering Society General Meeting*, 2007, pp.1-3.

[17] Emmanuel.J.Thalassinakis, Evangelos.N.Dialynas, "A Monte-Carlo Simulation Method for Setting the Under frequency Load Shedding Relays and Selecting the Spinning Reserve Policy in Autonomous Power Systems", *IEEE Transactions on Power Systems*, Vol. 19 No.4, 2004, pp. 2044 – 2052.

[18] S.H.Horowitz, A.Politis, A.F.Gabrielle, "Frequency Actuated Load Shedding and Restoration-Part II Implementation *IEEE Transaction on Power System*, Vol. 90, 1971, pp. 1460 – 1468.

Author's Biographies:

T.A. Rameshkumaar (1973) received Bachelor of Engineering in Electrical and Electronics Engineering (2002), Master of Engineering in Power System Engineering (2008) and he is working as Assistant Professor in the Department of Electrical Engineering, Annamalai University, Annamalainagar. He is currently pursuing Ph.D degree in Electrical Engineering from Annamalai University. His research interests are in Power Systems, Control Systems, and Electrical Measurements. (Electrical Measurements Laboratory, Department of Electrical Engineering, Annamalai University, Annamalainagar-608002, Tamilnadu, India, **tarpagutharivu@gmail.com**

I.A.Chidambaram (1966) received Bachelor of Engineering in Electrical and Electronics Engineering (1987), Master of Engineering in Power System Engineering (1992) and Ph.D in Electrical Engineering (2007) from Annamalai University, Annamalainagar. During 1988 - 1993 he was working as Lecturer in the Department of Electrical Engineering, Annamalai University and from 2007 he is working as Professor in the Department of Electrical Engineering, Annamalai University, Annamalainagar. He is a member of ISTE and ISCA. His research interests are in Power Systems, Electrical Measurements and Control Systems. (Electrical Measurements Laboratory, Department of Electrical Engineering, Annamalai University, Annamalainagar – 608002, Tamilnadu, India, Tel: - 91-04144-238501, Fax: -91-04144-238275) **driacdm@yahoo.com**.

Synchronization and Coordination among Multi-Agent Systems in Distributed Data Mining

Thulasi.Bikku, Asst.Professor and Dr. Ananda Rao . Akepogu, Professor

Computer Science Department, Narayana Engineering College, Andhra Pradesh, INDIA.

thulasi.bikku@gmail.com, thulasi_bikku@yahoo.com

Computer Science Department,JNTU Anantapur, Anantapur, Andhra Pradesh, INDIA.

akepogu_@yahoo.co.in, akepogu@gmail.com

Abstract: *Data mining technology has emerged, for identifying patterns and trends from large quantities of data. The Data Mining technology normally adopts data integration method to generate Data warehouse, which is used to gather all data into a central repository, and then run an algorithm against that data to extract the useful Patterns and knowledge evaluation. However, a single data-mining technique has not been proven appropriate for every domain and data set. Distributed data mining is originated from the need of mining over decentralized data sources. Multi-agent systems (MAS) often deal with complex applications that require distributed problem solving. In many applications the individual and collective behavior of the agents depends on the observed data from distributed data sources. Since multi-agent systems are often distributed and agents have proactive and reactive features which are very useful for Knowledge Management Systems, combining DDM with MAS for data intensive applications is appealing. The integration of multi-agent system and distributed data mining, also known as multi agent based distributed data mining.*

The increasing demand to extend data mining technology to data sets inherently distributed among a large number of autonomous and heterogeneous sources over a network with limited bandwidth has motivated the development of several approaches to distributed data mining and knowledge discovery, of which only a few make use of agents. We briefly review existing approaches and argue for the potential added value of using agent technology in the domain of knowledge discovery. In this paper we propose an approach to distributed data clustering, outline its agent-oriented implementation, and security attacks in which agents may incur. Its core problem concerns collaborative work of distributed software in design of multi-agent system destined for distributed data mining and classification.

Keywords: *Distributed Data Mining, Multi-Agent Systems, Multi Agent Data Mining, Multi-Agent Based Distributed Data Mining.*

1. Introduction

Data Mining (DM), originated from knowledge discovery from databases (KDD), the large variety of DM techniques which have been developed over the past decade includes methods for pattern-based similarity search, cluster analysis, decision-tree based classification, generalization taking the data cube or attribute-oriented induction approach, and mining of association rules [13]. Distributed data mining (DDM) mines data from data sources regardless of their physical locations. The need for such characteristic arises from the fact that data produced locally at each site may not often be transferred across the network due to the Excessive amount of data and security issues. Recently, DDM has become a critical component of knowledge based systems because its decentralized architecture reaches every network such as weather databases, financial data portals, or emerging disease information systems has been recognized by industrial companies as an opportunity of major revenues from applications such as warehousing, process control, and customer services, where large amounts of data are stored. Data Mining still poses many challenges to the research community. The main challenges in data mining are: 1) Data mining has to deal with huge amounts of data located at different physical locations. 2) Data mining is computationally intensive process involving very large data i.e. more than terabytes. So, it is necessary to partition and distribute the data for parallel processing to achieve acceptable time and space performance. 3) The data stored for particular domain the Input data changes rapidly. In these cases, knowledge has to be mined fast and efficiently in order to be usable and updated.

2. Multi-Agents Behavior

DDM is a complex system focusing on the distribution of data resources over the network as well as extraction of data from those resources. The very core of DDM systems is the scalability as the system configuration may be altered time to time, therefore designing DDM systems deals with great details of software engineer issues, such reusability, extensibility, compatibility, flexibility and robustness. For these reasons, agents' characteristics are desirable for DDM systems.

Autonomy of the system: A DM agent here is considered as a modular extension of a data management system to deliberatively handle the access to the data source in agreement with constraints on the required autonomy of the system, data and model. This is in full compliance with the paradigm of cooperative information systems [12].

Scalability of DM to massive distributed data: To reduce network and DM application server load, DM agents migrate to each of the local data sites in a DDM system on which they may perform mining tasks locally, and then either return with or send relevant pre-selected patterns to their originating server for further processing. Experiments in using mobile information filtering agents in distributed data environments are encouraging [16].

Multi-strategy DDM: For some complex application settings an appropriate combination of multiple data mining techniques may be more beneficial than applying just one particular one. DM agents may choose depending on the type of data retrieved from different sites and mining tasks to be done. The learning of multi-strategy selection of DM methods is similar to the adaptive selection of coordination strategies in a multi-agent system as proposed.

Collaborative DM: DM agents may operate independently on data they have gathered at local repositories, and then combine their respective patterns or they may agree to share potential knowledge as it is discovered.

Security and Trustworthiness: Any agent-based DDM system has to cope with the problem of ensuring data security and privacy. However, any failure to implement least privilege at a data source, that means endowing subjects with only enough permissions to discharge their duties, could give any mining agent unsolicited access to sensitive data. Moreover, any mining operation performed by agents of a DDM system lacking sound security architecture could be subject to eavesdropping, data tampering, or denial of service attacks. Agent code and data integrity is a crucial issue in secure DDM: Subverting or hijacking a DM agent places a trusted piece of (mobile) software. In addition, data integration or aggregation in a DDM process introduces concern regarding inference attacks as a potential security threat. Data mining agents may infer sensitive information even from partial integration to a certain extent and with some probability. This problem, known as the so called inference problem, occurs especially in settings where agents may access data sources across trust boundaries which enable them to integrate implicit knowledge from different sources using commonly held rules of thumb.

Furthermore, the decentralization property seems to fit best with the DDM requirement in order to avoid security treats. At each data repository, mining strategy is deployed specifically for the certain domain of data.

3. Open Problems Strategy

Several systems have been developed for distributed data mining. These systems can be classified according to their strategy to three types; central learning, meta-learning, and hybrid learning.

3.1 Central learning strategy is when all the data can be gathered at a central site and a single model can be build. The only requirement is to be able to move the data to a central location in order to merge them and then apply sequential DM algorithms. This strategy is used when the geographically distributed data is small. The strategy is generally very expensive but also more accurate[18]. The process of gathering data in general is not simply a merging step; it depends on the original distribution. Agent technology is not very preferred in such strategy.

3.2 Meta-learning strategy offers a way to mine classifiers from homogeneously distributed data. Meta-learning follows three main steps. 1) To generate base classifiers at each site using a classifier learning algorithms. 2) To collect the base classifiers at a central site, and produce meta-level data from a separate validation set and predictions generated by the base classifier on it. 3) To generate the final classifier (meta-classifier) from meta-level data via a combiner or an arbiter. Copies of classifier agent will exist or deployed on nodes in the network being used. Perhaps the most mature systems of agent-based meta-learning systems are: JAM system [19], and BODHI [19].

3.3 Hybrid learning strategy is a technique that combines local and centralized learning for model building [20]; for example, Papyrus [21] is designed to support both learning strategies. In contrast to JAM and BODHI, Papyrus can not only move models from site to site, but can also move data when that strategy is desired. Papyrus is a specialized system which is designed for clusters while JAM and BODHI are designed for data classification. The major criticism of such systems is that it is not always possible to obtain an exact final result, i.e. the global knowledge model obtained may be different from the one obtained by applying the one model approach (if possible) to the same data. Approximated results are not always a major concern, but it is important to be aware of that. Moreover, in these systems hardware resource usage is not optimized. If the heavy computational part is always executed locally to data, when the same data is accessed concurrently, the benefits coming from the distributed environment might vanish due to the possible strong performance degradation. Another drawback is that occasionally, these models are induced from databases that have different schemas and hence are incompatible.

Autonomous agent can be treated as a computing unit that performs multiple tasks based on a dynamic configuration. The agent interprets the configuration and generates an execution plan to complete multiple tasks. [13], [23], [14], [12], and [16] discuss the benefits of deploying agents in DDM systems. Nature of MAS is decentralization and therefore each agent has only limited view to the system. The limitation somehow allows better security as agents do not need to observe other irrelevant surroundings. Agents, in this way, can be programmed as compact as possible, in which light-weight agents can be transmitted across the network rather than the data which can be more bulky. Being able to transmit agents from one to another host allows dynamic organization of the system. For example, mining agent *MA*1, located at repository R1, posses algorithm *A1*. Data mining task T1 at repository *R*2 is instructed to mine the data using *A*1. In this setting, transmitting *A*1 to *R*2 is a probable way rather than transfer all data from *R*2 to *R*1 where *A*1 is available.

4. Agent-Based Distributed Data Mining (ADDM)

ADDM takes data mining as a basis foundation and is enhanced with agents; therefore, this novel data mining technique inherits all powerful properties of agents and, as a result, yields desirable characteristics. In general, constructing an ADDM system concerns three key characteristics: interoperability, dynamic system configuration, and performance aspects, discussed as follows. 1) Interoperability concerns, not only collaboration of agents in the system, but also external interaction which allow new agents to enter the system seamlessly. The architecture of the system must be open and flexible so that it can support the interaction including communication protocol, integration policy, and service directory. 2) Communication protocol covers message encoding, encryption, and transportation between agents. Integration policy specifies how a system behaves when an external component, such as an agent or a data site, requests to enter or leave. 3) In relation with the interoperability characteristic, dynamic system configuration, that tends to handle a dynamic configuration of the system, is a challenge issue due to the complexity of the planning and mining algorithms. A mining task may involve several agents and data sources, in which agents are configured to equip with an algorithm and deal with given data sets. In distributed environment, tasks can be executed in parallel, in exchange, concurrency issues arise. Quality of service control in performance of data mining and system perspectives is desired; however it can be derived from both data mining and agents' fields. An ADDM system can be generalized into a set of components and viewed as depicted in figure 4.1.We may generalize activities of the system into request and response, each of which involves a different set of components. Basic components of an ADDM system are as follows.

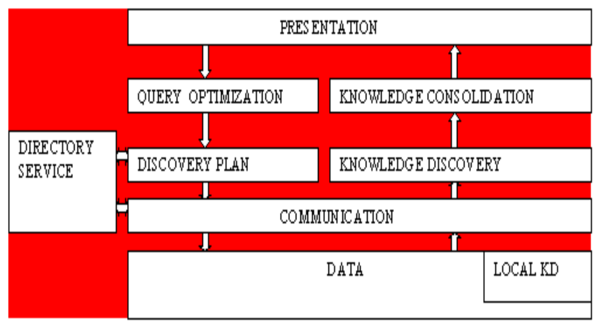


Fig. 4.1: Overview of ADDM system.

Data: Data is the foundation layer of the architecture. In distributed environment, data can be hosted in various forms, such as online relational databases, data stream, web pages, etc., in which purpose of the data might be varied.

Communication: The system chooses the related resources from the directory service, which maintains a list of data sources, mining algorithms, data schemas, data types, etc. The communication protocols may vary depending on implementation of the system, such as client-server, peer-to-peer etc.

Presentation: The user interface (UI) interacts with the user as to receive and respond to the user. The interface simplifies complex distributed systems into user-friendly message such as network diagrams, visual reporting tools, etc. On the other hand, when a user requests for data mining through the UI, the following components are involved.

Query optimization: A query optimizer analyses the request as to determine type of mining tasks and chooses proper resources for the request. It also determines whether it is possible to parallelize the tasks, since the data is distributed and can be mined in parallel.

Discovery Plan: A planner allocates sub-tasks with related resources. At this stage, mediating agents play important roles as to coordinate multiple computing units since mining sub-tasks performed asynchronously as well as results from those tasks. On the other hand, when a mining task is done, the following components are taken place,

Local Knowledge Discovery (KD): In order to transform data into patterns which adequately represent the data and reasonable to be transferred over the network, at each data site, mining process may take place locally depending on the individual implementation.

Knowledge Discovery: Also known as mining, it executes the algorithm as required by the task to obtain knowledge from the specified data source.

Knowledge Consolidation: In order to present to the user with a compact and Meaningful mining result, it is necessary to normalize the knowledge obtained from various sources. The component involves complex methodologies to combine knowledge/ patterns from distributed sites. Consolidating homogeneous knowledge/patterns is promising and yet difficult for heterogeneous case.

5. Proposed Schema

Here we propose a schema, the building and managing of large-scale distributed systems is becoming an increasingly challenging task. Continuous intervention by user administrators is generally limited in large-scale distributed environments. System support is also needed for configuration and reorganization when systems evolve with the addition of new resources. The primary goal of the management of distributed systems is to ensure efficient use of resources and provide timely service to users. Most of the distributed system management techniques still follow the centralized model that is based on the client-server model. Centralization have presented some problems, such as: 1) it could cause a traffic overload and processing at the manager node may affect its performance;2) it does not present scalability in the increase of the complexity of the network; 3) the fault in the central manager node can leave the system without a manager.

One model is the distributed management where management tasks are spread across the managed infrastructure and are carried out at managed resources. The goal is to minimize the network traffic related to management and to speed up management tasks by distributing operations across resources. The new trend in distributed system management involves using multi-agents to manage the resources of distributed systems. Agents have the capability to autonomously travel (execution state and code) among different data repositories to complete their task. The route may be predetermined or chosen dynamically depending on the results at each local data repository.

The concept of multi-agents promises new ways of designing applications that better use the resources and services of computer systems and networks. For example, moving a program
(e.g., search engine) to a resource (e.g., database) can save a lot of bandwidth and can be an enabling factor for applications that otherwise would not be practical due to network latency.

Conceptually, a multi-agent can migrate its whole virtual machine from host to host; it owns the code, not the resources. Multi-agents are the basis of an emerging technology that promises to make it very much easier to design, implement, and maintain distributed systems. We have found that multi-agents reduce network traffic, provide an effective means of overcoming network latency, and, perhaps most importantly, through their ability to operate asynchronously and autonomously of the process that created them help us to construct more robust and fault tolerant systems. The purpose of the proposed multi-agent system is to locate, monitor, and manage resources in distributed systems. The system consists of a set of static and mobile agents. Some of them reside in each node or element in the distributed system. There are two multi-agents named delegated and collector agents that can move through the distributed system. The role of each agent in the multi-agent system, the interaction between agents, and the operation of the system

5.1 Multi-Agent System Structure

The multi-agent system structure assumes that each node in the system will have a set of agents residing and running on that node[2]. These agent types are the following:

Client agent (CA) percepts service requests, initiated by the user, from the system. The CA may receive the request from the local user directly. In the other case, it will receive the request from the exporter agent coming from another node.

Service list agent (SLA) has a list of the resource agents in the system. This agent will receive the request from the CA and send it to the resource availability agent. If the reply indicates that the requested resource is local then the service list agent will deliver the request to the categorizer agent. Otherwise, it will return the request to the CA.

Resource availability agent (RAA) indicates whether the requested resource is free and available for use or not. It also indicates whether the requested resource is local or remote. It receives the request from the service list agent and checks the status of the requested resource through the access of the MIB. The agent then constructs the reply depending on the retrieved information from the database.

Resource agent (RSA) is responsible for the operation and control of the resource. This agent executes the on the resource. Each node may have zero or more RSAs.

Router agent (RA) provides the path of the requested resource on the network in case of accessing remote resources. Before being dispatched, the exporter agent will ask the router agent for the path of the requested resource. This in turn delivers it to the exporter agent.

Categorizer agent (CZA) allocates a suitable resource agent to perform the user request. This agent percepts inputs coming from the service list agent. It then tries to find a suitable free resource agent to perform the requested service.

Exporter agent (EA) is a mobile agent that can carry the user request through the path identified by the RA to reach the node that has the required resource. It passes the requested resource *id* to the RA and then receives the reply. If the router agent has no information about the requested resource, the EA will try to locate the resource in the system. There are also two additional mobile agent types exist in the system.

Delegated agent (DA) is a mobile agent that is launched in each sub network. It is responsible for traversing sub network nodes instead of the exporter agent to do the required task and carry results back to the exporter agent.

Collector agent (CTA) is a mobile agent that is launched from the last sub network visited by the exporter agent. It is launched when results from that sub network become available. This agent goes through the reversed itinerary of the exporter agent trip. The CTA collects results from the delegated agents and carries it to the source node. All mobile agents used here are of interrupt driven type.

5.2 System's operation

The activity cycle of our multi-agent system residing in a local data repository. The client agent receives the service requests either from the user or from an exporter agent. The client agent then asks the service list agent for the existence of a resource agent that can perform the request. The service list agent checks the availability of the required resource agent by consulting a resource availability agent to perform the requested service. The reply of the resource availability agent describes whether or not the resource is locally available and whether or not there is a resource agent that can perform the requested service. If the resource availability agent then accepts the request, the service list agent will ask the categorizer agent to allocate a suitable resource agent to the requested service and the resource agent will perform the requested service. Otherwise, the service list agent informs the client agent with the rejection and is passed to the exporter agent. The exporter agent asks the router agent for the path of the required resource agent. Once the path is determined, the exporter agent will be dispatched through the network channel to the destination node identified by that path. If the router agent has no information about the location of the required resource agent, the exporter agent will search the distributed system to find the location of the required resource agent and assign the required task to it.

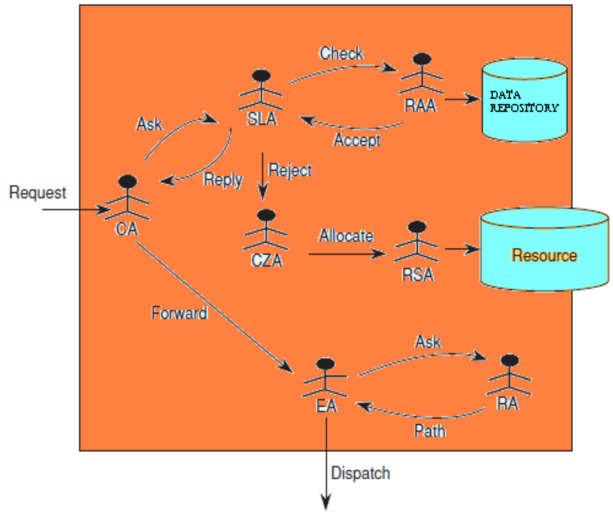


Fig. 5.2.1: Overview of Agents activity.

As shown in Fig. 5.2.2, the exporter agent traverses the sub networks of the distributed system through its trip. At each sub network, a delegated agent is launched to traverse the local nodes of that sub network doing the required task and carrying results of that task. The agents of the social interface described in Fig. 5.2.1 are implemented at each node in the system. There are two approaches to collect results of the required task and send these results back to the source.

In traditional agent-based management systems that use mobile agent, the exporter agent will wait at each visited sub network until the delegated agent finishes its work and obtains results. Then, the exporter agent will take these results and go to the next sub network in its itinerary. The exporter agent will return to its home sub network after visiting all the sub networks determined in the itinerary. The home sub network of the exporter agent is the sub network from which it was initially dispatched. The waiting of the exporter agent prevents execution of tasks to be started in the other sub networks. This approach is used in most of the previously developed management systems in which operation is based on mobile agents. In the proposed multi-agent management system, the exporter agent does not wait for results from each sub network. It resumes its trip visiting other sub networks, and at each sub network, another delegated mobile agent is launched to carry out management tasks instead of the exporter agent. The exporter agent will be killed at the last visited sub network in its itinerary. When results from the last visited sub network become available, another mobile agent called collector agent is launched or dispatched from this sub network to collect results from it and other sub networks. The collector agent goes through the reversed itinerary of the exporter agent trip carrying results to the home sub network. In this manner, operations can be done in a parallel fashion at different sub networks because there is no delay of the task submission to local data repositories of these sub networks.

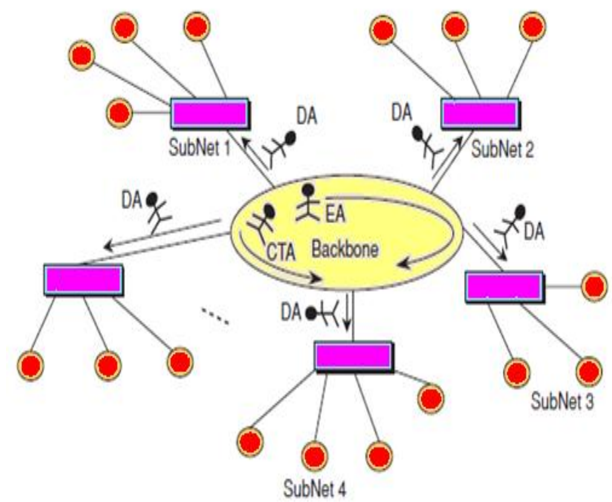


Fig. 5.2.2: Overview of network architecture ADDM.

Conclusions

Distributed management for distributed systems is becoming a reality due to the rapid growing trend in internetworking and the rapid expanding connectivity. This article describes a new multi-agent system for the management of distributed systems. The system is proposed to optimize the execution of management functions in distributed systems. The proposed system can locate, monitor, and manage resources in the system. The new technique in that system allows management tasks to be submitted to sub networks of the distributed system and executed in a parallel fashion. The proposed system uses two multi- agents. The first is used to submit tasks to the sub networks of the distributed system and the other collects results from these sub networks. The proposed system is compared against traditional management techniques in terms of response time, speedup, and efficiency. A prototype has been implemented using performance management as the case study. The performance results indicate a significant improvement in response time, speedup, efficiency, and scalability compared to traditional techniques. The use of JVM in the implementation of the proposed system gives the system a certain type of portability. Therefore, it is desirable to use the proposed system in the management of distributed systems. The proposed system is limited to be applied to high-speed networks that have bandwidth 100 Mb/s or more. Also, the system cannot work when a failure occurs. Future research will be related to the security of mobile agents and of hosts that receive them in the context of public networks. Mobile agents should be protected against potentially malicious hosts. The hosts should also be protected against malicious actions that may be performed by the mobile code they receive and execute. So, a detailed design and implementation of the whole secure system should be considered as a future work. Also, the high complexity of distributed systems could increase the potential for system faults. Most of the existing management systems assume that there is no fault in the system. It would be interesting to develop a fault tolerant management system that introduces safety in the system and attempts to maximize the system reliability without extra hardware cost.

References:

[1] T.M. Chen, and S.S. Liu, "A model and evaluation of distributed network management approaches," *IEEE J.Selected Areas Commun.*, vol. 20, no. 4,pp. 850–857, May 2002.

[2] T.C. Du, E.Y. Li, and A. Chang,"Mobile agents in distributed network management," *Commun. ACM*, vol. 46, no. 7, pp. 127–132, July 2003.

[3] H. Ku, G.W.R. Ludere, and B.Subbiah, "An intelligent mobile agent framework for distributed network management,"in *Proc. Globecom '97*Phoenix, pp. 160–164.

[4] N. R. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. Journal of Autonomous Agents and Multi-Agent Systems, 1(1):7–38, 1998.

[5] N. R. Jennings and S. Bussmann, "Agent-based control systems—Why are they suited to engineering complex systems?" *IEEE Control Syst.Mag.*, vol. 23, no. 3, pp. 61–73, Jun. 2003.

[6] W. Brennan, M. Fletcher, and D. H. Norrie, "An agent-based approach to reconfiguration of real-time distributed control systems," *IEEE Trans.Robot. Autom.*, vol. 18, no. 4, pp. 444–449, Aug. 2002.

[7] J. Kosakaya, A. Kobayashi, and K. Yamaoka, "Distributed supervisory system with cooperative multi-agent FEP," in *Proc. 22nd Int. Conf.Distrib. Comput. Syst. Workshops*, 2002, pp. 633–638.

[8] R. B. Patel, Neeraj Goel, "Mobile Agents in Heterogeneous Networks: A Look on Performance," Journal of Computer Science,2(11): 824-834, 2006.

[9] O'Hare G.M.P., Marsh D., Ruzzelli A., R. Tynan, "Agents for Wireless Sensor Network Power Management", in Proceedings of International Workshop on Wireless and Sensor Networks (WSNET-05), Oslo, Norway IEEE Press, 2005.

[10] Ajith Abraham, Crina Grosan, and Vitorino Ramos, editors. *Swarm Intelligence in Data Mining*, volume 34 of *Studies in Computational Intelligence*. Springer, 2006.

[11] Sung W. Baik, Jerzy W. Bala, and Ju S. Cho.Agent based distributed data mining. *Lecture Notes in Computer Science*, 3320:42–45,2004.

[12] S. Bailey, R. Grossman, H. Sivakumar, and A. Turinsky. Papyrus: a system for data mining over local and wide area clusters and super-clusters. In *Supercomputing '99: Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM)*, page 63, New York, NY, USA, 1999. ACM.

[13] R. J. Bayardo, W. Bohrer, R. Brice, A.Cichocki, J. Fowler, A. Helal, V. Kashyap, T. Ksiezyk, G. Martin, M. Nodine, and Others.InfoSleuth: agent-based semantic integration of information in open and dynamic environments. *ACM SIGMOD Record*, 26(2):195–206, 1997.

[14] F. Bergenti, M. P. Gleizes, and F.Zambonelli. *Methodologies And Software Engineering For Agent Systems: The Agentoriented Software Engineering Handbook*. Kluwer Academic Publishers, 2004.

[15] A. Bordetsky. Agent-based Support for Collaborative Data Mining in Systems Management. In *Proceedings Of The Annual Hawaii International Conference On System Sciences*, page 68, 2001.

[16] R. Bose and V. Sugumaran. IDM: an intelligent software agent based data mining environment. *1998 IEEE International Conference on Systems, Man, and Cybernetics*, 3, 1998.

[17] L. Cao, C. Luo, and C. Zhang. Agent-Mining Interaction: An Emerging Area. *Lecture Notes in Computer Science*, 4476:60, 2007.

[18] J. Dasilva, C. Giannella, R. Bhargava, H.Kargupta, and M. Klusch. Distributed data mining and agents. *Engineering Applicationsof Artificial Intelligence*, 18(7):791–807, October 2005.

[19] S. Datta, K. Bhaduri, C. Giannella, R. Wolff,and H. Kargupta. Distributed data mining in peer-to-peer networks. *Internet Computing,IEEE*, 10(4):18–26, 2006.

[20] W. Davies and P. Edwards. Distributed Learning: An Agent-Based Approach to Data-Mining. In *Proceedings of MachineLearning 95 Workshop on Agents that Learn from Other Agents*, 1995.

[21] U. Fayyad, R. Uthurusamy, and Others. Data mining and knowledge discovery in databases. *Communications of the ACM*,39(11):24–26, 1996.
[22] Vladimir Gorodetsky, Oleg Karsaev, and Vladimir Samoilov. Multi-agent technology for distributed data mining and classification.In *IAT*, pages 438–441. IEEE Computer Society, 2003.

[23] Sven A. Brueckner H. Van Dyke Parunak.Engineering swarming systems. *Methodologies and Software Engineering for Agent Systems*, pages 341–376, 2004.

Author Biographies

Dr. Anand Rao Akepogu. recieved B.Sc (M.P.C) degree from Sri Venkateswara University, Andhra Pradesh, India. He received B.Tech degree in Computer Science & Engineering from University of Hyderabad, Andhra Pradesh, India and M.Tech degree in A.I & Robotics from University of Hyderabad, Andhra Pradesh, India. He received Ph.D degree from Indian Institute of Technology, Madras, India. He is currently working as a Professor of Computer Science & Engineering Department and also as a Principal of JNTU College of Engineering, Anantapur, Jawaharlal Nehru technological University, Andhra Pradesh, India. Dr. Rao published more than twenty research papers in international journals and conferences. His main research interest includes software engineering and data mining.

Thulasi.Bikku received B.Tech degree in Information Technology from Jawaharlal Nehru Technological University, Hyderabad, Andhra Pradesh, India and M.Tech degree in Computer Science & Engineering from Jawaharlal Nehru Technological University,Kakinada, Andhra Pradesh, India. Her main research interest includes Distributed Data Mining and AI.

Software Reliability with SPC

Dr. R Satya Prasad[1], K Ramchand H Rao[2], Dr. R.R. L Kantham[3]

[1]Associate Prof, Dept. of Computer Science & Eng, Acharya Nagarjuna University, Guntur, INDIA, profrsp@gmail.com
[2] Dept. of Computer Science A.S.N. Degree College, Tenali, INDIA, ramkolasani@gmail.com
[3] Prof., Dept. of Statistics, Acharya Nagarjuna University, Guntur, INDIA, kantam_rrl@rediffmail.com

Abstract: *Software reliability is the probability of failure-free operation of software in a specified environment during specified time duration. Statistical Process Control can monitor the forecasting of software failure and thereby contribute significantly to the improvement of software reliability. Control charts are widely used for software process control in the software industry. Relatively little research work is, however, available on their use to monitor failure process of software. It is well known that Control charts can be used to analyze both small and large failure frequency. Some control charts can be used for monitoring the number of failures per fixed interval. However they are not effective especially when the failure frequency becomes small. To meet this desideratum, the control scheme adopted for our study is based on the cumulative data between observations of failure. It is proposed that the said control scheme can be easily and fruitfully applied to monitor the software failure process for Half Logistic Distribution based NHPP.*

Keywords: *Statistical Process Control (SPC), Software reliability, Control Charts, Probability limits, Half Logistic Distribution*

1. Introduction

The monitoring of Software reliability process is a far from simple activity. In recent years, several authors have recommended the use of SPC for software process monitoring. A few others have highlighted the potential pitfalls in its use[1].

The main thrust of the paper is to formalize and present an array of guidelines in a disciplined process with a view to helping the practitioner in putting SPC to correct use during software process monitoring.

Over the years, SPC has come to be widely used among others, in manufacturing industries for the purpose of controlling and improving processes. Our effort is to apply SPC techniques in the software development process so as to improve software reliability and quality [2]. It is reported that SPC can be successfully applied to several processes for software development, including software reliability process. SPC is traditionally so well adopted in manufacturing industry. In general software development activities are more process centric than product centric which makes it difficult to apply SPC in a straight forward manner.

The utilization of SPC for software reliability has been the subject of study of several researchers. A few of these studies are based on reliability process improvement models. They turn the search light on SPC as a means of accomplishing high process maturities. Some of the studies furnish guidelines in the use of SPC by modifying general SPC principles to suit the special requirements of software development [2] (Burr and Owen[3]; Flora and Carleton[4]). It is especially noteworthy that Burr and Owen provide seminal guidelines by delineating the techniques currently in vogue for managing and controlling the reliability of software. Significantly, in doing so, their focus is on control charts as efficient and appropriate SPC tools.

It is accepted on all hands that Statistical process control acts as a powerful tool for bringing about improvement of quality as well as productivity of any manufacturing procedure and is particularly relevant to software development also. Viewed in this light, SPC is a method of process management through application of statistical analysis, which involves and includes the defining, measuring, controlling, and improving of the processes[5].

2. Model Formulation.

Let $\left[\mathrm{N}(\mathrm{t}), \mathrm{t} \geq 0\right], \mathrm{m}(\mathrm{t}), \lambda(t)$ be the counting process, mean value function and intensity function of a software failure phenomenon.

The mean value function m(t) is finite valued, non decreasing, non negative and bounded with the boundary conditions

$$m(t) = \begin{cases} 0, & t = 0 \\ a, & t \to \infty \end{cases}$$

Here 'a' represents the expected number of software failures eventually detected. If $\lambda(t)$ is the corresponding intensity function. $\lambda(t)$ is a decreasing function of $m(t)$ as a result of repair action following early failures. A relation between $m(t)$ and $\lambda(t)$ is given by

$$\lambda(t) = \frac{b}{2a}\left[a^2 - m^2(t)\right]$$

where 'b' is a positive constant, serving the purpose of constant of proportional fall in $\lambda(t)$. This relation indicates a decreasing trend for $\lambda(t)$ with increase in $m(t)$

From the fact that $\lambda(t)$ is the derivative of $m(t)$ we get the following differential equation

$$\frac{dm(t)}{dt}=\frac{b}{2a}\left[a^2-m^2(t)\right]$$

whose solution is

$$m(t)=\frac{a\left(1-e^{-bt}\right)}{\left(1+e^{-bt}\right)} \qquad (2.1)$$

An NHPP with its mean value function given in equation (2.1). Its intensity function is

$$\lambda(t)=\frac{2abe^{-bt}}{\left(1+e^{-bt}\right)^2} \qquad (2.2)$$

3. Estimation Based on Inter Failure Times

The mean value function and intensity function of Half Logistic Model [6] are given by

$$m(t)=\frac{a\left(1-e^{-bt}\right)}{\left(1+e^{-bt}\right)} \quad ,a>0,b>0,t\geq 0 \qquad (3.1)$$

$$\lambda(t)=\frac{2abe^{-bt}}{\left(1+e^{-bt}\right)^2} \qquad (3.2)$$

The constants 'a', 'b' which appear in the mean value function and hence in NHPP, in intensity function (error detection rate) and various other expressions are called parameters of the model. In order to have an assessment of the software reliability 'a',' b' are to be known or they are to be estimated from a software failure data.

Suppose we have 'n' time instants at which the first, second, third..., n[th] failures of a software are experienced. In other words if S_k is the total time to the k[th] failure, s_k is an observation of random variable S_k and 'n' such failures are successively recorded. The joint probability of such failure time realizations $s_1, s_2, s_3, \ldots s_n$ is

$$L=e^{-m(s_n)}\cdot\prod_{k=1}^{n}\lambda(s_k) \qquad (3.3)$$

The function given in equation (3.3) is called the likelihood function of the given failure data. Values of 'a', ' b' that would maximize L are called maximum likelihood estimators (MLEs) and the method is called maximum likelihood (ML) method of estimation. Accordingly 'a', 'b' would be solutions of the equations

$$\frac{\partial \log L}{\partial a}=0\ ,\frac{\partial \log L}{\partial b}=0\ ,\ \frac{\partial^2 \log L}{\partial b^2}=0$$

Substituting the expressions for m(t), λ(t) given by equations (3.1) and (3.2) in equation (3.3), taking logarithms, differentiating with respect to 'a', 'b' and equating to zero, after some joint simplification we get

$$a=n\left[\frac{\left(1+e^{-bs_n}\right)}{\left(1-e^{-bs_n}\right)}\right] \qquad (3.4)$$

$$g(b)=\sum_{k=1}^{n}s_k-\frac{n}{b}-2\sum_{k=1}^{n-1}\frac{s_k.e^{-bs_k}}{\left(1+e^{-bs_n}\right)}-\frac{2s_n.e^{-bs_k}}{\left(1+e^{-bs_n}\right)}\left[1-\frac{n}{1-e^{-bs_n}}\right]=0 \qquad (3.5)$$

$$g^{|}(b)=\frac{n}{b^2}+2\sum_{K=1}^{n-1}\frac{s_k^2\, e^{-bs_k}}{(1+e^{-bs_k})^2} \qquad (3.6)$$

The value of 'b' can be obtained using Newton-Raphson method which when substituted in equation (3.4) gives value of 'a'.

4. Monitoring the time between failures using control chart

The selection of proper SPC charts is essential to effective statistical process control implementation and use. There are many charts which use statistical techniques. It is important to use the best chart for the given data, situation and need[7].

There are advances charts that provide more effective statistical analysis. The basic types of advanced charts, depending on the type of data are the variable and attribute charts. Variable control chats are designed to control product or process parameters which are measured on a continuous measurement scale. X-bar, R charts are variable control charts.

Attributes are characteristics of a process which are stated in terms of good are bad, accept or reject, etc. Attribute charts are not sensitive to variation in the process as variables charts. However, when dealing with attributes and used properly, especially by incorporating a real time pareto analysis, they can be effective improvement tools. For attribute data there are : p-charts, c-charts, np-charts, and u-charts. We have named the control chart as ***Failures Control Chart*** in this paper. The said control chart helps to assess the software failure phenomena on the basis of the given inter-failure time data[8].

4.1 Distribution of Time between failures

For a software system during normal operation, failures are random events caused by, for example, problem in design or analysis and in some cases insufficient testing of software. In this paper we applied *Half Logistic Distribution[6]* to time between failures data. This distribution uses cumulative time between failure data for reliability monitoring.

The equation for mean value function of Half Logistic Distribution from equation 2.1

$$m(t) = a\left[\frac{1-e^{-bt}}{1+e^{-bt}}\right]$$

Equate the pdf of above m(t) to 0.99865, 0.00135, 0.5 and the respective control limits are given by.

$$m(t) = a\left[\frac{1-e^{-bt}}{1+e^{-bt}}\right] = 0.99865$$

It gives

$$t = \frac{7.300122639}{b} = t_U \qquad (4.1)$$

Similarly

$$t = \frac{0.002700002}{b} = t_L \qquad (4.2)$$

$$t = \frac{1.098612289}{b} = t_C \qquad (4.3)$$

The control limits are such that the point above the $m(t_U)$ (4.1)(UCL) is an alarm signal. A point below the $m(t_L)$(4.2) (LCL) is an indication of better quality of software. A point within the control limits indicates stable process.

4.2 **Example**

The procedure of a failures control chart for failure software process will be illustrated with an example here. Table 1 shows the time between failures of a software product [8].

Table -1: Time between failures of a component[8]

Failure number	*Time between Failure (hrs)*	*Failure number*	*Time between Failure (hrs)*	*Failure number*	*Time between Failure (hrs)*
1	30.02	11	0.47	21	70.47
2	1.44	12	6.23	22	17.07
3	22.47	13	3.39	23	3.99
4	1.36	14	9.11	24	176.06
5	3.43	15	2.18	25	81.07
6	13.2	16	15.53	26	2.27
7	5.15	17	25.72	27	15.63
8	3.83	18	2.79	28	120.78
9	21	19	1.92	29	30.81
10	12.97	20	4.13	30	34.19

Table 2 shows the time between failures (cumulative) in hours, corresponding m(t) and successive difference between m(t)'s.

Table 2- Successive difference of mean value function (m(t))

Failure	*Time between Failure (hrs)*	*m(t)*	*Successive Difference of*	*Failure*	*Time between Failure (hrs)*	*m(t)*	*Successive Difference of*

number	(cumulative)		m(t)	number	(cumulative)		m(t)
1	30.02	2.364302301	0.112954122	15	136.25	10.35263373	1.078933884
2	31.46	2.477256423	1.75247064	16	151.78	11.43156761	1.720058168
3	53.93	4.229727062	0.105331424	17	177.5	13.15162578	0.181300217
4	55.29	4.335058486	0.265209098	18	180.29	13.332926	0.12414598
5	58.72	4.600267585	1.014117756	19	182.21	13.45707198	0.265317161
6	71.92	5.614385341	0.392552448	20	186.34	13.72238914	4.145675764
7	77.07	6.006937788	0.290696243	21	256.81	17.8680649	0.892060255
8	80.9	6.297634032	1.572927375	22	273.88	18.76012516	0.202130338
9	101.9	7.870561407	0.951568845	23	277.87	18.9622555	6.671109936
10	114.87	8.822130252	0.034170022	24	453.93	25.63336543	1.838854653
11	115.34	8.856300274	0.450773778	25	535	27.47222009	0.04287584
12	121.57	9.307074052	0.244275895	26	537.27	27.51509593	0.283919499
13	124.97	9.551349947	0.647546483	27	552.9	27.79901542	1.634249439
14	134.07	10.19889643	0.1537373	28	673.68	29.43326486	0.290356701
15	136.25	10.35263373	1.078933884	29	704.49	29.72362156	0.276439943

The values of 'a' and 'b' are computed by using the well know iterative Newton-Rapson method. These values are used to compute, T_u, T_L, T_c i.e. UCL, LCL, CL

The values of a and b are 31.524466 and 0.005006 and

$m(T_U)$/UCL = 31.48190797

$m(T_L)$/LCL = 0.042558035

$m(T_C)$/CL = 15.762233

The values of m(t) at T_c, T_u, T_L and at the given 30 inter-failure times are calculated. Then the m(t)'s are taken, which leads to 29 values. The graph with the said inter-failure times 1 to 30 on X-axis, the 29 values of m(t)'s on Y-axis, and the 3 control lines parallel to X-axis at $m(T_L)$, $m(T_U)$, $m(T_C)$ respectively constitutes failures control chart to assess the software failure phenomena on the basis of the given inter-failures time data.

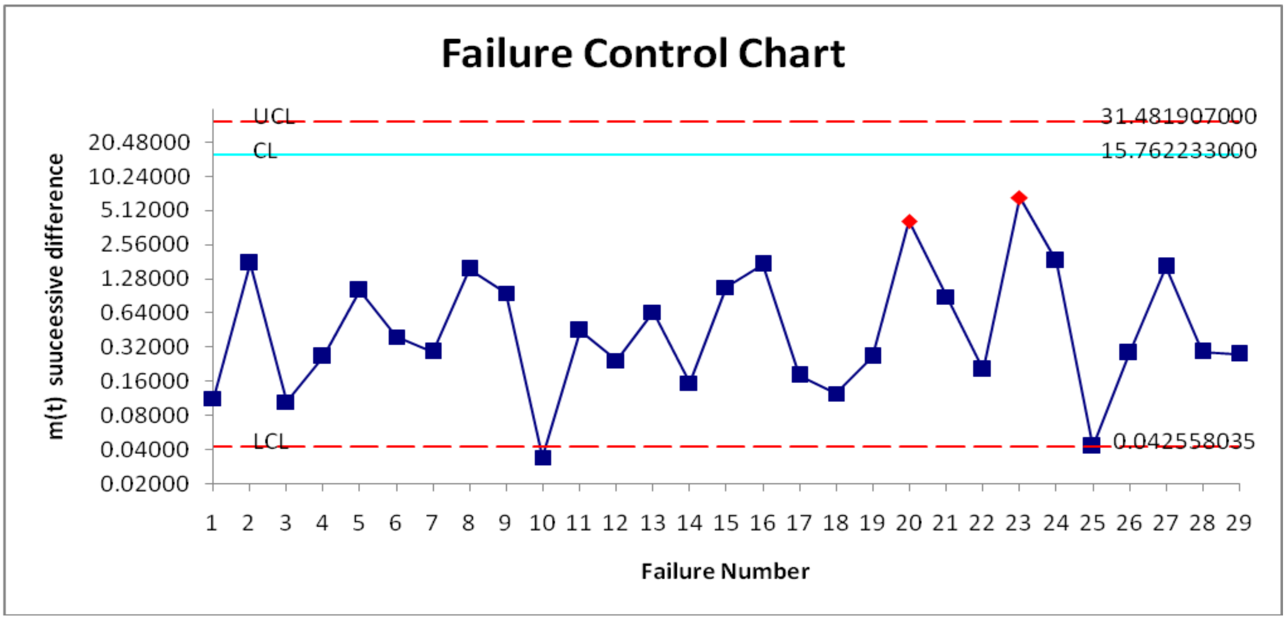


Figure 1: Failures Control Chart

5. Conclusion

This failures control chart (Figure 1) exemplifies that, the first out – of – control situation is noticed at the 10th failure with the corresponding successive difference of m(t) falling below the LCL. It results in an earlier and hence preferable out - of - control for the product. The assignable cause for this is to be investigated and promoted. In comparison, the time control chart for the same data given in Xie et a1 [8] reveals an out - of - control for the first time above the UCL at 23rd failure. Since the data of the time-control chart are inter-failure times, a point above UCL for time-control chart is also a preferable criterion for the product. The time control chart gives the first out - of - control signal in a positive way, but at the 23rd failure. Hence it is claimed that the proposed failures control chart detects out - of - control in a positive way much earlier than the time-control chart. Therefore, earlier detections are possible in failures control chart

References

[1] N. Boffoli, G. Bruno, D. Cavivano, G. Mastelloni; Statistical process control for Software: a systematic approach; 2008 ACM 978-1-595933-971-5/08/10.

[2]K. U. Sargut, O. Demirors; Utilization of statistical process control (SPC) in emergent software organizations: Pitfallsand suggestions; Springer Science + Business media Inc. 2006.

[3] Burr,A. and Owen ,M.1996. Statistical Methods for Software quality . Thomson publishing Company. ISBN 1-85032-171-X.

[4] Carleton, A.D. and Florac, A.W. 1999. Statistically controlling the Software process. The 99 SEI Software Engineering Symposimn, Software Engineering Institute, Carnegie Mellon University.

[5]Mutsumi Komuro; Experiences of Applying SPC Techniques to software development processes; 2006 ACM 1-59593-085-x/06/0005.

[6]Ronald P.Anjard;SPC CHART selection process;Pergaman 0026-27(1995)00119-0Elsevier science ltd.

[7]R.satyaprasad, Half Logistic Software Reliability Growth Model,Ph.D. Thesis,2007

[8]M.Xie, T.N. Goh, P. Rajan; Some effective control chart procedures for reliability monitoring; Elsevier science Ltd, Reliability Engineering and system safety 77(2002) 143- 150

Author Biographies

Dr. R. Satya Prasad received Ph.D. degree in Computer Science in the faculty of Engineering in 2007 from Acharya Nagarjuna University, Guntur, Andhra Pradesh, India. He have a satisfactory consistent academic track of record and received gold medal from Acharya Nagarjuna University for his out standing performance in a first rank in Masters Degree. He is currently working as Associate Professor and Head of the Department, in the Department of Computer Science & Engineering, Acharya Nagarjuna University. He has occupied various academic responsibilities like practical examiner, project adjudicator, external member of board of examiners for various Universities and Colleges in and around in Andhra Pradesh. His current research is focused on Software Engineering, Image Processing & Database Management System. He has published several papers in National & International Journals.

K Ramchand H Rao' received Master's degree in Technology with Computer Science from Dr. M.G.R University, Chennai, Tamilnadu, India, . He is currently working as Associate Professor and Head of the Department, in the Department of Computer Science, A.S.N. Degree College, Tenali, which is affiliated to Acharya Nagarjuna University. He has 18 years experience and 2 years of Industry experience at Morgan Stanly, USA as Software Analyst. He is currently pursuing Ph.D., at Department of Computer Science and Engineering, Acharya Nagarjuna University, Guntur, Andhra Pradesh, India. His research include on Software Engineering.

Design of Elliptical Air Hole PCF with Hybrid Square Lattice for High Birefringence and a Lower Zero Dispersion Wavelength

Ritu Sharma, Vijay Janyani, Anuradha Sharma
Department of electronics and communication Engineering
Malaviya National Institute of Technology, Jaipur, Rajasthan
ritusharma.mnit@gmail.com, anuabhi214@gmail.com

***Abstract**: In this paper a hybrid square-lattice PCF with circular and elliptical air holes has been designed and the numerical investigation shows its high birefringence and shifting of zero dispersion wavelengths toward lower wavelength range with the increase in the ellipticity of air holes. A full-vector TE, FDTD method is used.*

***Keywords:** Photonic Crystal Fiber (PCF), Total internal reflection (TIR), Effective Refractive index (neff), high birefringence (Hi-Bi), Finite Difference Time Domain(FDTD),Transparent boundary condition(TBC).*

1. Introduction

The Photonic crystal fibers (PCFs) are made from single material such as silica glass, with an array of microscopic air channels running along its length [1–6]. In this a defect can be created by removing the central air-hole with glass to guide light by total internal reflection (TIR) between the solid core and the cladding region. These index-guiding PCFs are also called m-TIR PCFs. Another PCFs that use a perfectly periodic structure exhibits photonic band gap (PBG) effect and these PBG–PCFs guide light in a low-index core region [1,7,8]. In recent years, photonic crystal fibers are attracting much attention because of their unique properties, such as dispersion and polarization properties, which cannot be realized in conventional optical fibers. In this paper, we will focus on index-guiding PCFs.

The polarization and dispersive properties of elliptical air hole PCFs (EHPCF) were investigated using full-vector TE FDTD method and compared with circular air hole PCF, keeping the area of air filling fraction same in both the cases. High level of birefringence in fiber optics is required to maintain the linear polarization state by reducing polarization mode dispersion. Recently, due to the large index contrast of photonic crystal fibers (PCFs) compared to the conventional fiber, Hi-Bi PCFs have been reported by researchers by breaking the circular symmetry, implementing asymmetric defect structures such as dissimilar air hole diameters along the two orthogonal axes [9-10], designing an air hole lattice or a microstructure lattice with inherent anisotropic properties such as the elliptical-hole PCF [11-12]. Modal birefringence in these Hi-Bi PCFs has been reported to have values of the order of 10^{-3} or 10^{-2} higher than that of the conventional High Birefringence fibers (10^{-4}) [13]. According to the Symmetry theory, the rectangular lattice is potentially more anisotropic than the triangular and honeycomb lattices [16]. The values of birefringence for basic rectangular lattice PCF depicted in [16] are of the order of 10^{-3}. By making a combination of the elliptical- air hole and rectangular lattice, the birefringence increases to an order of10^{-2} [14-15]. However, elliptical air holes are very difficult to control during the fabrication process. The dispersion properties of square-lattice PCF have been reported by Soan Kim [17] where a hybrid square lattice PCF is proposed which exhibits birefringence of the order of $10^{-3.}$

In this paper, the PCF structure is designed and analyzed by well-established method: FDTD (Finite difference time domain). A full vector TE mode is used to perform the modal analysis which generates the effective refractive index, which is further used to calculate the waveguide dispersion and Birefringence. A Hybrid Square -lattice PCF with circular air holes (Fig.1) is referred [17] and compared with elliptical air holes which exhibits high birefringence and lower dispersion. Also Zero dispersion wavelength gets shifted towards lower wavelength range.

2. Analysis of dispersion and Birefringence-

Theory basis—

The most interesting aspects of the work on PCFs are their propagation properties with respect to their possible application in modern fibers optical communication systems, such as dispersion and Birefringence [18].Some polarization maintaining fibers (PMFs) contains elliptical air holes in the cladding to produce a high birefringence, which is elliptical photonic crystal fiber (EPCF).The birefringence is defined as $\left| n^{x}_{eff} - n^{y}_{eff} \right|$ where n^{x}_{eff} and

n^{y}_{eff} are the effective indices of x-polarized mode and y-polarized mode, respectively.

The effective refractive index of the base mode is given as $n_{eff} = \beta/K_0$, where β is the propagation constant, $k_0=2\pi/\lambda$ is the free-space wave number. First the modal effective indexes n_{eff} are solved, and then the dispersion parameter D can be obtained [19]

$$D(\lambda) = -(\lambda/c)\,(d^2 n_{eff}/d\lambda^2) \qquad (1)$$

Where c is the velocity of the light in a vacuum and λ is the operating wavelength [19, 20].

The waveguide dispersion is strongly related to the design parameters of the PCFs and therefore can be optimized to achieve desired dispersion properties.

3. Design parameter and Simulation results

The cross section of a hybrid square-lattice PCF (using OPTI FDTD Simulator version 8) with circular air holes is shown in Fig.1. .

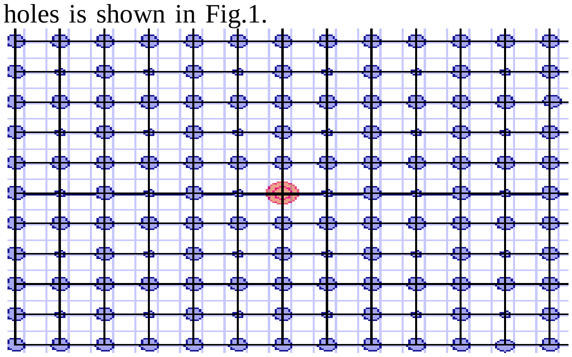


Fig1. A Hybrid Square -lattice PCF with circular air holes $d_c/d = 0.5$ µm, pitch (Λ) =2 µm, $d_c/\Lambda=0.2$ [17].

The wafer chosen is of pure (non dispersive) silica with refractive index 1.45 and the refractive index of air holes is 1. The wafer is designed for length = 26µm and width=22µm, ensuring high degree simulation accuracy. The diameters of the small and large air holes are d_c and d respectively. The pitch (Λ) which is center to center spacing between two nearest air holes gives the characteristics of a hybrid square-lattice PCF. The boundary conditions chosen are TBC. The mesh size is $\Delta x = \Delta z = 0.106$ µm for both direction.

In this paper, a hybrid square-lattice PCF is investigated with circular and elliptical air holes in order to control not only modal birefringence but also chromatic dispersion properties simultaneously. Taking reference of Fig 1(configuration I) with $d_c/d = 0.5$ µm, pitch (Λ) =2 µm, $dc/\Lambda=0.2$,the hybrid square lattice PCF with elliptical air holes structures were designed and are shown in Fig 2, 3, 4 respectively. Any stress or pressure intentionally or unintentionally may deform circular air hole to the elliptical shape. So the above investigation may be very applicable as far as fabrication is concerned. For elliptical air-holes the minor axis is defined as ‘a’ and major axis is defined as ‘b’ respectively. The ellipiticity is defined as the ratio of major axis and minor axis i.e. b/a. For the II, III, IV configuration the layout designs are shown in Fig. 2, 3, 4. For II configuration $d_c=0.4$µm, a=0.1µm, b=0.4µm for small air holes and a=0.3µm, b=0.533µm for large air holes and for III configuration a=0.1µm, b=0.4µm for small air holes and a=0.2 µm, b=0.8µm for large air holes, and for IV configuration a random distribution of circular and elliptical air holes are shown.

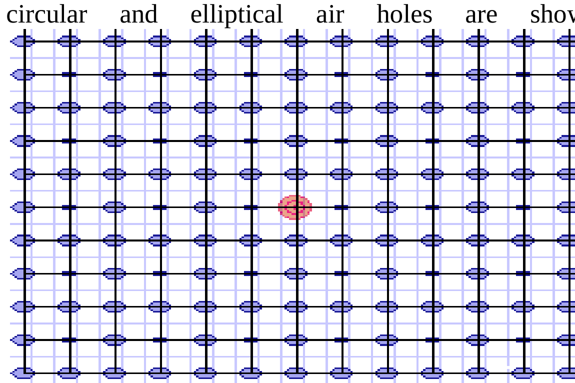


Fig 2 Layout design for a hybrid square lattice PCF with elliptical air holes, here a =0.1µm, b=0.4 µm for small air holes and a=0.3 µm, b=0.533 µm for large air holes

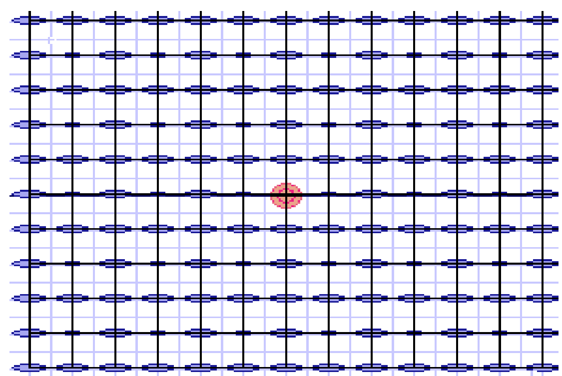


Fig 3 Layout design for a hybrid square lattice PCF with elliptical air holes, here a=0.1 µm, b=0.4 µm for small air holes and a=0.2 µm, b=0.8 µm for large air holes.

Table I - Dispersion in ps/nm.km

Wavelength	Dispersion (ps/nm.km) Conf I	Dispersion (ps/nm.km) Conf II	Dispersion (ps/nm.km) Conf III	Dispersion (ps/nm.km) Conf IV
(λ=1.5 µm)	10.42721	2.71813	-7.65905	12.40358

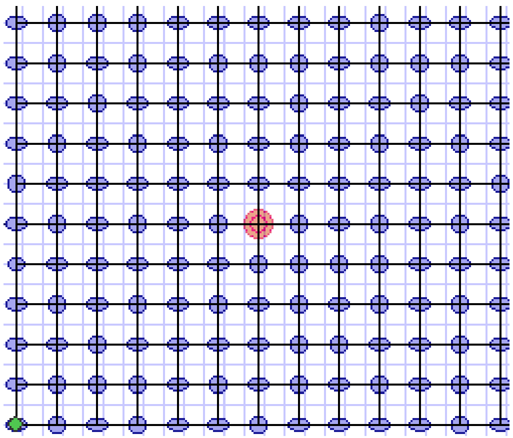


Fig 4 Layout design for a hybrid square-lattice PCF with a random distribution of circular and elliptical air holes.

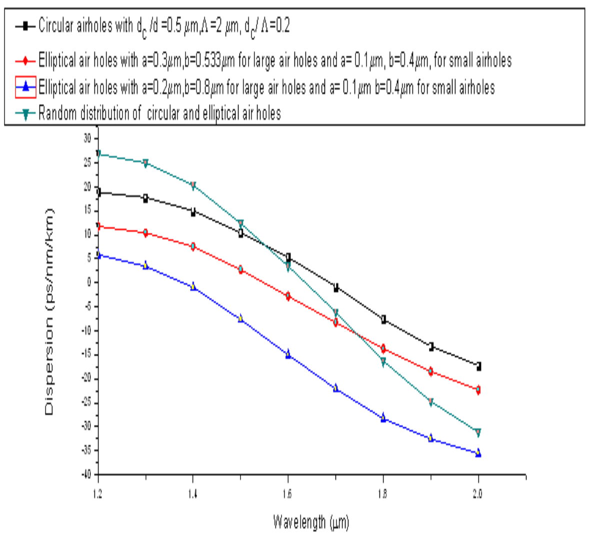


Fig 5 Waveguide dispersion of a hybrid square-lattice PCF varying ellipticity of air holes with $d_c/d = 0.5$,$d_c/ \Lambda = 2\mu m$, $\Lambda=2\ \mu m$, a=0.1 µm(red, blue)0.2 µm (blue),0.3 µm (red)and b=0.4 µm(red, blue),0.533 µm(red),0.8 µm(blue)

The dispersion properties of square lattice PCFs with elliptical air holes has been compared with circular air holes keeping the dimensions of the square Lattice same in Fig 5.The plots shows that dispersion gets decreased to -7.65905 ps / (nm.km) with increase in the ellipticity of air holes over the wavelength range 1.2 µm to 2 µm(incorporating optical range). Also Zero dispersion wavelength is shifted towards lower wavelength range. Table I shows a comparison Of Dispersion values at 1.5µm for all the stated configurations.

Table II- Zero Dispersion Wavelength

	Conf I	Conf II	Conf III	Conf IV
Zero Dispersion Wavelength	1.69 µm	1.475 µm	1.38 µm	1.645 µm

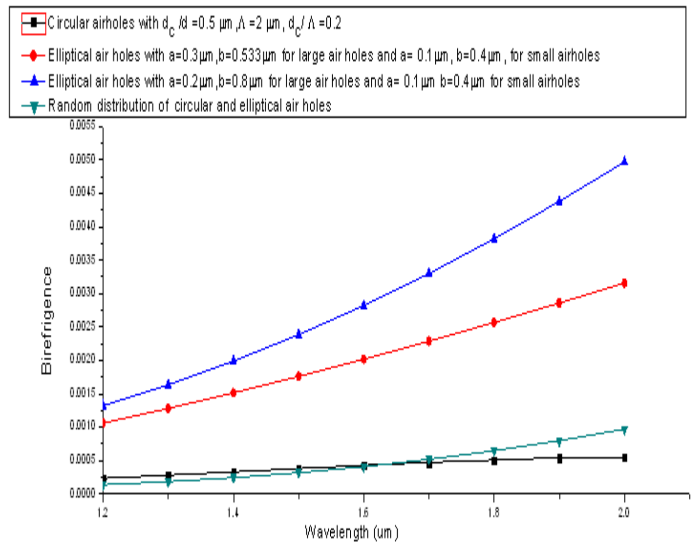


Fig 6 wavelength dependence of the modal birefringence of a hybrid square-lattice PCF Here with $d_c/d = 0.5$,$d_c/ \Lambda = 2\mu m$, $\Lambda=2\ \mu m$, a=0.1 µm (red, blue), 0.2 µm (blue), 0.3 µm (red) and b=0.4 µm(red, blue), 0.533 µm (red), 0.8 µm(blue)

The modal birefringence of a hybrid square-lattice PCF of elliptical air holes with $d_c/d = 0.5$, $d_c/ \Lambda = 2\mu m$, $\Lambda=2\ \mu m$, a=0.1 µm, b=0.4 µm for small air holes and a=0.2 µm, b=0.8 µm for large air holes as shown in Fig.6 increases to 4.9 $\times 10^{-3}$, comparing with the conventional square-lattice PCF with circular air holes 5.5 x 10^{-4}.

4. Conclusion

A Hybrid Square–lattice PCF with elliptical and circular air holes are compared and investigated for their dispersion and birefringence properties, with the design in fig 3(Conf III) , dispersion is decreased to -7.65905 ps / (nm.km) and zero dispersion wavelength gets shifted to 1.38 µm with the increase in the ellipticity of air holes. Thus the Zero dispersion wavelength is shifted towards lower wavelength range, the modal birefringence of a hybrid square-lattice PCF with elliptical air holes is investigated to be equal to 4.9 x10^{-3} at 2 µm wavelength which is very high as compared to circular air hole PCF. Therefore that the proposed Hybrid Square Lattice PCF with elliptical air hole can be used as high birefringence and low dispersion fibers.

References

[1] B. T. Kuhlmey, “Microstructured optical fibers: where is the edge?”Opt. Express, vol. 10, No. 1, pp. 54-59, Jan. 2002.

[2] Knight JC, Birks TA, Russell PSJ, Atkin DM. All-silica single mode optical fiber with photonic crystal cladding. Opt Lett1996;21:1547–9.

[3] Broeng J, Mogilevstev D, Barkou SE, Bjarklev A. Photonic crystal fibers: a new class of optical waveguides. Opti Fiber Technol 1999; 5:305–30.

[4] Reeves WH, Knight JC, Russell PStJ.Demonstration of ultraflattened dispersion in photonic crystal fibers. Opt Express2002;10:609–13.

[5] Birks TA, Knight JC, Russel PSJ. Endlessly single-mode photonic crystal fiber.Opt Lett 1997; 22:961–3.

[6] Saitoh K, Koshiba M. Leakage loss and group velocity dispersion in air-core photonic bandgap fibers. Opt Express 2003;11:3100–9.

[7] Knight JC,Broeng J, Birks TA, Russell PSJ. Photonic bandgap guidance in opticalfiber.Science 1998; 282:14768.

[8] Philip R. Photonic crystal fibers. Science 2003; 299:358–62.

[9] A. Ortigosa-Blanch, J. C. Knight, W. J. Wadsworth, J. Arriaga, B. J. Mangan, T. A. Birks, and P. S. J. Russell, “Highly birefringent photonic crystal fibers,” Opt. Lett. 25, 1325-1327(2000).

[10] J. Ju, W. Jin, M. S. Demokan, “Properties of a Highly BirefringentPhotonic Crystal Fiber,” IEEE Photon. Technol. Let15,1375-1377(2003).

[11] M. J. Steel, P. M. Osgood. Jr, “Elliptic-hole photonic crystalfibers,”Opt.Lett.26,229-231(2001).

[12] M. J. Steel and R. M. Osgood, “Polarization and dispersive propertiesof elliptical-hole photonic crystal fibers,” J.LightwaveTechnol.19,495-503(2001)

[13] J.Noda, K. Okamoto, and Y. Sasaki, “Polarization-maintaining fibers and their applications,”J.Lightwave Technol.4, 1071-1089(1986).

[14] M. Y. Chen and R. J. Yu, “Polarization properties of elliptical-hole rectangular lattice photonic crystal fibers,” J. Opt.A6, 512-515(2004) .

[15] Y. C. Liu and Y. Lai, “Optical birefringence and polarization dependent loss of square-and rectangular-lattice holey fibers with elliptical air holes: numerical analysis,” Opt.Express13,225-235(2004).

[16] A. Ferrando, and J. J. Miret, “Single-polarization single-mode intraband guidance in supersquare photonic crystal fibers,” Appl. Phys. Lett. 78, 3184-3186 (2001).

[17] Soan Kim, Chul-Sik Kee, Chung Ghiu Lee,”Hybrid Square-lattice Photonic Crystal Fiber” IEEE 978-1-14244-3830-3/09(2009).

[18] Broeng J, Mogilevstev D, Barkou SE, Bjarklev A. Photonic crystal fibers: a new class of optical waveguides. OptFiberTechnol.1999; 305.

[19] Ferrando A, Silvestre E, Andre´ s P, Miret JJ, Andre´s MV. Designing the properties of dispersion-flattened photonic crystalfibers.Opt.Exp.2001 ; 687.

[20] Shen LP, Huang W-P, Jian SS. Design and optimization of photonic crystal fibers for broadband dispersion compensation. IEEE Photon. Technol. Lett. 2003 ; 540.

Image Interpretation Based On Similarity Measures of Visual Content Descriptors – An Insight

Mungamuru Nirmala
Lecturer, Department of Computer Science,
Eritrea Institute of Technology, Asmara, State of Eritrea.
Email:nirmala.mungamuru@gmail.com

Kaliyaperumal Karthikeyan,
Lecturer, Department of Computer Science,
Eritrea Institute of Technology, Asmara, State of Eritrea.
Email:kirithicraj@gmail.com

Sreedhar Appalabatla
Faulty, Department of Information and Communication Technology
Zoba Maekel, Ministry of Education, Asmara, Eritrea, North East Africa
Email: appalabatla.s@gmail.com

Raja Adeel Ahmed
Lecturer, Department of Computer Science,
Eritrea Institute of Technology, Asmara, State of Eritrea.
Email: adeelrajj@yahoo.com

Abstract: *Efficient and effective retrieval techniques of images are desired because of the explosive growth of digital images. Content based image retrieval is a promising approach because of its automatic indexing and retrieval based on their semantic features and visual appearance. Interest in the potential of digital images has increased enormously over the last few years. Content-Based Image Retrieval (CBIR) is desirable because most webs based image search engines rely purely on the meta-data which produces a lot of garbage in the result.*

In the content-based image retrieval, the search is based on the similarity of the content of the images such as color, texture and shape. However, "a picture is worth a thousand words." Image contents are much more versatile compared with text, and the amount of visual data is already enormous and still expanding very rapidly.

Most content based image retrieval systems focus on overall and qualitative similarity of scenes. Conventional information retrieval is based solely on text, and these approaches to textual information retrieval have been transplanted into image retrieval in a variety of ways, including the representation of an image as a vector of feature values. By measuring the similarity between image in the database and the query image using a similarity measure, one can retrieve the image. This paper describes Content-Based Image Retrieval methods using visual content descriptors.

Keywords: *CBIR (Content Based Image Retrieval), visual content descriptors, similarity measure, Digital image, (CBVIR) Content-Based Visual Information Retrieval, TBIR (Text Based Image Retrieval)*

1. Introduction

Content-Based Visual Information Retrieval (CBVIR) is an application of the computer vision to the problem of digital pictures retrieval in a large data base. Information contained in an image can be visual information or semantic information. The visual information can be stated in general contexts in form of colors, textures, shapes, spatial relations, or in other specified forms which valid in the domain of certain problems. CBIR is a retrieval technique which uses the visual information by retrieving collections of digital images. The visual information are then extracted and stated as a feature vector which in the sequel then forms a feature database.

The retrieval of data stored in the form of text or documents is carried out by giving the keywords in the search engine. The keywords are compared with the text of the documents in the database. Based on the degree of comparison, the most pertinent documents are retrieved. Since 1990s the content-based image retrieval system has been a fast advancing research area and remarkable progress has been achieved in theoretical research and in developing the retrieval system. The ideal CBIR system from a user perspective would involve what is referred to as *semantic* retrieval. An advance in CBIR was marked by commercial development of image retrieval system for government organization, private institutions and hospitals.

2. Content Based Image Retrieval

CBIR has been an interesting topic of research that attracts many researchers since the early of 90's. In the last decade, a lot of progress attained in theoretical research CBIR or in the development of the CBIR system. But, up to now there are still many challenging problems in the field of CBIR which attracts attention of many scientists from various disciplines.

Late of 1970's is the beginning era of research in CBIR. On 1979 there was conference on application of database technique in image, held in Florence. Since then, the application of image database management technique became an area of research that attracted many scientists. The technique used in the beginning of CBIR did not use the visual content yet, but relied on the textual information of each image -Text Based Image Retrieval (TBIR). On the other hand, additional textual information is needed of every image before retrieval. Then the images can be retrieved by using the textual approach DBMS. Text based image retrieval uses the traditional database technique to manage the image database. Through textual description, image can be organized based on topic or level of hierarchies to make the navigation process and browsing easier.

Generally in an image retrieval system, the visual content of images is stored in a multi dimension feature vector. To retrieve an image, users enter inputs of the form image query or sketch. Then the CBIR system computes the feature vector of the query images or sketch. Similarity between the feature vectors of query image/sketch is obtained based on a measure of distance or index scheme.

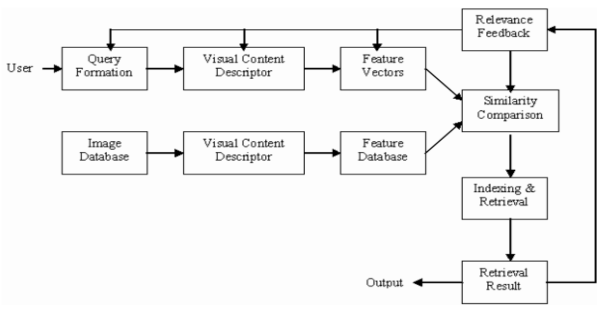
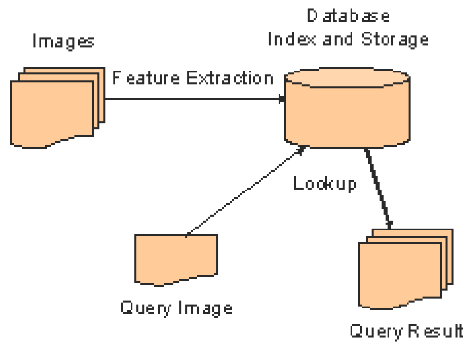


Fig 1: Content based image retrieval System overview

Some of the commercial CBIR systems are IBM's QBIC, Virage's VIR Image Engine, and Excalibur's Image Retrieval Ware. The current CBIR system is limited to effectively operate at the primitive feature level. They are not efficient to search for some semantic queries. For example, say, a photo of a cat. But some semantic queries can be handled by specifying them in terms of primitives.

3. Literature Review

An image is called 'the same' with an image in the database if the value of similarity measure is 'small'. This means that a good CBIR retrieval system must be



supported by an accurate similarity measure. The Robust Distance for Similarity Measure of Content Based Image Retrieval, Dyah E. Herwindiati and Sani M Isa [3]

Fig 2: Typical flow of CBIR

Shape matching is an important ingredient in shape retrieval, recognition and classification, alignment and registration, and approximation and simplification. Various aspects that are needed to solve shape matching problems: choosing the precise problem, selecting the properties of the similarity measure that are needed for the problem, choosing the specific similarity measure, and constructing the algorithm to compute the similarity.
Remco C. Veltkamp, "Shape Matching: Similarity Measures and Algorithms," smi, pp.0188, International Conference on Shape Modeling & Applications, 2001[7]

4. Visual Contents

Visual contents are divided as primitive features and semantic features. Primitive features are the low-level visual features such as color, texture, shape and spatial relationships - directly related to perceptual aspects of image content. It is usually easy to extract and represent these features and fairly convenient to design similarity measures. System of image retrieval generally stores visual contents of an image in a multi dimension feature vector. It is a very challenging task to extract and manage meaningful semantics and to make use of them to achieve more intelligent and user-friendly retrieval. In an image retrieval process, users enter inputs of the form query image. Depending upon the image a suitable content can be chosen and a descriptor is defined for retrieval.

4.1 Content Descriptor

A *descriptor* is the collection of features or attributes of an object. A good visual content descriptor must be invariant to the variance caused by the process of image formation or local. The global descriptor makes use visual information from the whole images, meanwhile the local descriptor makes use visual information of image region to describe the visual content of images. Any object or image should satisfy the criteria of both local and global features. To obtain the local descriptors, the image is partitioned into parts of equal size and shape. A better method is to use some criteria to divide the image into homogeneous regions or a region segmentation algorithm. A complex method of dividing an image is to consider a complete object segmentation to obtain semantically meaningful objects like dog, ball, donkey etc. Depending upon the image, the description of the global features and the local features will differ with respect to the visual content under consideration.

4.2 Feature Vectors

CBR system computes the feature vector of the query image. Similarity between the feature vectors of the query images is obtained based on the measurement of distance of index scheme. The features of the visual contents are extracted from the image. The collection of features of the contents is known as a *feature vector*. Therefore, they are multi-dimensional vectors. Feature vectors describe particular characteristics of an image based on the nature of the extraction method. For feature extraction, a number of extraction algorithms have been proposed. Visual feature extraction is the basis of any content-based image retrieval technique. For example, properties of a bounding box, a curvature, spherical functions etc. The feature vectors are used for indexing and image retrieval using a similarity measure. The collection of feature vectors is termed as *feature database* of the images in the database. Feature vector and the different approaches on feature-based similarity search techniques are discussed in [4] and [5].

5. Color

Color provides a significant portion of visual information to human beings and enhances their abilities of object detection. Each pixel in an image can be represented as a point in a 3D color space. The common color spaces used for image retrieval are RGB, CIE L*a*b*, CIE L*u*v*, HSV (or HSL, HSB), and opponent color space. The perceptual attributes of color are brightness, hue and saturation. Brightness is the luminance perceived from the color. Hue refers to its redness, greenness etc. Uniformity of a color is the desirable characteristics of an appropriate color space for image retrieval. If two color pairs are equal in similarity distance in a color space then they are perceived as equal by the viewer.

5.1 Color Descriptor

Color is widely used for content-based image and video retrieval in multimedia databases. The retrieval method based on the content, *color*, uses the similarity measure of features derived from color. For example, the proportion of the colors in the images is computed for the sample image. Then the images in the database having more or less the same proportion of the colors can be retrieved using the similarity measures. The descriptors are defined based on the attributes of color. The color descriptors are Color Histogram, Color Moments, Color Coherence Vector, and Color Correlogram. The descriptors color histogram and color moments do not include spatial color distribution.

5.2 Color Histogram

Color histogram is the most commonly used descriptor in image retrieval. Images added to the collection are analyzed to compute the color histogram, which shows the proportion of pixels of each color within the image. The color histogram is stored in the database. The user has to specify the proportion of each color to search for a desired image. The user can also input an image from which a color histogram is calculated. The matching process retrieves those images whose color histograms match with those of the query images very closely.

Further the retrieval system includes: (i) the definition of adequate color space with respect to a specific application (ii) an appropriate extraction algorithms (iii) evaluation of similarity measures.
The color histogram extraction algorithm can be divided into three steps:

(i) Partition of the color space into cells
(ii) Associate each cell to a histogram bin
(iii) Count the number of image pixels of each cell and store this count in the corresponding histogram bin.

The descriptor is invariant to translation and rotation.

5.3 Color Moments

Color moments have been used in many retrieval systems, when the image contains just the object. The first three moments have been proved to be efficient and effective in representing color distributions of images. They are defined as:

$$\mu_i = \frac{1}{N}\sum_{j=1}^{N} x_{ij}$$

$$\sigma_i = \left(\frac{1}{N}\sum_{j=1}^{N}(x_{ij}-\mu_i)^2\right)^{1/2}$$

$$s_i = \left(\frac{1}{N}\sum_{j=1}^{N}(x_{ij}-\mu_i)^3\right)^{1/3}$$

Where x_{ij} is the value of the ith color component of the image pixel j and N is the number of the pixels in the image.

5.4 Color Coherence Vector (CCV)

This descriptor incorporates spatial information into the color histogram. Each pixel of the image is classified as either coherent or incoherent, depending upon whether or not it is a part of a “large” similarly-colored region. A region is assumed to be large if its size exceeds a fixed user-set value. By counting coherence and incoherent pixels separately, the method offers a finer distinction between images than color histograms.
For each color c_i the number of coherent pixels, α_{ci}, and the number of non-coherent pixels, β_{ci}, are computed; each component in the CCV is a pair (α_{ci}, β_{ci}), called a *coherence pair*. The coherence vector is given by

$$V_c = <(\alpha_{c1},\beta_{c1}),(\alpha_{c2},\beta_{c2}),...,(\alpha_{cN},\beta_{cN})>$$

The sum $\alpha_{ci} + \beta_{ci}$ is the number of pixels of color c_i present in the image; the set of sums for i = 1,2,…,N represents the color histogram.

5.5 Color Correlogram

The color correlogram encodes the spatial correlation of colors. The first and second dimension of the three-dimensional histogram is the colors of any pixel pair and the third dimension is their spatial distance. The color correlogram is a table indexed by color pair, where the k-th entry for <i, j> specifies the probability of finding a pixel of color j at a distance k from a pixel of color i, in the image.
Let I represent the entire set of pixels of the image and $I_{c(i0}$ represent the set of pixels whose colors are *c(i).*
The color correlogram is defined as:

$$r_{i,j}^{(k)} = \Pr_{p_1 \in I_{c(i)}, p_2 \in I} [p_2 \in I_{c(j)} / | p_1 - p_2 | = k]$$

Where i, j $\in$ {1, 2,…, N}, k$\in$ {1,2,…,*d*} and $|p_1-p_2|$ is the distance between pixels p_1 and p_2. If all possible combinations of color pairs are considered the size of the color correlogram will be very large ($O(N^2, d)$). But instead, color auto-correlogram captures the spatial correlation between identical colors, hence the size is reduced to $O(Nd)$. Even though the auto-correlogram is having high dimensionality when compared to color histogram and CCV, its computational cost is very high.

6. Texture

Texture is a low-level descriptor for image search and retrieval applications. There are three texture descriptors considered in MPEG-7. It describes spatial relationships among grey-levels in an image. Texture is observed in the structural patterns of surfaces of objects such as wood, grain, sand, grass and cloth. The ability to match on texture similarity can be useful in distinguishing between areas of images with similar color such as sea and sky, grass and leaves. A variety of techniques have been used for measuring texture similarity. A technique named as modeling-from-reality [5] has been proposed for creating geometric models of virtual objects, and is used for texture mapping of color images. The color and texture descriptors [6] of MPEG-7 standard are described and the effectiveness of these descriptors in similarity retrieval is also evaluated.

6.1 Texture Descriptors

A basic texture element is known as texels. A Texel contains several pixels. The placement of the texel could be periodic or random. Natural textures are random and artificial textures are periodic or deterministic. Texture can be smooth, regular, irregular, linear, rippled etc. Texture is broadly classified into two main categories, namely *Statistical* and *Structural*. Textures that are random in nature are suitable for statistical categorization. Structural textures are deterministic texels, which repeat according to some placement rules, deterministic or random. A Texel is isolated by identifying a group of pixels having certain invariant properties, which repeat in the given image. The pixel can be defined by its gray level, shape, or homogeneity of some local property like, size, orientation or concurrence matrix. The placement rules are the spatial relationship of the pixels. In deterministic rule, the spatial relationships may be expressed in terms of adjacency, closest distance, etc and the texture is said to be strong.
For randomly placed texels, the associated texture is said to be weak and the placement rules may be expressed in terms of edge density, run length of maximally connected texels, relative extreme density. Apart from these two classifications there is another model called *Mosaic model*, which is a combination of both statistical and structural approaches.
The texture is described based on its features such as, coarseness, contrast, directionality, likeliness, regularity and roughness. These features are designed based on the psychological studies on the human perception of texture.

6.2 Coarseness

Coarseness is a measure of the granularity of the texture. Coarseness is calculated using moving averages. Let $M_k(x, y)$ denote the moving average at each pixel (x, y) in the window of size 2^k x $2^{k,}$ where k = 0, 1, …, 5)

$$M_k(x,y) = \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} g(i,j)/2^{2k}$$

Where g(i, j) is the pixel intensity at (i, j). The differences between the pairs of non-overlapping moving averages in the horizontal and vertical directions for each pixel are computed. The value of k, which maximizes, the difference calculated in either direction is used to set the best size (say) *S* for each pixel.

$$S_{best}(x, y) = 2^k$$

The coarseness is computed by averaging S_{best} over the entire image.

$$coarseness = \frac{1}{mxn}\sum_{i=1}^{m}\sum_{j=1}^{n} \mathrm{S_{best}}(i, j)$$

6.3 Contrast

Contrast may be defined as the difference in perceived brightness. Detection of light spots depends on the brightness, size of the space and duration as well as the contrast between the spot and the background.

$$contrast = \frac{\sigma}{\alpha_4^{1/4}}$$

Where the kurtosis $\alpha_4 = \mu_4/\sigma^4$, μ_4 is the fourth central moment, and σ^2 is the variance. The contrast can be computed using this formula for both the entire image and a region of the image.

7. Shape Retrieval

The shape of an object refers to its profile and physical structure. These characteristics can be represented by the boundary, region, moment, and structural representations. Shapes are divided into two main categories namely, static shapes and dynamic shapes. Static shapes are rigid shapes. They do not change due to deformation or articulation. For example the shape of a rigid object like a water jug is a static shape. The object like human face is a dynamic shape as the shape of the human face changes with the change in expressions and actions of the human being. A number of features of the object shape can be computed for each stored image. The same set of features is computed for the query image. The images that are having close similarity with the query image are retrieved from the database. The various aspects that are needed to solve shape matching problems like choosing precise problem, selection properties of similarity measure are stated in [7].

7.1 Shape Descriptors

Shape descriptors are classified into boundary-based and region-based methods. This classification takes into account whether shape features are extracted from the contour or from the whole shape region. That can be further divided into structural (local) and global descriptors. If the shape is represented by segments or sections, it is structural and if it is from the whole shape region, it is global. Another classification categorizes the shape description into spatial and transform domain techniques, depending on whether direct measurements of the shape are used or a transformation is applied. The literature on content-based retrieval methods are evaluated [8] with respect to several requirements of the retrieval system.

7.2 Shape Matching

The retrieval method based on shape can be divided into three broad categories: (1) feature based methods, (2) graph based method and (3) other methods.
Feature Based Methods: The shape feature can be classified into regenerative features and measurement features. Boundaries, regions, moments, structural and syntactic features are identical to the regenerative features. Geometry and Moments are paired with measurement features. In 3D shapes, features denote geometric and topological properties of 3D shapes. Based on the type of shape feature used, the feature based method can be divided into: (1) global features, (2) spatial maps, (3) global feature distributions and (4) local features. The first three represent features of a shape using a single descriptor. The descriptor is a vector of *n*-dimension, and *n* is fixed for all shapes.
The global features are used to characterize the overall shape of the objects. These methods use the global features like that of area; volume, statistical moments, and Fourier transform coefficients. These methods do not discriminate above the object details. These methods support the user feed back. The global feature distribution method is a refinement of the global feature method. Spatial maps are representations that use the spatial location of an object. The entries in the map are locations of the object and are arranged in the relative positions of the features in an object. Local features are derived from the segment or part of the image. At the outset, the image is partitioned using a suitable criterion into equal sizes or meaningful objects. Then based on the features, similarity is measured.

Graph Based Methods: In graph based method, the geometric meaning of a shape is extracted and a graph is constructed. The graph represents the shape components and the links. The graph based method includes (1) model graph, (2) Reeb graph, and (3) Skeletons. Model based graph are useful to 3D solid models created using CAD systems. This method is difficult to find the similarity for models of natural shapes like humans and animals. Because of the fact that the shape should be solid, it is difficult to represent the natural shapes as similar to a sphere, a cylinder, or a place. The skeletal points are connected in an undirected acyclic shape graph, using *Minimum Spanning Tree algorithm*. The shape information of the 3D objects [9] are used to form a skeletal graph and a graph matching technique is used for retrieval Moreover, as an extension of the skeleton method of shape comparison the objects may be segmented as semantically meaningful parts (shapes), the skeleton of the parts can be used for similarity measure. The skeleton of the parts can be converted into graph, and graph matching techniques can be used for comparison.

8. Similarity Measure

The measure of similarity between two objects is obtained, based on the distance between pairs of descriptors using a dissimilarity measure. If the distance is small, then it means small dissimilarity and large similarity. The dissimilarity measure can be defined as a non-negative valued function.
Let *d* be the dissimilarity measure on a set S. *d*: S x S → R+ U {0}. The following properties are defined on *d*.

(i) Identity: For all x Є S, *d*(x, x) = 0
(ii) Positivity: For all x, y Є S, *d*(x, y)>0, where x ≠y
(iii) Symmetry: : For all x, y Є S, *d*(x, y) = *d*(y, x)
(iv) Triangle Inequality: For all x, y, z Є S, *d*(x, z) = *d*(x, y) + *d*(y, z)
(v) Transformation Invariance: For any chosen transformation group T,
for all x, y Є S, t Є T, *d*(t(x),t(y)) = *d*(x, y).

The identity property states that the descriptor is completely similar to itself. The positive property implies that different descriptors are never completely similar. This is a very strong property for a high-level descriptor and it is rarely contented. This will not affect the result much if the dissimilarity is on the negligible part of the image.
According to human perception, a visual content of the image, say, shape is not always similar. Human perception does not find a shape x similar to y, as y is similar to x. If partial matching of objects is used, the Triangle Inequality is not satisfied. Since the part of the object is matched the distance between the object would be very small.
Transformation Invariance should be satisfied in all types of descriptors, as the comparison and the extraction process of the descriptors are independent of the place, orientation and scale of the object in the Cartesian coordinate system. If a dissimilarity measure is affected by any transformation, an alternative formulation may be used. For example, (v) can be defined as

(v) Transformation Invariance: For any chosen transformation group T,
for all x,y Є S, t Є T, *d*(t(x),y) = *d*(x,y).

If all the properties (i)-(iv) hold, then dissimilarity is called a *metric*. It is called *pseudo-metric* if (i), (iii) and (iv) hold and *semi-metric* if only (i), (ii) and (iii).
Future Work:
The future study involves in deriving a retrieval method for natural objects in images. It includes comparison of different similarity search methods and the feature extraction methods for the shape of the objects in the image.

Some of the Challenges are:

- Semantic gap
    - The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation.
    - User seeks semantic similarity, but the database can only provide similarity by data processing.
- Huge amount of objects to search among.
- Incomplete query specification.
- Incomplete image description.

9. Conclusion

In this paper, the basic concept of Content-Based Image Retrieval methods using visual content is described. The descriptors should be transformation invariance to measure the similarity between any two descriptors. Instead of comparing the query image as a whole with the images in the database, the descriptors are compared for retrieval. With an extension of this comparison the feedback of the retrieved images can be used to further refine the retrieval process. Based on the feedback the descriptor can be redefined and an iterative similarity checking can be implemented to improve the retrieval of proper images with reference to the query image. Another approach in CBIR system is that a priori feature extraction is defined. The features are selected from the predefined set of features. This method uses a set of primitive features and logical features. Based on both primitive and logical features, the query is processed. This may result in the reduction of cost in feature extraction.

References

1. Eakins, John & Margaret Graham: Content-Based Image Retrieval. *JISC Technology Applications.* Report 39: 1-65 (1999).

2. Benjamin, Bustos, Keim Daniel, Saupe Dietmar & Schreck Tobias: Content-based 3D Object Retrieval. (2007).

3. Isa Dyah E. Herwindiati and Sani M: The Robust Distance for Similarity Measure of Content Based Image Retrieval. Proceedings of the World Congress on Engineering 2009.

4. Vol II WCE 2009, July 1 - 3, 2009, London, U.K.Benjamin, Bustos, Keim Daniel, Saupe Dietmar, Schreck Tobias & Dejan V. Vranic: Feature-Based Similarity Search in 3D Object Databases. *ACM Computing Surveys (CSUR)*, 37: 345-387 (2005).

5. Benjamin, Bustos, Keim Daniel, Saupe Dietmar & Dejan V. Vranic: An Experimental Effectiveness Comparison of Methods for 3D Similarity Search. *International Journal on Digital Libraries, Special Issue on Multimedia Contents and Management*, 6(1): 39-54. (2006).

6. Kurazume Ryo, Ko Nishino, Zhengyou Zhang & Katsushi Ikeuchi: Simultaneous 2D images and 3D geometric model registration for texture mapping utilizing reflectance attribute. Paper presented in the *5th Asian Conference on Computer Vision*, 23–25 Melbourne, Australia (2002).

7. Manjunath, B. S., Jens-Rainer Ohm, Vinod V. Vasudevan, Akio Yamada: Color and Texture Descriptors. *IEEE* Transactions on Circuits and Systems for Video Technology 11/6: 703-714 (2001).

8. Remco C. Veltkamp, "Shape Matching: Similarity Measures and Algorithms," smi, pp.0188, International Conference on Shape Modeling & Applications, 2001 Veltkamp. C. Remco (2001) Shape Matching: Similarity Measures and Algorithms *SMI* (2001).

9. Tangelder, Johan W.H. & Remco C. Veltkamp: A Survey of Content Based 3D Shape Retrieval Methods.
www.cs.princeton.edu/~funk/cacm05.pdf (2005).

Modified Anderson Darling Test for Wind Speed Data

Atul Viraj Wadagale[1], P.V. Thatkar[2], R.K. Dase[3] and D.V. Tandale[4]

[1] Assistant Professor (Statistics), MIMSR Medical College, Latur (MS), INDIA
[2] Lecturer, MIT College of Arts & Science, Alandi (D), Pune (MS), INDIA
[3] Assistant Professor (Statistics), MGM Medical College, Aurangabad (MS), INDIA
[4] Assistant Professor, Padmashree Dr. D.Y. Patil University, Navi Mumbai (MS), INDIA

Corresponding authors Email: atulvw@gmail.com, thatkar@indiatimes.com

Abstract: In this article we collect daily wind speed of three stations in Marathwada region for one year duration. By analyzing collected data it seems that Weibull distribution is best fitted for the wind speed data. The given data is compared by other continues distribution like Beta, Exponential, Gamma, Lognormal, Uniform. Maximum likelihood method is used for estimating parameters of weibull distribution. . Maximum likelihood method is used for estimating parameters of weibull distribution. We calculate Goodness of fit of these continuous distributions by comparing Anderson Darling test with modified Anderson Darling test.

Keywords: weibull distribution, beta distribution, Modified Anderson Darling normality test, maximum likelihood method, wind speed data.

Introduction:

India is fastest growing and developing country. Population of India is nearly equal to China. So, need of basic things demanded more like energy but the present energy sources are limited for used.[7] In Marathwada region (situated in middle of Maharashtra state with eight districts) people face 8-12 hours load shading in rural & urban area due to shortage of regular energy resources. This effect on their day-to-day life, agriculture and yield product and also on progress. As wind energy is renewable, environmental, cost effective energy so it may help for the people situated in this region. Wind speed is most important parameter of wind energy[2,5]. For evaluating the values of wind speed, we need an accurate probability distribution. As the values of wind speed are continuous one so we come to know that we must used continuous type of probability distribution such as, in literature, the Weibull distribution is used for wind energy. We compare Anderson-Darling normality test with Modified Anderson-Darling test. It can be shown that Weibull distribution is best fitted for wind speed data but, sometimes probability density function of wind speed data is not statistically accepted as weibull pdf so we used other continuous distribution like Beta, Exponential, Gamma, Lognormal and Uniform with Weibull pdf.

Material and Method:

We collected wind speed data from 3 different places from Marathwada region. We daily collect 1 sample from each station from January 2009 to Dec 2009. Therefore, there are 1035 total samples collected from all the stations. Table [1] shows the station, district and duration of records from 3 places.

Station	District	Latitude		Longitude		Annual Wind Speed	Duration
		Deg	Min	Deg	Min		
Kankara	A'bad	19	59	75	27	5.40	Jan'09-Dec'09
Sautada	Beed	18	48	75	20	5.72	
Rohina	Latur	18	27	76	57	5.57	

Table [1]: Study data from three stations in Marathwada region[1]

Statistical Distributions:

Here, we deal with different statistical continuous probability distributions with their probability density function.

Beta Distribution:

Probability density function for Beta distribution is,

$$f(x) = \frac{1}{B(\mu, v)} x^{\mu-1}(1-x)^{(v-1)}\ ;\ \ (\mu, v) > 0,\quad 0 < x < 1$$

Exponential Distribution:

Probability density function for Exponential distribution is,

$$f(x,\theta) = \theta e^{-\theta x}\,,\ x \geq 0$$

Gamma Distribution:

Probability density function for Gamma distribution is,

$$f(x) = \frac{e^{x}x^{\lambda-1}}{\Gamma(\lambda)}\ ;\ \lambda > 0, 0 < x < \infty$$

Uniform (Rectangle) Distribution:

Probability density function for Uniform distribution is,

$$f(x) = \frac{1}{b-a}\ ; if\ a < x < b$$

Lognormal Distribution:

The pdf of two parametric Lognormal distribution is given as follow:

$$f(x) = \frac{1}{x\sigma\sqrt{2\pi}} exp\left(-\frac{1}{2\sigma^2}(log(x)-\mu\right)^2$$

Weibull Distribution:

The analysis of wind speed data can be computed by weibull function. The formula for the probability density function of the general weibull distribution is,

$$f(x) = \frac{\gamma}{\alpha}\left(\frac{x-\mu}{\alpha}\right)^{(\gamma-1)} exp\left(-\left(\frac{(x-\mu)}{\alpha}\right)^{\gamma}\right)\ wherex \geq 0, \gamma > 0$$

Annual mean wind speed at 10m Ht	Index value of wind resource
Below 4.5(m/s)	Poor
4.5-5.4(m/s)	Marginal
5.4 – 6.7(m/s)	Good to very good
Above 6.7(m/s)	Exceptional

Table 2 : Significance of wind energy according to speed[3]

where γ is the shape parameter , μ is location parameter and α is the scale parameter. The case where μ= 0 & α=1 is called standard weibull distribution.
We put μ = 0, it is called the two-parameter weibull distribution.

The equation for the standard weibull distribution reduces to,

$$f(x) = \gamma x^{(\gamma-1)} exp\big(-(x^{\gamma})\big)\ where\ x \geq 0; \gamma > 0$$

Analysis showed that the result of wind speed could be represented by cumulative distribution function. The formula for cumulative distribution function of Weibull distribution is,

$$F(x) = 1 - e^{-(x^{\gamma})}\ where\ x \geq 0; \gamma > 0$$

Maximum likelihood method:

Maximum likelihood estimation begins with the mathematical expression known as a likelihood function of the sample data. The likelihood of a set of data is the probability of obtaining that particular set of data given the chosen probability model. This expression contains the unknown parameters. Those values of the parameter that maximize the sample likelihood are known as the maximum likelihood estimates.[4,6]

MLE for Weibull Distribution:

The scale and shape parameters were estimated using the method of maximum likelihood estimation (MLE) as follows:

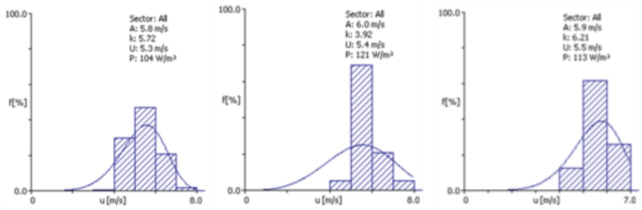
$$\prod_{i=1}^{n} P(t_i|\gamma,\alpha)$$

Taking log of this gives,

$$ln\left(\prod_{i=1}^{n}\big(P(t_i|\alpha,\beta)\big)\right)$$

Hence we derive the following expression describing the likelihood function,

$$\sum_{i=1}^{n} ln\big(P(t_i|\alpha,\beta)\big)$$



Figure(1): Histogram for three stations A'bad, Beed, Latur respectively
Report produced by WAsP OWC Wizard (version 2.1.2)

Method	Parameter	
MLE	Shape(γ)	Scale(α)
	9.9658	5.8488

Table 3: Estimation values of shape and scale parameters of weibull obtained with MLE

Goodness-of-fit test:

1) **Anderson-Darling Normality test:**

The A-D test for normality can be computed as follow:

H_0:The data follow the specified distribution.
H_A: The data do not follow the specified distribution.
The hypothesis regarding the distributional form is rejected at the chosen significance level (*alpha*) if the test statistic, A^2, is greater than the critical value obtained from a table.

AD = -N-S

where,

$$S=\sum_{i=1}^{N}\frac{2i-1}{N}\left[\ln F(Y_i)+ln\left(1-F(Y_{N+1-i})\right)\right]$$

therefore,

$$AD=-N-\frac{2i-1}{N}\left(ln\left(F(Y_i)\right)+ln\left(1-F(Y_N+1-i)\right)\right)$$

The critical values for the Anderson-Darling test are dependent on the specific distribution that is being tested. Tabulated values and formulas have been published for a few specific distributions (normal, lognormal, exponential, Weibull, logistic, extreme value type 1)[8,9]

Distribution	Anderson-Darling Test				Modified Anderson-Darling Test			
	AD_c	AD_α	p value	Result	AD_c^*	AD_α^*	p value	Result
Beta	2.76	2.49	0.0364	Reject	3.06	2.49	0	Reject
Exponential	111	2.49	0	Reject	123	2.49	0	Reject
Gamma	6.54	2.49	0.0005	Reject	7.26	2.49	0	Reject
Lognormal	14.5	2.49	0	Reject	16.11	2.49	0	Reject
Uniform	49.4	2.49	0	Reject	54.88	2.49	0	Reject
Weibull	1.36	2.49	0.213	Accept	1.51	2.49	0.0072	Accept

Table 4 : Test of Goodness of fit

Modified Anderson-Darling Normality Test:

The Modified A-D test is nothing but the imposed form of A-D test. It can state as follow:

$$AD^*=AD\left(1+\frac{0.75}{N}+\frac{2.25}{N^2}\right)$$

Station	Measured	Weibull fit	Discrepancy
A'bad	5.40	5.34	1.14%
Beed	5.72	5.42	5.24%
Latur	5.57	5.52	0.99%

Table 5: Site Description for three stations

Report produced by WAsP OWC Wizard (version 2.1.2)

Result:

The results shown in present study are statistically significant. The two parametric weibull distribution is best fitted for wind speed data. Annual average wind speed of three stations is more than 5 m/s while it is observed that the monthly mean wind speed data of three stations in Marathwada are fitted to the Weibull distribution. Weibull is best fitted distribution as considering to MLE. Weibull distribution is accepted in both the normality tests. It seems that Modified Anderson Darling test is better than Anderson Darling test because the outcome of modified A-D test is better than A-D test.

Acknowladgement:

Energy department, Govt. of Maharashtra, India.

References:

1. Energy department, Govt. of Maharashtra, India.
2. Stevens, M. and Smulders, P. [1979], The estimation of the parameters of the Weibull wind speed distribution for wind energy utilization purposes, *Wind Engng* 3, 132-145.
3. Ramachandra, T.V.; Subramanian, D.K. and Joshi, N.V. [1997], Wind Energy Potential Assessment in UttaraKanada, District of Karnataka, India, *Renewable Energy* 10(4), 585-611.
4. Jaynes, E. T. [1983] Papers on Probability, Statistics, and Statistical Physics. Netherlands: *D. Reidel, Dordrecht*.
5. Li M., Li X.[2005] Investigation of wind characteristics and assessment of wind energy potential for Waterloo region, *Energy conversion management*, 46, 3014-3033.
6. Gracia, A., Torres, J.L., Prieto, E., and DeFrancisco, A. [1998]. Fitting wind speed distribution: A case study, *Solar Energy* 2, 139-144.
7. Veysel YILMAZ, H. Eray CELIK [2008]. A Statistical approach to estimate the wind speed distribution: The case of Gelibolu region, *Doğuş Üniversitesi Dergisi* 9(1),122-132.
8. Anderson, T. W. and Darling, D. A. [1954]. A test of goodness of fit, *J. Amer. Statist. Assoc.*, 49, 765–769.
9. Chen, C. [2006]. Test of fit for the three-parameter lognormal distribution, *Computational Statistics & Data Analysis*, 50, 1418-1440.

Real Time Object Tracking Using Image Based Visual Servo Technique

M. A. El-Bardini, * E. A. Elsheikh, and M. A. Fkirin

Dept. Of Industrial Electronics and Control Eng. Faculty of Electronic Engineering, Menouf, 32952, Menoufya University, Egypt.
dr_moh2000@yahoo.com, mafkirin@yahoo.com
* *Corresponding author. Tel.:+2-048-366-0716 ; fax:+2-048-366-0716;* e-mail: eng_emadelsheikh@yahoo.com

Abstract: In this paper a practical real time object tracking using Image Based Visual Servoing (IBVS) is implemented. The proposed algorithm uses the modified frame difference as an object detection technique integrated with (area, center of gravity and color) as a feature based to segment the detected object. The real position of the interested object is estimated from its features based on camera calibration. A DC-motor position and direction Fuzzy Logic Position Controller (FLC) is implemented to maintain this object in the center of the camera view. Experimental results show that, the developed design is valid to detect and track objects in real time.

Keywords: Real Time Object Tracking, Visual Servoing, Image Processing, Fuzzy Logic Controller, Position Control, DC Motor.

1. Introduction

The Object tracking using camera is an important task within the field of computer vision. The increase of high-powered computers, the availability of high quality inexpensive video cameras and the increasing need for automated video analysis has generate a great deal of interest in object tracking algorithms [1][2]. The real time tracking algorithm consists of two modules; the **first** module is designed to acquire the images, detect the interested moving objects, segment the object or track of such object from frame to frame and estimate the position of this object, then deliver this position as a desired value to the **second** module, which is designed as a *position controller* to maintain the object in the camera view [2][9]. In this case, the information extracted from the vision sensor is used as a feedback signal in servo control, this called Visual Servoing (VS), also known as Vision Based Control [3][4]. The Visual Servoing techniques are broadly classified into, Image Based (IBVS) and Position Based visual servoing (PBVS) [3][4][5][6]. In PBVS, the image features are extracted from the image and a model of the scene and the target is used to determine the pose of the target with respected to the camera [4][7]. While in IBVS, it is also referred to as a feature based technique, because the algorithm uses the features extracted from the image to directly provide a command to motors. And the control law is directly expressed in the sensor space (image space). Many authors consider that the IBVS approach is better of the two, because the image-based approach may reduce computational delay, eliminate the necessity for image interpretation and eliminate errors in sensor modeling and camera calibration [5][6][8].

Various approaches to detect and segment objects has been proposed for visual servoing including, *statistical model based*, *Template-based*, *Background subtraction* based, *feature-based*, *gradient based*, and *Optical flow* object detection methods [10][11][12]. Statistical models are quite computationally demanding in most cases as mentioned in [11]. Template-based method, carries both spatial and appearance information. Templates, however, is suitable only for track objects whose poses does not vary considerably during the course of tracking, where the object appearance is generated from single view [2][13]. Feature-based and Gradient-based or frame difference methods are affected by noise due to intensity changes and some false detection appears in the segmented image [2][11][14].

After segment the object, the camera calibration technique is used to estimate the position of the object [3][15][16]. This position is used as a desired position to the servo controller. The objective of the controller in a vision control system is to maintain a position set point at a given value and able to accept new set point values dynamically. Modern position control environments require controllers are able to cope with parameter variations and system uncertainties [18-21]. One of the intelligent techniques is a Fuzzy Logic [22][23]. The Fuzzy controllers have certain advantages such as; simplicity of control, low cost and the possibility to design without knowing the exact mathematical model of the process [17][24][25].

This work, describes real time vision tracking system which uses a closed-loop control algorithm to track static and moving objects using a single camera mounted on a DC-motor driven platform based on IBVS. The algorithm is designed as two stages, the *first stage* depends on the frame difference with modification in threshold method integrated with (color, area and center of gravity (cog)) as features of the object, to detect and segment moving and static objects. The *second stage* is the proposed servo controller which is designed as a Proportional Derivative Fuzzy Logic Controller (PD-FLC) to overcome the non-linearity in the system. The system was tested in different cases such as; static object static camera, moving object static camera and moving object moving camera for single and multi objects. The practical results show the effectiveness of the proposed technique.

This paper is organized as follows: Section2 describe the problem statement. The Image analysis and feature extraction are explained in Section3, while in Section4 the proposed PD-FLC position controller is illustrated. The practical implementation and results are shown in Section5, and finally conclusions are given in section6.

2. Problem statement

To track moving objects in real time, there are two problems are affects in the system:

1. Problem due to intensity changes

The false detections are appeared in segmented images due to intensity changes.

2. Computational time Problem;
   2.1. The time required to detect moving objects is minimized by using frame difference method, because its time is smaller than compared to optical flow methods [26] and Statistical models [11].
   2.2. The time required to maintain the object in the center of camera view, is minimized by using PD-FLC as a position controller because it accelerate the transient response of the system .Also the Triangular type membership functions are used to fuzzify their inputs and output due to computation simplicity and ease in real time [17][24]. Analysis of these problems are explained in the following sections.

3. Image analysis and features extraction

The problem due to intensity changes is solved using the proposed algorithm as follows:

1- Acquiring the current Images.
2- Smoothing the acquired images: using median filter [27] to reduce noise and color variation. In this case, the algorithm able to detect and track objects with specified color.
3- *Frame Difference*: the frame difference serves as an important function to eliminate the background and any stationary objects from the filtered image. It attempt to detect moving regions by subtracting two or three consecutive frames in a video sequence [11] [28], using absolute frame difference as;

$$Diff\,(k) = \left| I_1((i,j),k) - I_2((i,j),k) \right| \quad (1)$$

Where $I_1\,(i, j)$ and $I_2\,(i, j)$ are the current and previous frames respectively.
4-*Threshold*: by applying threshold operation on the difference image to decrease information and eliminate pixels with small intensity values. The threshold value is calculated from the difference image as in [27][29]. But, the noise and false detections are appeared in the output binary image [30]. Therefore, there is a modification in the threshold process by adding two threshold values as follows;

- Applying constant threshold on the difference image to remove noise due to camera and intensity changes. This threshold is determined experimentally. And the output image is a grayscale or RGB depending on the original image, as:

  If diff (i, j) > constant threshold 1
  Then Out_image (i, j) = diff (i, j),
  Else
  Out_image (i, j) =0
  End

- The second threshold is estimated for the resulted image after removing the constant noise. The output image is binary image without noise as shown in the results below. Each pixel is calculated as;

$I_{th}\,(x, y) = 1$ *if Out_image (i, j) > thresh2*
$= 0$ *otherwise*

5- *Noise removing*: by applying binary area open morphological operation [27], to remove small objects in the segmented image.
6- *Edge detection*: the edge detection approach used, is the Canny Edge detector [30].
7- *Position Estimation:* After segment the interested object based on (color and area) features, the position of the object is calculated with respected to the center of the camera view (**cov**), which used to track the object [28]. It is calculated in pixel as :

$$Position\,(x, y) = cov\,(x, y) - cog\,(x, y) \quad (2)$$

Where *cog* is the center of gravity for the object (pixel). This position is converted into angular position (degree) and becomes a desired position (θd) for the fuzzy logic position controller.

4. Controller Design

The Fuzzy Logic Controller (FLC) is designed as a Proportional, Derivative (PD) controller type to control the position and direction of the DC-Motor according to the desired (estimated) position. The discrete time form of the PD-Controller is:

$$u(k) = k_p\,.e(k) + k_d\,.\Delta e(k). \quad (3)$$

Where $U\,(k)$, $e\,(k)$ and $\Delta e\,(k)$ are the control signal, error signal and error change(Ce) at the sample instant k respectively. And K_P and K_D are the controller parameters. The error is defined as a difference between estimated (reference position $(\theta_d\,(k))$ and previous estimated position ($\theta_d\,(k\text{-}1)$) in degree. $e\,(k\text{-}1)$ is the error value at the previous sample time. For the k[th] sampling instant, $(e\,(k))$ and $(Ce\,(k))$ can be expressed as:

$$e(k) = \theta_d(k) - \theta_d(k-1)$$

$$\Delta e(k) = e(k) - e(k-1) \quad (4)$$

4.1. Fuzzification

The inputs for the FLC are the error and change of error. The fuzzification strategy converts the crisp input variables into suitable linguistic variables (e.g. Zero "Z", Positive Small "PS"… etc.). The action of the stage is primarily depending on the membership functions (µfs) that describe the input linguistic terms [23]. *Triangular* type membership functions are used to fuzzify the error(e), change of error(Ce) and output control signal(U). Graphical illustration of (µfs) and their ranges for (err, Ce and U) signals are shown in Figure 1 respectively. The µfs are; NVH: Negative Very High, NH: Negative High, NM: Negative Medium, NS: Negative Small, Z: Zero, PS: Positive Small, PM: Positive Medium, PH: Positive High, and PVH: Positive Very High. The ranges for universe of discourses are selected according to the maximum error and change of error values occurred. Therefore, these ranges are in the intervals of [-60 ^{o}, 60 ^{o}] for error, [-20 ^{o}, 20 ^{o}] for change of error and [-1 , 1] for the output control signal (U).

4.2. Rule base design

The relationship between input and output of the system, is called the *rule-base* or *If - then* rules [17]. This relationship must be obtained correctly to improve the performance of the fuzzy logic based control system. This process determine the rules used to track the reference position set point with minimum steady state error. For instance, If the Error (e) is Negative High (NH) and Change of Error (Ce) is Zero (Z) then the output (U) is Negative Very High (NVH). The full set of rules are summarized in the table 1.

4.3. Defuzzification process

At the final step, the determination of both the applicable rules for any fuzzy controller inputs (*e*, *Ce*) and the fuzzy control output (*U*) are performed by inference engine strategy. The fuzzy results obtained from the above rules should be defuzzified to convert them into meaningful crisp values. For this purpose, the center of area (*COA*) method, proposed by [23] is used. It is expressed as;

$$UCOA = \frac{\sum_{i=1}^{n} \mu i(u).u}{\sum_{i=1}^{n} \mu i(u)} \qquad (5)$$

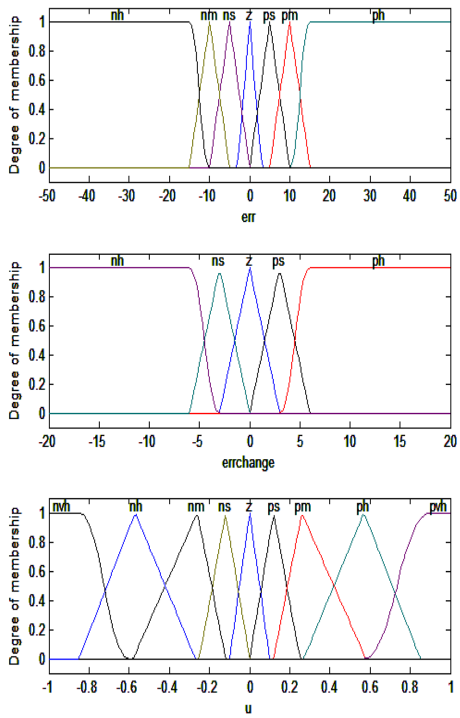


Figure 1. Membership functions for error, change of error and output control signal (u), respectively.

Table 1. Summary of Fuzzy Rules

Ce \ e	Nh	Nm	Ns	Z	Ps	Pm	Ph
Nh	Nvh	Nvh	Nh	Ps	Pm	Pm	Ph
Ns	Nvh	Nh	Nm	Ps	Ps	Pm	Pvh
Z	Nvh	Nh	Nm	Z	Pm	Ph	Pvh
Ps	Nvh	Nm	Ns	Ns	Pm	Ph	Pvh
Ph	Nh	Nm	Ns	Ns	Ph	Pvh	Pvh

5. Experimental results

5.1. System description

The implementation of the visual servoing system is divided into **hardware and software**. The hardware used in the proposed architecture includes the following items:

- PC Pentium 4. CPU 1.8 GHz/512 Mb cache, 512 MB RAM,
- Interface, Motor Drive Circuits and Power Supply Unit, (+5 and +12 VDC).
- Brush type DC-Motor (12 v) and Incremental Encoder with (*105* Slot with *Resolution = 3.429 degree/slot*) to measure the actual position of the camera.
- CCD web camera (¼" color progressive CMOS,640×480pixels) as a vision sensor.

Software implementation: The algorithms are implemented using Matlab software. The block diagram of the system is shown in Figure 2.

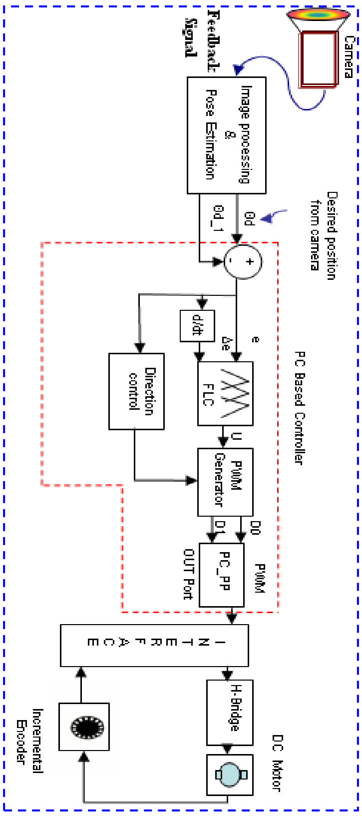


Figure 2. Block diagram of the PC- based DC motor visual servoing system

5.2. Practical results

To test the noise effects on the scene, different experiments are carried to test the proposed algorithm to track objects in the *image frames without controller* to identify any noise in the scene (static scene static camera). Then some testes are carried to track objects in real time (dynamic scene moving camera).

5.2.1. Static scene static camera

a) *Comparison between conventional and proposed frame differences***.**

The conventional frame difference (FD) is depends on one threshold value to produce the inter-frame difference image. When the threshold value is small, some noisy pixels appeared. These noisy pixels will give rise to erroneous object tracking. When the threshold value is large, some moving pixels are lost. The noise is detected in the case of static scene Figure (3-a). But, when applying the proposed algorithm, the noise is removed from the image as shown in Figure (3-b). When the moving object appears into the frame, the noise is decreased, this is because the intensity changes due to moving object is larger than intensity changes due to noise, as shown in Figure (3-c, d).

b) White background, single object detection.

This test is used to detect objects based on object features such as color feature, because the frame difference method cannot detect static objects. Figure 4, shows different locations for the object and their Segmentation. Estimated positions of the object (based on camera calibration technique) with respect to real world positions are summarized in table.2. All values are measured with respect to center of camera view. Table 2 shows that the estimated positions(θe) are close to measured positions(θm). The average processing time in the case of static scene about (0.54 sec).

c) Complex background, multiple object detection.

The modified frame difference is integrated with the object features (color and area) to detect and segment moving and static objects from multiple objects. As shown in Figure (5-a), the algorithm can detect and segment the red color object with specified area from multiple moving objects are appears in the scene. The edge detection for the segmented object at each frame is shown in Figure (5-b). Estimated position (degree) in (x, y) directions are shown in Figure (6-a,b).

5.2.2. Dynamic Scene and Moving Camera (Visual Servoing).

After testing the algorithm to track moving and static objects in image frames (static scene static camera), the algorithm is able to track moving objects in real time. To track objects in real time, the camera is mounted on the shaft of DC-motor. In this case the estimated position is used as a desired angle for the PD-FLC controller. The output of the FLC is a Dynamic Level Control Signal(DLCS).

a) Step Response test to track objects in real time.

In order to test the performance of the proposed real time tracking algorithm, the incremental encoder mounted on the shaft of the motor is used to measure the actual position of the camera. The required object is the red color object, as shown in Figure7. The algorithm detects the red object from the first frame, and then calculates the desired position, as shown in Figure (7-a). The desired position is (-19.2^{o}) into CCW. The FLC generate the DLCS as shown in Figure (7-b), which used to generate Pulse Width Modulated (PWM) signal. The PWM signal is used to control in the speed of the motor to direct the camera into the desired position (*cog*) of the object smoothly without overshoot. As shown in Figure (7-c), the controller able to generate number of loops to apply the PWM signal number of times on the motor, to reach the desired position in very short time. The average processing time observed to maintain the object in the center of view is about (1.3 sec).

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

Figure 3. a. Conventional FD static scene, b. Proposed FD, c. Conventional FD for moving object, d. Proposed FD.

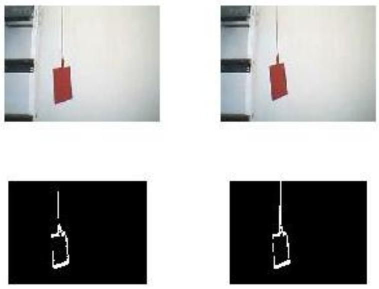
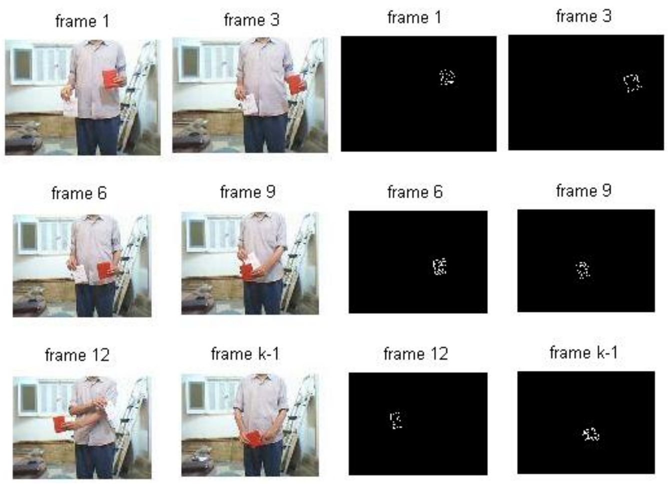


Figure 4. Single object Red color segmentation.

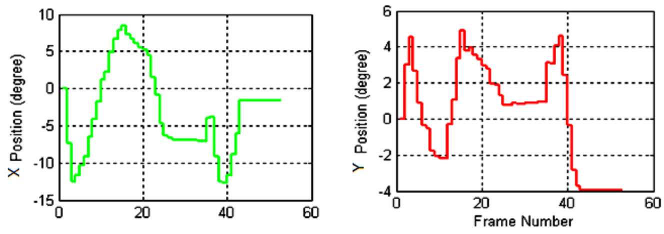
Table 2. Summary for estimated and measured position values.

Position in X-direction (cm)		Position in Y-direction (cm)	
θe	**θm**	**θe**	**θm**
0	0	0	0
9	9.5	9.094	9.5
17.48	18.5	8.65	8.5
37.5	38.5	18.91	18
49	51	27.47	27
67	68	17	16.5



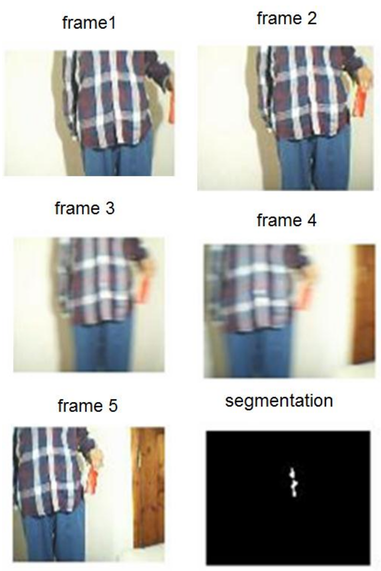
(a) **(b)**

Figure5.a. Motion of multiple objects, b. red color segmentation.

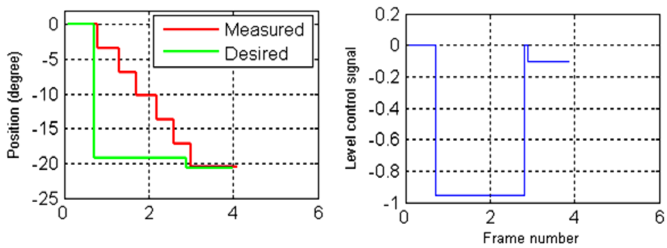


(a) **(b)**

Figure 6. a. Estimated position X-direction (Degree). b. Estimated position in Y direction (Degree).



a. Step response for object tracking



b. Step response and FLC Control signal

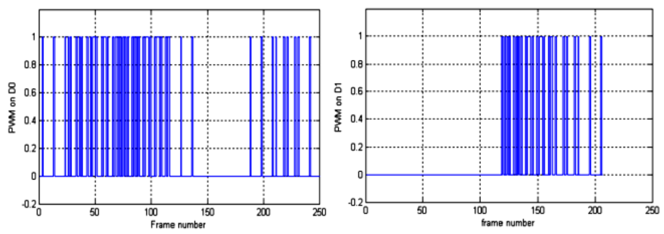


Figure 7 .c- PWM signal

b) *Tracking moving objects in real time*

This test is used to detect and track red object from complex background, as shown in Figure8. The algorithm tries to maintain the object in the center of camera view from frame 2 until the object stops its movement and becomes in the center of view at frame 95. At frame 118, the red object moved out form the camera view in the other direction instantaneously and then stop. As shown in Figure 8, at frames (141 to 164) the algorithm tries to maintain the object in the camera view and the object becomes in the center at frame 210. Estimated position of the object at each frame with respect to measured position using encoder are shown in Figure 9. The DLCS output control signal of the PD-FLC position controller is used to generate the PWM signal. The average processing time to track objects about (0.98 sec to 3.93 sec).

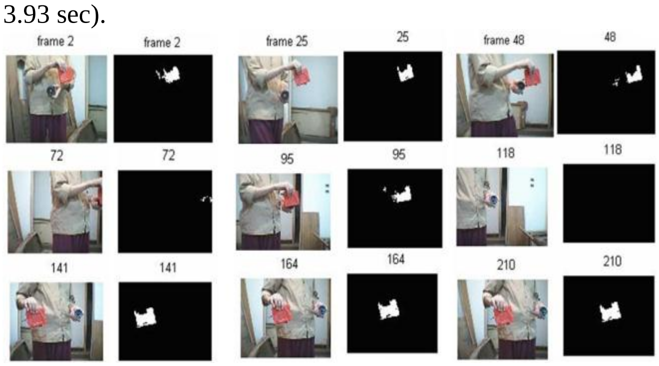


Figure 8. Complex background, Sample of frames to detect and track the red color object.

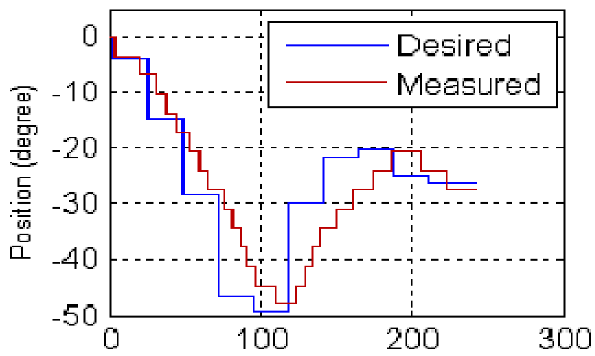


Figure 9. Estimated position w.r.t. measured position.

6. Conclusions

The real time object tracking based on IBVS is implemented. The algorithm merge the feature based and the modified frame difference technique to detect, segment and track Static and Moving objects in real time. The PD-FLC position controller was designed and implemented to generate a PWM signal that used to control the position and direction of the DC-Motor, to maintain the tracked object in the center of camera view, via hardware circuits. A PWM software program has been developed to control the speed of the motor to reach the desired position smoothly without overshoot. Different tests are carried to verify the validity of the proposed algorithm such as, static scene static camera and moving object moving camera. The results show that, the static and moving objects can be detected and segmented without noise and false detections in the scene. The average processing time was observed since the image acquired until the object becomes in the center of camera view is between **0.54 : 3.93** sec. It is depending on the position of the object with respect to the camera view.

References

[1] N. Zuech "Understanding and applying machine vision", Marcel Dekker Inc., 2nd edition Revised and Expanded, 2000.

[2] A. Yilmaz, O. Javed and M. Shah, "Object Tracking: A Survey", ACM Computing Surveys, Vol. 38, No. 4, Article 13, December 2006.

[3] P. I. Corke, "Visual Control of Robots: high performance visual servoing", Research Studies Press Ltd., 1996.

[4] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control", IEEE Trans. on Robotics and Automation, Vol. 12, No. 5, pp. 651-670, October 1996.

[5] D. Kragic and H. Christensen, "Survey on Visual Servoing for Manipulation", Computational Vision and Active Perception Laboratory, Stockholm, Sweden, 2002.

[6] F. Chaumette and S. Hutchinson, "Visual Servo Control Part I: Basic Approaches", IEEE Robotics and Automation magazine, pp. 82-90, December 2006

[7] V. Lippiello, B. Siciliano and L. Villani, "Position-Based Visual Servoing in Industrial Multirobot Cells Using a Hybrid Camera Configuration", IEEE Trans. On Robotics, Vol. 23, No. 1, February 2007.

[8] D. Fioravanti, B. Allotta and A. Rindi, "Image based visual servoing for robot positioning tasks",Springer Science, Meccanica, Vol. 43, pp. 291–305, 2008.

[9] C. Micheloni, G. Foresti "Real-Time Image Processing For Active Monitoring Of Wide Areas", J. Vis. Commun. Image R. Vol. 17, pp. 589–604, 2006.

[10] Y. Yoon, G. DeSouza and A. kak, "Real Time Tracking And Pose Estimation For Industrial Objects Using Geometric Features", IEEE Int. Conf. On Robotics & Automation, Taipci Taiwan, pp.3473-3478, September 14-19, 2003.

[11] P.Y. Yeoh and S.A.R. Abu-Bakar, "Accurate Real-Time Object Tracking With Linear Prediction Method", IEEE Int. Conf. In Image Processing ICIP 2003, Vol. 3, pp. 941-944, September 14-17, 2003.

[12] A. Elgammal, D. Harwood, L. S. Davis, "Non-parametric Model for Background Subtraction" IEEE ICCV99 Frame Rate Workshop. IEEE 7th Int. Conf. on Computer Vision. Kerkyra, Greece, September 1999.

[13] S. Theodoridis and K. Koutroumbas, "Pattern Recognition", Elsevier Academic Press, 2nd edition, 2003.

[14] G. De Cubber, S. Berrabah, H. Sahli, "Color-based visual servoing under varying illumination conditions", Robotics and Autonomous Systems, vol. 47, pp 225–249, 2004.

[15] X. Wan and G. Xu, "Camera Parameters Estimation and Evaluation in Active Vision System", Elsevier Science Ltd, Pattern Recognition, Vol. 29, No. 3, pp. 439-447, 1996.

[16] Z. Zhang, "Flexible Camera Calibration By Viewing a Plane From Unknown Orientations", The Proceedings of the Seventh IEEE International Conference on Computer Vision, Vol. 1 pp. 666-673. 1999.

[17] Paul I- Hai Lin, Santai Wing and John Chou , " Comparison on Fuzzy Logic and PID Control for a DC Motor Position Controller ",IEEE, Industry Applications Society Annual Meeting, volume 3,pp 1930 – 1935, 1994.

[18] G. Cheng and K. Peng "Robust Composite Nonlinear feedback Control with Application to a Servo Positioning System ", IEEE, Trans. On Ind. Elec., vol. 54, No 2, pp 1132-1140 April, 2007.

[19] Tolgay Kara, Ilyas Eker "Nonlinear Modeling and Identification of a DC motor for Bidirectional Operation with Real Time Experiments", Energy Conversion and Management, volume 45, pp 1087-1106, 2004.

[20] M.A. Fkirin, K. N. Khatab and A. A. Montasser "Identification of Stochastic Processes in Presence of Input Noise", Cybernetics and Systems, an International Journal, Volume 20, pp 489-500, 1989.

[21] M. A. Fkirin "Fixed-interval Smoother in the Identification of Time-Varying Dynamics", INT. J. System SCI., volume 20, NO. 11, pp 2267-2273, 1989.

[22] L. A. Zadeh "A Fuzzy-Algorithmic approach to the Definition of Complex or Imprecise Concepts", Int. J. Man-Machine Studies, pp 249-291, 1976.

[23] D. Driankov, H. Hellendoorn and M. Reinfrank, "An Introduction to Fuzzy Control ", Springer – Verlag Berlin Heidelberg, 1993.

[24] Tipsuwan, Y. Mo-Yuen Chow "Fuzzy Logic Microcontroller Implementation for Motor Speed Control" , The 25th Annual Conference of the IEEE, Industrial Electronics Society, IECON '99 Proceedings, volume 3, pp 1271- 1276, 1999.

[25] C. Pėrez, O. Reinoso and M. Vicente, "Robot hand visual tracking using an adaptive fuzzy logic controller ", WSCG Posters proceedings, February 2-6, 2004.

[26] C. Cėdras, M. Shah " Motion based recognition : a survey ", Image and Vision Computing, Vol. 13, No. 2, March 1995.

[27] Rafael C. Gonzalez and Richard E. Woods "Digital Image Processing", Prentice-Hall, Inc., 2nd edition, 2001.

[28] Y. Shiao and S. Road, "Design and implementation of real time tracking system based on vision servo control", Tamkang Journal of science and Engineering, Vol. 4, No. 1, pp. 45-58, 2001.

[29] R. C. Gonzalez, R. E. Woods and S. L. Eddins, "Digital Image Processing Using MATLAB", Prentice-Hall, Pearson Education, Inc., 2004.

[30] F. Cheng,Y. Chen, " Real time multiple objects tracking and identification based on discrete wavelet transform", Elsevier Science Ltd, Pattern Recognition, 39 , pp. 1126 – 1139, 2006.

Integrating MATLAB with Verification HDLs for Functional Verification of Image and Video Processing ASIC

Dhaval Modi[1], Harsh Sitapara[2], Rahul Shah[3], Ekata Mehul[4], Pinal Engineer[5]

Communication System Engineering, L.D. College of Engineering, Ahmedabad[1.]
Microprocessor Engineering, M.S. University, Baroda[2]
ASIC Division, EINFOCHIPS Pvt. Ltd., Ahmedabad[3]
ASIC Division, EINFOCHIPS Pvt. Ltd., Ahmedabad[4]
S.V.N.I.T., Surat[5]
dhavalmodi045@yahoo.com[1], harsh.sitapara@einfochips.com[2], rahulv.shah@einfochips.com[3] , ekata.mehul@einfochips.com[4] , pje@eced.svnit.ac.in[5]

Abstract - *The ultimate Aim of ASIC verification is to obtain the highest possible level of confidence in the correctness of a design, attempt to find design errors and show that the design implements the specification. Complexity of ASIC is growing exponentially and the market is pressuring design cycle times to decrease. Traditional methods of verification have proven to be insufficient for Digital Image processing applications. We develop a new verification method based on SystemVerilog verification with MATLAB to accelerate verification. The co-simulation is accomplished using MATLAB and SystemVerilog coupled through the DPI. I will be using the Image Resize design as case study by using co-simulation method between SystemVerilog and MATLAB. Golden reference will be made using MATLAB In-built functions, while rest of the Verification blocks are in SystemVerilog. The goal is to find more bugs from Image resizing Design as compared to traditional method of Verification, reduce time to verify video processing ASIC, reduce debugging time, and reduce coding length.*

Key words: *Code base, API, DPI, Design cycle time*

1. Introduction

Leading chip development teams report that functional verification has become the biggest bottleneck, consuming approximately 70% of chip development time and efforts. For Image and video processing application, Register transfer level test-benches have become too complex to manage. New method has to be discovered to reduce verification cycle.

Image processing designs easily coded in MATLAB. Therefore, we believe that verification could be significantly improved and accelerated by reusing these golden references models in MATLAB. In this paper we explore combining the power of MATLAB, for Image processing application, signal content generation, spectral analysis, spectrum and waveform display and SystemVerilog, for random stimuli generation. MATLAB and SystemVerilog correlated with each other through the SystemVerilog DPI interface.

1.1 Verification Architecture using co-simulation interface

The MATLAB environment is a high-level technical computing language for algorithm development, data visualization, data analysis and numerical computing [1]. MATLAB also included the Simulink graphical environment used for multi-domain simulation and model-based design. Image processing designers take advantage of Simulink as it offers a good platform for preliminary algorithmic exploration and optimization.

First in MATLAB, algorithmic level model is developed. Second step is to start RTL level implementation. To verify this research work, we use SystemVerilog. SystemVerilog has become a concrete RTL level verification language used by many industries. One of the good capabilities of SystemVerilog is to generate random stimuli.

Here golden reference is in MATLAB and design Under Test is in HDL. Scoreboard compares the output of golden reference and DUT. We have to require an efficient transition between algorithmic level and RTL level design. Thus, we need a co simulation between the MATLAB environment and SystemVerilog.

SystemVerilog language doesn’t provide any facility to directly call the MATLAB engine. The SystemVerilog Direct Programming Interface (DPI) is basically an interface between SystemVerilog and a foreign programming language, in particular the C language. It allows the designer to easily call C functions from SystemVerilog and SystemVerilog function from c. We make the same C program to call MATLAB Engine library.

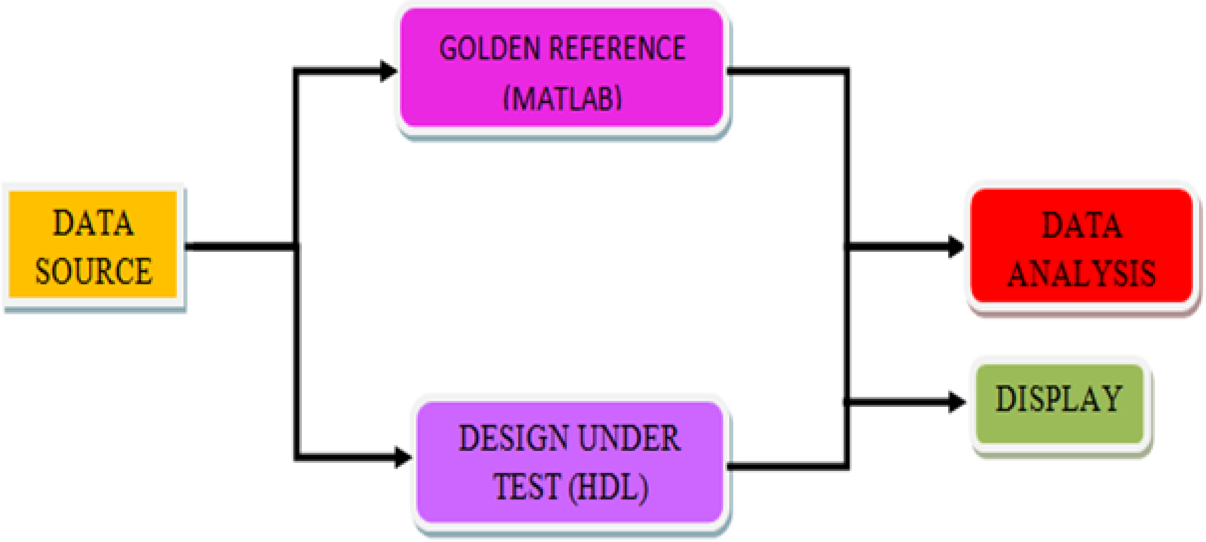


Figure 1. Verification Architecture

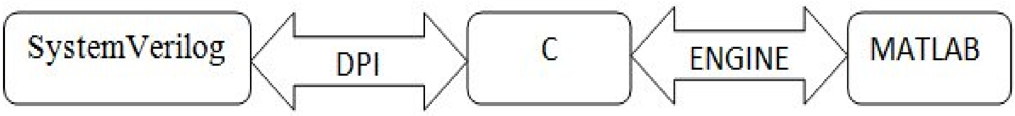


Figure 2. Interface between SystemVerilog and MATLAB

Once a link has been established between SystemVerilog and MATLAB, it opens up a wide range of additional capability to SystemVerilog, like stimulus generation and data visualization. The first advantage of our technique is to use the right tool for the right task. Complex stimulus generation and signal processing visualization are carried out with MATLAB while hardware verification is performed with SystemVerilog verification standard. The second advantage is to have a SystemVerilog centric approach allowing greater flexibility and configurability.

2. Co-simulation between System Verilog and MATLAB

2.1 Simulation between MATLAB and C

2.1.1 The MATLAB Engine Library

To enable C to call MATLAB, we use 'engine' library available within MATLAB. The MATLAB engine library contains routines that allow us to call MATLAB software from our own program. Engine programs are standalone C/C++ or FORTRAN programs that communicate with a separate MATLAB process via pipes. MATLAB provides a library of functions that allows us to start and end the MATLAB process, send data to and from MATLAB, and send commands to be processed in MATLAB. The MATLAB engine operates by running in the background as a separate process from our own program.

The MATLAB language works with only a single object type: MATLAB array. These arrays are manipulated in C using the 'mx' prefixed application programming interface (API) routines included in the MATLAB engine. This API consists of over 60 routines to create access, manipulate, and destroy mxArrays.

The engine library is part of the MATLAB C/C++ and Fortran API Reference. It contains routines for controlling the computation engine. The function names begin with the three-letter prefix "eng". MATLAB libraries are not thread-safe. If you create multithreaded applications, make sure only one thread accesses the engine application.

2.1.2 How to communicate with MATLAB

In this paragraph we show detail about how to write our application with use of MATLAB engine library. Write your application in C/C++ using any of the engine routines to perform computations in MATLAB. Use the mex script to compile and link engine programs. mex has a set of switches you can use to modify the compile and link stages.

Table 1. MATLAB Engine routines to communicate with C

Function	Purpose
engOpen	Start up MATLAB engine
engClose	Shut down MATLAB engine
engGetVariable	Get a MATLAB array from the MATLAB engine
engPutVariable	Send a MATLAB array to the MATLAB engine
engEvalString	Execute a MATLAB command
engOutputBuffer	Create a buffer to store MATLAB text output
engOpenSingleUse	Start a MATLAB engine session for single, nonshared use
engGetVisible	Determine visibility of MATLAB engine session
engSetVisible	Show or hide MATLAB engine session

MATLAB supplies a mex options file to facilitate building MEX applications. This file contains compiler-specific flags that correspond to the general compile, prelink, and link steps required on your system. If you want to customize the build process, you can modify this file. The MATLAB Engine Library is an external library; several steps have to be taken in order to utilize its capability within SystemVerilog. The Linker path has to be modified. Use of the MATLAB Engine Library also requires modifications to the C compile command line.

2.1.3 Compiling and Linking MATLAB Engine Programs

Step: 1 Write your application in C/C++ or FORTRAN using any of the engine routines to perform computations in MATLAB.

Step: 2 Build the Application. Use the mex script to compile and link engine programs.

Step: 3 Use of MEX Options File:- MATLAB supplies an options file to facilitate building MEX applications. This file contains compiler-specific flags that correspond to the general compile, prelink, and link steps required on your system. If you want to customize the build process, you can modify this file.

Step: 4 Building an Engine Application on LINUX Systems.

Build the executable file using the ANSI compiler for engine stand alone programs and the options file engopts.sh:

optsfile = [matlabroot '/bin/engopts.sh'];

mex('-f', optsfile, 'engdemo.c');

Verify that the build worked by looking in your current working folder for the file engdemo:

dir engdemo

To run the demo in MATLAB, make sure your current working folder is set to the one in which you built the executable file, and then type:

!engdemo

We can change compiler using **mex -setup**. We can choose GCC or LCC compiler for our application. MATLAB provides inbuilt compiler LCC for C/C++ programs.

2.2 Simulation between SystemVerilog and C

2.2.1 Introduction about Direct Programming Interface

Direct Programming Interface (DPI) is an interface between SystemVerilog and a foreign programming language. It consists of two separate layers: the SystemVerilog layer and a foreign language layer. Both sides of DPI are fully isolated. The motivation for this interface is two-fold. The methodological requirement is that the interface should allow a heterogeneous system to be built (a design or a testbench) in which some components can be written in a language (or more languages) other than SystemVerilog, hereinafter called the foreign language. On the other hand, there is also a practical need for an easy and efficient way to connect existing code, usually written in C or C++, without the knowledge and the overhead of PLI or VPI.

2.2.2 Import Method

Methods implemented in C and given import declarations in SystemVerilog can be called from SystemVerilog, such methods are referred to as imported methods. Imported tasks or functions are similar to SystemVerilog tasks or functions. Imported tasks or functions can have zero or more formal input, output, and inout arguments. Imported tasks always return an int result as part of the DPI-C disable protocol and, thus, are declared in foreign code as int functions.

The syntax import method:

import {"DPI-C"}[context|pure][c_identifier =] [function task] function_ identifier|task _identifier] ([port_list]);

2.2.3 Export Method

Methods implemented in SystemVerilog and specified in export declarations can be called from C, such methods are referred to as exported methods. Syntax of export method is same as import method.

The syntax import method:

export {"DPI-C"}[context|pure][c_identifier =] [function task][function_ identifier|task _identifier] ([port_list]);

2.3 Co-simulation between SystemVerilog and MATLAB

2.3.1 Combining power of SystemVerilog and MATLAB using DPI

Here I combine the SystemVerilog DPI and MATLAB application programming interface. I use wrapper of C around MATLAB Engine and use of DPI to communicate with SystemVerilog as shown in the figure 2.3.

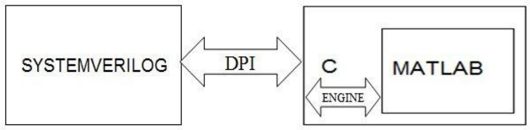


Figure 3. Co-Simulation between SystemVerilog and MATLAB

I make code in which I call "engdemo.c" from SystemVerilog using import DPI method. Here first SV code is executing and with import DPI engdemo.c is executing. Output is combination of both MATLAB and SystemVerilog.

To compile above program I use following command:

Irun dpic.sv try1.c -I/opt/matlab2008/extern/include -L/opt/matlab2008/bin/glnx86 -leng

When the SystemVerilog compiler while encountering the C code, It calls GCC compiler to compile the C code in background. The final control however remains within the SV compiler.

2.3.3 Flow Chart of co-simulation

A block diagram of the flow used for the Object tracking project is shown in the following diagram. The items existing in the SystemVerilog environment are in the left column. The middle two columns show tasks existing in the two interface C-layers. The far right column shows tasks existing in the MATLAB workspace.

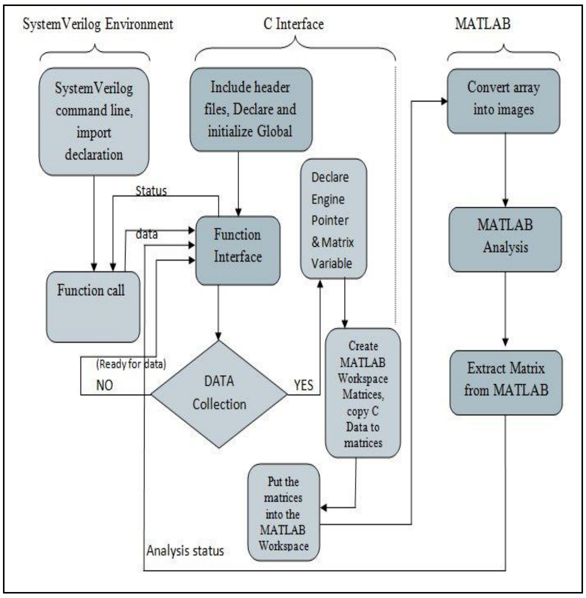


Figure 4. Flow chart of Co-simulation

3 Image Resizing as a case study

3.1 Overview and Objective

Image resizing includes image enlargement and image shrinking. The resizing of image is required in different applications. Now a days, in every television sets Picture-in-Picture (PIP) facility is available. In that a small image of one channel is shown in the main image of current channel. This facility uses the fundamentals of image resizing. The image of other channel is shrunk in smaller size and shown in current channel screen as a small window. The image resizing is also used in all media players available in PC now days.

3.2 Image Resizing Interface

The top level Input and Output pin diagram for IMR DUT device is shown below. It shows both host interface side and memory interface side input/output signals.

Input Interface:-

Imr_host_clk_i

Imr_host_reset_ni

Imr_host_addr_i[15:0]

Imr_host_wr_data_i

Imr_host_wr_en_i

Imr_host_rd_en_i

Output Interface:-

Imr_clk_i

Imr_reset_ni

Imr_wr_hold_ni

Imr_wr_grant_i

Imr_rd_hold_ni

Imr_rd_grant_i

Imr_wr_brust_o

Imr_rd_brust_o

imr_wr_ald_o[31:0]

imr_rd_ald_o[31:0]

3.3 Verification Architecture for Image Processing Application

As we know, Image processing Application is easily made in MATLAB. Due to visualization capability of MATLAB, it is very easily checked by human beings. So I can make Scoreboard is in MATLAB.

The output of Scoreboard and DUT in checker is compared & with the help of co-simulation, the output of MATLAB is transferred in SystemVerilog. So checker is also in SystemVerilog. Rest of the blocks is in SystemVerilog.

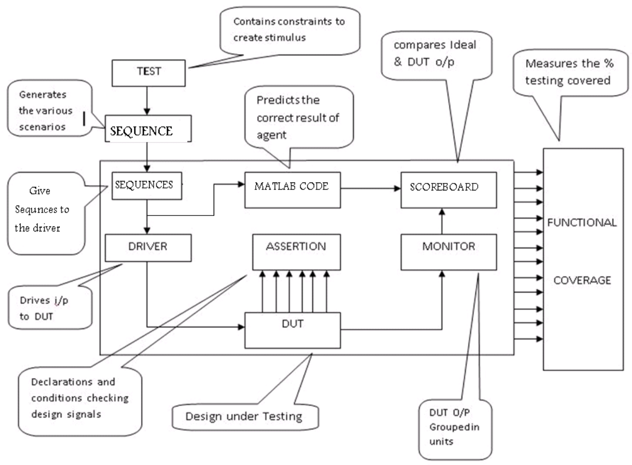


Figure 5. Image Resizing Verification Environment

Table 2. Verification Architecture using Co-simulation

Sr No.	OVM Component	Coding Languages
1	Packet	SystemVerilog
2	Sequence	SystemVerilog
3	Sequences	SystemVerilog
4	Driver	SystemVerilog
5	Monitor	SystemVerilog
6	Golden reference	MATLAB
7	Scoreboard	SystemVerilog
8	Coverage	SystemVerilog

3.4 Image Resizing OVM Verification Components:-

OVM is a complete verification methodology that codifies the best practices for development of verification environments. OVM supports the transaction level modeling with built in class which reuse easily by extending it.

OVM uses a SystemVerilog implementation of standard TLM interfaces for modular communication between components. The architecture of OVM is same as shown in SystemVerilog for my case study of Image Resizing. But communication is simple and easy.

3.4.1.1 Top level:-

Blocks:-

- Interface

- Design instance

- Clock Declaration and Generation

Description:- It has Description of Interface and Top module and mapping the interface with DUT and with testbench environment block.

3.4.2 Class Environment extends ovm_env:-

Methods:-

Build()

Connect()

Description:- Environment class is used to implement verification environments in OVM. It is extension on ovm_env class. The testbench simulation needs some systematic flow like building the components, connection the components, starting the components etc. ovm_env base class has methods formalize the simulation steps.

3.4.3 Class Packet extends ovm_sequence_item:-

Members: - rand bit [7:0] sa;

rand bit [7:0] da;

rand bit [31:0] rowpixel;

rand bit [31:0] colpixel;

rand bit [3:0] hodis;

rand bit [3:0] verdis;

Description: - One way to model Packet is by extending ovm_sequence _item. ovm_sequence_item provides basic functionality for sequence items and sequences to operate in a sequence mechanism. Packet class should be able to generate all possible packet types randomly. To define copy, compare, record, print and sprint methods, we will use OVM field macros.

3.4.4 Class Sequencer extends ovm_sequencer #(Packet):-

Methods: - end_of_elaboration();

OVM macro:-
`ovm_sequencer_utils(Sequencer)

Description:- A Sequencer is defined by extending ovm_sequencer. ovm_sequencer has a port seq_item_export which is used to connect to ovm_driver for transaction transfer.

3.4.5 Class sequence extends ovm_sequence #(Packet):-

OVM macros: -
`ovm_sequence_utils(sequence , sequencer)

Description:- A sequence is defined by extending ovm_sequence class. This sequence of transactions should be defined in budy() method of ovm_sequence class. OVM has macros and methods to define the transaction types.

3.4.6 Class Driver extends ovm_driver #(Packet):-

Methods: - build();

end_of_elaboration()

reset_dut()

cfg_dut();

drive(Packet pkt)

run()

OVM macros: - ovm_analysis_port #(Packet) Drvr2Sb_port

`ovm_component_utils(Driver)

Description:- Driver is defined by extending ovm_driver. Driver takes the transaction from the sequencer using seq_item_port. This transaction will be driven to DUT as per the interface specification. After driving the transaction to DUT, it sends the transaction to scoreboard using ovm_analysis_port.

3.4.7 Class Receiver extends ovm_component:-

Methods: - build()

end_of_elaboration()

run()

OVM macros: - `ovm_component_utils(Receiver)

Description: - Receiver collects the data bytes from the interface signal. Receiver class is defined by extending ovm_component class. It will drive the received transaction to scoreboard using ovm_analysis_port.

3.4.8 Class Scoreboard extends ovm_scoreboard:-

OVM macros: - `ovm_component_utils(Scoreboard)

Description:- Scoreboard compare the output of golden reference and DUT.

3.5 Advantages compare to traditional method of Verification

Here Golden reference is in MATLAB for Image Resizing ASIC. So we can reduce code length for the same Image Resizing logic if we can code in verification HDLs. Another advantage is reduction in debugging time. We can use the MATLAB inbuilt function In our Verification architecture. So we have advantages of both SystemVerilog and MATLAB. This way we can reduce ASIC design Cycle for Image processing ASIC.

4 Conclusions

In this project a verification environment based on co-simulation interface between SystemVerilog and the MATLAB environment has been presented. The DPI C-layer can be used to interface to a wide variety of C base libraries and also the MATLAB Engine Library. The simulation stimulus could be generated from SystemVerilog; this would allow more robust image. Use of the MATLAB graphics capabilities could be more fully utilized. A more complete testbench can be build up in a shorter period of time than with traditional methods. Use of the SystemVerilog and MATLAB could be extended in a variety of directions for various applications.

5 Future Works

The co-simulation between SystemVerilog and MATLAB has been used in the digital image processing application project. Instead of creating time consuming stimuli in SystemVerilog, data generated from MATLAB environment is used to drive the testbench. Further using the MATLAB, a golden reference model is created. This Golden reference model is used in SystemVerilog environment to compare behavior of the Design under verification.

6 References

1. Compiling and Linking MATLAB Engine Programs, which is available online at http://www.mathworks.com/help/techdoc/matlab_external/f39903.html
2. MATLAB Application Program Interface Guide (December 1996).
3. Calling existing C code from MATLAB which is available online at http://www.mathworks.com/support/compilers/interface_r13.html#Call_MATLAB_from_C
4. SystemVerilog Language Reference Manual by Accellera's Extension to Verilog, 2002, 2003.
5. Irun-user guide from cadence, product version 9.2, july 2010.
6. Brian Bailey, "CoVerification: From Tool to Methodology," white paper, www.mentor.com, June 2002.
7. Jean-François Boland "USING MATLAB AND SIMULINK IN A SYSTEMC VERIFICATION ENVIRONMENT" McGill University, QC, Canada
8. John Stickley, & Wade Stone, "Accelerated Verification of a MATLAB-Driven Digital FIR Filter RTL Design Using Veloce and TBX" Mentor Graphics Corporation
9. A Zuloaga, J. L. Martín, U. Bidarte, J. A. Ezquerra "VHDL test bench for digital image processing systems using a new image format" Department of Electronics

and Telecommunications, University of the Basque Country
10. Janick Bergeron. Writing Test benches using SystemVerilog. Springer, 2006.
11. William K. Lam. Hardware Design Verification. Pearson Education, Inc., 2005.
12. Jean Francois Boland, Cosimulation of Matlab with system C, Mcgill University, Canada, 2004.
13. SystemVerilog for Verification: A Guide to Learning the Testbench Language Features, 2006 by Chris Spear. ISBN:0387270361, Publisher:Springer
14. Hardware Verification with System Verilog, 2007 by Mintz, Mike, Ekendahl, Robert ISBN: 978-0-387-71738-8, Publisher:Springer
15. Writing Testbenches using SystemVerilog, 2006 by Bergeron, Janick ISBN: 978-0-387-29221-2.
16. The Art of Verification with SystemVerilog Assertions, 2006 ISBN-13: 978-0-9711994-1-5

Implementation of Sensors using Ptolemy

Stuti V.Shah [1] ,Ekata Mehul [2]

[1] Computer Science and Engineering Department,Nirma University,Ahmedabad,India

[2] ASIC Division,eInfochips Private Limited,Ahmedabad,India

stuti.shah@einfichips.com [1] ,ekata.mehul@einfochips.com [2]

Abstract - Based on the complexity of Wireless Sensor Networks and their large deployment requirements, it becomes essential to model the system be- fore its implementation.Wireless Sensors systems includes different types of sensors with different characteristics and real constraints such as power , energy etc.This paper mainly focuses on implementation of sensors on Ptolemy and incorporating that sensors in to user library of Ptolemy which further can be re use for modeling similar types of system. Based on the type of sensor, the methodology changes since the requirement changes.Couple of sensors which are used for health monitoring systems such as blood pressure sensor, Glucose sensor , Temperature sensor , Bed and chair occupancy sensor with primarily learning method of artificial intelligence are implemented and added to Ptolemy library,example of sen- sors which are added to Ptolemy library are glucose sensor, Temperature sensor and Blood pressure sensor.

Keywords: *Ptolemy, Sensor, Health, Monitoring Systems, Security*

1. Introduction

This paper gives reader a quick overview of how different sensors are modeled in Ptolemy. Approach adopted in this paper is couple of sensors are taken as an example for showing their interaction with Ptolemy. Implementation of the sensors is done using visual sense which is a tool for modeling wireless sensor networks on top of Ptolemy. After implementation the sensors can be added to user library of Ptolemy as example Glucose sensor, Blood pressure sensor and Temperature sensor. The sensor also gives parameters which can be configured at the time the component is dragged from library. Sensors are implemented using primary learning method of artificial intelligence which is learning from past records.

Section 1 give details on features of sensors which are implemented. Section 2 elaborates stepwise details used for implementing controlling device and different sensors,flow chart showing steps of implementation for controlling device and implemented sensors. Section 3 elaborates the way artificial intelligence and parameter configuration is incorporated in sensor implementation.

- **Features of Sensors:**

1.1 Glucose sensor,Blood Pressure Sensor, Temperature Sensor

- This sensors are working on timers
- They will have timer running on their device and when at every specified time interval it will send signal to Wall mounted device which will contain record value.
- Device should have capability to find whether value received is within range or not.
- Parameter configuration facility is require so initial value for range can be set by expert.
- The sensor should be added to Ptolemy library so then it can be directly dragged from library for reusable purpose.
- Location variable should move sensors randomly.
- Single output port will be used for sending values to controlling device

1.2 Bed and Chair Occupancy Sensor

- Used for finding whether person has gone out of bed and return within specified period of time not
- Location for this sensor is fixed.
- The sensor will have timer running on it. When person gets out of bed it will keep time details and will start timer at that time.
- If the person is back within time the timer will be stop else if time expires it will send signal to controlling device
- This will generate interrupt to controlling device as immediate action needs to be taken by controlling device

1.3 Motion Sensor

- Used to track persons movement in the room

- The sensor will continuously generate values of x,y coordinate and that signal will be sent to controlling device timely
- The controlling device will have plot functionality by which it will plot points on graph and it will use mathematical formula for finding distance traveled by person.
- If the person is back within time the timer will be stop else if time expires it will send signal to controlling device
- If the person is moving too faster or too slower that can be found out by controlling device.

2 Implementation of Sensors

Ptolemy provides large number of library components which can be used to implement sensors. The generic method followed for implementation of all the sensors is described.

Communication between Monitoring device and sensors is done with wireless channel. For implementing all the sensors discrete event director of Ptolemy is used so that can be further used under wireless director of Ptolemy for system level implementation.

2.1 Steps used for implementing Monitoring device

1. Monitoring device needs to maintain database for different sensors incom- ing records.
2. Ptolemy's Database manager library of components can be used for main- tain database. It is available with Ptolemy 8.0.1. It provides different components for interacting with sql database. It provides facility of in- serting record in to database, fetching record from database and deletion of records.
3. All the sensors implemented in this paper is having clock running on it.

   It will send timely report to Monitoring device

4. When monitoring device will receive message on input port it will read the message and it will find that from which sensor record is coming .message separator functionality can be achieved by Ptolemy distributor compo- nent.

5. For different sensors Different database will be used

6. When it receives record it will check that record against the range set with the use of primarily artificial intelligence method described in section 3. Ptolemy Boolean switch component is used for checking record against desire range
7. If record is within specified range then that record will be stored in database and if record is outside desired range then monitoring device will take action
8. The action by monitoring device can be call to doctor with help of sensing message to care taker
9. If there is some fault in sensor and if it is not working properly it will not be able to send timely record to monitoring device this will be tracked by monitoring device by running timer on it. If after certain time expires if sensor has not set any record, it will try to connect it and it will find whether it is working properly or not.

The way of implementation of monitoring device can be shown with flow chart as shown in the figure 1.Implemented Monitoring Device is shown in Figure
2.

1. All the sensors implemented here are working on clock. So the clock is generating an event at specified time interval. Different type of clocks such as triggered clock, Poisson clock, Discrete clock are available with Ptolemy. The time interval for clock can be configured according to requirement
2. Once Clock generates event at that time the sensor should generate value and send it to monitoring device
3. For generating value it will use range variable which is set by primarily artificial intelligence method described in section 3.
4. For simulation sensor node is deciding whether to generate value inside range or value which is not in range
5. Use of random number generator component is done for deciding whether to generate values inside range or not. Different types of random number generators such as Uniform,Triangle,Gaussian,Bernoulli are available with Ptolemy random library. It gives flexibility to specify range in which the numbers needs to be generated.

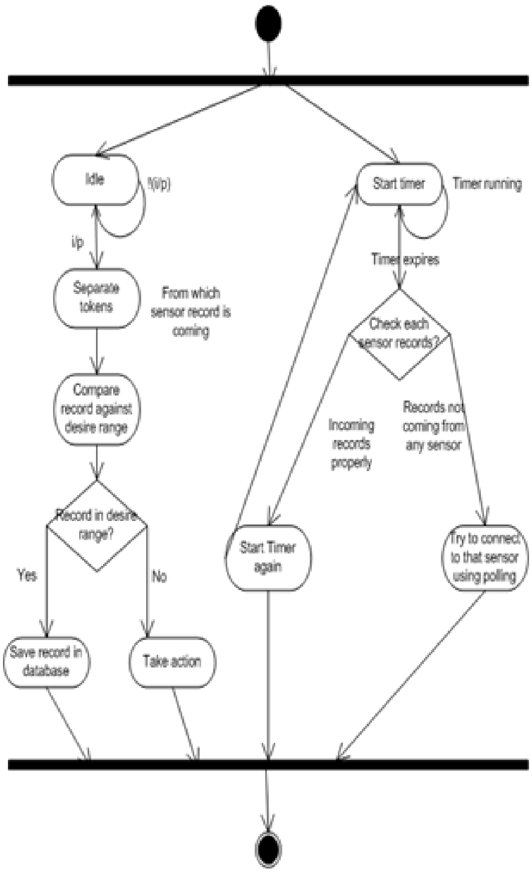


Figure 1: Flow Chart for Implementation of Monitoring Device

2.2 Steps Used for implementing Sensors

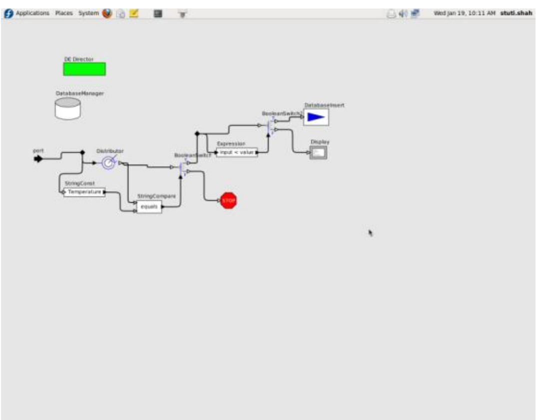


Figure 2: Implemented Monitoring Device in Ptolemy

6. Once random number is generated it is compared with Boolean switch to find out whether to generate value inside range or value outs ide range.

7. Again random number component is used after taking decision to generate actual values. If value need to be generated in the range then it will take range variable values set by primary artificial intelligence method described in section III and it will set that variable for generating values else it will take the range and by some simple mathematical addition or subtraction it will take range which is not actual range parameter and generate values in that range.

8. It will form packet which includes the token identifying the sensor name so controlling device can use this token to identify from which sensor record is coming and record value which is generated by random number. Merge component will combine both record and string so the packet is formed

9. It will use out port to send that packet to controlling device. The packet is broadcasted to all nodes but packet will be used by controlling device.

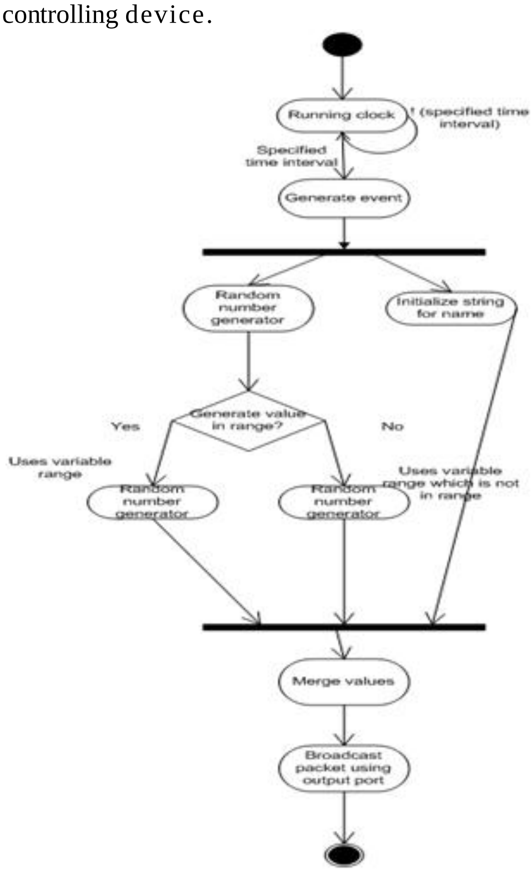


Figure 3: Flow Chart for Implementation of Sensor

The way of implementation of sensors can be shown with flow chart in figure3.

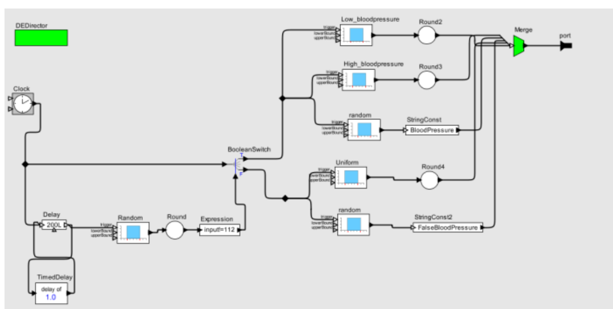


Figure 4: Sensor Implementation in Ptolemy

Figure 5: Sensor Implementation in Ptolemy

2.3 Adding Sensor to Ptolemy library

Ptolemy provides facility to add actor in user library by just simple right clicking on the model built and click on save actor into user library. Once this library is formed it will appear for every time Ptolemy graphical user interface is opened shown in the below figure.

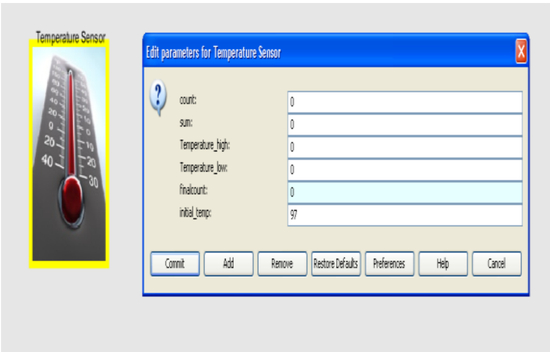


Figure 6: Sensor Added in Ptolemy Library.

3 Incorporating Artificial Intelligences Method and Flexibility in Parameter Configuration

1. For generating values the sensors are using variables that can be set by parameter.

2. Ptolemy provides way to create parameter which can be specified according to requirements and when that component is added to library or it is used again it can take values specified by user.

3. The use of parameter configuration in implementation of this sensors are for example the range of glucose level changes from person to person so that parameters can be set by doctor. So the values which are generated is in the range of that values

4. The sensors which are implemented uses primary artificial learning method which is learning from past records

5. Once initial range of glucose level is set by doctor sensor would start generating values in that specified range. The model will have one counter running on it. First ten records will be added and when count turns to eleven the average of the records are taken and it will change the range from old one set by doctor to newly calculated one.

6. This value range is broadcasted to controlling device so that controlling device can also set values according to the range values coming from in- coming port.

7. Once range parameter is calculated random number generator will use that range to generate values.

Snap shots for parameter configuration and sensor implementation with artificial intelligence is shown in figure7.

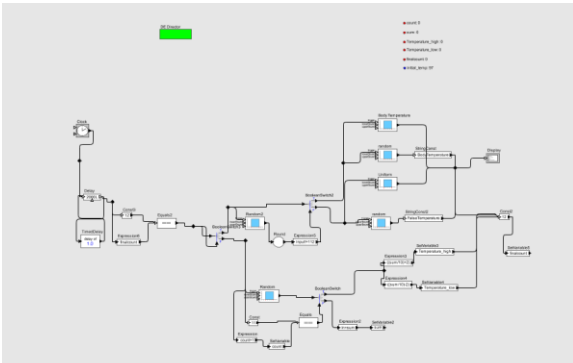


Figure 7: Implementation of Sensor Network with Artificial Intelligence

4.Conclusion

Wireless sensors implementation on Ptolemy provides a good way of realizing actual sensor implementation. It provides flexibility for configuring parameters and adding that to user library so the sensors can be re use for similar types of system. The sensors implemented in this paper are all working on clock bases and used for health monitoring systems. Similarly different sensors used for different purpose can be implemented on Ptolemy.

5.Future work

Primary artificial intelligence method for sensor implementation is used. Different learning algorithms such as neural networks, genetic algorithms can be implemented for more accurate calculation of range parameter.

References

1. Aleksandar Milenkovic,Chris Otto,Emil Jovanov,"Wireless sensor networks for personal health monitoring: Issues and an implementation",Computer Communication,Science Direct,Volume 29,Issues 13-14,p.2521-2533,21 Au- gust, 2006

2. Katsushi Ikeuchi,Takeo Kanade,"Modelling sensors: Toward automatic generation of object recognition program", Computer Vision,Graphics and Image Processing,Science Direct,Volume 48,Issue 1,p.50-79,October 1989

3. Philip Baldwin,Sanjeev Kohli,Edward A. Lee,Xiaojun Liu,Yang Zhao, "Modeling of sensor nets in Ptolemy II", 3rd international symposium on Information processing in sensor networks,ACM,2004

4. Kay Romer,Friedemann Mattern,"The Design Space of Wireless Sensor Networks",Wireless Communication,IEEE,Dec 2004

5. Sanjib Kumar Panda,"Assistive Technology-For Personnel Wellness and Quality Living of Ageing Population"

6. Korhonen I.,Parkka J., Van Gils M.,"Health monitoring in the home of the future", Engineering in Medicine and Biology Magazine,IEEE,Volume 22,Issue 3,p.66-73,May-June,2003

7. Jafari R,Encarnacao A,Zahoory A.,Dabiri F,Noshadi H,Sarrafzadeh M,"Wireless sensor networks for health monitoring",Second International Conference on Mobile and Ubiquitous Systems,IEEE,2005

8. Toshiyo Tamura,Tatsuo Togawa,Mitsuhiro Ogawa,Mikiko Yoda,"Fully au- tomated health monitoring system in the home", Medical Engineering and Physics,Volume 20,Issue 8,October 1998,p.573-579

9. Upkar Varshney,"Pervasive Healthcare and Wireless Health Monitoring",Mobile Networks and Applications,Springerlink,Volume 12,p.113-127,2007

10. Edward A. Lee,Stephen Neuendorffer,"Building Ptolemy II Models Graph- ically",EECS,October 31,2007

Simulation of unified architecture of IEEE 802.11a and 802.16a PHY layers using MATLAB

Devashish Raval[1], Nilesh Ranpura[2], Ekata Mehul[3], Zuber Saiyed[4]

[1] *Department of Electronics & Communication Engineering, Charotar Institute of Technology, Changa, Gujarat, India*
[2,4] *Department of Electronics Engineering, Sardar Vallabhbhai National Institute of Technology, Surat, Gujarat, India*

[3] *ASIC Division, Einfochips Pvt. Ltd., Ahmedabad, Gujarat, India*

Abstract*— WiFi and WiMAX are widely used wireless technologies for accessing internet. This paper elaborates the simulation of unified architecture physical layers of WiFi and WiMAX technologies which are compliant to IEEE standards 802.11a and 802.16a respectively. The PHY specifications of these standards are described with block schematics. SIMULINK based simulation of the block schematics of these two technologies is done and the results of packet error rate are studied as compared to the data rates specified in the standards. A comparative analysis is done based on the OFDM parameters and their variations in the two technologies. The scope of this paper is limited to the digital signal processing involved in the PHY layers of WiFi and WIMAX technologies.*

Keywords: *OFDM, WiFi, WiMAX, SIMULINK*

1. Introduction

WiFi and WiMAX are the well developed and standardized technologies working on OFDM platform. Their physical layer architecture are much similar accept WiMAX physical layer has RS encoder & Decoder at transmitter and receiver respectively. In this paper the physical layer specification similarities and differences of IEEE 802.11a (WiFi) and IEEE 802.16a (WiMAX) are discussed. We used the conventions for WiFi as 11a and WiMAX as 16a for the entire paper.The architecture of PHY layer of the IEEE 802.16a is similar to IEEE 802.11a except some differences are stated as below.

Table 1. Architectural differences between IEEE 802.11a and IEEE 802.16a

Parameters	IEEE 802.11a	IEEE 802.16a
Scrambler	7 bits	15 bits
FEC coder (Reed Solomon) encoder profile (N,K,T) supported profiles (N,K,T)	Not present	(255,239,8) (12,12,0), (32,24,4), (40,36,2), (64,48,8), (80,72,4), (108,96,6), (120,108,6)
FEC coder (Convolutional coder) Code rates	1/2, 2/3 , 3/4	1/2, 2/3, 3/4, 5/6
Interleaver Block Size	48, 96, 192, 288	192, 384, 768, 1152
Pilot/Guard Insertion Pilot indices Guard indices	-21, -7, 7, 21 -32 to -27, 0, 27 to 31	-88, -63, -38, -13, 13, 38, 63, 88 -128 to -101, 0, 101 to 127
IFFT Size	64	256
CP insertion CP size Short Preamble Long Preamble	16 (32 for preamble) 8 16-sample symbols 2 64-sample symbols	8, 16, 32, 64 4 64-sample symbols 2 128-sample symbols
S/P symbol / CP sizes	64 / 16	256 / 8, 16, 32, 64
FFT Size	64	256
Deinterleaver Block Size	48,96,192,288	192, 384, 768, 1152
Reed Solomon Decoder Reed Solomon Settings	Not present	Same as Reed Solomon Encoder
Descrambler LFSR settings	Same as Scrambler	Same as Scrambler

The organization of paper is as follows. Section 1 contains the comparison of physical layers of 11a and 16a. Section 2 contains the description of block diagrams of SIMULINK model for unified architecture of 11a and 16a. We conclude in section 3.

2. SIMULINK based simulation of unified architecture of 802.11a and 802.16a PHY layers

Description of each block of the SIMULINK system and their respective hierarchy levels are described below. Each block is shown by its hierarchy level that denotes the level at which the block is placed in the system. Hierarchy level 0 is Top level containing the top parent blocks. The first level subsystem blocks are shown as Hierarchy level 1. The level 2 subsystems are the child blocks of Hierarchy level 1.

1) Variable data rate block:

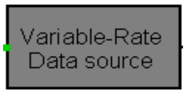


Figure 1. Variable rate data source(Hierarchy level 0)

- Generates random data input: 6,9,12,18,24,36,48,54 Mbps for 11a and 12,18,24,36,48,72,96,108 Mbps for 16a configuration.

2) Convolutional Encoder Bank**:**

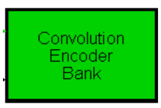


Figure 2. Convolution Encoder Bank(Hierarchy level 0)

- Contains CC coder blocks with different code rates. This block performs Forward error correction operation on input data stream.

The internal blocks of this block are shown as below.

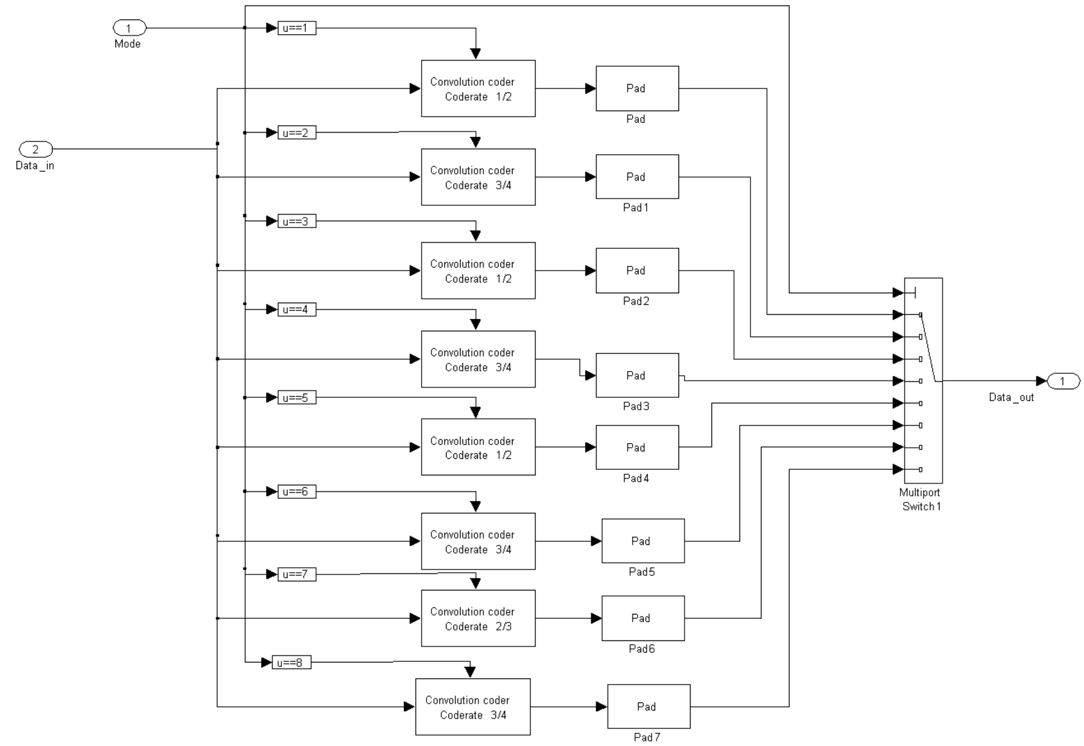


Figure 3. Internal blocks of Convolution Encoder Bank (Hierarchy level 1)

- As per the specification of 802.11a and 16a the convolutional encoder used works on code rates of 1 / 2 , 3 / 4, 2/3, 5/6(for 16a).
- The multiport switch block chooses between a number of inputs.
- The pads are used to match dimensions of the output of CC coder on every pins of the multi port switch.

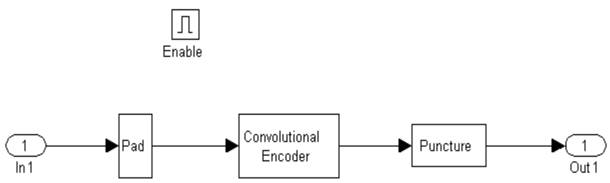


Figure 4. Internal blocks of Convolutional encoder(Hierarchy level 2)

- The convolutioal encoder

performs the encoding operation. The puncture block selects a particular code rate.

3) Interleaver Bank:

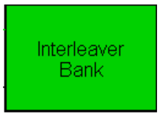


Figure 5. Interleaver Bank(Hierarchy level 0)

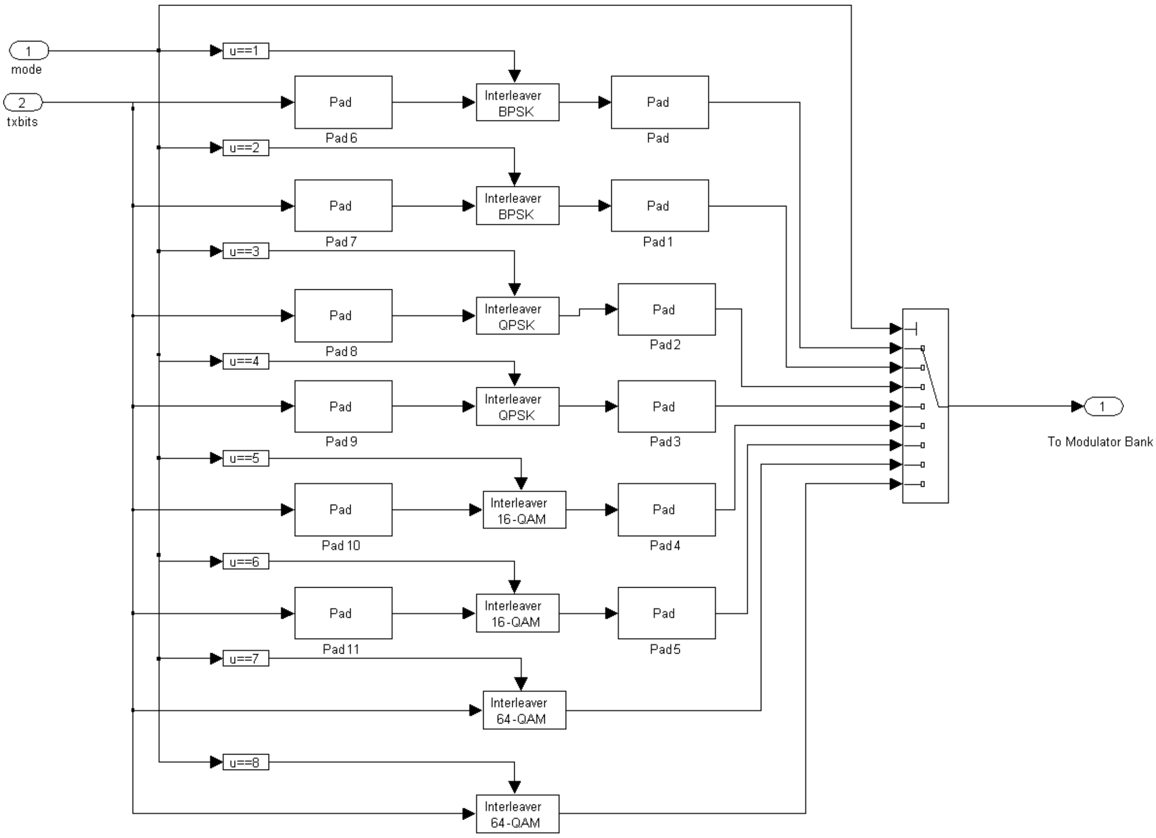


Figure 6. Internal blocks of Interleaver bank(Hierarchy level 1)

- This block performs interleaving operation on the encoded bit stream.

The internal blocks of this block are shown below.

- The pads are used for matching the dimensions of inputs of interleaver blocks.

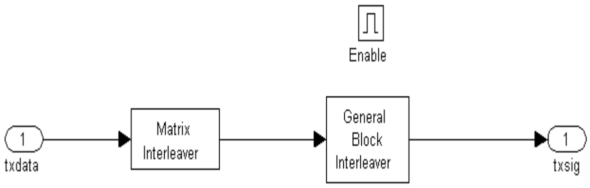


Figure 7. Internal blocks of interleaver(Hierarchy level 2)

- Inside the interleaver blocks two interleaver blocks are cascaded.
  - Matrix interleaver
  - Block interleaver

4) Modulator bank:

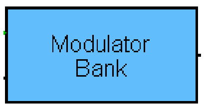


Figure 8. Modulator Bank(Hierarchy level 0)

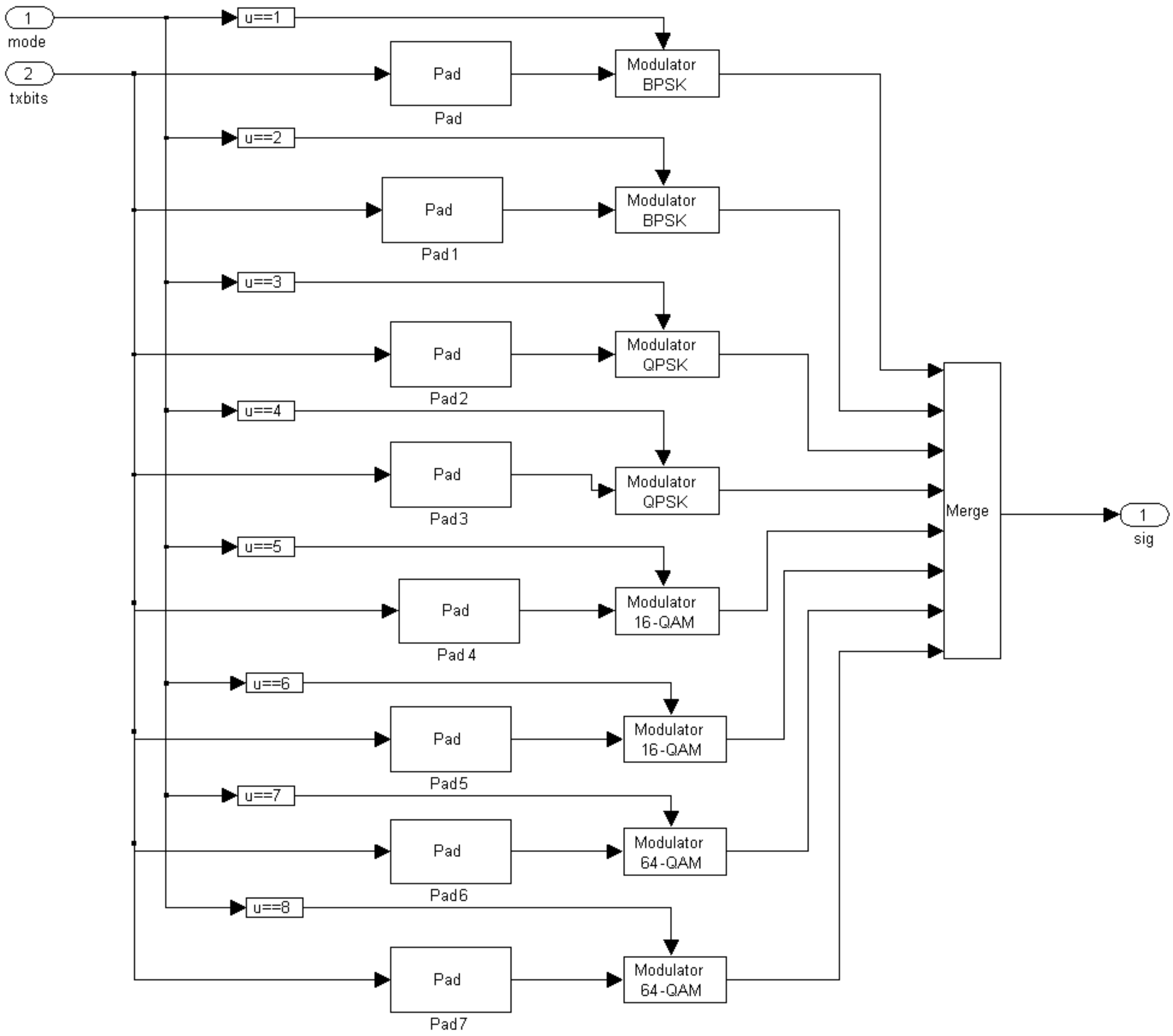


Figure 9. Internal Blocks of Modulator Bank(Hierarchy level 1)

- This block performs QAM modulation on the interleaved bit stream. It will output complex numbers assigned to a group of bits.

The internal blocks of this block are as shown below.

- Each modulator block is named as per the M-ary modulation performed by the modulator block. Depending on mode the m-ary number is changed from 2,4,16 and 64 referring to BPSK, QPSK, 16 QAM and 64-QAM. The mapping schemes for 11a and 16a are same.
- The Merge block combines its inputs into a single output line whose value at any time is equal to the most recently computed output of its driving blocks.

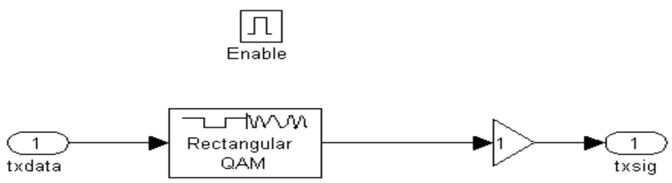


Figure 10. Internal Blocks of Modulator(Hierarchy level 2)

5) Serial to parallel block:

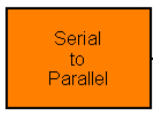


Figure 11. Serial to parallel converter block(Hierarchy level 0)

- This blocks is used to insert the pilot carriers and DC components to the bit stream and to convert the serial data into parallel data carriers which can be given as input to the IFFT block.

Internal block diagram of this block is as given below.

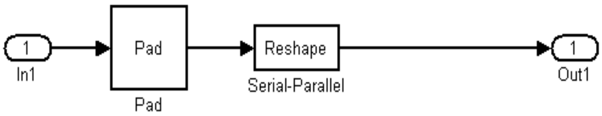


Figure 1 Internal blocks of serial to parallel converter block(Hierarchy level 1)

- The Reshape block changes the dimensionality of the input signal to a dimensionality that you specify, using the block's Output dimensionality parameter.

6) IFFT block:

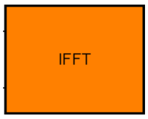


Figure 2 IFFT block(Hierarchy level 0)

- The IFFT block computes the inverse fast Fourier transform (IFFT) of each channel of a P-by-N or length-P input, u. When the Inherit FFT length from input dimensions check box is selected, the input length P must be an integer power of two, and the FFT length M is equal to P.

The internal blocks of this block are as shown below

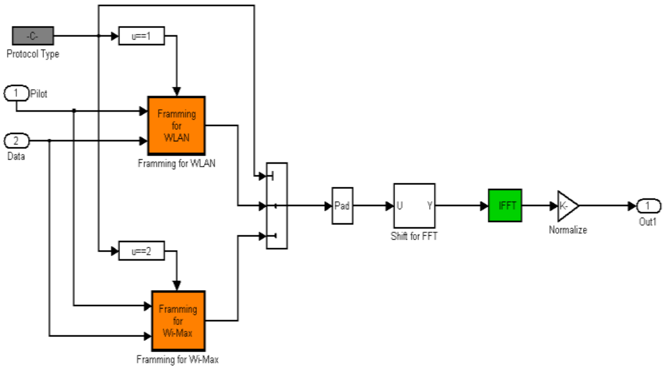


Figure 5 Internal blocks of IFFT blocks(Hierarchy level 1)

- Depending on the protocol type set in the system settings block the framing blocks are chosen. The pad provides the length of data compatible with IFFT.
- The IFFT points are inherited from the data

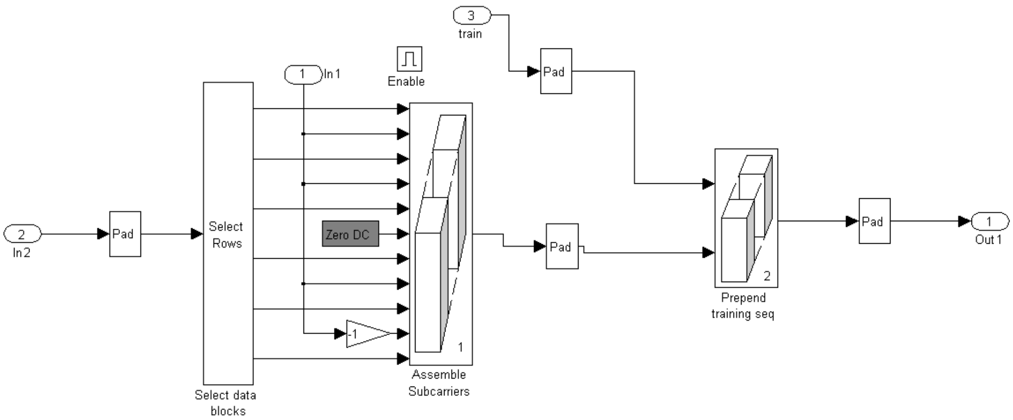


Figure 3 Internal blocks of Framing for WLAN(Hierarchy level 2)

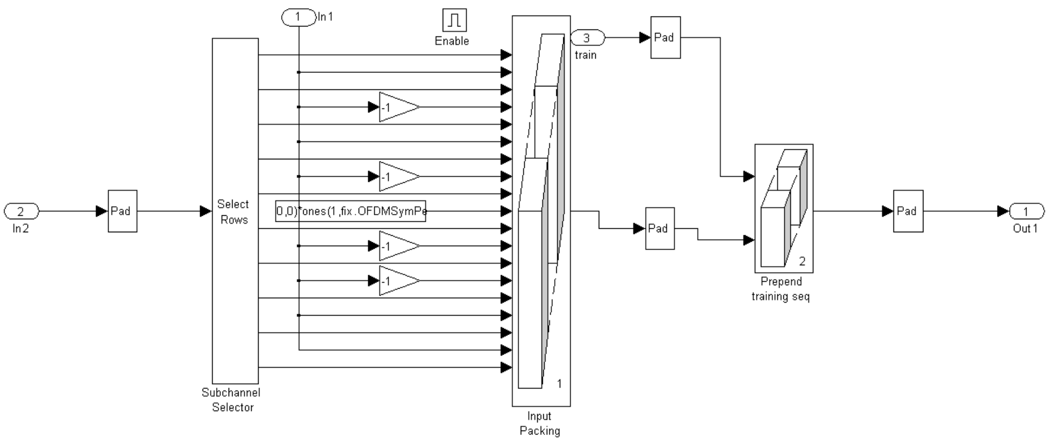


Figure 4 Internal blocks of Frammig for 16a(Hierarchy level 2)

dimensions. The framming blocks make the data dimension 64 and 256 for 11a and 16a respectively.

- The select rows block selects the inputs of concatenate block(Assemble subcarrier). The concatenate block combines the data carriers, pilot carriers and training carriers.

- The Framing blocks are used to make the parallel bit stream to be compatible with the Inputs of the IFFT block. The Pad adds extra zero bits to the input pins of IFFT to assure same length of data at each pins of IFFT.

7) Cyclic prefix insertion block:

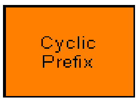


Figure 6 Cyclic prefix block(Hierarchy level 0)

- This block adds the portion of input frame to its front end. Thus inserting a guard interval. For 11a 16 bits and for 16a 64 bits are the CP lengths.

The internal block of cyclic prefix is as shown in figure.

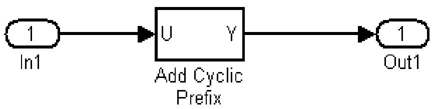


Figure 7 Internal block of Cyclic prefix block(Hierarchy level 1)

- Selector Block: The Selector block generates as output selected or reordered elements of an input vector, matrix, or multidimensional signal. A Selector block accepts vector, matrix, or multidimensional signals as input.

8) Parallel to serial block:

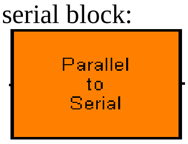


Figure 8 Parallel to serial block(Hierarchy level 0)

The internal block diagram of this block is as shown below.

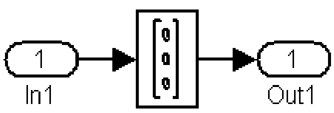


Figure 9 Internal blocks of Parallel to serial block(Hierarchy level 1)

- The reshape function is used to convert the parallel data streams to serial bit stream.

9) Wireless channel:

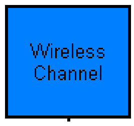


Figure 10 Wireless channel (Hierarchy level 0)

- The wireless channel is used to insert noise in the OFDM modulated data. There are three options
  - No noise
  - Non dispersive multipath noise
  - Dispersive multipath noise

The internal blocks of this block is shown as below

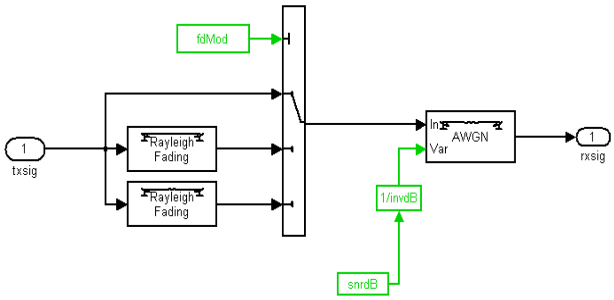


Figure 11 Internal blocks of Wireless channel(Hierarchy level 1)

- As shown in figure the multi port switch selects one of the three options stated above. After this the AWGN noise is added to the bit stream with user defined level of noise power (in dB).
  1) Multipath Rayleigh Fading channel
  2) AWGN channel(Addative White Gaussian Noise)

10) Serial to parallel block at receiver:

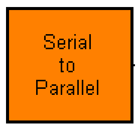


Figure 12 Serial to parallel block at receiver(Hierarchy level 0)

- This block performs inverse operation of parallel to serial block of transmitter.

The internal blocks are as shown below.

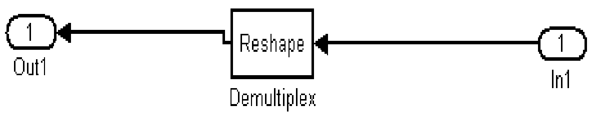


Figure 13 Internal blocks of serial to parallel block at receiver(Hierarchy level 1)

- Reshape is a library block that converts the dimension of a matrix to the input dimensions

11) Remove cyclic prefix block:

Figure 14 Remove cyclic prefix block(Hierarchy level 0)

- This block removes the cyclic prefix part from the input bit stream.

The internal blocks of this block are as shown below.

Figure 15 Internal blocks of remove cyclic block(Hierarchy level 1)

- This block uses a selector block which will select only the desired part of the input bit stream and drop the cyclic prefix part.

12) FFT block:

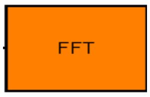


Figure 16 FFT block(Hierarchy level 0)

- This block performs the inverse operation of the IFFT block and will convert the time domain data to frequency domain.

The internal block diagram is as shown below.

- This block also removes the DC zeros contained in the input data. The pilots are extracted and discarded. The training sequence given is multiplied by the input data. The result is given to equalizer gain block which performs the averaging operation and stabilizes the constellation. The data is further sent to the next block.

13) Disassemble OFDM frames:

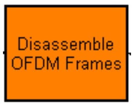


Figure 29 Disassemble OFDM frames block(Hierarchy level 0)

- This block converts the parallel data carriers to serial carrier

The internal block diagram is as shown below.

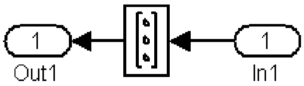


Figure 10 Parallel to serial converter(Hierarchy level 1)

- It performs the parallel to serial operation.

14) Demodulator Bank:

Figure 21 Demodulator Bank(Hierarchy level 0)

- The demodulator bank recovers digital bits from the modulation symbols.

The internal diagram of this block is as shown below

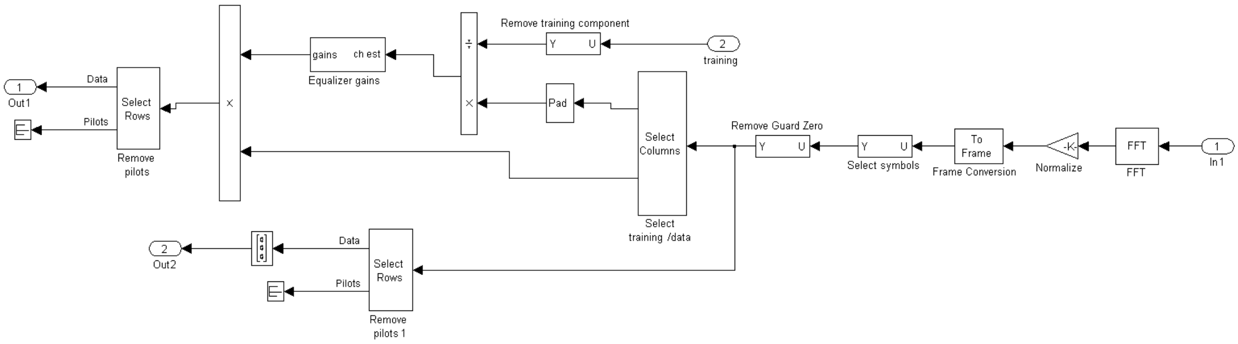


Figure 28 Internal block diagram of FFT block(Hierarchy level 1)

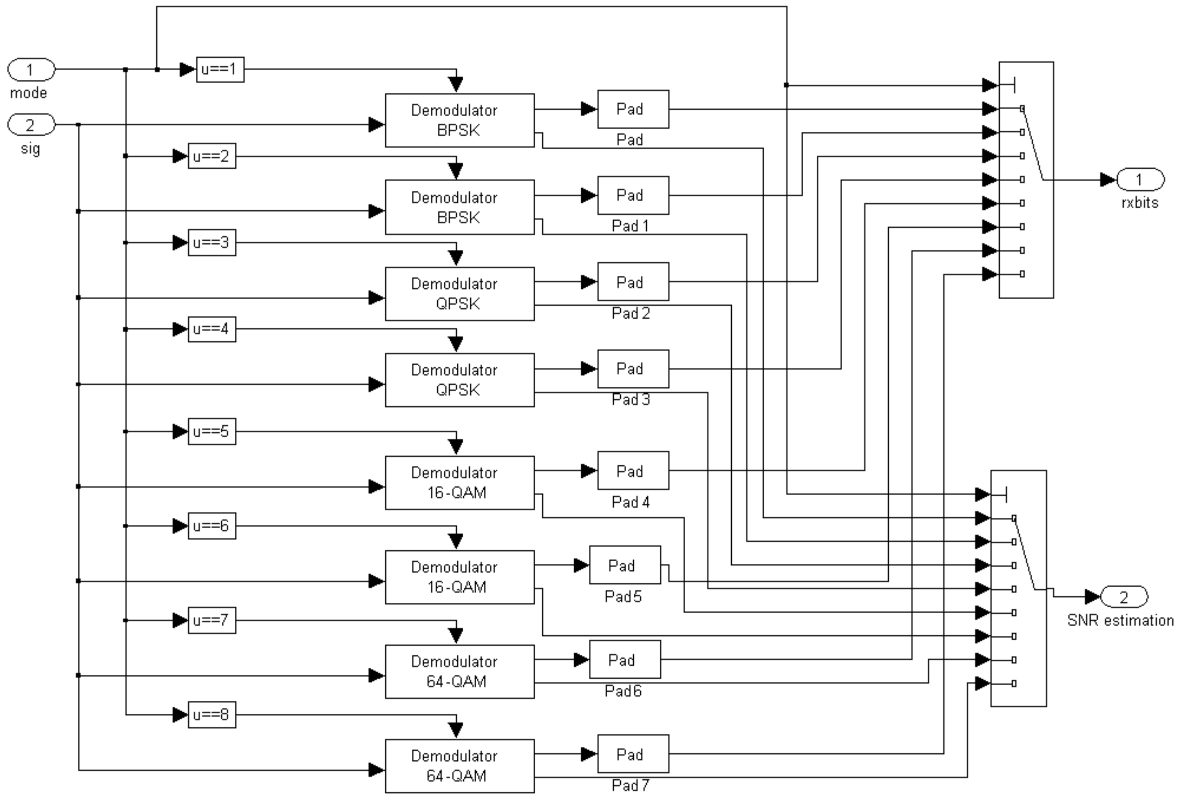


Figure 42 Internal diagram of Demodulator bank(Hierarchy level 1)

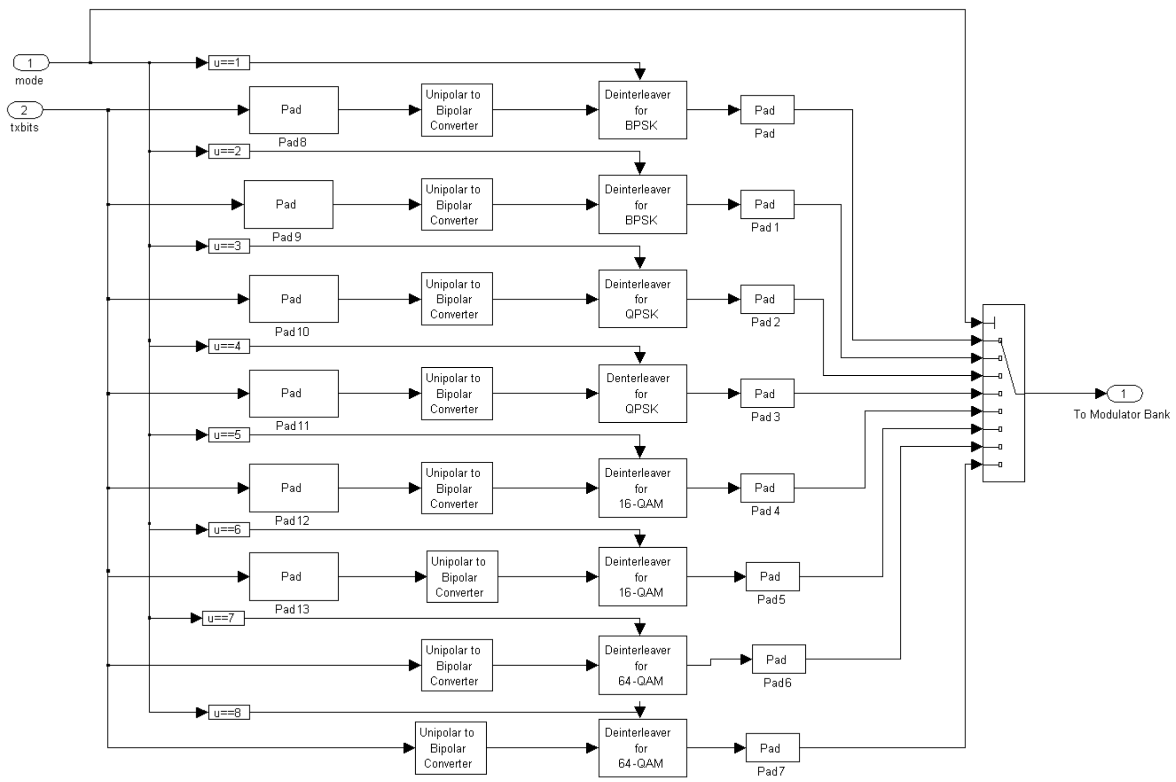


Figure 35 Internal block diagram of Deinterleaver bank(Hierarchy level 1)

- As shown in figure 2.1.27, various demodulators such as BPSK, QPSk, 16-QAM and 64 QAM are used to demodulate the data. As per the mode the respective demodulator bank. The demodulator blocks are enabled sub systems which are controlled by the function blocks, u==1, u==2, etc. The Pads are used to equalise the output of Demodulator blocks. This is done because the multiport switch can not take different dimensions of data.

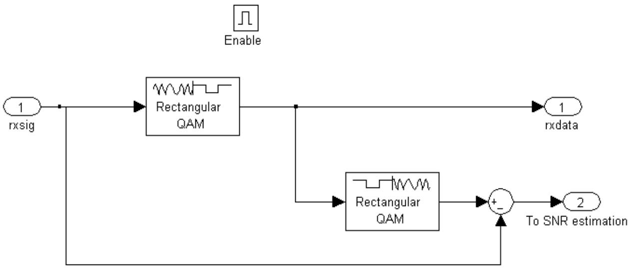


Figure 53 Internal blocks of Demodulator block(Hierarchy level 2)

- Here the demodulated data is given to deinterleaver bank as well as to SNR estimation block.
- The rectangular QAM block performs the demapping of data and recovers the original bitstream. The bitstream is again modulated and subtracted with the received data. This data is given to SNR estimation block.

15) Deinterleaver Bank:

Figure 64 Deinterleaver Bank block(Hierarchy level 0)

- This block performs the deinterleaving operation on the demodulated data.

The internal block diagram is as shown below

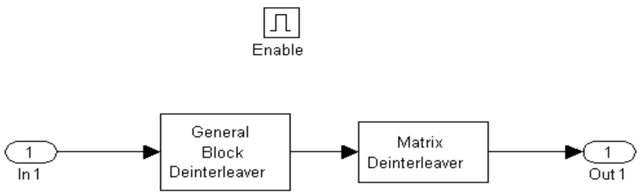


Figure 36 Internal blocks of Deinterleaver(Hierarchy level 2)

- The general and matrix interleaver performs two permutation (rearranging) of data to get the encoded data.

16) Viterbi decoder Bank:

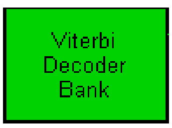


Figure 37 Viterbi Decoder Bank block(Hierarchy level 0)

- This block recovers the original information data inserted to by random data generator, by decoding the coded bitstream using viterbi decoder blocks.The internal block diagram is shown as below.
- As per the mode, respective viterbi decoder will be selected. The viterbi block estimates the correct data based on maximum likelihood algorithm. Various code rates are 1/2, 2/3, 3/4 and 5/6, where 5/6 is only used in 16a implementation. For code rates 2/3, 3/4 and 5/6 depuncturing is done before applying them to viterbi decoder block

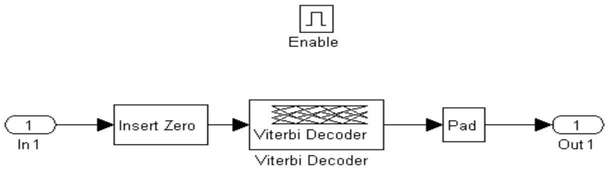


Figure 39 Internal blocks of Viterbi decoder(Hierarchy level 2)

- The insert zero block performs the de-puncturing operation. The viterbi decoder block estimates the correct data using trellis diagram.

17) Mode Control:

Figure 70 Mode Control(Hierarchy level 0)

- This block takes input from SNR estimation block and changes the mode of the system accordingly. When SNR is higher then mode will be increased towards 8 and when SNR is lower the mode is decreased towards 1.

The internal block diagram of this block is as shown below.

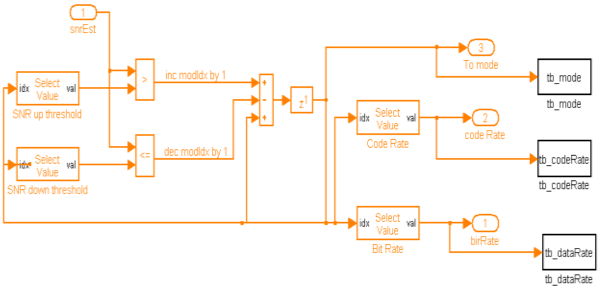


Figure 81 Internal blocks of Mode control block(Hierarchy level 1)

- The select value blocks select a value from snrup matrix and snrdn matrix.
- These matrices contains SNR values for which the data rates are supposed to changes.
- The comparison block gives incremented or decremented index value. This index value actually selects a matrix value from Mode, Code rate , data rate and snrup and snrdn

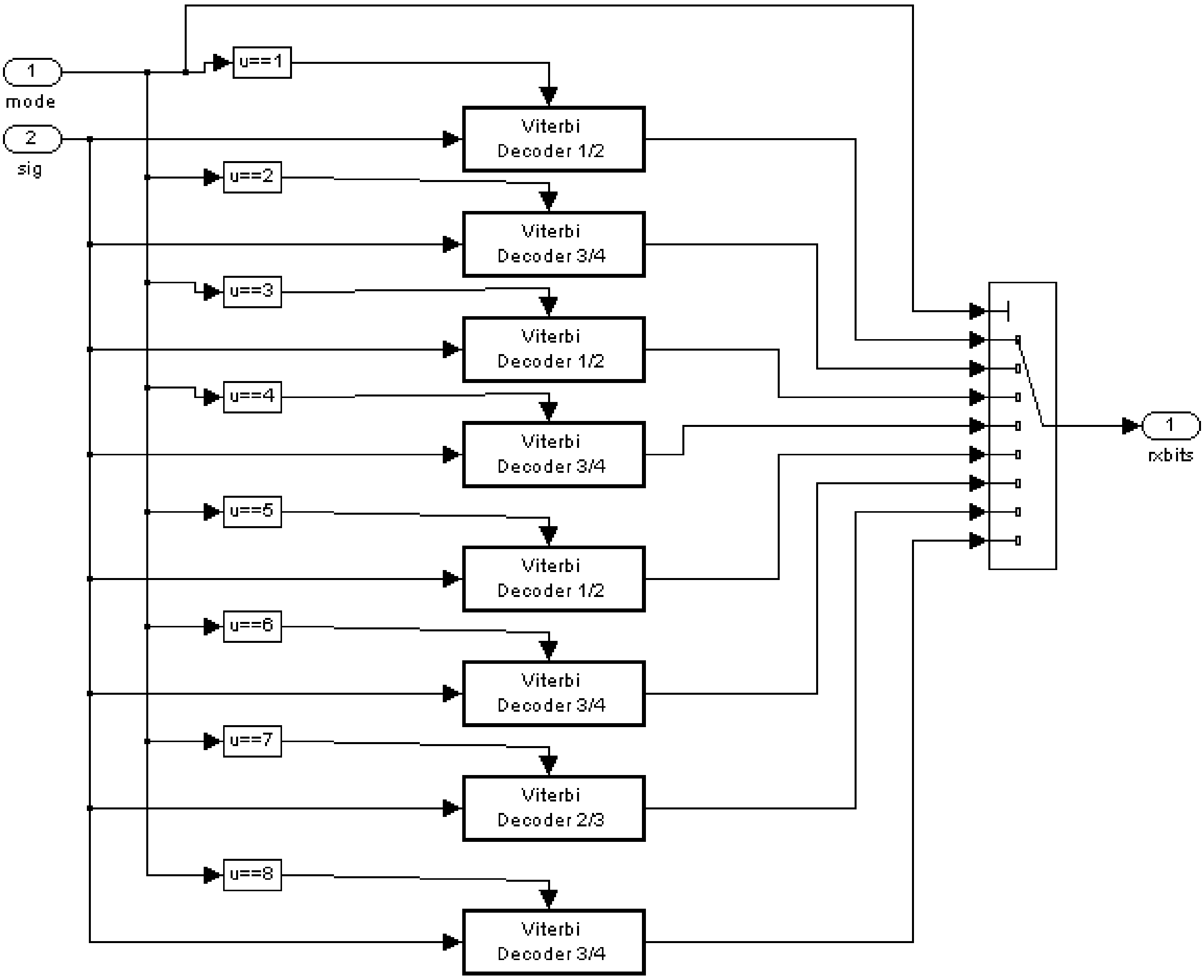


Figure 38 Internal blocks of Viterbi decoder Bank(Hierarchy level 1)

matrices.

- Mode output is used to configure Variable data rate source, Convolution encoder bank, Interleaver bank and Modulator bank at transmitter and to demodulator bank, deinterleaver bank and viterbi decoder bank at receiver.
- The tb_mode, tb_codeRate and tb_datarate are To workspace blocks used to export the simulation data to excel spreadsheet.

The example of simulation data porting to excel spreadsheet is as shown below.

Table 2 Simulation data poted to Excel worksheet

Mode	Code_Rate	Data_Rate	Modulation_Scheme
6	0.75	36	16-QAM
5	0.5	24	16-QAM
4	0.75	18	QPSK
3	0.5	12	QPSK
2	0.75	9	BPSK
1	0.5	6	BPSK
1	0.5	6	BPSK
1	0.5	6	BPSK
1	0.5	6	BPSK
1	0.5	6	BPSK
1	0.5	6	BPSK
1	0.5	6	BPSK
1	0.5	6	BPSK

1. SIMULINK model for IEEE 802.11a and 802.16a

Fig 2.1 shows the SIMULINK model compatible for 11a and 16a PHY layers. The System settings block configures the model in 11a or 16a with number of OFDM symbols to be processed at a time.

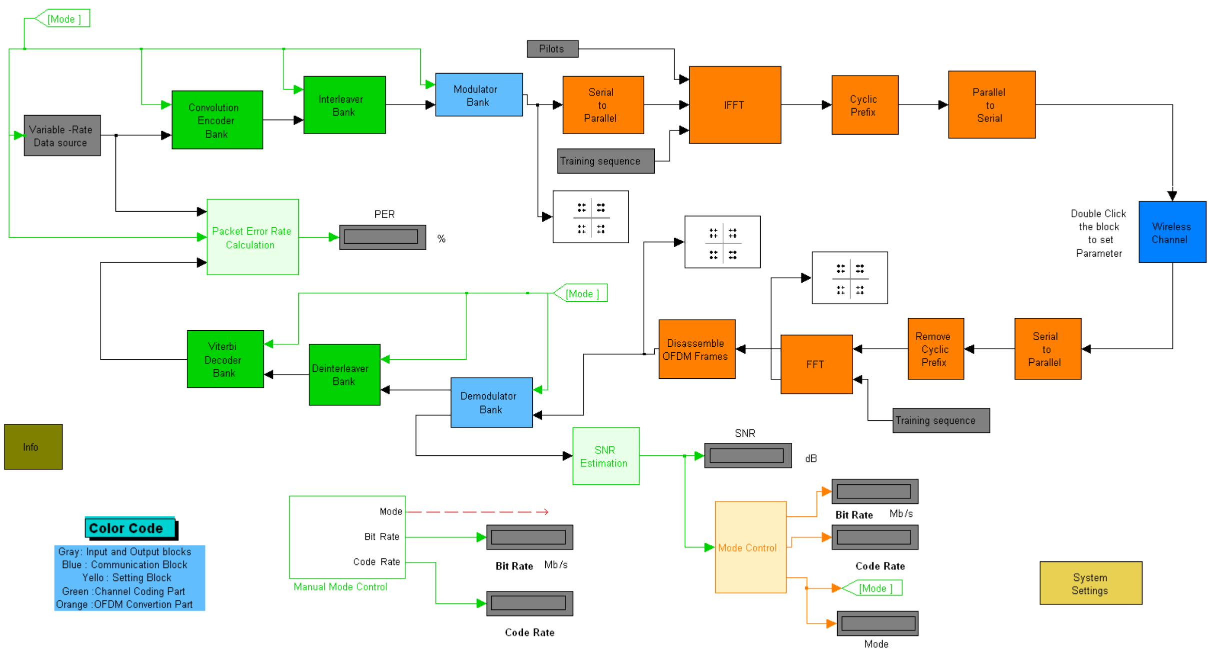


Figure 42. SIMULINK Schematic for Combined Configuration of 11a and 16a

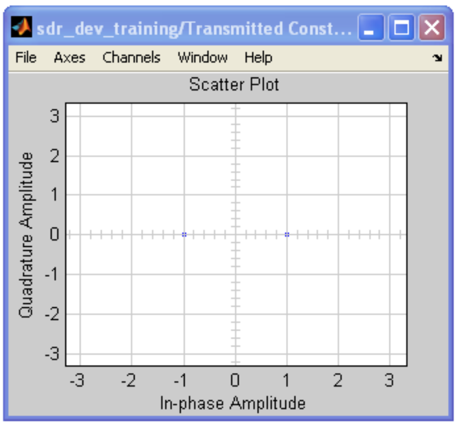


Figure 43(a). Transmitted Constellation

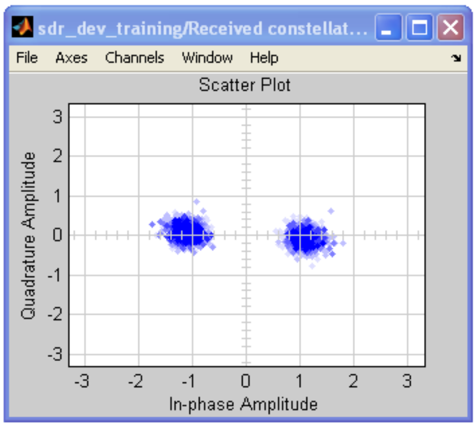


Figure 43(b). Received Constellation with equalization

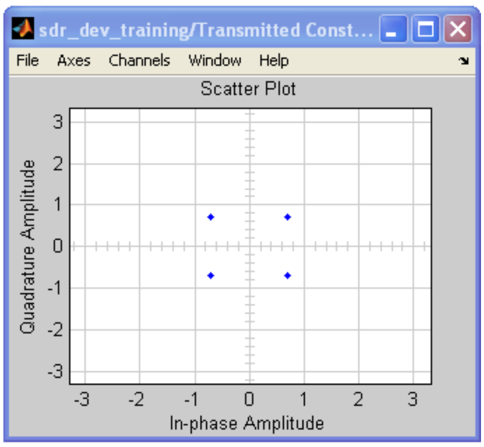


Figure 44(a). Transmitted Constellation

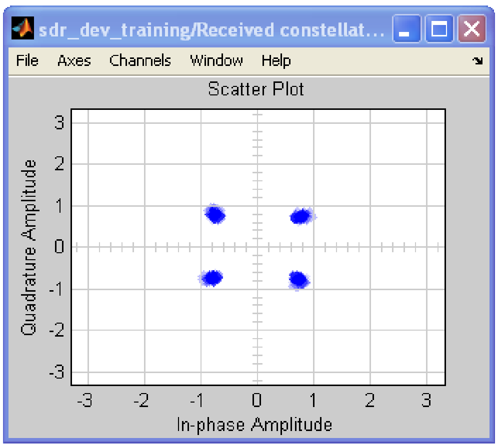


Figure 44(b). Received Constellation with equalization

The colors show the classification of the block in

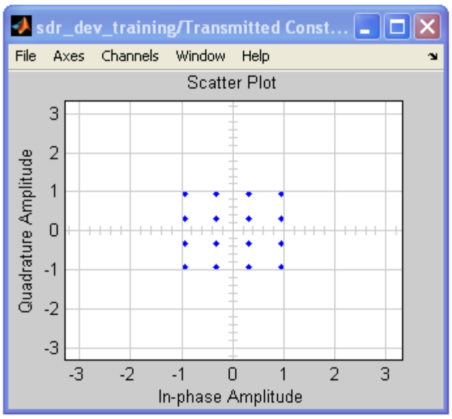


Figure 45(a). Transmitted Constellation

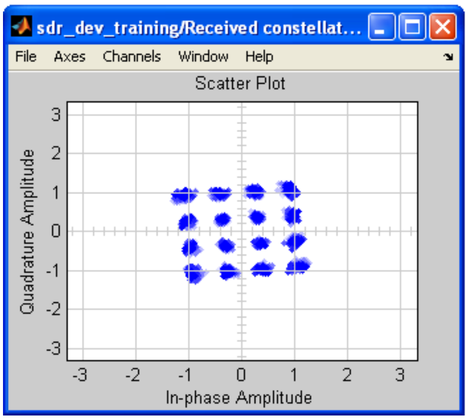


Figure 45(b). Received Constellation with equalization

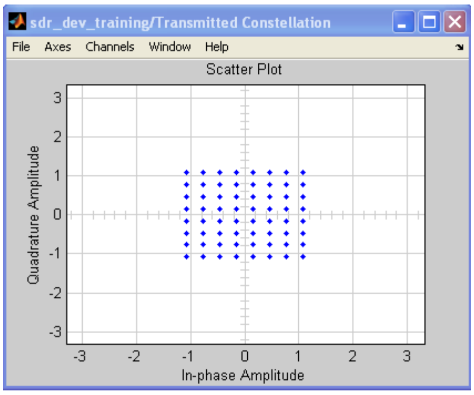


Figure 46(a). Transmitted Constellation

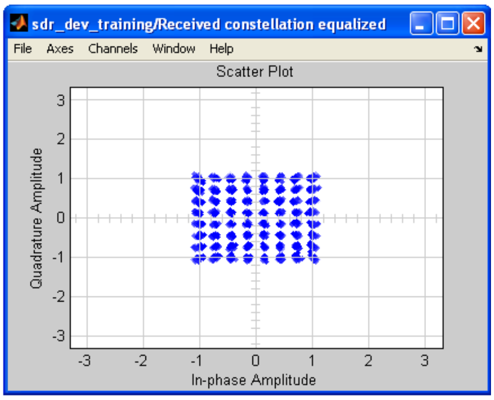


Figure 46(b). Received Constellation with equalization

Input and Output blocks, Communication blocks, setting blocks, Channel coding blocks and OFDM conversion block. Color codes for SIMULINK system are as follows

1) Gray : Input and output block
2) Blue: Communication block
3) Yellow: System setting block
4) Green: Channel coding block
5) Orange: OFDM Block

Above schematic is also supported by SDR_settings.m file which includes the initialization of the 11a and 16a blocks. Here the scrambler is not included for simulation purpose because the source itself is a random source and long strings of 0s and 1s are avoided.
Constellations Transmitted and received:

For BPSK (Data rate 6 and 9 for 11a and 12 and 18 for 16a) as shown in Figure 43(a, b)

For QPSK (Data rate 12 and 18 for 11a and 24 and 36 for 16a) as shown in Figure 44(a, b)

For 16-QAM (Data rate 12 and 18 for 11a and 24 and 36 for 16a) as shown in Figure 45(a, b)

For 64-QAM (Data rate 12 and 18 for 11a and 24 and 36 for 16a) as shown in Figure 46(a, b)

2. Conclusion

This work derives the necessary results for the final unified hardware to be implemented in FPGA. Following are the key points to summarize the work done so far.

1) Understanding of PHY layers of 11a and 16a standards is achieved.
2) SIMULINK based simulation provides the reference for the final design.
3) In the SIMULINK system, the mappings of BPSK, QPSK, 16QAM and 64 QAM are achieved at the receiver in presence of Rayleigh multipath fading.
4) For statistical analysis the mechanism of exporting the simulation data to the Excel worksheet is developed.
5) The entire system works according to IEEE 802.11a and 802.16a physical layer specifications as per the choice selected in the system settings blocks. Hence unification is achieved.

The simulation is performed for both 11a and 16a PHY configurations.

References

For a standard citation

1. IEEE Std 802.11a, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, High-Speed Physical Layer in the 5 GHz Band. 2007.
2. IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems, 2009.

For a book citation

3. J. Terry and J. Heiskala, *OFDM WIRELESS LANs: A Theoretical and Practical Guide*. SAMS, 2001.
4. Matthew Gast, *Wireless Networks,* O'Reilly, 2002.

For a paper citation

5. S. A. Fechtel, OFDM: From the Idea to Implementation, 2005.
6. Choi, S., Philips Research Labs, OPNET Model of the 802.11a Protocol, 15 November, 2000.
7. Prasad, R., van Nee, R., *OFDM for Wireless Multimedia Communications*, Artech House, 2000.
8. Chow, C. C. and Leung, V. C. M., "Performance of IEEE 802.11 Medium Access Control Protocol Over a Wireless Local Area Network with Distributed Radio Bridges,".
9. Alden J. Doyle, K. Han, S. Nadkarni, K. Seshadrinathan, R. Simha, Ian C. Wang, "Performance Evaluation of the IEEE 802.16a PHY Layer Using Simulation", EE381K-11 (14980) wireless communications project report.
10. Carl Eklund, Roger B. Marks, Kenneth L. Stanwood and Stanley Wang, "IEEE Standard 802.16: A Technical Overview of the WirelessMAN Air Interface for Broadband Wireless Access", IEEE Communications Magzine, vol.40, no.6, pp.98-107, June 2002.

A Software Product Line Methodology for Development of E-Learning System

Faheem Ahmed[1], Imran A. Zualkernan[2]
[1]College of Information Technology, UAE University, Al Ain, UAE
[2]Computer Science and Engineering, American University of Sharjah, Sharjah, UAE
f.ahmed@uaeu.ac.ae, izualkernan@aus.edu

Abstract: *Software product line has become one of the most promising software practices with the potential to substantially increase the development productivity in the software industry. Learning objects can be defined along the three dimensions of pedagogy, technology and the domain. We put forward a methodology to develop a software product line of E-Learning systems. This model identifies and elaborates the essential phases and activities. The model is divided into two phases, Domain Engineering and Application Engineering. Domain engineering phase consists of two views, Product Line Infrastructure View and E-Learning Analysis View. Application Engineering phase has Product Line Application View and Core Assets Development View. Various activities related to each view are identified to enhance the reuse development process for E-Learning software product line. The methodology demonstrates the use of explicit variability definition in a learning object at various levels of the model including Site, Structure, Skin, Services, Space Plan and Stuff. The methodology is validated against a commercial e-learning course in Six Sigma.*

1. Introduction

Software product line aims at curtailing the concept of "reinventing the wheel" in software development. It accentuates on consolidating the software assets in prescribed and a systematic way rather than an ad hoc and need to know basis. Software product line provides likelihood to accommodate the changing needs of the customers into new products by competently using software assets and allows capturing market segments for profitable business. Definition of software product line terminology has been widely explored by researchers [1, 2, 3, 4] to narrate an in-depth philosophy behind this approach. Synonyms of software product line terminology have also been widely used in Europe, for example "Product Families", "Product Population" and "System Families" etc. [5, 6]. The economic potentials of software product line have long been recognized in software industry [5, 7]. Software product line engineering is gaining popularity in the software industry. Some of the potential benefits of this approach include cost reduction, improvement in quality and a decrease in product development time. Software organizations are improving business operations such as technology, administration, and product development process in order to capture a major portion of the market share to be profitable. One of their major concerns is the effective utilization of software assets, thus reducing considerably the development time and cost of software products. Many organizations that deal in wide areas of operation, from consumer electronics, telecommunications, and avionics to information technology, are using software product lines practice, because it deals with effective utilization of software assets. Software product lines are promising, with the potential to substantially increase the productivity of the software development process and emerging as an attractive phenomenon within many organizations that deal with the software development. Several studies have been done, COPA [8], FAST [9], FORM [10], KobrA [11] and PuLSE [12] etc. to elaborate the software product line process. Independent work carried out in software reusability, object-oriented, and software architecture has reached a point at which many activities can be integrated to yield a new coherent approach to product-line integration. Traditional software life-cycle models do not encourage reusability within their phases. A product line can be built around *E-Learning system* by analyzing the products to determine the common and variable features. The product structure and implementation strategy around *E-Learning system* prepares a platform for several products, which aligns with the concept of software product line.

Recently, software development trends have caused single product development to evolve into "software product line architecture" (SPLA) which integrates lines of resulting products. The main objective of software product line is to reuse the architecture for successive product development. Clements [13] defines the term "software product line" as a set of software intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment and are developed from a common set of core assets in a prescribed way." The software product line is receiving an increasing amount of attention from software development organizations because of the promising results in cost reduction, quality improvements and reduced delivery time. Clement et al. [14] report that SPL engineering is a growing software engineering sub-discipline and many organizations, including Philips®, Hewlett-Packard®, Nokia®, Raytheon®, and Cummins®, are using it to achieve extraordinary gains in productivity, development time, and product quality.

Since the acceptance of object-oriented paradigm in the early 1980's, concepts of software architectures have evolved significantly. As Garlan and Perry [15] point out, traditionally software architecture includes the structure of the components of a program or system, their interrelationships, and the principles and guidelines governing their design and evolution. However, more recently, software architecture is being restructured towards a SPLA, where the focus is not on single product development but rather on multiple product development. In a SPLA, all of the products share the same architecture. Pronk [16] defines SPLA as a system of reuse in which the same software is recycled for an entire class of products, with only minimal variations to support the diversity of individual product family members. According to Jazayeri et al. [17], SPLA defines the concepts, structures and textures necessary to achieve variation in the features of diverse products while ensuring that the products share the maximum amount of parts in the implementation. Mika and Tommi [18] point out that SPLA can be produced in three different ways: from scratch, from an existing product group or from a single product. Hence, software product line architecture is an effective way to minimize risks and to take advantage of opportunities such as complex customer requirements, business constraints and technology.

Digital learning objects are a complex amalgamation of learning content, pedagogy and technology and represent the building blocks of any online course. While there are many views on what a learning object is [19] [20], in practice, a learning object typically contains learning objectives, reusable information objects and formative and/or summative assessments [21]. The reusable information objects can range from images, text or videos to games or simulations. Assessment objects, on the other hand, range from simple multiple-choice questions to adaptive testing techniques. Much like earlier days of object-oriented design, most discussions on learning objects have revolved around micro-level reuse; how to re-use a learning object in a different learning context. In the mean time, significant success has been achieved in macro-level reuse in the context of traditional object-oriented design. In specific, one promising area for macro-level re-use has been that of product-line engineering [13]. The commonality and variability characteristics of digital learning objects makes a clear case of software product line architecture development, which can be used to come up with multiple product development based on business case engineering.

1.1 Research Motivations

Digital learning objects are composite structure of learning content, pedagogy and technology. The use of the Internet further accelerates the popularity and significance of learning objects design and implementation at an unprecedented rate of growth. Conceptually different products using digital learning objects share commonality and variability up to certain extend. Software product line provides an opportunity to explicitly identify commonality and variability first at the architecture level and later at the implementation. Figure 1 shows the systematic view of generic product line architecture.

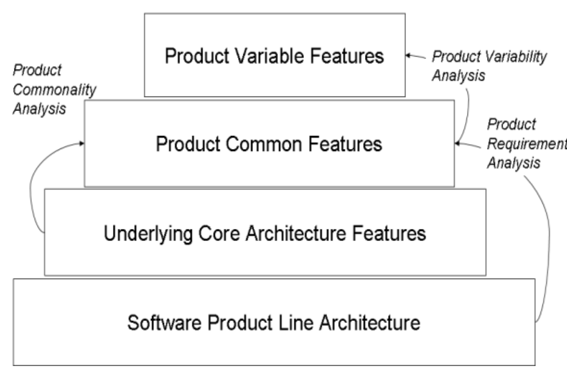


Figure 1: Software Product Line Architecture: A Structural View

A well-established requirements management activity for the software product line assists in understanding the scope and boundaries of the products to be developed and it helps in establishing the underlying core architecture features in terms of functionalities and their structure. Product line requirements deal with features or functionalities common to all the products belonging to that family. Product line requirements are composed of a constant and a variable part. The constant part comes from product line requirements and deals with features common to all the products belonging to the family. The variable part represents those functionalities that can be changed to differentiate one product from another. This activity defines the variable part of the product requirement. The product commonality analysis provides a set of features that are common to all products, whereas product variability analysis identify explicit variation points where we can introduce changes to develop new set of products. Since the popularity of both the use of digital learning objects and software product line, there is a need to establish a process methodology for the learning objects to make use of software product line in order to get benefits of the product line approach in terms of cost, quality and reduction of development time. The objective of this study is to present a methodology to establish a software product line for E-learning system. The methodology will concentrates in identifying core architecture features of digital learning objects along with explicit definition of commonality and variability.

Given the phenomenal success and popularity of both software product line and E-learning system's development paradigms, we assert that a process guidance model can help to identify and understand the activities and tasks that need to be undertaken in order to successfully develop a software product line for E-learning system. In order to address this gap, we propose a model of developing software product line based on E-learning by incorporating several concepts that characterizing various aspects of software product line and learning objects. The proposed model identifies the interdependency of various activities of software product line and learning objects and describes different ways of exploiting the relationships between those activities in order to guide the process of developing learning objects based software product line. It should be clarified that such a process guidance model will not aim to replace existing software product line development and maintenance models and frameworks such as reported in [1] [22]. Rather, this model complements those frameworks for establishing and maintaining software product line in E-Learning context. Since Software architecture and its related

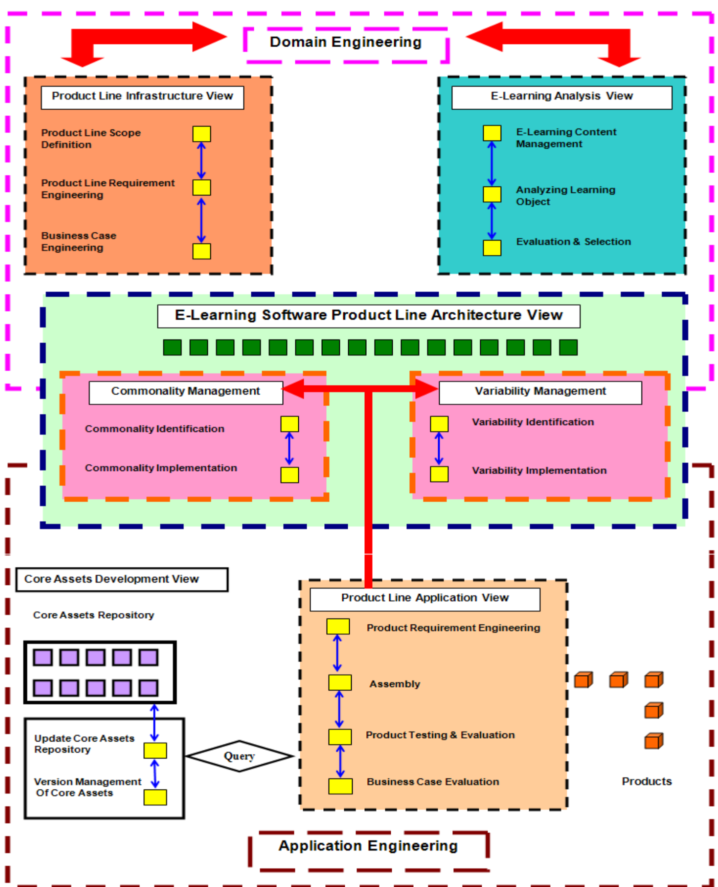


Figure 2: The Software Product Line Development Methodology for E-Learning System

issues are considered of paramount importance in the successful development and maintenance of a software product line [1] [23], this model emphasises the vital role of software architecture in developing E-learning based software product line.

2. Software Product Line Methodology for E-Learning System

The methodology shown in Figure 2 describes a complete life cycle for a software product line development for E-Learning system starting from conceptualization, initiation and development. This process methodology focuses on and encourages software reuse from repository within an application domain. The process methodology has two phases, *Domain Engineering* and *Application Engineering*. The *Domain Engineering* consists of two views, *Product Line Infrastructure View* and *E-Learning Analysis View*. Similarly the *Application Engineering* has two views, *Product Line Application View* and *Core Assets Development View*. Each view describes the development process with respect to its perspective and identifies the core activities to be performed in that view.

2.1 Domain Engineering

The *Domain Engineering* phase of the proposed methodology helps in establishing an infrastructure for software product line and constructs a core assets repository for product development. During the *Domain Engineering* phase, we initiate *Product Line Infrastructure View* and *E-Learning Analysis View*. The iterations in the activities of both views provide feedback to one another. The aim is to generate a core assets repository and a base line software product line architecture, which fulfills the product line requirements and meets the production constraints. In *Domain Engineering* phase we concentrate on the "E-Learning" and carry out activities in both the views.

2.1.1 Product Line Infrastructure View

Product Line Infrastructure View involves the activities related to conceptualization and initiation of software product line within an organization. This view performs activities that establish an infrastructure for a software product line. The *Product Line Infrastructure View* constantly provides feedback to *E-Learning Analysis View* for effective definition, identification, evaluation, selection and catalog/storage of E-learning content management. Software product line scope definition activity iteratively provides feedback to E-learning contents' definition and identification activity in *E-Learning Analysis View*. This way it ensures that all the material for content management are consistent with the scope of product line. Product line requirements deal with features or functionalities common to all the products belonging to that family. The requirement engineering for product line gives feedback to analyzing learning object activity in the *E-Learning Analysis View* to generate a candidate list of learning object's presentation that meets the product line requirements. The identification of business cases helps in evaluating identified learning objects in *E-Learning Analysis View* in order to meet the production criteria and the product requirements.

2.1.2 E-Learning Analysis View

E-Learning Analysis View is responsible for building up a core assets repository for the e-learning based software product line and provides base line information for the software product line architecture by specifically dealing with commonality and variability management. It communicates with *Product Line Infrastructure View* to generate content management. Initially the *E-Learning Analysis Engineer* identifies potential contents from the software product line requirements and scope. The definition and identification process yields a number of potential contents that can be used in the development of various products in a software product line. Those contents need to be evaluated at the individual level as well as at the product line level before they are selected for use in a software product line development. The selected contents are cataloged and stored in the repository with enough information so that they can be easily traced and retrieved as and when required for assembly. The *E-Learning Analysis View* uses support methodology to analyze, evaluate and select the contents. In this paper we are using the ED^2 Model [24] [25] for analysis, evaluation and selection of the contents discussed in detail in later part of this paper. The support methodology to analyze, evaluate and selection provides foundations for the commonality and variability selections in the software product line architecture.

2.2 Application Engineering

In the *Application Engineering* phase of the proposed methodology shown in Figure 2, actual products are developed using software product line architecture and core assets repository. In this phase, activities of the *Product Line Application View* interact with the activities of the *Core Assets Development View* to produce required products. The application engineering phase provides feedback to domain engineering phase through business case evaluation to accommodate changing needs of the product line.

2.2.1 Product Line Application View

Product Line Application View interacts with *Product Line Infrastructure View* to identify potential business cases to capture market segment. The *Product Line Application View* generates the product requirements of the potential business case and provides feedback to *E-Learning Analysis View* to find out contents to be used in the product development. Product requirements are composed of a constant and a variable part. The constant part comes from product line requirements in the *Product Line Infrastructure View* and deals with features common to all the products belonging to the family. The variable part represents those functionalities that can be changed to differentiate one product from another. This activity defines the variable part of the product requirement. The assembly activity involves the development of product. The product requirements guide the assembly process to get feedback from the query activity of *Core Assets Development View* to find out those potential contents suitable to be assembled in order to produce the product. In product testing and evaluation, products developed from software product line are tested to analyze whether they meet the product line testing and evaluation criteria or not. Business case evaluation compares the proposed business case strategy with the outcome of the development and deployment process of products.

2.2.2 Core Assets Development View

Core Assets Development View is responsible for providing required components from core assets repository for developing products. *Core Assets Development View* interacts with *Product Line Application View* to receive product. In the query activity of the *Core Assets Development View*, components are searched from the core assets repository in order to develop the product. A well-catalogue core assets repository reduces the efforts to trace the suitable components for assembly. The product requirements serve as an input to the query activity, and continuously traversing core assets repository yields the required components, exactly matched, partially matched or not matched. The components, after adaptation, generate versions, which are documented in this activity. A comprehensive version management and dependency link strategy for components and products in the software product line engineering provides us with vital information about components and products having a relationship of composition and utilization. A software product line develops an initial core assets repository in the *Domain Engineering* phase. As a product line gets matured in its lifecycle, new core assets or even new versions of existing core assets are produced, which must be added to the core assets repository so that they can be reused in later products. The core assets repository is dynamic and continues increasing its size with the addition of new core assets.

2.3 Software Product Line Architecture

The proposed model emphasizes the importance of developing a software product line architecture based on e-learning product. The junction of *Domain Engineering phase* and *Application Engineering phase* produces a suitable product line architecture based on existing e-learning components. Overall *Software Product Line Architecture* can be produced in three different ways; it can be developed from scratch, it can be based on the existing product group, or it can be built based on a single existing product [7]. The proposed methodology emphasizes the approach of developing *Software Product Line Architecture* based on a single existing product. The junction of *Domain Engineering* phase and *Application Engineering* phase produces the *Software Product Line Architecture* based on the first product developed. The *Domain Engineering* phase provides product line requirements. The *Application Engineering* phase accommodates those requirements along with product specific requirements to establish the *Software Product Line Architecture*. The *Software Product Line Architecture* reflects the commonalities among the products and variation points where products differ from each other. All the resulting products from the product line share this common architecture. The iterative approach of methodology refines the *Software Product Line Architecture* after successive development of products.

3. The ED^2 Model: E-Learning Analysis View Support Activity

The ED^2 model for analyzing learning objects is presented in [24] [25]. The ED^2 model presents a comprehensive framework for thinking about variability and change in any learning object. The model has two dimensions. The first dimension views variability in terms of pedagogy, technology and domain of learning. The second dimension looks at variability from the perspective of architectural layers of a learning object. The first dimension of the ED^2 model deals with the three dimensions of pedagogy, technology and the domain. The first source of variability is the domain of learning where different types and levels of knowledge may be included in a learning object. Similarly, the second source of variability is the technological basis of a learning object; one may decide to use an HTML-based learning object as opposed to a Flash-based one, for example. Finally, the pedagogy underlying the learning object can vary from tell-and-test to a socio-constructivist framework emphasizing the social nature of learning. In many cases, product families are established on top of a successful pilot product [26].

The second dimension of the ED^2 model deals with layers of change within a learning object. This dimension was originally conceived to think about layers of change in the field of architecture and later applied to learning objects [27] [28]. Brand [27] contends that a building can be viewed to consist of multiple layers that slide along each other. Each layer is designed to vary on a different time-scale. The layer that varies at the slowest speed is called the *Site.* This layer represents the physical location of a building and is consequently the most stable. The next level of variability is represented by the *Structure* of a building. The Structure may consist of the walls and roof of the building. In buildings, structures can last from 30 to 300 years and are hence less stable. *Skin* layer represents the exterior of a building and typically changes over a period of 20 years or so. The infrastructure inside a building represents the *Services* layers. This may include lighting, air-conditioning and plumbing etc. Services typically change over a life-span of decades. The *Space Plan* layer consists of the internal walls and the layout of the building. This layer can be changed every few years or so. Finally, the *Stuff* layer represents what is inside a building. For example, furniture represents one type of stuff. Stuff can be changed on a weekly or monthly basis. For a learning object, ED^2 combines the two dimensions as described below.

3.1 Site

Technologically, *Site* represents a choice of the lowest virtual machine being used to deploy a learning object. For example, one could choose Windows operating system, Java virtual machine or Adobe Flash virtual machine as the *Site*. Pedagogically, *Site* represents choosing an epistemological orientation towards learning. For example, one could choose an instructivist or constructivist pedagogy. From the domain of learning, *Site* represents choosing what constitute fundamental and immutable principles of a domain; homeopathic verses allopathic, for example.

3.2 Structure

Structure from a technology perspective, involves choosing learning design architecture like SCORM [29] or IMS-LD [30]. Pedagogically, the choice is about the nature of learning design; problem-based learning verses informal learning, for example. From a domain perspective, the choice is about embedding a particular domain ontology that emphasizes only particular views of a domain [31].

3.3 Services

Choices in technology-oriented services in a learning object may include authentication, login, tracking, archiving, and book-marking. Choice in pedagogical services, on the other hand, is represented by a level of understanding being delivered by a learning object. For example, one may use Bloom's taxonomy [32] to specify that a learning object delivers a "recall" level of understanding as opposed to "analysis." Domain services provide choices in learning objectives; what is to be learned in a learning object, for example.

3.4 Space Plan

Technologically, *Space Plan* of a learning object represents choices of personalization and customization. For example, the linear sequencing in SCORM [29] supports a flexible *Space Plan*. Pedagogically, the *Space Plan* represents a choice on the degree of adaptive-ness of the learning design; does the learning object use learning styles and preferences to determine learning paths for individual learners? Domain-wise, a *Space Plan* represents choices regarding multiple types (or roles) of learners; is the learning object designed for a technician, an engineer or both?

3.5 Skin

Technologically, *Skin* of a learning object represents choices in user-interface technologies; HTML, DHTML, XUP or SVG, for example. Pedagogically, *Skin* represents the choice of a learning template. For example, Cisco's methodology [21] specifies that teaching a concept requires a definition, an example and a non-example. Domain-wise, *Skin* represents choices in presentation styles and colors; each domain has its own presentation norms.

3.6 Stuff

Technologically, *Stuff* represents choice or selection of specific *assets* [29]; text, PDF files, Flash movies and Java scripts, for example. Pedagogically, *Stuff* represents choices about including specific instances of pedagogical primitives like objectives, topics, lessons, assessments, games, simulations, and activities. Domain-wise, *Stuff* represents choices in which of the specific definitions, concepts, processes, and principles to include in a specific learning object.

4. Application of the Methodology: A Case Study

We applied the proposed methodology software product line development to creating learning objects for the Six Sigma quality methodology [33]. As a part of implementing Six Sigma within an organization, the Six Sigma training needs to be delivered at various levels for various stakeholders. The level of training can vary from yellow-belt training to black-belt or master black-belt training. Intermediate levels of green belt and orange-belt have also been introduced. In addition to various levels of training, the Six sigma methodology has various variants like DMAIC (Define, Measure, Analyze, Implement and Control) or DFSS (Design for Six Sigma), for example. These variants can be further customized; some companies may choose to combine the Define step with Measure to just include MAIC. Another source of variability is the diverse set of conceptual and statistical tools that can be optionally used to implement this methodology; for example [34] lists about one hundred tools. The high variability in various aspects of a Six Sigma course makes it a good candidate for a SPL. A business cases for this product line is clearly justified by the need for various specialized versions of yellow belt, green-belt and black-belt courses in Six Sigma.

In the Domain Engineering we perform the activities of the *Product Line Infrastructure View* and *E-Learning Analysis View* to establish product line architecture. This can be achieved by highlighting the commonality and variability features in the products. The scope definition of product line yields that a six sigma course learning system that supports various courses and can be accessible on the WWW for delivery, simulation, discussion, and examination purposes. In order to carry out the task of product line requirements engineering based on the business case of "six sigma course product line" the activities in the *E-Learning Analysis View* starts analyzing the contents and the way the contents are presentable. The Analyzing Learning Object activities in the *E-Learning Analysis View* help in establishing the Figure 3. Figure 3 shows a simplified feature model the six sigma course learning object in terms of ED^2 model. The first variability is at the top level which is the training levels of six sigma courses of yellow, black, orange and green levels. The commonality exists at the six level of ED^2 model which are Site, Services, Structure, Space plan, Skin and Stuff. In other words, any Six Sigma course must include (and choose) aspects from each of these six. Commonality management in software product line architecture deals with all product aspects that are common across all the various ED^2's categories. The structure of ED^2 dictates that commonality should be high (less variability) at the lower layers (like the Site) and low (high variability) in the higher layers (like the Stuff). Commonality analysis ensures that the selections made for each layer of commonality according to the ED^2 model are mutually consistent. For example, since a pure Flash framework was fixed for the Site, it is consistent with the Stuff being constrained to use common Flash buttons; a use of Windows buttons for the Stuff, would obviously be inconsistent. Detailed example of commonality among successive products of six sigma learning objects are shown in Table 1 which is a result of the product line requirements engineering in the *Product Line Infrastructure View*.

The product requirement engineering activity in *Product Line Application View* provides information about as opposed to commonality; variability explicitly models what can be changed. It highlights in the core architecture where we can introduce changes to come up with new products based on the business case. Again, ED^2 provides a structured approach to identifying variability. The Figure 2 clearly highlights variations in features in terms of ED^2 model. For example, in case of Structure some products may use SCORM and others may use LD. Whereas in case of Site, some product may use HTML, DHTML, or XUP.

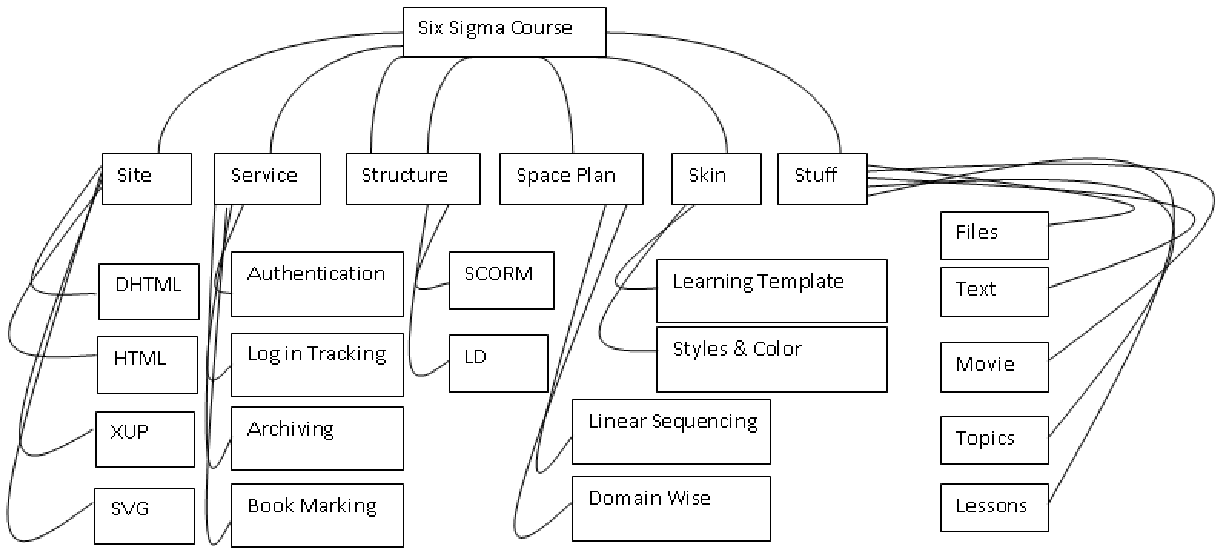


Figure 3: Feature Model of Six Sigma Course Product

Like commonality analysis, variability analysis also needs to ensure that there are no contradictions. For example, the variability in the *Services* layer (no audio) is consistent with the Flash engine's capabilities (to skip audio for a Tab) at the *Site* layer. Some more examples of variability analysis are shown in Table 2.

During *Application Engineering* phase product s are developed using the product line architecture. The Figure 4 and 5 show screen shots for a yellow-belt and orange-belt modules as a part of two successive products that are developed from the Six Sigma courses software product line. These two products consist of sets of Adobe Flash modules that constitute each course. The commonality is present in all the six factors of ED^2 model including Site, Structure, Services, Space plan, Skin and Stuff. Both courses use the same Adobe Flash Engine that interacts with a SCORM-based Learning Management System using and AICC interface to support authentication, tracking and book-marking services. Both products have the same look and feel in term of the various buttons and menus on the screen. Both products support pages and sub-pages and an audio interface. The variability arises in the depth of the presentation content. As Figure 3 shows, the yellow-belt course is mostly descriptive and introduced Six Sigma at a basic level. The orange-belt course, on the other hand, goes into the details of technical analyses like gage analysis. There is variability from a presentation perspective as well. For example, in Figures 4 and 5, the yellow belt example shows a simple table while, in the orange-belt example, the three graphs can be selected one at a time by the user by clicking on it. There is also wide variability in the two courses with respect to the pedagogy. While the yellow-belt course relies primarily on tell and test (knowledge level, in terms of Bloom's taxonomy), the orange-belt course uses scenario-based learning as well. For example, a culturally relevant example of how to select a mobile phone is used where the user can choose a particular mobile phone and try to understand the reasons behind their selection by using a factorial design approach.

Table 1: Commonality Identification Examples

Dimension	Commonality Identification
Site	Use the Adobe Flash-based engine for all products. This engine defines a common virtual layer for all the products. The engine needs to support interface to a Learning Management System. In addition, the product will use a traditional teacher-centric pedagogy.
Structure	All products will contain modules where each module will present objectives, followed by a mix of information and assessment items, followed by a conclusion. Each product will support multiple learning styles by including images, text and audio. Each product will be structured to include sub-pages in the form of Tabs. Traditional Six Sigma methodology will be used (excluding Lean concepts).

Services	All products will have login, book-marking and audio services. In addition, all products will have a help facility.
Space Plan	All modules will have pages and each page will have tabs for sub-items within a page.
Skin	All products will carry the same broiler plate with the company's logo. All products will have a forward-backwards button, an audio panel including on/off, pause and repeat buttons, a drop-down menu to access various pages of the product and a panel showing where the learner is (sub-page number) with respect to the complete module.
Stuff	All products will use common Flash buttons for forward and backwards, drop-down menus and Tab objects.

Table 2: Variability Identification Examples

Dimension	Variability Identification
Site	Arbitrary Flash movies can be embedded within the generalized Adobe Flash engine. The pedagogy can be changed to emphasize problem-based learning in a social context.
Structure	The content inside each Tab in a product can be an arbitrary image, text or another Flash movie. The depth of content presentation can be varied for yellow to black-belt.
Services	Audio can be skipped for some tabs. Similarly, tracking is optional. The learning objectives are different for yellow or black-belt.
Space Plan	The placement of text, images, and movies within a Tab can be varied. In other words, different configuration of text, images and movies can be used. The order of modules and sub-pages to include will depend on whether yellow or black-belt training is being delivered.
Skin	Colors and look & feel can be changed.
Stuff	All the content including text, images and specific audio is variable. The specific stuff to include will depend on which of the various levels of training is being imparted.

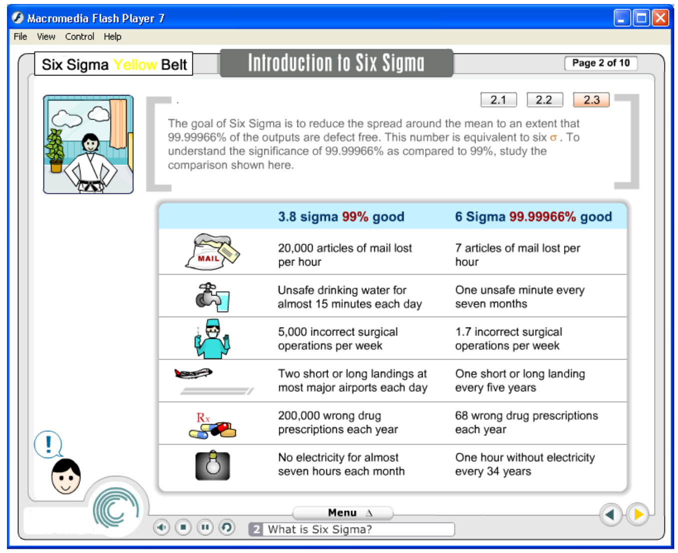


Figure 4: Screen shot of one module in a yellow-belt course

Source: w.knowledgeplatform.com

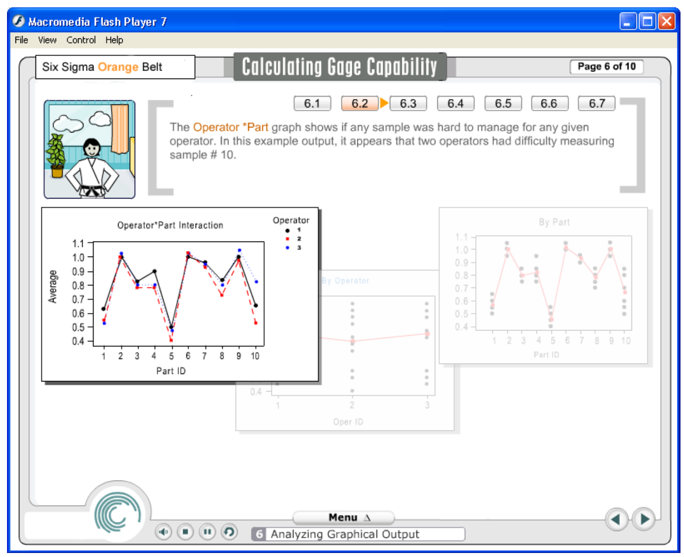


Figure 5: Screen shot of one module in a black-belt Course

Source: www.knowledgeplatform.com

5. Conclusion

Design and development of digital learning objects is gaining popularity due to significantly large increase in online-learning. Software products line engineering curtails the development time and further avoids reinventing the wheel in software development. The objective of this study was to investigate the use of product line approach in developing digital learning objects and put forward a methodology. The proposed methodology highlights various activities of software product line and E-Learning system development. The model integrates the concept of software product line and learning object to come up with a prescribed way of establishing a software product line for E-Learning system capable of producing multiple products within an application domain. In order to validate the model, we developed a software product line for six sigma application domain, which reveals that productivity in terms of cost, time and quality would be increased if we follow the proposed methodology. Additionally, the methodology provides an efficient way of integrating the approaches of software product line and E-learning system development process.

References

[1] Clement P. and Northrop L., Software Product Lines: Practices and Pattern, Addison Wesley, 2001.

[2] M.L., Implementing Product-Line Features with Component Reuse, Proceedings of 6th International Conference on Software Reuse: Advances in Software Reusability, pp. 137-152, Lecture Notes in Computer Science, Springer Verlag, 2000.

[3] Griss D. M. Weiss and Lai C. T., Software Product-Line Engineering: A Family-Based Software Development Approach, Addison-Wesley, 1999.

[4] Griss M.L., Product-Line Architectures, G. T. Heineman and W. L Councill (Eds.) Component-Based Software Engineering, pp. 405-419, Addison-Wesley, 2001.

[5] Linden F. van der, Software Product Families in Europe: The Esaps & Café Projects,

IEEE Software, Vol. 19, No. 4, pp. 41-49, 2002.

[6] Ommering R.V. (2000) Beyond Product Families: Building a Product Population, Proceedings of the Conference on Software Architectures for Product Families, Lecture Notes in Computer Science, Springer - Verlag, pp.187-198.

[7] Buckle G.; Clements P.; McGregor J.D.; Muthig D. and Schmid K., Calculating ROI for Software Product Lines, IEEE Software, Vol. 21, No. 3, pp. 23-31, 2004.

[8] P. America, H. Obbink, J. Muller, and R. van Ommering, "COPA: A Component-Oriented Platform Architecting Method for Families of Software Intensive Electronic Products,". Denver, Colorado: The First Conference on Software Product Line Engineering, 2000.

[9] David M. Weiss and Chi Tau Robert Lai. Software Product-Line Engineering: A Family-Based Software Development Process. Addison-Wesley, 1999.

[10] K. C. Kang, S. Kim, J. Lee, K. Kim, E. Shin, and M. Huh, "FORM: A Feature-Oriented Reuse Method with Domain- Specific Reference Architectures," Annals of Software Engineering, vol. 5, 1998, pp. 143 - 168.

[11] Atkinson, C., Bayer, J., Muthig, D., Component-Based Product Line Development: The KobrA Approach, Proceedings, 1st International Software Product Line Conference,2000, pp.289-309.

[12] Bayer, J.; Flege, O.; Knauber, P.; Laqua, R.; Muthig, D.; Schmid, K.; Widen, T.; and DeBaud, M.;, (1999) PuLSE: A Methodology to Develop Software Product Lines, Proceedigns of the 5th ACM SIGSOFT Symposium on Software Reusability, pp. 122-131.

[13] Clements, P. C. (2001). "On the importance of product line scope," in: Proceedings of the 4th International Workshop on Software Product Family Engineering, pp. 69-77.

[14] Clements, P. C., Jones, L. G. Northrop, L. M. & McGregor, J. D. (2005). "Project management in a software product line organization," IEEE Software, vol. 22, no. 5, pp. 54-62.

[15] Garlan, D. and Perry, D. (1995). "Introduction to the special issue on software architecture," IEEE Transactions on Software Engineering, vol. 21, no. 4, pp. 269-274.

[16] Pronk, B. J., (200) "An interface-based platform approach," in: Proceedings of the 1st Software Product Lines Conference, pp. 331-352.

[17] Jazayeri, M., Ran, A. and Van der Linden, F. (2000). Software Architecture for Product Families: Principles and Practice, Addison Wesley.

[18] Mika, K. and Tommi, M. (2004). "Assessing systems adaptability to a product family," Journal of Systems Architecture, pp. 383-392.

[19] Wiley, D., Ed. (2000). The Instructional Use of Learning Objects, (Available online at http://www.reusability.org/read/) [Accessed November 14, 2008]

[20] McGreal, R., Ed., 2004, Online Education Using Learning Objects. Open and Distance Learning Series, Routledge/Falmer, London.

[21] Cisco. (2003). Reusable Learning Objects Authoring Guidelines: How to Build Modules, Lessons and Topics, Cisco Systems, Inc., San Jose, California.

[22] F.v.d. Linden, K. Schmid, and E. Rommes, Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering. 2007: Springer.

[23] J. Bosch, Design & Use of Software Architectures: Adopting and evolving a product-line approach. 2000: Addison-Wesley.

[24] Zualkernan, I. A. (2006). "Stability Analysis of Learning Objects in Engineering Education," in Proceedings of the 2nd International Conference on Engineering Education, Kuwait.

[25] Zualkernan, I. A. (2008). Eighteen Design Decisions, Computer Science and Engineering Department, Technical Report # TR-02-08, American University of Sharjah.

[26] M. Korhonen and T. Mikkonen, Assessing Systems Adaptability to a Product Family, Journal of Systems Architecture, No. 50, pp. 383-392, 2004.

[27] Brand, S. (1994). How Buildings Learn: What Happens After They're Built. New York: Penguin Books.

[28] Gibbons, S., Nelson, J. and Richards, R., 2000, "The nature and origin of instructional objects," In D. A. Wiley (Ed.), The Instructional Use of Learning Objects, (Available online at http://www.reusability.org/read/) [Accessed November 14, 2006]

[29] IMS-SCORM, 2005, IMS Content Packaging Use Case Descriptions, Version 1.2, (Available at http://www.imsglobal.org/content/packaging/cpv1p2pd/imscp_usecv1p2pd.html) [Accessed November 14, 2008]

[30] IMS-LD, 2008, IMS Learning Design Specification, (Available at http://www.imsglobal.org/learningdesign/) [Accessed November 14, 2010].

[31] Gomez-Perez A., Fernandez-Lopez, M. and Corcho, O. (2004). Ontological Engineering, Springer, London.

[32] Bloom, B. S., Ed. (1956). Taxonomy of Educational Objectives: Book 1, Cognitive domain. New York: Longman.

[33] Pyzdek, T. (2003). The Six Sigma Handbook, McGraw-Hill, New York.

[34] George, M. L., Rowlands, D., Price, M. and Mazey, J. (2005). The Lean Six Sigma Pocket ToolBook, McGraw-Hill.

A Unified Network Security and Fine-Grained Database Access Control Model

Amit Kuraria M.E. (computer science) IVth semester Shri Ram Institute of Technology,Jabalpur(M.P.),INDIA

Prof.Vikram Jain, Shri ram Institute of technology ,Jabalpur(M.P),INDIA (Mtech. VIT,vellore.Thesis Co-guide)
Prof. Sweta Modi, Shri ram Institute of technology ,Jabalpur(M.P) ,INDIA (Mtech ,IIT Roorkee,Thesis Guide)

Abstract—With the development of Internet and Intranet, Web and distributed databases have been used more and more widely. It is important to properly handle network and Web database security issues including authentication, denial of service, and fine-grained access control. When database access control and the network security are addressed separately, the security systems are not optimized sufficiently as a whole. This paper proposes a method of integrating network security with criterion based access control to handle network security and the fine-grained Web database access control simultaneously. To improve efficiency, the model adopts two step access controls. The first preliminary access control is combined with the firewall function, and the second fine-grained access decisions are determined by the user's digital credentials as well as other factors such as his/her IP address.

Keywords: *Network Security, fine-grained access control, Web and distributed database, criterion-based access control*

1. Introduction

Web applications become wide-spread and more and more companies take the advantage of them to increase their revenues. Web and distributed databases play the key role in most of these Web applications and thus it is critical to protect them from unauthorized access and malicious attacks. Web and distributed database security has aroused many researchers interests. Because of the high accessibility, Web and distributed databases tend to be more vulnerable and expose to various attacks from wide variety of sources. To address this issue, a more efficient and flexible security mechanism is required to systematically authenticate users, control network traffic, and provide efficient fine-grained access control. Web and distributed databases need a strong authentication system. In the Internet environment, the possibilities of impersonation increase. The identity of a remote user must be verified based on his/her IP address, password, and credentials to combat the repudiation attack. Since denying the requests at early stage can significantly increase the efficiency of the network, the required firewall system should not only filter the network traffic but also provide the preliminary access control. Because a remote user's permissions depends on his/her credentials as well as other factors including his/her IP address (location), the security mechanism should further refine the user's permissions based on all of these factors. The security mechanism should also provide the fine-grained access control based on these refined permissions. Currently, network security and database security are often addressed separately and therefore the security system is not optimized properly as a whole. Besides, the computational cost of fine-grained access control is high if the access control system directly implements the organization's security policy. This is because different organizations usually have different security policy and there are big semantic gaps between security policy and implementation. To improve the efficiency of the fine-grained access control, a criterion-based multilevel database access control approach has been proposed. The approach transforms security policy into security criterion expressions and security criterion subsets upon which the fine-grained access control is achieved. Although this approach is not specifically designed to address the security requirements of Web and distributed database, it can actually be applied to these situations and combined with network security mechanisms.

This paper presents how to apply the criterion-based access control to Web and distributed database, and explores the method of developing a unified network and database security system. The rest of the paper is organized as follows. Section 2 briefly overviews the previous works of network security and fine-grained database access control. In section 3, the criterion based access control is reviewed. Many important concepts of this model are discussed. Section 4 presents a unified network and fine-grained Web and distributed database security model. Section 5 concludes the paper.

2. Overview of Network Security and Fine-Grained Database Access Control

Computer network security has been intensively studied for several decades. The first step of network security is to authenticate users. The authentication can be done based on one or more factors such as what you know (e.g. password), what you have (e.g. smart card), and what you are (e.g. fingerprint). After authentication, a firewall enforces access policies such as what services are allowed to be accessed by the network users. The types of firewall include packet filter, application gateway, circuit-level gateway, and proxy server. The third technique for the network security is the intrusion detection system which monitors the network and detects and stops the unexpected traffics and abnormal behaviors. To protect the information in transition, it can be encrypted by public or private key encryption.

Multilevel database security has also attracted a lot of attention. C. Pfleeger and S. Pfleeger presented many important multilevel database security solutions, including partitioning model, encryption, integrity and sensitivity locks, trusted front-end, and view etc. Partitioning model divides the database into separate databases according to the security level. Each of the divided database stays at a specific security level. However, this solution is against the basic database principle of elimination redundancy. Encryption model uses a unique key to encrypt data of each security level. The problem with this solution is the high overhead when processing a query because data need to be decrypted first. In the proposal of integrity and sensitivity lock, each data item has a lock which is the combination of a unique identifier and unforgettable, unique, and concealed label. The lock is used to handle access control. The disadvantages of this solution include inefficiency, high storage cost, and Trojan horse attack. Trusted front-end solution adds a trusted front-end between users and DBMS. A disadvantage of this method is the complexity of the front-end system and the separation of the database. The view solution uses a view to represent and filter a user's subset of database. The drawback of this method is the complexity of creating and maintaining the views. There are also many other solutions. L. Null etc. proposed a method of combining trusted filter and an inference engine T. Didriksen presented a rule based database access control. He partitioned a database into fragments and extended SQL to specify data fragmentation and access control. He also adopted "meta" table to hold the security policy.

In our criterion-based method, the security policy is transformed into security criterion expressions without partitioning the database or introducing the "meta" table. Meanwhile, users' security attributes are specified by the security criterion subsets. The fine-grained access control is achieved by evaluating security criterion expressions with user's security criterion subset. This method is further explained in the next section.

3. The Criterion-Based Access Control

A. Basic Concepts

The criterion-based access control approach was first proposed to integrate with role-based access control model to deal with multilayer security of multimedia applications .The approach also works well independently for fine-grained database access control. In this approach, security criteria, security criterion expressions, and security criterion subsets are introduced. Security criterion expressions and security criterion subsets serve as locks and keys, respectively. Each object or sub object is embedded into a lock and each user (subject) is assigned a set of keys. The user's keys are used to actuate the locks and the state of the locks determines whether the user has access to an object or sub object. A security criterion is a criterion used to both specify the user's security attributes and define the object's (and the sub object's) security attributes. Each security criterion is represented by a symbol s_i . Security criteria are abstracted from authorization rules. An authorization rule specifies who is authorized to do what. For example, an authorization rule may specify that a junior bank teller do not have access to customers' mortgage information. From this authorization rule, we introduce a security criterion s_3 to indicate both users of junior bank teller and objects (sub objects) of mortgage information. From the whole set of authorization rules, a set of security criteria s_1 , s_2 ,..., s_n in an application domain can be abstracted. The collection of all security criteria, their complement counterparts (s_j), constant false F and true T form a set which is called the security criterion set, and is denoted as SCS, that is, SCS={F, T, s_1 , s_2 ,..., s_n , s_1 , s_2 ,..., s_n}. A user may have more than one security attributes. So, several security criteria are often required to specify the user's security attributes. The set composed of these security criteria is called a security criterion subset (SCSS). When a user is associated with a security criterion subset (SCSS), he/she is enhanced to be a secure user (SU). For example, a secure user's security criterion subset is { s_1 , s_2 , s_3 }, where s_1 , s_2 , s_3 represent "employee," "not a manager", and "a junior bank teller," respectively. The special null security criterion subset indicates that there are no authorization rules confining the user accessing any part of the objects. To precisely reflect authorization rules, objects (and the sub objects), as well as their security attributes, are defined by security criterion expressions. A security criterion expression (SCE) is a Boolean expression in terms of security criteria. A Boolean expression is considered to be a security criterion expression if it reflects one or more authorization rules. Following are legal security criterion expressions:

(1) A constant true, T, or false, F

(2) A Boolean expression derived directly from an authorization rule

(3) Logical "OR" of (1) and (2)

The constant true, T, or false, F, represents special cases. When an (sub) object needs unconditional protection, its corresponding security criterion expression should be the constant true, T (reserving T for completeness in theory). On the other hand, when the security criterion expression is the constant false, F, the related (sub) object is accessible in any circumstances. To support fine-grained multilevel access, in an object, each part (i.e. sub object) with different security attributes and thus of different security levels has an embedded security criterion expression to specify its security attributes. A sub object with an embedded security criterion expression is a secure sub object. In a relational database, the objects that need to be protected include tables, views, logs and so forth. Because views and logs can be considered as special tables, to simplify the discussion, we confine the objects to tables. A sub object can be a cell, a row or a column in a table. If the whole database is regarded as an object, the table can also be treated as a sub object.

Example 1 presents a secure object: In a bank system, some information is sensitive and inaccessible to some employees according to the security policy. The sensitivity needs to be specified. To specify the security attributes and the security level of the data in a table, a special row and a column are added to hold corresponding security criterion expressions of each row and column. Non-sensitive information corresponds with a special security criterion expression, constant F, and sensitive information corresponds with more complex security criterion expressions abstracted from authorization rules. Table 1 shows the secure object (table of customer records). Its first row and last column are used to contain corresponding security criterion expressions. In the first row, a security criterion expression in certain column (e.g. Column 6) specifies the security attributes and security level of the data in that column (e.g. Mortgage). In the same way, the security criterion expression in certain row (e.g. the fourth row) and last column specifies the security attributes and security level of the data in that row (e.g. William Wilson). The security attributes and security level of the cell (4, 6) (e.g. 40,000) is the logical "OR" of these two security criterion expressions (e.g. ($s_1 \wedge s_2) \vee (s_1 \wedge s_3)$).

B. *Security criterion abstraction and secure object and secure user generations*

A systematic method has been developed to abstract security criteria from authorization rules, to transform authorization rules into security criterion expressions, and to generate security criterion subset based on the authorization rules . To save space, only the summary of the major steps is presented here. For details, please reference .

Step 1. Transform conditions required to specify users' security attributes into security criteria from authorization rules.

The basic idea is that only those conditions required to specify the users' security attributes are abstracted. One security criterion is abstracted to represent one specific condition. For example, two security criteria s_1 , s_3 are abstracted to specify "employee" and "junior bank teller" from the authorization rule "junior bank teller do not have access to the mortgage information."

Step 2. Creating secure object

Step 2.1 Using security criterion expressions to represent the authorization rule(s). Each authorization rule can be expressed by a security criterion expression. For example, the security criterion expression corresponding to the above authorization rule is $s_1 \wedge s_3$, which means "employees of junior bank teller don't have access to the Mortgage information." If more than one authorization rules are relevant to an (sub) object, the logical "OR" of the security criterion expressions with respect to different authorization rules is the final security criterion expression for that (sub) object.

Step 2.2 upgrade the object The table (object) is extended to insert a row and a column to which the relevant security criterion expressions are added (see table 1).

Step 3. Secure user generation The security criterion subset associated with a user is generated according to the user's security attributes. In most cases, the security criterion subset can be generated from the authorization rules directly (see step 1). However, the security subset may have one or more concealed security criteria which must be identified by analyzing the relationship among the authorization rules .The security criterion subset for the above authorization rule should be { s_1 , s_2 , s_3 } rather than { s_1 , s_3 } , where s_2 represents "not a manager.".

TABLE I. A SECURE OBJECT OF CUSTOMER TABLE

F	F		F	F	$s_1 \wedge s_3$	$(s_1 \wedge s_2) \vee (s_1 \wedge s_3)$, $s_1 \wedge s_3$, $s_1 \wedge s_2$
Name	Address	…	Check account	Investment	Mortgage	F
Shawn Smith	1234 Ontario Street, Nile, IL	…	102.68	15,000	81,000	F
William Wilson	568 35th street, Nile IL	…	1,218	40,000	123,000	$s_1 \wedge s_2$
…	…	…	…	…	…	…
Jim Johnson	5969 College Ave. Nile, IL	…	250.25	32,000	54,000	F
Hannah Howard	7785 Beach St. Nile, IL	…	89.35	0	68,000	$s_1 \wedge s_2$
Wendy White	332 Fisher St. Nile, IL	…	3,022	100,000	228,000	F

C. Achieving fine-grained access control

In the Criterion-Based Access Control model, an (sub) object's security attributes and security level are implied by indicating users who do not have access rather than explicitly defining them. The system becomes simpler because one mechanism is used to define both the user's security attributes and the (sub) object's security attributes.

The security criterion expressions embedded in a secure object can be regarded as locks, while the security criteria in the security criterion subset can be considered as keys. When a secure user accesses a secure object, he/she uses the available keys to actuate the locks. Whether the secure user is allowed to access the secure sub object (the cell, column, or row) depends on the state of the corresponding locks.

A security criterion expression is evaluated in the following two steps. First, substitute all the security criteria in the security criterion expression with true, T, or false, F, according to the following rules: all the security criteria in a security criterion expression that also appear in the secure user's security criterion subset

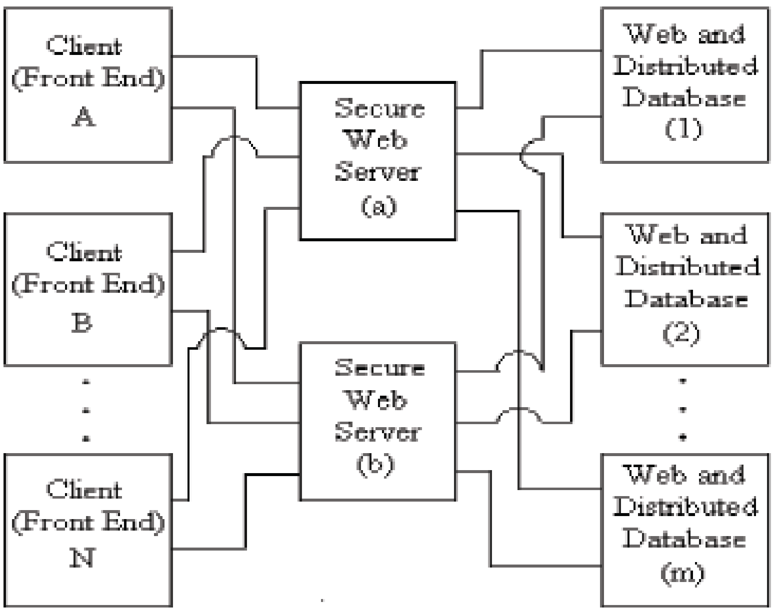


Figure 1. Logical structure of the unified network and database security system

have the value true, T, while other security criteria have the value false, F. Second, the security criterion expression is evaluated according to the normal evaluation procedure in Boolean algebra. The evaluation value T of a security criterion expression implies that users with security attributes specified by these security criteria are not allowed to access the corresponding secure sub object, according to the sub object's security criterion expression transformed from the authorization rules. On the contrary, a false evaluation value, F, of the security criterion expression implies that the security criterion expression (actually the authorization rules) does not prevent these secure users from accessing this sub object.

The fine-grained access is achieved as following. If the evaluation value of a security criterion expression in a column (row) is true, T, the column (row) is not accessible to the user. Therefore, the column (row) is filtered and not be sent to the user. To improve the efficiency of the system, we usually first evaluate three special security criterion expressions stored in the cell of first row and the last column . If the evaluation value of the first one is false F, the whole table is accessible, and the rest of the security criterion expressions need not be evaluated. This is because the first security criterion expression is the logical "OR" of all the security criterion expressions in different columns and rows. When the evaluation value of this security criterion expression is false F, the evaluation value for every of its term must also be false F. On the other hand, if the evaluation value of the first one is true T, there must be at least one security criterion expression in a column or a row whose evaluation value is true T. therefore, the rest of the security criterion expressions should be evaluated. We can evaluate the second security criterion expression to see if any column is inaccessible. If the evaluation value of this one is true T, we need to evaluate the security criterion expressions in different columns one by one to find out all the columns that are inaccessible. In the same way, we can find out the inaccessible rows.

Example 2: Suppose a junior bank teller requests to access the table 1. As discussed above, the user's security criterion subset is { s_1 , s_2 , s_3 }.The rows and columns with security criterion expressions of constant false F are by default accessible and need not be evaluated. When the security criterion subset is used to evaluate the other security criterion expressions, the evaluation values are true T. Therefore, the corresponding column (mortgage) and rows (records of William Wilson and Hannah Howard) are inaccessible. By filtering these inaccessible sub objects, the user gets the following table 2.

TABLE II ACCESSIBLE COLUMNS AND ROWS FOR A BANK TELLER

Name	Address	…	Check account	Investment
Shawn Smith	1234 Ontario Street, Nile, IL	…	102.68	15,000
…	…	…	…	…
Jim Johnson	5969 College Ave. Nile, IL	…	250.25	32,000
Wendy White	332 Fisher St. Nile, IL	…	3,022	100,000

4. A Unified Network and Fine-Grained Web and Distributed Database Security Model

The proposed model includes three tires: client tire, secure Web server tire, and Web and distributed database tire. The client tire is the front end of the system on which client software such as browser and c applications are run. Remote users scattered in different locations send their requests from the client tire. The Web and distributed database tire is on the other end. All of the databases have been upgraded by inserting a row and a column and embedding security criterion expressions derived from the authorization rules (security policy). The secure Web server tire sits between the client tire and the Web and distributed database tire. The secure Web server tire provides the function of

Web server and security services. Figure 1 shows the logical structure of the model. The redundant connections between the tires increase the ability of fault tolerance and resisting the denial-of-service attack.

Discussing the Web server function is out of the scope of this paper. The following discussion focuses on the security service function of the secure Web server. When the user's request is received, the first job a secure Web server does is to authenticate the user. The authentication is performed based on multiple factors including user name, password, and digital credential. Once the user is authenticated successfully, the user's request, IP address, and digital credential are forward to the preliminary access control system. The preliminary access control can be achieved with an upgraded firewall system, which filters the traffic based on the user's request, IP address, and digital credential. For example, although a bank teller (possesses the credential of bank teller) has access to the customer saving information in the local databases, he/she does not have the permission to access the similar information located in the remote databases.

This indicates that user's permissions to a database vary when he/she is in different places. The preliminary access control is a valid way to improve the system efficiency because the user's disallowed requests are terminated at the early stage. If the user's request is allowed by the preliminary access control, the user's request is forwarded to the databases. Meanwhile, a component in the secure Web server generates the security criterion subset for the user based on his/ her IP address and digital credential (as well as other factors required by the security policy).

The component is actually a well-designed program which implements the function of efficient abstracting the security criteria from the authorization rules. When the result of the request comes back to the secure Web server, based on the security criterion subset and the embedded security criterion expressions, the fine-grained access control is achieved by the secure Web server as described in section 3.

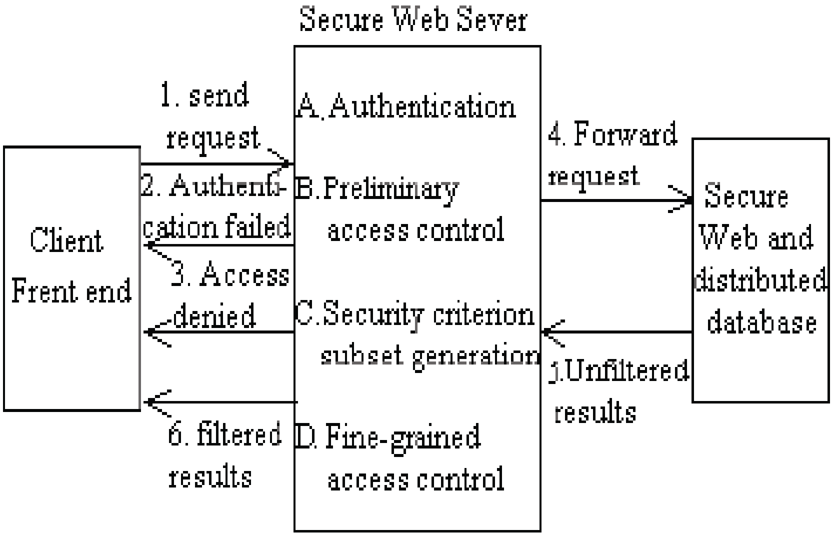


Figure 2. Working procedure of the proposed security system

Figure 2 summarizes the work of the proposed system.

Note: 1. Abstracting security criteria in secure Web server prevents users from modifying their security criterion subsets.

2. The authorization rules (security policy) may be different for different Web and distributed databases. It is convenient to implement flexible fine-grained access control for each database based on different authorization rules.

5. Conclusions

Addressing network security and database security simultaneously leads to efficient unified security system. The information used for authentication can be reused for the preliminary access control and fine-grained access control. The termination of the users' requests at the early stage avoids to unnecessarily process the requests further. The proposed model can be applied to many areas such as finance, health care, government, and military.

References

[1] Ponemon Institute LLC, "Database Security 2007: Threats and Priorities within IT Database Infrastructure", June 4, 2007,http://www.appsecinc.com/techdocs/whitepapers/2007-Ponemon- Database-Security-Study-Sponsored-by-Application-Security-Inc.pdf

[2] L.M. Null and J. Wong "A unified approach for multilevel database security based on inference engines" ACM SIGCSE Bulletin, Volume 21 , Issue 1 (February 1989), Pages: 108 – 111

[3] D. E. Denning, Cryptographic Checksums for Multilevel Database Security, 1984 IEEE Symposium on Security and Privacy P. 52..

[4] R. Schell and M. Heckman, "Views for Multilevel Database Security" 1986 IEEE Symposium on Security and Privacy p. 156

[5] E. Fernández-Medina and M. Piattini "A Methodology for Multilevel Database Design" http://ftp.informatik.rwthaachen.de/Publications /CEUR-WS/Vol 74/files/FORUM_09.pdf

[6] E. Bertino, B. Catania and E. Ferrari, "A nested transaction model for multilevel secure database management", ACM Transactions on Information and System Security (TISSEC), Volume 4 , Issue 4 (November 2001), Pages: 321 – 370

[7] W. Rjaib, **"**An introduction to multilevel secure relational database management systems", Proceedings of the 2004 conference of the Centre for Advanced Studies on Collaborative research, Markham, Ontario, Canada, Pages: 232 – 241.

[8] W. Itani, A. Kayssi and A. Chehab, **"**An enterprise policy-based security protocol for protecting relational database network objects" Proceedings of the 2006 international conference on Wireless communications and mobile computing, Vancouver, British Columbia, Canada, SESSION: T1-B: computer and network security symposium, Pages: 343 – 348.

[9] T. Didriksen, "Rule based database access control—a practical approach" Proceedings of the second ACM workshop on Role-based access control, Fairfax, Virginia, United States, Pages: 143 – 151.

[10] Charles P. Pfleeger and Shari Lawrence Pfleeger. *Security in Computing*, Fourth Edition. Prentice Hall, 2007.

[11] Pan, L., Zhang, C.N., and Yang, C. (Spring 2006 issue) A Role-Based Multilevel Security Access Control Model. The Journal of Computer Information Systems. Volume XLVI, Number 3, Pages: 1-10.

[12] Pan, L., and Zhang, C. N. (will be published in 2007) A Web-Based Multilayer Access Control Model for Multimedia Applications in MPEG-7. International Journal of Network Security. Vol.4, No.2, Pages: 155–165.

[13] Pan, L., and Zhang, C. N. (September 2006) Achieving Multilayer Access Control for Role-Based, Mandatory, and Discretionary Access control Models. Accepted by Information Systems Security.

[14] Pan, L., and Zhang, C. N. (2006) A Criterion-Based Role-Based Multilayer Access Control model for multimedia Applications. IEEE International Symposium on Multimedia (ISM2006). San Diego, Pages: 145-152..

[15] Pan, L., and Zhang, C. N, "A Criterion-Based Multilayer Access Control Approach for Multimedia Applications and the Implementation Considerations". Accepted by ACM Transactions on Multimedia Computing, Communications and Applications (ACM TOMCCAP), March, 2008.

[16] Pan, L, Using Criterion-Based Access Control for Multilevel Database Security. Proceedings of The International Symposium on Electronic Commerce and Security, Guangzhou, China, August 3-5, 2008, 518-522.

[17] Brands S (1999) A technical overview of digital credentials, http://www.xs4all.nl/#brands/

[18] M.R. Neilforoshan Network security architecture, Journal of Computing Sciences in Colleges, Volume 19 , Issue 4 (April 2004), Pages: 307–313

[19] D. Davis, R. Swick, Network security via private-key certificates, ACM SIGOPS Operating Systems Review Volume 24 , Issue 4 (October 1990) Pages: 64 – 67.

[20] K. Sadasivam, B. Samudrala, T. A. Yang, Design of network security projects using honeypots, Journal of Computing Sciences in Colleges, Volume 20 , Issue 4 (April 2005), Pages: 282 – 293

Predict the onset of diabetes disease using Artificial Neural Network (ANN)

Manaswini Pradhan[1] , Dr. Ranjit Kumar Sahu[2]

[1]*Lecturer, P.G. Department of Information and Communication Technology, Fakir Mohan University, Odisha, India*

[2] *Consultant, Tata Memorial Cancer Hospital (TMH), Mumbai, India*

[1]ms.manaswini.pradhan@gmail.com, [2]drsahurk@yahoo.co.in

Abstract: Diabetes Mellitus is a chronic metabolic disorder, where the improper management of the blood glucose level in the diabetic patients will lead to the risk of heart attack, kidney disease and renal failure. Data Classification is a prime task in Data mining. Accurate and simple data classification task can help the clustering of large dataset appropriately. In this paper we have experimented and suggested an Artificial Neural Network (ANN) based classification model as one of the powerful method in intelligent field for classifying diabetic patients into two classes. For achieving better results, genetic algorithm (GA) is used for feature selection. The GA is used for optimally finding out the number of neurons in the single hidden layered model. Further, the model is trained with Back Propagation (BP) algorithm and GA (Genetic Algorithm) and classification accuracies are compared. The designed models are also compared with the Functional Link ANN (FLANN) and several classification systems like NN (nearest neighbor), kNN(k-nearest neighbor), BSS(nearest neighbor with backward sequential selection of feature, MFS1(multiple feature subset) , MFS2(multiple feature subset) for Data classification accuracies. It is revealed from the simulation that our suggested model is performing better compared to NN(nearest neighbor), kNN(k-nearest neighbor), BSS(nearest neighbor with backward sequential selection of feature, MFS1(multiple feature subset) , MFS2(multiple feature subset) and FLANN model and it can be a very good candidate for many real time domain applications as these are simple with good performances.

Key words: ANN, Genetic Algorithm, Data classification, FLANN

1. Introduction

Diabetes is one of the most deadly, disabling, and costly diseases observed in many of the nations at present , and the disease continues to be on the rise at an alarming rate .Women tend to be hardest hit by diabetes with 9.6 million women having diabetes. This represents 8.8% of the total women adult population of the 18 years of age and above in 2003 and this is nearly a two fold increase from 1995 (4.7%). Women of minority racial and ethnic groups have the highest prevalence rates with two to four times the rates of the white population. With the increased growth of minority populations, the number of women in these groups who are diagnosed will increase significantly in the coming years. By 2050, the projected number of all persons with diabetes will have increased from 17 million to 29 million. Diabetes is metabolic disorder where people suffering from it either have a shortage of insulin or have a decreased ability to utilize their insulin. Insulin is a hormone that is produced by the pancreas and allows glucose to be converted to energy at the cell level. Diabetes that is uncontrolled, that is consistently high levels of blood glucose (> 200mgl/dL) leads to micro and macro vascular disease complications, such as, blindness, lower extremity amputations, end stage renal disease, and coronary heart disease and stroke. Diabetes is found in approximately one in ten individuals, but the chances increases to one in five as the age group increases to 65 years of age and above.

DIABETES Mellitus (DM) is a chronic and progressive metabolic disorder and according to the World Health Organization there are approximately million people in this world suffering from diabetes. The number of diabetic patients is expected to increase by more than 100% by the year 2030 [1]. Common manifestations of diabetes are characterized by insufficient insulin production by pancreas, ineffective use of the insulin produced by the pancreas or hyperglycemia. Causes like obesity, hypertension, elevated cholesterol level, high fat diet and sedentary lifestyle are the common factors that contribute to the prevalence of diabetes. Development of renal failure, blindness, kidney disease and coronary artery disease are types of the severe damage which are resulted by improper management and late diagnosis of diabetes. Even though there is no established cure for diabetes, indeed, the blood glucose level of diabetic patients can be controlled by well-established treatments, proper nutrition and regular exercise [1]-[3].

Data classification is a classical problem extensively studied by statisticians and machine learning researchers. It is an important problem in variety of engineering and scientific disciplines such as biology, psychology, medicines, marketing, computer vision, and artificial intelligence [1]. The goal of the data classification is to classify objects into a number of categories or classes. Given a dataset, its classification may fall into two tasks. First, supervised classification in which given data object is identified as a member of predefined class Second, unsupervised classification (or also known as Clustering) in which the data object is assigned to an unknown class. Supervised classification (here onwards to be referred as classification) algorithms have been widely applied to speech, vision, robotics, diseases, and artificial intelligence applications etc where real time response with complex real world data is necessity. There have been wide ranges of machine learning and statistical methods for solving classification problems. Different parametric and non-parametric classification algorithms have been studied [4-11]. Some of the algorithms are well suited for linearly separable problems. Non-linear separable problems have been solved by neural networks [12], support vector machines [13] etc. However, in many cases it is desired to find a simple classifier with simple architecture. There has been wide spectrum of work on developing ANN based classification models consisting of many hidden layers and large number of neurons in the hidden layers. It is obviously understood from the literature of ANN that more number of hidden layers and large number of neurons may sometimes present a good solution for the problem but at the expense of computational cost. There have been also some attempts made to use FLANN [15] for classification purpose. In Functional Link Artificial Neural Networks (FLANNs) the hidden layer is removed without giving up non-linearity by providing the input layer with expanded inputs that are constructed as the functions of original attributes [15]. Removal of hidden layer makes these networks extremely simple and computationally cheap. Identification of nonlinear processes using FLANNs has been reported by researchers [16]-[19]. FLANNs have an inherent limitation, of not guarantying universal approximation, which has deterred interest in them. Only a few applications using FLANNs are available in literature. In this paper ANN (Artificial Neural Network) model is suggested in which we have proposed a ANN model having m-n-p as the model parameters wherein m is the number of inputs (based on the dataset under investigation) , n is the number of neurons in hidden layer (only one hidden layer is used to minimize the computational complexity) and p is the number of output neurons (based on the dataset under investigation). The optimal numbers of neurons in the hidden layer are chosen by Genetic Algorithm (GA) [14]. The weights of the novel ANN are tuned by BP algorithm and GA for different datasets and results are compared. Our results are also compared with FLANN based models, NN (nearest neighbor), kNN(k-nearest neighbor), BSS(nearest neighbor with backward sequential selection of feature, MFS1(multiple feature subset) , MFS2(multiple feature subset). It is found that the models suggested by us are less in complexity and better in performance. The paper is sequentially arranged in the following order. Section II comprises related survey works. In section III, Back Propagation (BP) algorithm is briefly described, which is used to train the SANN. Section IV describes the GA approach for optimally finding the values for the number of neurons in the hidden layer. FLANN basics are described in section V. In section VI the dataset is described. Section VII discusses the simulation and results. Conclusion and future research are given in Section VIII.

2. Related Works

Apart from the works mentioned above, a lot of research has been done specifically using ANN in diagnosing diabetes mellitus and some approaches are discussed below.

Siti Farhanah, Bt Jaffar and Dannawaty Mohd [20] proposed a method for diagnosing diabetes. The diagnosis is accomplished using back propagation neural network algorithm. The inputs to the system are plasma glucose concentration, blood pressure, triceps skin fold, serum insulin, Body Mass Index (BMI), diabetes pedigree function, number of times a person was pregnant and age. The biggest challenge to this method was the missing values in the data set. This system was later modified and presented by T.Jayalakshmi and Dr.A.Santhakumaran[21].They have proposed an idea to overcome the missing values that was not addressed by Siti Farhanah Bt Jaafar [20] and this included constructing the data sets with reconstructed missing values, thereby increasing the classification accuracy[21]. They have also proposed an alternate method to overcome missing value by performing data pre-processing, which also speeds up the training process by reducing the actual learning time. Various missing value techniques and pre-processing methods were analyzed. By adopting these modifications, the results improved and achieved a classification accuracy of 99% [20]. Rajeeb Dey [22] proposed a method to predict diabetes mellitus using back propagation algorithm of Artificial Neural Network (ANN). The problem of diagnosing diabetes has been treated as a binary classification, i.e., those predicted to be diabetic falls under category 1 and others falls under category 0. The basic architecture of ANN used for accomplishing this classification task is a supervised multilayer feed-forward network with back propagation learning algorithm. The parameters considered in this system to diagnose diabetes are Random Blood Sugar test result, Fasting Blood Sugar test result, Post Plasma Blood Sugar test, age, sex and their occupation. The performance has been measured in terms of absolute error calculated between network response and desired target. Classification performance achieved using the system is 92.5%. Eng Khaled Eskaf [23] proposed a method for managing diabetic patients by trying to predict their glucose levels in the near future on the basis of current levels of glucose. The prediction is done using Artificial Neural Network (ANN). Feature extraction procedure was implemented on diabetic blood glucose time series. Blood glucose values of diabetic patients are recorded for 24 hours for about one week with a sampling frequency of 5 minutes. A dynamic model is used as an extraction procedure in order to extract different values from a blood glucose test. ANN was then trained using this knowledge to predict the blood glucose levels of a diabetic patient with reasonable accuracy. Gregory Hastings [24] proposed a hybrid model for diagnosis of diabetes mellitus by integrating three different data mining techniques using supervised and unsupervised learning algorithm. The inputs were processed using Support Vector Machine (SVM) and rules were extracted using electic approach. Real time data set is taken as input from which rules are formulated to describe relationships between input features and output class labels and thus diagnosis is done based on the rules extracted through electic approach. Of the two rules generated by the electic approach one was found to be inconsistent with generally acceptable medical knowledge in diagnosis of diabetes. The above survey clearly highlights that ANN technique for diagnosis of diabetes gives much better results than other existing techniques. Also, when considering the inputs to all these systems, there is at least one input value for which the patient should get the help of a doctor or a hospital staff. The proposed system aims to avoid the patients from undergoing blood tests, checking diastolic and systolic blood pressure etc, thus creating a user friendly environment without the need for a doctor or a hospital staff. The inputs designed are based on the symptoms which could appear during the early stages of diabetes and based on the physical conditions.

3. Back Propagation Training of ANN

An MLP (Multi-Layer Perceptron) network with 2-3-3 neurons (2, 3 and 3 denote the number of neurons in the input layer, the hidden layer and the output layer respectively) with the back-propagation (BP) learning algorithm, is depicted in Figure1.

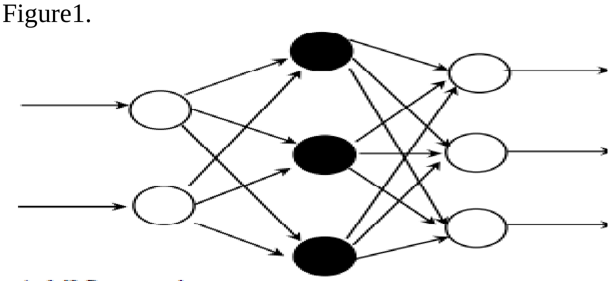


Figure 1: MLP network

Initialize the weights: In BP algorithm, initially the weights are initialized to very small random numbers. Each unit has a bias associated with it. The biases are similarly initialized to small random numbers.

Propagate the inputs forward: First, the training tuple is fed into the input layer of the network. The inputs pass through the input units, unchanged. Next, the net input and output of each unit in the hidden and output layers are computed. The net input to a unit in the hidden or output layers is computed as a linear combination of its inputs. To compute the net input to the unit, each input connected to the unit is multiplied by its corresponding weight, and is summed. Given a unit in a hidden layer or output layer, the net input, I_j, is

$$I_j = \sum_i W_{ij} O_i + \theta_j$$

(1)

where W_{ij} is the weight of the connection from the unit I in the previous layer to unit j .

O_i output of unit I from the previous layer

θ_j is the bias of the unit

The bias acts as a threshold in that it serves to vary the activity of the unit.

Each unit in the hidden and output layers takes its net input and then applies an activation function. The function symbolizes the activation of neuron represented by the unit. The logistic or sigmoid function is used and the output of unit j, is computed as

$$O_j = \frac{1}{1+e^{-I_j}}$$

(2)

This function is also referred to as squashing function, because it maps a large input domain onto the smaller range of 0 to 1.

Back propagate the error:The error is back propagated backward by updating the weights and biases to reflect the networks prediction. For a unit j in the output layer, the error is computed by

$$Error_j = O_j(1-O_j)(T_j - O_j)$$

(3)

where O_j is the actual output of unit j, and T_j is the known target value of the given training tuple. To compute the error of a hidden layer unit j, the weighted sum of the errors of the units connected to unit j in the next layer is considered. The error of a hidden layer unit j is,

$$Error_j = O_j(1-O_j)\sum_k Error_k W_{jk}$$

(4)

where W_{jk} is the weight of the connection from unit j to a unit k in the next higher layer, and $Error_k$ is the error of unit k.

The weights and biases are updated to reflect the propagated errors. Weights are updated by following equation,

$$W_{ij} = W_{ij} + (l) Error_j O_i$$
(5)

The variable *l* is the learning rate, a constant having a value between 0 to 1. Back propagation learns using a method of gradient descent to search for a set of weights that fits the training data so as to minimize the mean squared error. The learning rate helps avoid getting stuck at a local minimum and encourages finding global minimum.
Biases are updated by following equation,

$$\theta_j = \theta_j + (l) Error_j$$
(6)

Terminating condition:
Training stops when
i) All weights in the previous epoch were so small as to below some specific threshold
ii) The percentage of tuples misclassified in the previous epoch is below some threshold
iii) A prespecified number of epochs has expired.

A classifier which gives a higher accuracy value is considered as a good classifier.
Algorithm:
1. Initialize weights and biases in the network.
2. Propagates inputs forward in the usual way, i.e. All outputs are computed using sigmoid threshold of the inner product of the corresponding weight and input vectors. All outputs at stage n are connected to all the inputs at stage n+1
3. Propagates the errors backwards by apportioning them to each unit according to the amount of this error the unit is responsible for.
4. Terminating condition (Error is minimum or till the iterations are exhausted).

4. GA for optimally finding Neurons in hidden layer of ANN

Genetic algorithms (GA) are an evolutionary optimization approach which is an alternative to traditional optimization methods. GAs is most appropriate for complex non-linear models where location of the global optimum is a difficult task. It may be possible to use GA techniques to consider problems which may not be modeled as accurately using other approaches. Therefore, GA appears to be a potentially useful approach. GA follows the concept of solution evolution by stochastically developing generations of solution populations using a given fitness statistic (for example, the objective function in mathematical programs). They are particularly applicable to problems which are large, non-linear and possibly discrete in nature, features that traditionally add to the degree of complexity of solution. Due to the probabilistic development of the solution, GA does not guarantee optimality even when it may be reached. However, they are likely to be close to the global optimum. This probabilistic nature of the solution is also the reason they are not contained by local optima. The standard GA process consists of an initialization step and the iterative generations [14]. The Genetic Algorithm (GA) process is described below. First, a population of chromosomes is created. Second, the chromosomes are evaluated by a problem defined fitness function. Third, some of the chromosomes are selected for performing genetic operations. Fourth, genetic operations of crossover and mutations are performed. The offspring produced out of genetic operations replace their parents in their initial population. This GA process repeats until a user defined criterion is met.

Procedure Genetic Algorithm

```
begin
i=0 /* i: number of iteration*/
initialize P(i) /* P(i): population for iteration I */
compute f(P(i)) /* f: fitness function */
  perform until (non termination condition)
        begin
        i=i+1;
        choose two parents P1 and P2 from P(i-1)
        perform genetic operations
        {
        crossover;
         mutation;
        }
      reproduce a new P(i)
      compute f(P(i))
  end-perform
end
```

In our work chromosomes are nothing but string of integers within a random range depicting the possible values for the number of neurons in the hidden layer. Depending on the dataset complexity i.e the dimension and number of data objects in the dataset this range has been chosen for simulation. The fitness value is nothing but the classification accuracy of the model. More the percentage of correct classifications better the model. The population size is chosen based on the dataset. However, a population size up to 30 is a good choice for our simulations. In our work as we are aiming for ANN having only one hidden layer, the binary GA is implemented for the simulation. In Binary GA the chromosomes initialized by integer is converted to binary values for applying the GA operators. In entire of our work the one point cross over is adopted with 0.8 crossover probabilities and with 0.01 mutation probability.

5. Basics of FLANN

FLANN architecture for predicting a diabetic patient is a single layer feed forward neural network consisting of one input layer and an output layer.

The FLANN generates output (positive '1' or negative '0') by expanding the initial inputs (drivers) and then processing to the final output layer. Each input neuron corresponds to a component of an input vector. The output layer consists of one output neuron that computes diabetic positive/negative as a linear weighted sum of the outputs of the input layer. Pao[15] originally proposed the FLANN architecture. They have shown that, their proposed network may be conveniently used for function approximation and pattern classification with faster convergence rate and lesser computational load than an MLP structure. The FLANN is basically a flat net and the need of the hidden layer is removed and hence, the learning algorithm used in this network becomes very simple. The functional expansion effectively increases the dimensionality of the input vector and hence the hyper planes generated by the FLANN provide greater discrimination capability in the input pattern space.

To bridge the gap between the linearity in the single layer neural network and the highly complex and computation intensive multi layer neural network, the FLANN architecture is suggested. The FLANN architecture uses a single layer feed forward neural network and to overcome the linear mapping, functionally expands the input vector. Let each element of the input pattern before expansion be represented as $z(i), 1 < i < d$ where each element $z(i)$ is functionally expanded as $z_n(i)$, $1 < n < N$, where N=number of expanded points for each input element. Expansion of each input pattern is done as follows.

$$x_1(i) = z(i),\ x_2(i) = f1(z(i)),....,\ x_N(i) = fN(z(i))$$

where, $z(i), 1 < i < d$, d is the set of features in the dataset.

These expanded input patterns (shown in Figure 2) are then fed to the single layer neural network and the network is trained to obtain the desired output. The set of functions considered for function expansion may not be always suitable for mapping the nonlinearity of the complex task. In such cases few more functions may be incorporated to the set of functions considered for expansion of the input dataset. However dimensionality of many problems itself are very high and further increasing the dimensionality by to a very large extent may not be an appropriate choice. So, it is advisable to choose a small set of alternate functions, which can map the function to the desired extent.

Architecture of FLANN:

The FLANN network can be used not only for functional approximation but also for decreasing the computational complexity. This method is mainly focused on functional approximation. In the aspect of learning, the FLANN network is much faster than other network. The primary reason for this is that the learning process in FLANN network has two stages and both stages can be made efficient by appropriate learning algorithms. In this study the Functional Link Artificial Neural Network (FLANN) model for the task of pattern classification in data mining is evaluated. The FLANN model functionally expands the given set of inputs. These inputs are fed to the single layer feed forward ANN. A single layer model based on trigonometric expansion is presented. Let each element of the input pattern before expansion be represented as $z(i), 1 < i < I$ where each element $z(i)$ is functionally expanded as $z_n(i), 1 < n < N$, where N = number of expanded points for each input element. In this, N = 5 and I = total number of features in the dataset as been taken. Expansion of each input pattern is done as follows:

$$x\ (i) = z(i),\ x\ (i)= \sin\ \pi\ (z(i)),$$
$$x\ (i)=\sin 2\pi\ (z(i)),\ x\ (i)= \cos\ \pi\ (z(i)),$$
$$x\ (i)= \cos 2\pi\ (z(i))$$

where, $z(i), 1 < i < d$, d is the set of features in the dataset.

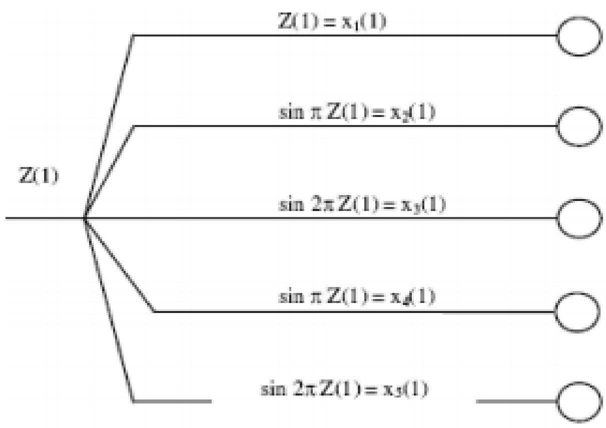


Figure 2: Functional expansion of each unit

These nonlinear outputs are multiplied by a set of random initialized weights from the range [-0.5, 0.5] and then summed to produce the estimated output. This output is compared with the corresponding desired output and the resultant error for the given pattern is used to compute the change in weight in each signal path P, given by

$$\Delta W_j(k) = \mu * xf_j(k) * e(k) \quad (7)$$

where, $xf_j(k)$ is the functionally expanded input at kth iteration.

Then the equation, which is used for weight update, is given by

$$W_j(k+1) = W_j(k) + DW_j(k) \quad (8)$$

where, $W_j(k)$ is the jth weight at the kth iteration, m is the convergence coefficient, its value lies between 0 to 1 and $1<j<J$, J = M´d. M is defined as

the number of functional expansion unit for one element.

$$e(k) = y(k) - \hat{y}(k) \quad (9)$$

where, y(k) is the target output and $\hat{y}(k)$ is the estimated output for the respective pattern an and is defined as:

$$\hat{y}(k) = \sum xf_j(k).w_k \quad (10)$$

where, xf_j is the functionally expanded input at kth iteration and $W_j(k)$ is the jth weight at the kth iteration and $W_j(0)$ is initialized with some random value from the range [-0.5, 0.5]. The FLANN for classification shown in Figure 3. FLANN is a single layer nonlinear network. Let *k* be number of input-output pattern pairs to be learned by the FLANN. Let the input pattern vector $\boldsymbol{X_k}$ be of dimension *n*., and the output y*k* be a scalar. The training patterns are denoted by { X_k, y_k}. A set of *N* basis functions

$$\phi(X_k) = [\phi(X_k)\phi(X_k)\phi(X_k)...\phi(X_k)]^T \quad (11)$$

are adopted to expand functionally the input signal $X_k = [x_1(k)x_2(k)...x_n(k)]^T$.
(12)
These *N* linearly independent functions map the n-dimensional space into an *N*-dimensional space, that is $R^n \rightarrow R^N, n < N$. The linear combination of these function values can be presented in its matrix form, that is $S = W\phi$. Here $[S_k = S_1(k)S_2(k)....Sm(k)]^T$, W is the *m X N* dimensional weight matrix. The matrix S_k is input into a set of nonlinear function $\rho(.) = \tanh(.)$ to generate the equalized output $\hat{Y}=[\hat{y}_1, \hat{y}_2 ......\hat{y}_m]$, $\hat{y}_j = \rho(S_j), j = 1,2,...,m.$ The major difference between the hardware structures of ANN and FLANN is that FLANN has only input and output layers and the hidden layers are completely replaced by nonlinear mappings. In fact, the task performed by the hidden layers in an ANN is carried out by functional expansions in FLANN. Being similar to ANN, the FLANN also uses BP algorithm to train the neural networks.

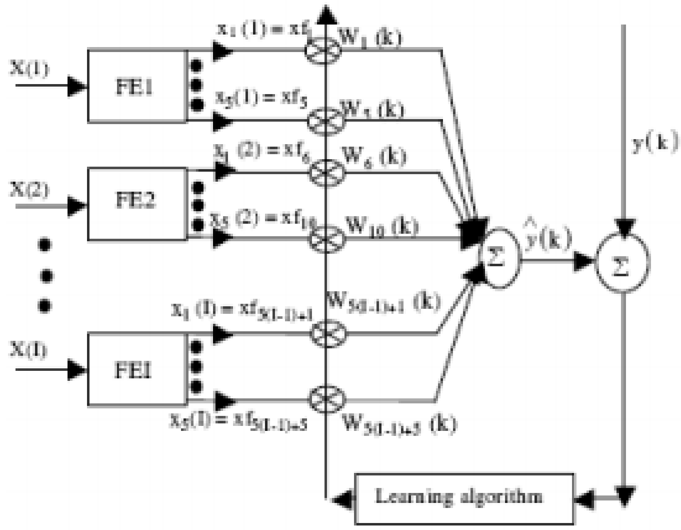


Figure 3: FLANN nonlinear model for classification

6. Dataset Description

In our work we have used Pima Indian Diabetes data sets [25] for training and testing the neural network model.
PIMA INDIAN DIABETES Dataset:
Several constraints were placed on the selection of these instances from a larger database. In particular, all patients here are females at least 21 years old of Pima Indian heritage.

Attribute Information:

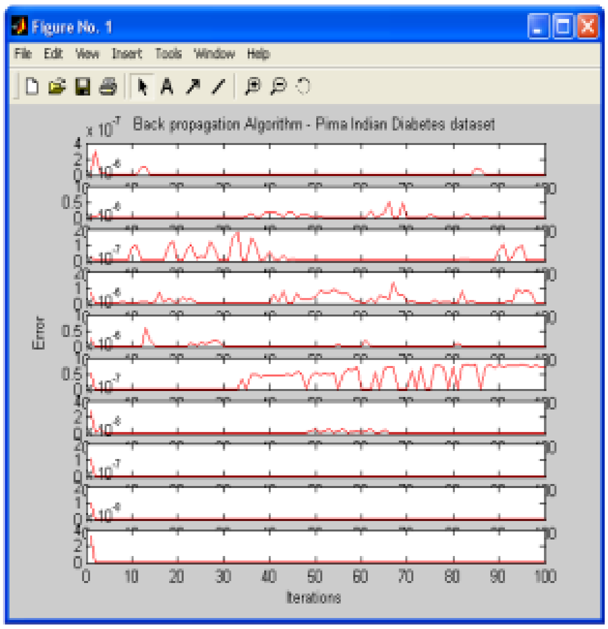
1. Number of times pregnant
2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3. Diastolic blood pressure (mm Hg)
4. Triceps skin fold thickness (mm)
5. 2-Hour serum insulin (mu U/ml)
6. Body mass index (weight in kg/(height in m)^2)
7. Diabetes pedigree function
8. Age (years)
9. Class variable (0 or 1)

7. Simulation and Results

The neural network, which we are using in back propagation algorithm, is m-n-1 type network. It represents that input layer would contain 'n' nodes, which will be equal to the number of attributes in the dataset we are using. Say in Pima Indian Diabetes dataset we have 8 attributes so of construction neural network for this we require 8 nodes in its input layer. In the above notation m represents nodes in hidden layer (only one hidden layer we are considering). We can have any number of nodes in hidden layer and in output layer we are considering only one node.
For training and testing we have adopted 10-fold cross validation for Pima Indian Diabetes, in which we divide tuples in the dataset into 10 equal

divisions. We apply back propagation algorithm for first 9 divisions and train the network for certain number of iterations. After training the network we then apply same algorithm without propagation of errors back and find the accuracy of the 10th division. We repeat this process for all the remaining divisions by placing the last division on the top moving down the remaining tuples Such that each division will take part in training the network.

GA is used for optimally selecting the n values. For Pima dataset the ANN gives the best accuracy with 5 neurons in the hidden layer. Best accuracy being 72% with average accuracy of 72.2%. The MSE is at 1.6838e- 004. The error plot is shown below.



Back propagation algorithm X: Y –>Iteration versus Error

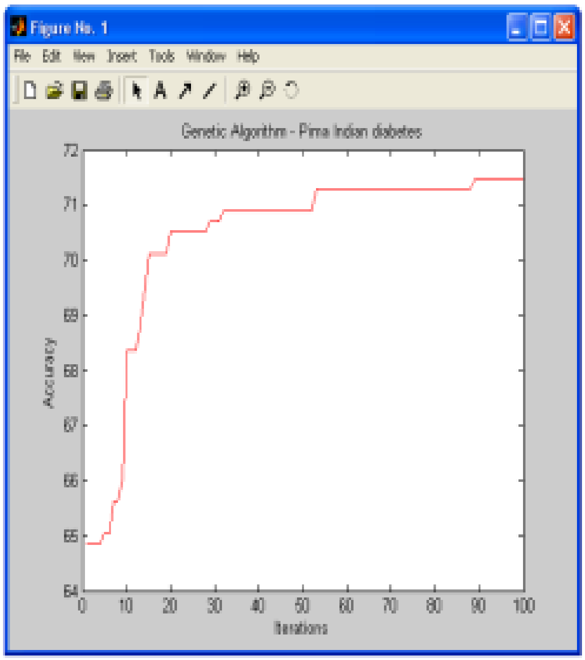
The results obtained using BP algorithm for all the investigated dataset reveal that the networks fall into the local minima. The results can be improved if some randomized optimization techniques is used for training the ANN i.e optimizing weights of the ANN. This has motivated us to explore the use of GA for optimizing the weights of ANN. In our experiments we have used the same ANN which we have obtained with GA and trained with BP. For Pima the ANN is 8-5-1. For example, in the Pima Indian diabetes dataset ANN model total weights to optimized are (8*5+5*1) =45. In our simulation the population size is taken to be 30. The performance of the model is shown in the table 1 given below.

Table 1: Performance of Pima Indian Diabetes model

Model/Dataset	**Best Accuracy**	**Average accuracy(standard)**
8-5-1/Pima	73.438	71.212(0.324)

The weights of these networks are optimized using GA. The chromosomes of GA are the strings of real numbers randomly chosen in a range. The length of chromosome is determined by the number of weights to be optimized. The number of weights to be optimized is (m*n+n*p)
The performances of these models are shown in the table 1 given below.

The results reveal GA based optimization for ANN is far more accurate with comparison to BP based training. The fitness curves for the Pima dataset is given below.



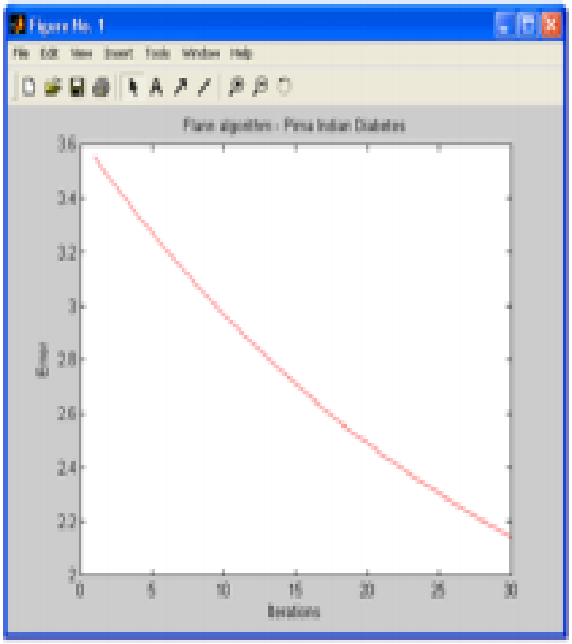
Genetic Algorithm X: Y –>Iteration versus Accuracy

The simulated result of FLANN model is shown below in a tabular format in Table 2. The experiment is done for 30 epochs and 10 such simulations are considered for finding average correct classifications.

Table 2: Simulation result of FLANN Pima Model

Dataset	**Convergence coefficient**	**Best accuracy**	**Average accuracy**
Pima Indian Diabetes	8e-006	71.845	59.76

The error curves using FLANN for the Pima model shown below is x: y is iterations verses error



FLANN algorithm X: Y –>Iteration versus Error

Comparison with other models:

The results obtained for the Pima Indian Diabetes Database dataset were compared with the results described in (Aha, D.W. and R.L. Bankert, 1994. Feature selection for case-based classification of cloud types: An empirical comparison. Proc. Am. Assn. for Artificial Intelligence (AAAI-94)-Workshop Case-Based Reasonings, pp: 106-112.) where the performance of several models is presented: NN(nearest neighbor), kNN(k-nearest neighbor), BSS(nearest neighbor with backward sequential selection of feature, MFS1(multiple feature subset) , MFS2(multiple feature subset).Table 3 presents the results obtained by the various different models.

Table 3: Comparison of the average performance of several other classification systems

Models	**Pima Indian diabetes dataset**
NN	65.1%
kNN	69.7%
BSS	67.7%
MFS1	68.5%
MFS2	70.5%
Novel ANN	73.4%
FLANN	59.8%

8. Conclusions

In recent years several researches have been conducted to classify and show that who is diabetic or not. Shanker used neural networks (NN) to predict the diabetic person and also showed that neural networks obtained a better accuracy which was higher than other methods like logistic regression, feature selection, decision tree etc. This paper has explored the design of a novel ANN for data classifications. We have evaluated the ANN model for the task of pattern classification in data mining. The novel ANN is envisioned by using GA for optimally deciding the number of neurons in single hidden layer architecture. The weights of such novel ANN is trained using BP algorithm and GA algorithm respectively. The experimental studies demonstrated that the ANN model performs the pattern classification task quite well. Again, this gives a very clear impression of the simplicity of the model without sacrificing at the cost of accuracy. This model proved to be better than other models and algorithms with which it was compared. The performance of this model is remarkable in terms of processing time, which is treated as one of the crucial aspect in data mining. Contrary to the views of many researchers it is felt that we can have novel ANN model with only one hidden layer and having very few neurons in the hidden layer. Even our designed novel ANN outperforms NN(nearest neighbor), kNN(k-nearest neighbor), BSS(nearest neighbor with backward sequential selection of feature, MFS1(multiple feature subset) , MFS2(multiple feature subset) [26] and FLANN classification models in the dataset .Due to less computational cost at the hidden layer novel ANN can have applications in the real time domain.
However, this study is nascent to claim the universalness of the model. It is open for further study to examine how the other models like Bayesian classifier, Decision tree, etc, behave with comparison to this suggested models. It can also be further investigated to improvise the classification accuracies using some other randomized optimization techniques. Performance comparisons with some other well known approach for data classification can also be a better direction for future work.

References

[1] World Health Organization. Available: http://www.who.int
[2] American Diabetes Asscociation. Available: http://www.diabetes.org
[3] Gan, D. editor. Diabetes atlas, 2nd ed. Brussels: International Diabetes Federation, 2003. Available at http://www.eatlas.idf.org/webdata/docs/Atlas%202003-Summary.pdf2]
[4] R.O.Duda and P.E.Hard, Pattern classification and Scene Analysis, John wiley & Sons, NY, USA, 1973.
[5] Breitman,L.,Friedman,J.H.,Olshen,R.A.,C. J.,Classification and Regression tress,Wadsworth,Belmont,CA, 1984.
[6] Buntine,W.L., Learning classification trees, Statistics and Computing, 1992,pp. 63-73.
[7] Cover,T.M.,Hart,P.E., Nearest neighbors pattern classification, IEEE Trans on Information Theory, vol. 13, ,1967,pp. 21-27.
[8] Hanson R.,Stutz,J.,Cheeseman,P., Bayesian classification with correlation and

inheritance, Proceedings of the 12 th International Joint Conference on Artificial Intelligence 2, Sydney,Australia,Morgan KaufSANN, 1992,pp. 692-698.

[9] Michie,D. et al , Machine Learning, Neural and Statistical Classification, Ellis Horwood,1994.

[10]Richard,M.D, LippSANN,R.P., Neural network classifiers estimate Bayesian a-posterior probabilities, Neural Computation ,vol.3, ,1991,pp. 461-483.

[11] Tsoi, A.C et al, Comparison of three classification Techniques, CART,C4.5 and multilayer perceptrons ,Advances in Neural Information Processing Systems, vol. 3, 1991 pp.963-969.

[12]C.Bishop,Neural Networks for Pattern Recognition. New York: Oxford Univ. Press, 1995.

[13] V.N.Vapnik, A.Y. Chervonenkis, On the uniform convergence of relative frequencies of events to IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.10, October 2009 115 their probabilities, Theory of Probability and its Applications, 1971, pp.264-280.

[14] D.E.Goldberg, "Genetic Algorithms in Search, Optimization and machine Learning", AddisionWesley, New York, 1989.

[15] Y. H. Pao, Adaptive Pattern Recognition and Neural Networks, Reading, MA: Addison-Wesley, 1989.

[16] S. Chen and S. A. Billings, "Neural networks for non-linear dynamic system modeling and identification," *Int. J. of Control*, vol. 56, no. 2, pp. 319-346, 1992.

[17] J. C Patra., R. N. Pal and B. N.Chatterji, "Identification of non-linear dynamic systems using functional link artificial neural networks," *IEEE Trans. on Neural Networks*, vol. 29, no. 2, pp. 254-262, 1999.

[18] A Ugena., F.de Arriaga and M. El Alami, "Speaker-independent speech recognition by means of functional-link neural networks," *Int. Conf. on* Pattern Recognition (ICPR'00)-Vol. 2, Barcelona, Spain (2000)

[19] L.H.P Harada.; A.C Da Costa.; R.M Filho., "Hybrid neural modeling of bioprocesses using functional link networks," *Applied Biochemistry and Biotechnology*, vol. 98, no. 1-3, pp. 1009-1024, 2002.]

[20] Siti Farhanah Bt Jaafar and Dannawaty Mohd Ali, "Diabetes mellitus forecast using artificial neural networks", Asian conference of paramedical research proceedings, 5-7, September, 2005, Kuala Lumpur, MALAYSIA

[21] T.Jayalakshmi and Dr.A.Santhakumaran, "A novel classification method for classification of diabetes mellitus using artificial neural networks". 2010 International Conference on Data Storage and Data Engineering.

[22] Rajeeb Dey and Vaibhav Bajpai and Gagan Gandhi and Barnali Dey, "Application of artificial neural network technique for diagnosing diabetes mellitus", 2008 IEEE Region 10 Colloquium and the Third ICIIS, Kharagpur, INDIA December 8-10.

[23] Eng Khaled Eskaf, Prof. Dr. Osama Badawi and Prof. Dr. Tim Ritchings," Predicting blood glucose levels in diabetes using feature extraction and artificial neural networks".

[24]Gregory Hastings, Nejhdeh Ghevondian, "A selforganizing estimator for hypoglycemia monitoring in diabetic patients", 20 th annual international conference of IEEE engineering in medicine and biology society, Vol. 20, No 3, 1998.

[25]UCI machine learning repository and archive.ics.uci.edu/ml/datasets.html

[26] Bay, S.D., 1999. Nearest neighbor classification from multiple feature subset. Intell. Data Anal. , 3: 191-209.

Authors Profile

Manaswini Pradhan received the B.E. in Computer Science and Engineering, M.Tech in Computer Science from Utkal University, Orissa, India.She is into teaching field from 1998 to till date. Currently she is working as a Lecturer in P.G. Department of Information and Communication Technology, Orissa, India. She is currently persuing the Ph.D. degree in the P.G. Department of Information and communication Technology, Fakir Mohan University, Orissa, India. Her research interest areas are neural networks, soft computing techniques, data mining, bioinformatics and computational biology.

Dr Ranjit Kumar Sahu,, M.B.B.S, M.S. (General Surgery), M. Ch. (Plastic Surgery). Worked as an Assistant Surgeon in post doctoral department of Plastic and reconstructive surgery, S.C.B. Medical College, Cuttack, Orissa, India. Presently working as a Consultant, Tata Memorial Cancer Hospital (TMH), Mumbai, India, He has five years of research experience in the field of surgery and published many national and international papers in Medical field.

On (1,2)*-Semi-Generalized-Star Homeomorphisms

O.Ravi[1], S. Pious Missier[2], T. Salai Parkunan[3]
And K.Mahaboob Hassain Sherieff[4]

[1]Department of Mathematics, P. M. Thevar College, Usilampatti, Madurai Dt, Tamilnadu, India, Email : siingam@yahoo.com
[2]Department of Mathematics, V. O. Chidambaram College, Thoothukudi, Tamilnadu, India, Email : spmissier@yahoo.com
[3]Department of Mathematics, Arul Anandar College, Karumathur, Madurai Dt, Tamilnadu, India, Email :parkunan@yahoo.com
[4]Department of Mathematics, S.L.S. Mavmm AV College, Kallampatti, Madurai Dt., Tamilnadu, India, Email : rosesheri14@yahoo.com

Corresponding Author: T.SALAI PARKUNAN parkunan@yahoo.com

***Abstract*:** *The aim of this paper is to introduce the concept of (1,2)*-semi-generalised-star closed sets (briefly (1,2)*-sg*-closed sets) and study some of its properties. Their corresponding pre-(1,2)*-sg*-closed maps and (1,2)*-sg*-irresolute maps are defined and studied in this paper.*

***Keywords*:** *(1,2)*-sg*-closed set, (1,2)*-sg*-open set, pre-(1,2)*-sg*-closed map, (1,2)*-sg*-irresolute map.*

2000 Mathematics Subject Classification .54E55.

1. Introduction

The study of bitopological spaces was first initiated by Kelly [2] in the year 1963. Recently Ravi, Lellis Thivagar, Ekici and Many others [3, 5 - 14] have defined different weak forms of the topological notions, namely, semi-open, preopen, regular open and α-open sets in bitopological spaces.

In this paper, we introduce the notion of (1,2)*-semi-generalized-star closed (briefly, (1,2)*-sg*-closed) sets and investigate their properties. By using the class of (1,2)*-sg*-closed sets, we study the properties of (1,2)*-sg*-open sets, pre-(1,2)*-sg*-closed maps and (1,2)*-sg*-irresolute maps. In most of the occasions our ideas are illustrated and substantiated by some suitable examples.

2. Preliminaries

Throughout this paper, X and Y denote bitopological spaces (X, τ_1, τ_2) and (Y, σ_1, σ_2) respectively, on which no separation axioms are assumed.

Definition 2.1

Let S be a subset of X. Then S is called $\tau_{1,2}$-open [13] if S = A ∪ B, where $A \in \tau_1$ and $B \in \tau_2$.

The complement of $\tau_{1,2}$-open set is called $\tau_{1,2}$-closed.

The family of all $\tau_{1,2}$-open (resp. $\tau_{1,2}$-closed) sets of X is denoted by (1,2)*-O(X) (resp. (1,2)*-C(X)).

Example 2.2

Let X = {a, b, c}, $\tau_1 = \{\varphi, X, \{b\}\}$ and $\tau_2 = \{\varphi, X, \{c\}\}$. Then the sets in {φ, X, {b}, {c}, {b, c}} are $\tau_{1,2}$-open and the sets in {φ, X, {a}, {a, b}, {a, c}} are $\tau_{1,2}$-closed.

Definition 2.3

Let A be a subset of a bitopological space X. Then

(i) the $\tau_{1,2}$-closure of A [12], denoted by $\tau_{1,2}$-cl(A), is defined by $\cap$ {U: A ⊆ U and U is $\tau_{1,2}$-closed };

(ii) the $\tau_{1,2}$-interior of A [12], denoted by $\tau_{1,2}$-int(A), is defined by ∪ {U: U ⊆ A and U is $\tau_{1,2}$-open}.

Remark 2.4

Notice that $\tau_{1,2}$-open subsets of X need not necessarily form a topology.

Now we recall some definitions and results, which are used in this paper.

Definition 2.5

A subset S of a bitopological space X is said to be (1,2)*-semi-open [12] if S ⊆ $\tau_{1,2}$-cl($\tau_{1,2}$-int(S)).

The complement of (1,2)*-semi- open set is called (1,2)*-semi-closed.

The family of all (1,2)*-semi-open sets of X will be denoted by (1,2)*-SO(X).

The (1,2)*-semi-closure of a subset S of X is, denoted by (1,2)*-scl(S), defined as the intersection of all (1,2)*-semi-closed sets containing S.

Definition 2.6

A subset S of a bitopological space X is said to be a (1,2)*-sg-closed [10] if (1,2)*-scl(S) ⊂ U whenever S ⊂ U and U ∈ (1,2)*-SO(X).

Definition 2.7

A subset S of a bitopological space X is said to be a (1,2)*-g-closed [14] if $\tau_{1,2}$-cl(S) ⊂ U whenever S ⊂ U and U ∈ (1,2)*-O(X).

The complement of (1,2)*-g-closed set is (1,2)*-g-open.

Definition 2.8

A map f : X → Y is called

(i) (1,2)*-continuous [12] if $f^{-1}(V)$ is $\tau_{1,2}$-closed in X for every $\sigma_{1,2}$-closed set V in Y.

(ii) (1,2)*-open map [11] if the image of every $\tau_{1,2}$-open set is an $\sigma_{1,2}$-open.

(iii) (1,2)*-irresolute [9] if the inverse image of (1,2)*-semi-open set is (1,2)*- semi-open.

Definition 2.9

A map f : X → Y is called (1,2)*-homeomorphism [11] if f is bijection, (1,2)*-continuous and (1,2)*-open.

3. (1,2)*-Semi-Generalised- Star-Closed Sets

Definition 3.1

A subset A of a bitopological space X is called a (1,2)*-semi-generalised-star-closed (briefly, (1,2)*-sg*-closed) if $\tau_{1,2}$-cl(A) ⊂ U whenever A ⊂ U and U is (1,2)*-semi-open in X.

Example 3.2

Let X = {a, b, c}, τ_1 = {φ, X, {b}} and τ_2 = {φ, X, {c}}. Then the sets in {φ, X, {b}, {c}, {b, c}} are $\tau_{1,2}$-open. Clearly the sets in {φ, X, {a}, {a, b}, {a, c}} are (1,2)*-sg*-closed.

Theorem 3.3

Every $\tau_{1,2}$-closed set is (1,2)*-sg*-closed.

Proof:

Let A be a $\tau_{1,2}$-closed subset of X. Let A ⊂ U and U be (1,2)*-semi-open. $\tau_{1,2}$-cl(A) = A, since A is $\tau_{1,2}$-closed. Therefore $\tau_{1,2}$-cl(A) ⊂ U. Hence A is (1,2)*-sg*-closed.

Remark 3.4

The converse of Theorem 3.3 need not be true as shown in the following example.

Example 3.5

Let X = {a, b, c}, τ_1 = {φ, X, {a, b}} and τ_2 = {φ, X}. Then the set {a, c} is (1,2)*-sg*-closed but not $\tau_{1,2}$-closed.

Theorem 3.6

Every (1,2)*-sg*-closed set is (1,2)*-g-closed .

Proof:

Let A be a (1,2)*-sg*-closed subset of X. Let A ⊂ U and U be $\tau_{1,2}$-open. Then U is (1,2)*-semi-open since every $\tau_{1,2}$-open set is (1,2)*-semi-open. Since A is (1,2)*-sg*-closed and U is (1,2)*-semi-open, we have $\tau_{1,2}$-cl(A) ⊂ U. Hence A is (1,2)*-g-closed .

Remark 3.7

The converse of Theorem 3.6 need not be true as shown in the following example.

Example 3.8

Let X = {a, b, c, d}, τ_1 = {φ, X, {a, b}} and τ_2 = {φ, X, {b, c}, {a, b, c}}. Then the sets in {φ, X, {a, b}, {b, c}, {a, b, c}} are $\tau_{1,2}$-open. Then the set {b, d} is (1,2)*-g-closed but not (1,2)*-sg*-closed.

Remark 3.9

(1,2)*-semi-closed sets and (1,2)*-sg*-closed sets are independent of each other.

Example 3.10

Let X = {a, b, c}, τ_1 = {φ, X, {a}, {b}, {a, b}} and τ_2 = {φ, X, {a, c}}. Then the sets in {φ, X, {a}, {b}, {a, b}, {a, c}} are $\tau_{1,2}$-open. Clearly the set {b} is (1,2)*-semi-closed set but not (1,2)*-sg*-closed.

Example 3.11

Let X = {a, b, c, d}, τ_1 = {φ, X, {a, b}} and τ_2 = {φ, X, {b, c}, {a, b, c}}. Then the sets in {φ, X, {a, b}, {b, c}, {a, b, c}} are $\tau_{1,2}$-open. Clearly the set {a, c, d} is (1,2)*-sg*-closed set but not (1,2)*-semi-closed.

Remark 3.12

Union of two (1,2)*-sg*-closed sets need not be a (1,2)*-sg*-closed.

Example 3.13

Let X = {a, b, c}, τ_1 = {φ, X, {a}, {c}, {a, c}, {b, c}} and τ_2 = {φ, X, {a}, {a, b}}. Then the sets in {φ, X, {a}, {c}, {a, b}, {a, c}, {b, c}} are $\tau_{1,2}$-open and the sets in {φ, X, {a}, {b}, {c}, {a, b}, {b, c}} are (1,2)*-sg*-closed. But {a} ∪ {c} = {a, c} is not (1,2)*-sg*-closed.

Theorem 3.14

A (1,2)*-sg*-closed set which is (1,2)*-semi-open is $\tau_{1,2}$-closed.

Proof:

Let A be a (1,2)*-sg*-closed set which is (1,2)*-semi-open. We have $A \subset A$ and A is (1,2)*-semi-open. Since A is (1,2)*-sg*-closed, $\tau_{1,2}$-cl(A) $\subset$ A. It is well known that A $\subset$ $\tau_{1,2}$-cl(A). Hence A is $\tau_{1,2}$-closed.

Result 3.15

Being (1,2)*-semi-open is a sufficient condition for a (1,2)*-sg*-closed set to be $\tau_{1,2}$-closed. However this condition is not necessary. There are (1,2)*-sg*-closed sets which are $\tau_{1,2}$-closed but not (1,2)*-semi-open.

Example 3.16

Let X = {a, b, c}, τ_1 = {φ, X, {a}, {c}, {a, c}} and τ_2 = {φ, X, {c}}. Then the sets in {φ, X, {a}, {c}, {a, c}} are $\tau_{1,2}$-open.Clearly the set {b} is (1,2)*-sg*-closed set and $\tau_{1,2}$-closed but not (1,2)*-semi-open.

Theorem 3.17

If A is (1,2)*-sg*-closed and A $\subset$ B $\subset$ $\tau_{1,2}$-cl(A), then B is (1,2)*-sg*-closed.

Proof:

Let A be a (1,2)*-sg*-closed subset of X. Since A $\subset$ B $\subset$ $\tau_{1,2}$-cl(A), we have $\tau_{1,2}$-cl(B) $\subset$ $\tau_{1,2}$-cl(A). Let B $\subset$ U and U be (1,2)*-semi-open. Then A $\subset$ U, $\tau_{1,2}$-cl(A) $\subset$ U since A is (1,2)*-sg*-closed. Hence $\tau_{1,2}$-cl(B) $\subset$ U. Hence B is (1,2)*-sg*-closed.

Theorem 3.18

Let A be (1,2)*-sg*-closed in X but not $\tau_{1,2}$-closed. Then for every $\tau_{1,2}$-open set U $\subset$ A, there exists an $\tau_{1,2}$-open set V such that A intersects V and $\tau_{1,2}$-cl(U) does not intersect V.

Proof:

Assume that A is (1,2)*-sg*-closed but not $\tau_{1,2}$-closed. Let U $\subset$ A and U be $\tau_{1,2}$-open. We claim that A $\not\subset$ $\tau_{1,2}$-cl(U). If A $\subset$ $\tau_{1,2}$-cl(U), then U $\subset$ A $\subset$ $\tau_{1,2}$-cl(U) and U is $\tau_{1,2}$-open. Hence A is (1,2)*-semi-open. Therefore A is (1,2)*-sg*-closed and (1,2)*-semi-open which implies A is $\tau_{1,2}$-closed. But A is not $\tau_{1,2}$-closed. Hence A $\not\subset$ $\tau_{1,2}$-cl(U). Hence there exists x ∈ A and x ∉ $\tau_{1,2}$-cl(U). Let V = $(\tau_{1,2}\text{-cl}(U))^c$. Then V is $\tau_{1,2}$-open and $\tau_{1,2}$-cl(U) does not intersect V. Since x ∉ $\tau_{1,2}$-cl(U), we have x ∈ $(\tau_{1,2}\text{-cl}(U))^c$. Hence x ∈ V. Since x ∈ A and x ∈ V, A ∩ V ≠ φ, A intersects V and $\tau_{1,2}$-cl(U) does not intersect V.

Definition 3.19

Let X be a bitopological space and A $\subset$ X. Then (1,2)*-frontier of A, denoted by (1,2)*-Fr(A), is defined to be the set $\tau_{1,2}$-cl(A) \ $\tau_{1,2}$-int(A).

Theorem 3.20

Let A be (1,2)*-sg*-closed and A $\subset$ U where U is $\tau_{1,2}$-open. Then (1,2)*-Fr(U) $\subset$ $\tau_{1,2}$-int(A^c).

Proof:

Let A be (1,2)*-sg*-closed and let A $\subset$ U where U is $\tau_{1,2}$-open. Then $\tau_{1,2}$-cl(A) $\subset$ U. Take any x ∈ (1,2)*-Fr(U).We have x ∈ $\tau_{1,2}$-cl(U) \ $\tau_{1,2}$-int(U). Hence x ∈ $\tau_{1,2}$-cl(U) \ U since U is $\tau_{1,2}$-open. Hence x ∉ U. Therefore x ∉ $\tau_{1,2}$-cl(A). Hence x ∈ $(\tau_{1,2}\text{-cl}(A))^c$. Therefore x ∈ $\tau_{1,2}$-int(A^c). Hence (1,2)*-Fr(U) $\subset$ $\tau_{1,2}$-int(A^c).

Definition 3.21

A bitopological space X is called RM-space if every subset in X is either $\tau_{1,2}$-open or $\tau_{1,2}$-closed.

Theorem 3.22

In a RM-space X, every (1,2)*-sg*-closed set is $\tau_{1,2}$-closed.

Proof:

Let X be a RM-space. Let A be a (1,2)*-sg*-closed subset of X. Then A is $\tau_{1,2}$-open or $\tau_{1,2}$-closed. If A is $\tau_{1,2}$-closed, then nothing to prove. If A is $\tau_{1,2}$-open, then A is (1,2)*-semi-open. Since A is (1,2)*-sg*-closed and A is (1,2)*-semi-open, by Theorem 3.14, A is $\tau_{1,2}$-closed.

Remark 3.23

The converse of Theorem 3.22 need not be true as shown in the following example.

Example 3.24

Let X = {a, b, c}, τ_1 = {φ, X, {a}} and τ_2 = {φ, X}. Then the sets in {φ, X, {a}} are $\tau_{1,2}$-open and the sets in {φ, X, {b, c}} are (1,2)*-sg*-closed. Therefore every (1,2)*-sg*-closed set is $\tau_{1,2}$-closed. But X is not a RM-space.

Definition 3.25

A subset A of a bitopological space X is said to be (1,2)*-semi-generalised-star-open (briefly, (1,2)*-sg*-open) if A^c is (1,2)*-sg*-closed.

Theorem 3.26

Every $\tau_{1,2}$-open set is (1,2)*-sg*-open.

Proof:

Let A be an $\tau_{1,2}$-open set of X. Then A^c is $\tau_{1,2}$-closed. Also A^c is (1,2)*-sg*-closed since every $\tau_{1,2}$-closed set is (1,2)*-sg*-closed. Hence A is (1,2)*-sg*-open.

Remark 3.27

The converse of Theorem 3.26 need not be true as shown in the following example.

Example 3.28

Let X = {a, b, c, d}, τ_1 = {φ, X, {a, b, d}} and τ_2 = {φ, X, {b, c, d}}. Then the sets in {φ, X, {a, b, d}, {b, c, d}} are $\tau_{1,2}$-open. Clearly the set {b} is (1,2)*-sg*-open but not $\tau_{1,2}$-open.

Theorem 3.29

Every (1,2)*-sg*-open set is (1,2)*-g-open.

Proof:

Let A be a (1,2)*-sg*-open set of X. Then A^c is (1,2)*-sg*-closed. Also A^c is (1,2)*-g-closed, since every (1,2)*-sg*-closed set is (1,2)*-g-closed . Hence A is (1,2)*-g-open.

Remark 3.30

The converse of Theorem 3.29 need not be true as shown in the following example.

Example 3.31

Let X = {a, b, c}, τ_1 = {φ, X, {a}} and τ_2 = {φ, X, {a, c}}. Then the sets in {φ, X, {a}, {a, c}} are $\tau_{1,2}$-open. Then the set {c} is (1,2)*-g-open but not (1,2)*-sg*-open.

Remark 3.32

Intersection of two (1,2)*-sg*-open sets need not be a (1,2)*-sg*-open.

Example 3.33

Let X = {a, b, c}, τ_1 = {φ, X, {a}, {c}, {a, c}, {b, c}} and τ_2 = {φ, X, {a}, {a, b}}. Then the sets in {φ, X, {a}, {c}, {a, b}, {a, c}, {b, c}} are $\tau_{1,2}$-open and the sets in {φ, X, {a}, {c}, {a, b}, {a, c}, {b, c}} are (1,2)*-sg*-open. But {a, b} ∩ {b, c} = {b} is not (1,2)*-sg*-open.

Theorem 3.34

If A is (1,2)*-sg*-open and $\tau_{1,2}$-int(A) ⊂ B ⊂ A. Then B is (1,2)*-sg*-open.

Proof:

Let A be (1,2)*-sg*-open. Hence A^c is (1,2)*-sg*-closed. Since $\tau_{1,2}$-int(A) ⊂ B ⊂ A, $(\tau_{1,2}\text{-int}A)^c \supset B^c \supset A^c$. Therefore $A^c \subset B^c \subset \tau_{1,2}$-cl($A^c$). Hence by Theorem 3.17, B^c is (1,2)*-sg*-closed. Hence B is (1,2)*-sg*-open.

Theorem 3.35

If A is (1,2)*-sg*-open and A ⊃ F where F is $\tau_{1,2}$-closed then (1,2)*-Fr(F) ⊂ $\tau_{1,2}$-int(A).

Proof:

Given that A is (1,2)*-sg*-open and A ⊃ F where F is $\tau_{1,2}$-closed. Then A^c is (1,2)*-sg*-closed, $A^c \subset F^c$ and F^c is $\tau_{1,2}$-open. By Theorem 3.20 (1,2)*-Fr(F^c) ⊂ $\tau_{1,2}$-int(A). Hence (1,2)*-Fr(F) ⊂ $\tau_{1,2}$-int(A) since (1,2)*-Fr(F^c) = (1,2)*-Fr(F).

Theorem 3.36

In a RM-space X, every (1,2)*-sg*-open set is $\tau_{1,2}$-open.

Proof:

Let X be a RM-space. Let A be a (1,2)*-sg*-open subset of X. Then A^c is (1,2)*-sg*-closed. Since X is a RM-space, A^c is $\tau_{1,2}$-closed. Hence A is $\tau_{1,2}$-open.

Remark 3.37

The converse of Theorem 3.36 need not be true as shown in the following example.

Example 3.38

Let X = {a, b, c}, τ_1 = {φ, X, {b}} and τ_2 = {φ, X}. Then the sets in {φ, X, {b}} are $\tau_{1,2}$-open and the sets in {φ, X, {b}} are (1,2)*-sg*-open. Therefore every (1,2)*-sg*-open set is $\tau_{1,2}$-open. But X is not a RM-space.

Theorem 3.39

Any singleton set is either (1,2)*-semi-closed or (1,2)*-sg*-open.

Proof:

Take {x}, if it is (1,2)*-semi-closed then nothing to prove. If it is not (1,2)*-semi-closed, then $\{x\}^c$ is not (1,2)*-semi-open. Therefore X is the only (1,2)*-semi-open set containing $\{x\}^c$. Therefore $\{x\}^c$ is (1,2)*-sg*-closed. Hence {x} is (1,2)*-sg*-open. Therefore {x} is (1,2)*-semi-closed or (1,2)*-sg*-open.

4.(1,2)*-Semi-Generalised-Star Homeomorphisms

Definition 4.1

A function f: X → Y is called a (1,2)*-closed if f(V) ∈ (1,2)*-C(Y) for every $\tau_{1,2}$-closed set V in X.

Theorem 4.2

A function f : X → Y is (1,2)*-closed if and only if $\sigma_{1,2}$-cl[f(A)] ⊆ f [$\tau_{1,2}$-cl(A)] for every A ⊆ X.

Proof:

Let f be (1,2)*-closed and let A ⊆ X. Then f [$\tau_{1,2}$-cl(A)] ∈ (1,2)*-C(Y). But f(A) ⊆ f [$\tau_{1,2}$-cl(A)]. Then $\sigma_{1,2}$-cl[f(A)] ⊆ f [$\tau_{1,2}$-cl(A)]. Conversely, let A ⊆ X be a $\tau_{1,2}$-closed set. Then by assumption, $\sigma_{1,2}$-cl[f(A)] ⊆ f [$\tau_{1,2}$-cl(A)] = f(A). This shows that f(A) ∈ (1,2)*-C(Y). Hence f is (1,2)*-closed.

Definition 4.3

A function f: X → Y is called a (1,2)*-sg*-continuous if f^{-1} (V) is (1,2)*-sg*-closed in X for every $\sigma_{1,2}$-closed set V of Y.

Theorem 4.4

Let f: X → Y be a (1,2)*-homeomorphism. Then a subset A is (1,2)*-sg*-closed in Y ⇒ f^{-1} (A) is (1,2)*-sg*-closed in X.

Proof:

Let f: X → Y be a (1,2)*-homeomorphism. Let A be a (1,2)*-sg*-closed subset of Y. Let B = f^{-1} (A). Now to prove that B is (1,2)*-sg*-closed in X. Let U be any (1,2)*-semi-open set with B ⊂ U. Then f(B) ⊂ f(U). Therefore f(f^{-1} (A)) ⊂ f(U). Since f is (1,2)*-bijective, f(f^{-1} (A)) = A. Therefore A ⊂ f(U). We claim that f(U) is (1,2)*-semi-open. Since U is (1,2)*-semi-open, U ⊂ $\tau_{1,2}$-cl($\tau_{1,2}$-int(U)). Then f(U) ⊂ f($\tau_{1,2}$-cl($\tau_{1,2}$-int(U))) ⊂ $\sigma_{1,2}$-cl(f($\tau_{1,2}$-int(U))), since f is (1,2)*-continuous and f(U) ⊂ $\sigma_{1,2}$-cl($\sigma_{1,2}$-int f(U)) since f is (1,2)*-open. Therefore f(U) is (1,2)*-semi-open. Since A ⊂ f(U), f(U) is (1,2)*-semi-open and A is (1,2)*-sg*-closed. Therefore $\sigma_{1,2}$-cl(A) ⊂ f(U). Hence f^{-1}($\sigma_{1,2}$-cl(A)) ⊂ f^{-1} (f(U)). Since f is a (1,2)*-homeomorphism, f^{-1}($\sigma_{1,2}$-cl(A)) = $\tau_{1,2}$-cl(f^{-1} (A)). Therefore $\tau_{1,2}$-cl (f^{-1}(A)) ⊂ f^{-1}(f(U)). Therefore $\tau_{1,2}$-cl(B) ⊂ U. It means B is (1,2)*-sg*-closed in X. Hence f^{-1} (A) is (1,2)*-sg*-closed.

Theorem 4.5

Let f: X → Y be a (1,2)*-homeomorphism. A subset A is (1,2)*-sg*-open in Y ⇒ f^{-1} (A) is (1,2)*-sg*-open in X.

Proof:

A is (1,2)*-sg*-open in Y ⇒ A^c is (1,2)*-sg*-closed in Y ⇒ f^{-1} (A^c) is (1,2)*-sg*-closed in X ⇒ [f^{-1} (A)]^c is (1,2)*-sg*-closed in X ⇒ f^{-1} (A) is (1,2)*-sg*-open in X.

Theorem 4.6

Let f: X → Y be a (1,2)*-homeomorphism. A subset A is (1,2)*-sg*-closed in X ⇒ f(A) is (1,2)*-sg*-closed in Y.

Proof:

Let f: X → Y be a (1,2)*-homeomorphism. Assume that A is (1,2)*-sg*-closed in X. Let B = f(A). Now to prove that B is (1,2)*-sg*-closed in Y. Let U be a (1,2)*-semi-open set with B ⊂ U. That is f(A) ⊂ U. Hence f^{-1} (f(A)) ⊂ f^{-1} (U). Since f is (1,2)*-bijective, f^{-1}(f(A)) = A. Therefore A ⊂ f^{-1} (U). Since U is (1,2)*-semi-open and f is a (1,2)*-homeomorphism, f^{-1} (U) is (1,2)*-semi-open, we have A ⊂ f^{-1}(U), f^{-1} (U) is (1,2)*-semi-open and A is (1,2)*-sg*-closed. Therefore $\tau_{1,2}$-cl(A) ⊂ f^{-1} (U). Hence f($\tau_{1,2}$-cl(A)) ⊂ f(f^{-1} (U)). Since f is a (1,2)*-closed map, $\sigma_{1,2}$-cl(f(A)) ⊂ f($\tau_{1,2}$-cl(A)). Therefore $\sigma_{1,2}$-cl(f(A)) ⊂ f[f^{-1} (U)]. Hence $\sigma_{1,2}$-cl(B) ⊂ U. It means B is (1,2)*-sg*-closed in Y. Therefore image of a (1,2)*-sg*-closed set is (1,2)*-sg*-closed.

Theorem 4.7

Let f: X → Y be a (1,2)*-homeomorphism. A is (1,2)*-sg*-open in X ⇒ f(A) is (1,2)*-sg*-open in Y.

Proof:

A is (1,2)*-sg*-open in X ⇒ A^c is (1,2)*-sg*-closed in X.
⇒ f(A^c) is (1,2)*-sg*-closed in Y.
⇒ [f(A)]^c is (1,2)*-sg*-closed in Y.
⇒f(A) is (1,2)*-sg*-open in Y.

Definition 4.8

Let X and Y be two bitopological spaces. A map f: X → Y is called a pre (1,2)*-sg*-closed if for each (1,2)*-sg*-closed set A in X, f(A) is (1,2)*-sg*-closed in Y.

Theorem 4.9

Every (1,2)*-homeomorphism is a pre (1,2)*-sg*-closed map.

Proof :

It follows from Theorem 4.6.

Definition 4.10

Let X and Y be two bitopological spaces. A map $f : X \to Y$ is called a pre- (1,2)*-sg*-open if for each (1,2)*-sg*-open set A in X, f(A) is (1,2)*-sg*-open in Y.

Theorem 4.11

Every (1,2)*-homeomorphism is a pre-(1,2)*-sg*-open map.

Proof:

It follows from Theorem 4.7.

Definition 4.12

Let X and Y be two bitopological spaces. A map $f : X \to Y$ is called (1,2)*-sg*-irresolute if for each (1,2)*-sg*-closed set A in Y, $f^{-1}(A)$ is (1,2)*-sg*-closed in X.

Theorem 4.13

Every (1,2)*-homeomorphism is (1,2)*-sg*-irresolute map.

Proof:

It follows from Theorem 4.4.

Theorem 4.14

$f: X \to Y$ is (1,2)*-sg*-irresolute if and only if inverse image of every (1,2)*-sg*-open set in Y is (1,2)*-sg*-open in X.

Proof:

A is (1,2)*-sg*-open in Y $\Leftrightarrow A^c$ is (1,2)*-sg*-closed in Y

$\Leftrightarrow f^{-1}(A^c)$ is (1,2)*-sg*-closed in X.

$\Leftrightarrow [f^{-1}(A)]^c$ is (1,2)*-sg*-closed in X.

$\Leftrightarrow f^{-1}(A)$ is (1,2)*-sg*-open in X.

Definition 4.15

Let X and Y be two bitopological spaces. A map $f : X \to Y$ is called a (1,2)*-sg*-homeomorphism if f is (1,2)*-bijective, f is (1,2)*-sg*-irresolute and f^{-1} is (1,2)*-sg*-irresolute.

Theorem 4.16

If $f: X \to Y$ is (1,2)*-bijective, then the following are equivalent.

1. f is (1,2)*-sg*-irresolute and f is pre-(1,2)*-sg*-closed.
2. f is (1,2)*-sg*-irresolute and f is pre-(1,2)*-sg*-open.
3. f is (1,2)*-sg*-homeomorphism.

Proof:

1 $\Rightarrow$ 2. We have $f: X \to Y$ is (1,2)*-bijective, f is (1,2)*-sg*- irresolute and f is pre-(1,2)*-sg*-closed. Since f is pre-(1,2)*-sg*-closed, A is (1,2)*-sg*-open in X $\Rightarrow A^c$ is (1,2)*-sg*-closed in X.

$\Rightarrow f(A^c)$ is (1,2)*-sg*-closed in Y.

$\Rightarrow [f(A)]^c$ is (1,2)*-sg*-closed in Y.

$\Rightarrow f(A)$ is (1,2)*-sg*-open in Y.

Hence f is a pre-(1,2)*-sg*-open map.

2 $\Rightarrow$ 3. We have $f: X \to Y$ is (1,2)*-bijective, f is (1,2)*-sg*-irresolute and f is pre-(1,2)*-sg*-open. Since f is pre-(1,2)*-sg*-open, A is (1,2)*-sg*-open in X $\Rightarrow$ f(A) is (1,2)*-sg*-open in Y $\Rightarrow (f^{-1})^{-1}(A)$ is (1,2)*-sg*-open in Y.

Hence f^{-1} is (1,2)*-sg*-irresolute. Hence f is a (1,2)*-sg*-homeomorphism.

3 $\Rightarrow$ 1. We have $f: X \to Y$ is (1,2)*-bijective, f is (1,2)*-sg*-irresolute and f^{-1} is (1,2)*-sg*-irresolute. Now f^{-1} is (1,2)*-sg*-irresolute $\Rightarrow$ f is pre-(1,2)*-sg*-closed.

Definition 4.17

Let X and Y be two bitopological spaces. A map $f : X \to Y$ is called (1,2)*-sg*-closed map if for each $\tau_{1,2}$-closed set F of X, f(F) is (1,2)*-sg*-closed in Y.

Definition 4.18

Let X and Y be two bitopological spaces. A map $f: X \to Y$ is called (1,2)*-sg*-open if for each $\tau_{1,2}$-open set U of X, f(U) is (1,2)*-sg*-open in Y.

Theorem 4.19

Every (1,2)*-homeomorphism is a (1,2)*-sg*-homeomorphism.

Proof:

It follows from the fact that every (1,2)*-continuous map is (1,2)*-sg*-continuous [10] and every (1,2)*-open map is a (1,2)*-sg*-open map [11].

Remark 4.20

The converse of Theorem 4.19 need not be true as shown in the following example.

Example 4.21

Let X = {a, b, c} = Y, τ_1 = {φ, X, {a}, {a, b}} and τ_2 = {φ, X, {b}}. Then the sets in {φ, X, {a}, {b}, {a, b}} are $\tau_{1,2}$-

open. Let $\sigma_1 = \{\varphi, Y, \{a\}\}$ and $\sigma_2 = \{\varphi, Y, \{a, b\}\}$. Then the sets in $\{\varphi, Y, \{a\}, \{a, b\}\}$ are $\sigma_{1,2}$-open. Define $f : X \rightarrow Y$ by $f(a) = a$, $f(b) = b$ and $f(c) = c$. Clearly f is not a (1,2)*-homeomorphism since f is not an (1,2)*-open map. However f is (1,2)*-sg*-homeomorphism.

Theorem 4.22

If $f : X \rightarrow Y$ is an (1,2)*-irresolute (1,2)*-closed map, then F is (1,2)*-sg*-closed in X $\Rightarrow$ f(F) is (1,2)*-sg*-closed in Y.

Proof:

Let F be a (1,2)*-sg*-closed subset of X. Now to prove that f(F) is (1,2)*-sg*-closed in Y. Let $f(F) \subset U$ and U be (1,2)*-semi-open. Then $F \subset f^{-1}(U)$. Since U is (1,2)*-semi-open and f is (1,2)*-irresolute. Therefore $f^{-1}(U)$ is (1,2)*-semi-open. Since F is (1,2)*-sg*-closed, $F \subset f^{-1}(U)$ and $f^{-1}(U)$ is (1,2)*-semi-open, $\tau_{1,2}\text{-cl}(F) \subset f^{-1}(U)$. Hence $f(\tau_{1,2}\text{-cl}(F)) \subset f(f^{-1}(U)) \subset U$. Since f is a (1,2)*-closed map, $\sigma_{1,2}\text{-cl}(f(F)) \subset f(\tau_{1,2}\text{-cl}(F))$. Hence $\sigma_{1,2}\text{-cl}(f(F)) \subset U$. Therefore f(F) is (1,2)*-sg*-closed.

References

[1] N.Bourbaki, “General Topology”, Part I, Addison-Wesley, 1996.

[2] J.C.Kelly, “Bitopological spaces”, Proc. London Math. Soc. Vol.13, pp. 71-89, 1963.

[3] M.Lellis Thivagar, M.Margaret Nirmala, R.Raja Rajeshwari and E.Ekici, “A Note on (1,2)-GPR-closed sets”, Math.Maced Vol.4, pp. 33-42, 2006.

[4] M. Murugalingam, “A Study of Semi-Generalized Topology”, Ph.D. Thesis, Manonmaniam Sundaranar University Tirunelveli, Tamil Nadu, 2005.

[5] O.Ravi, M.Lellis Thivagar and E.Ekici, “On (1,2)*-sets and decompositions of bitopological (1,2)*-continuous mappings”, Kochi J.Math., Vol. 3, pp. 181-189, 2008.

[6] O.Ravi, M.Lellis Thivagar and E.Hatir, “Decomposition of (1,2)*-continuity and (1,2)*-α-Continuity”, Miskolc Mathematical notes, Vol. 10, No. 2, pp. 163-171, 2009.

[7] O.Ravi, K. Mahaboob Hassain Sherieff and M.Krishna Moorthy, “On decompositions of bitopological (1,2)*-A-continuity” (To appear in International Journal of Computer Science and emerging Technologies).

[8] O.Ravi, G.Ram Kumar and M.Krishna Moorthy, “Decompositions of (1,2)*-α-continuity and (1,2)*-αgs-continuity” (To appear in International Journal of computational and applied mathematics).

[9] O.Ravi and M.Lellis Thivagar, “Remarks on λ-irresolute functions via (1,2)*-sets”, Advances in Applied Mathematical Analysis, Vol. 5, No. 1 pp. 1-15, 2010.

[10] O.Ravi and M.Lellis Thivagar, “A bitopological (1,2)*-semi-generalized continuous maps”, Bull. Malays. Math. Sci. Soc. Vol. 2, No. 29(1), pp. 79-88, 2006.

[11] O.Ravi, S.Pious Missier and T.Salai Parkunan, “On bitopological (1,2)*-generalized Homeomorphisms”, Internat. J. Contemp. Math. Sci. Vol 5, No. 11, pp. 543-557, 2010.

[12] O.Ravi, M. Lellis Thivagar, M.E.Abd El-Monsef, “Remarks on bitopological (1,2)*-quotient mappings”, J. Egypt Math. Soc. Vol. 16, No. 1, pp. 17-25, 2008.

[13] O.Ravi, M.Lellis Thivagar, M.Joseph Israel, K.Kayathri, “Mildly (1,2)*-Normal spaces and some bitopological functions”, Mathematica Bohemica Vol. 135, No. 1, pp. 1-13, 2010.

[14] O.Ravi, M.Lellis Thivagar, “On stronger forms of (1,2)*-quotient mappings in bitopological spaces”, Internat. J. Math. Game theory and Algebra. Vol. 14. No.6, pp. 481-492, 2004.

[15] M.K.R.S. Veerakumar, “ $\hat{g}$ -closed sets in topological spaces”, Bull. Allah. Math. Soc., Vol. 18, pp. 99-112, 2003.

Survey on - Can we make File Secure in Linux?

Shrinivas Karwa[1], Anand Bisen[2], Dr. B.B. Meshram[3]
Veermata Jijabai Technological Institute, Mumbai
shrikarwa1@gmail.com[1], anandbisen2008@gmail.com[2], bbmeshram@vjti.org.in[3]

Abstract: ***With the rapid growing of internet and networks applications, data security becomes more important than ever before. The Computer is inevitable thing in our life. The most important thing in computer system is file management. It is very important to keep file safe. Existing cryptographic file systems for UNIX do not take into account that sensitive data must often be shared with other users, but still kept secret. Cryptographic technique gives a promising way to protect our files from unauthorized user. We have made the critical literature survey about cryptographic file systems, the design goals for file security and available solutions for securing file. Later on we compared the available cryptographic file systems with respect to mentioned design goals.***

***Keywords*: File Security, FUSE, Cryptography, vnode.**

1. Introduction

File System is the only module of the operating system that is most visible to the user. It deals with the easy storage and fast retrieval of the data without user actually knowing the details of the storage device.

Now a day's private information might be stored on the laptops or computers shared by several users. Humans have a great dependence on computer and network, and the security of computer related to the whole world and everybody. Unfortunately, these systems are usually susceptible to attack by techniques that circumvent the restrictions of such security features as logon authentication or file permission implemented by access control lists (ACLs). The attacker can start up a different operation system, or remove the hard drives and place them in another system. Then he can gain access to all the stored files. So sensitive data will certainly be exposed.

Data security in modern distributed computing systems is a difficult problem. Network connections and remote file system services, often make it possible for an intruder to gain access to sensitive data by compromising only a single component of a large system. Because of the difficulty of reliably protecting information, sensitive files are often not stored on networked computers, making access to them by authorized users inconvenient and putting them out of the reach of useful system services such as backup. (Of course, off line backups are themselves a security risk, since they make it difficult to destroy all copies of confidential data when they are no longer needed.)

Operating system provides a way to protect directories, folder and files through list of the user and user group that have permission to access these resources. Attributes can also be associated with directories, folder and files to manage access and support the creation of backups.

Cryptographic techniques offer a promising approach for protecting files against unauthorized access. When properly implemented and appropriately applied, modern cipher algorithms are widely believed sufficiently strong to render encrypted data unavailable to virtually any adversary who cannot supply the correct key.

The outline of this paper is as follows: **Section 2** discusses about the available cryptographic file system and how it provides file security; **Section 3** explains Design Goals for File security. Thereafter we discuss the available solution and comparison among the available cryptographic file system with respect to design goals in **Section 4** and **Section 5** and finally we conclude the paper in **Section 6**.

2. Literature Review

Following are the cryptographic file system available to securing file under Linux environment.

2.1. CFS (Cryptographic File System)

CFS is a cryptographic file system that is implemented as a user-level NFS server. It discusses the advantage and disadvantage of putting CFS at user-level and system-level. It lies in between user level and system level. An NFS layer implemented the encryption, decryption, and key management locally on a trusted client: files were encrypted while in transit between the trusted component the untrusted component. It uses DES to encrypt the contents of file. All the encrypted files are mounted under **/crypt** directory and can be seen in clear text form in that directory. [4]

2.2. TCFS (Transparent Cryptographic File System)

It improves on the CFS design by removing the NFS client encryption layer. TCFS requires the installation of modules and tools on the client, as well as a special attributes daemon on the server. TCFS is a modified client-side NFS that communicates with a remote NFS server. It gives user the choice to choose the cryptographic algorithm of his/ her choice to encrypt the data. [5]

2.3. CryptFS

Cryptfs is implemented as a kernel-resident file system. Cryptfs is implemented as a stackable vnode interface. A Virtual Node or "vnode" is a data structure used within Unix-based operating systems to represent an open file, directory, device, or other entity (e.g., socket) that can appear in the file system name-space. It uses UID and Session ID combinations in order ensure Key Management. It uses Blowfish cryptographic Algorithm to encrypt the data as it is simple, fast and compact. Files name are also encrypted so that the attacker is not able to guess the type of data that file contains. For encrypting File names it converts every 3 byte encrypted sequence into a 4 byte sequence of ASCII characters. [6]

2.4. encFS

EncFS is free FUSE-based cryptographic file system that transparently encrypts files, using an arbitrary directory as storage for the encrypted files. Two directories are involved in mounting an EncFS filesystem: the source directory, and the mountpoint. Each file in the mountpoint has a specific file in the source directory that corresponds to it. The file in the mountpoint provides the unencrypted view of the one in the source directory. Files are encrypted using a volume key, which is stored encrypted in the source directory. A password is used to decrypt this key. Filenames are encrypted in the source directory. It uses AES for encryption and decryption. Filenames in the source directory can be encrypted in block or stream mode. [11]

2.5. Encrypt-FS

Encrypt-FS uses symmetric and Asymmetric Cryptography to provide security. It works on Linux Virtual File System (VFS) layer. It uses two types of cryptographic algorithms to encrypt a file, 2 kinds of encryption keys exist in Encrypt-FS: FEK and UEK. FEK is called File Encryption Key and is used to encrypt the content of file. Many block cryptographic algorithm are available to encrypt the file content. FEK is randomly generated by Encrypt-FS. Each encrypted file has an exclusive and different FEK, that's makes attacker to guess FEK.UEK is User Encryption Key. Some time encrypted files may be shared among several users, it is not wise to tell the FEK to each person. Encrypt-FS encrypt FEK using public key cryptography and is stored with the file itself. It uses public key cryptography for encrypting FEK. [10]

3. Design Goals for File Security

Following are the Design goals that should be taken into consideration for designing robust and reliable cryptographic file system.

3.1. Read/ Write Control (RW)

This design goal will take care that only Authorized User should be able to Read and Write content to secured Directory. Only AU should be able to change permission related to secure Directory. If an unauthorized user tries to read or write content of secured file he should be denied from doing this.

3.2. Modification (delete, rename) Protect (MP)

This design goal is about protecting the secured directory from modification. After protecting no one should be able to delete or rename file. Only the user who has created file should be able to delete or rename the file.

3.3. Anti-copy Protect (ACP)

No one should be able to copy the file. It may happen that attacker can collect many encrypted files and try to decrypt it to get the actual content by taking the file to other operating system.

3.4. Integrity Protect (IP)

No one should be able to change the content of secured file. Only the user who has the right to make changes to file should be allowed to make changes to its contents.

3.5. Data Encrypt (DE)

This design goal is about making the content of file unreadable (Human shouldn't understand that). Encryption algorithm used to encrypt file content should be fast, secure, and compact. Along with file data encryption, file name and directory name should be encrypted which will create confusion and diffusion.

3.6. Application Transparent (AT)

User should not know that the other application is used to secure its data. He should feel like working in the same environment.

3.7. Key Management (KM)

Cryptographic systems restrict access to sensitive information through knowledge of the keys used to encrypt the data. Clearly, to be of any use at all, a system must have some way of obtaining the key from the user. But this need not be intrusive; encryption keys should not have to be supplied more than once per session.

3.8. Protection of Sensitive Meta-Data (PSM)

Considerable information can often be derived from a file system's structural data; these should be protected to the extent possible. In particular, file names should not be visible without the key. User often chooses comfortable file and directory names describing the nature of the data stored within. An attacker who discovers the names of the file even if they cannot access the content of the file data; but can refer the nature of the data itself.

3.9. Portability (PO)

The encryption system should exploit existing interfaces wherever possible and should not rely on unusual or special-purpose system features. Furthermore, encrypted files should be portable between implementations; files should be usable wherever the key is supplied.

3.10. Concurrent Access (CA)

It should be possible for several users (or processes) to have access to the same encrypted files simultaneously.

4. Available Solutions

Following are the ways in which you can implement a file system to make it secure.

4.1. Modify following System Calls and related Code (MSC) [1,2]

Changing the system calls and related files require kernel recompilation that is very time consuming process.

We may have to change the call like read(): Read file, open(): Open File, close(): Close File, chmod(): Change File attribute, write(): Write data to file, rm(): Remove file, rmdir(): remove Directory, ls(): List content of file/ directory, rename(): rename file, copy(): copy file/ directory, move(): move file/ directory

Modifying system call and related code requires deep knowledge of kernel programming and illogical changes may lead to kernel panic.

4.2. Use FUSE (FUSE) [12, 13, 14, 15]

Filesystem in Userspace (FUSE) is a hybrid approach that consists of two parts:

1) A standard kernel-level file system which passes calls to a user-level demon, and
2) A library to easily develop file system-specific FUSE demons.

It also implements a special-purpose device which can be opened by a user-space process. It then spends its time accepting filesystem requests, translating them into its own protocol, and sending them out via the device interface. Responses to requests come back from user space via the FUSE device, and are translated back into the form expected by the kernel.

With FUSE it is possible to implement a fully functional file system in a userspace program. Features include: Simple library API, Simple installation (no need to patch or recompile the kernel), Secure implementation, Userspace - kernel interface is very efficient and Usable by non privileged users.

By using FUSE all the system call that user indirectly calls them by executing commands on directory mounted using FUSE goes thorough FUSE code. This allows users to do additional tasks like decrypt and encrypt the directory while mounting and un-mounting. Thus user can provide security using FUSE. Running a FUSE code doesn't require kernel compilation and even if a something gets corrupted while doing operation using FUSE it's only at the user level doesn't affect kernel level code.

FUSE performance is limited due to crossing the user-kernel boundary.

4.3. Add Vnode (VNODE) [3]

Vnode provided a layer of abstraction that separates the core of the OS from file systems. Each file is represented in memory by a vnode. A vnode has an operations vector that defines several operations that the OS can call, thereby allowing the OS to add and remove types of file systems at runtime. Most of the Operating systems use something similar to the vnode interface, and the number of file systems supported by the OS has grown accordingly.

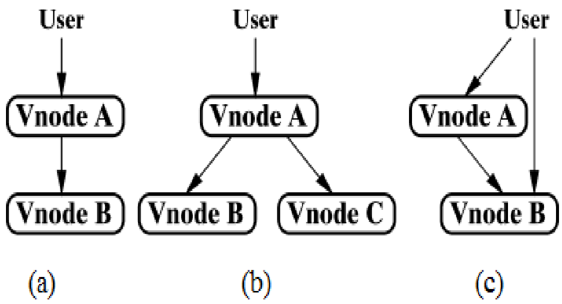


Figure 1: Types of layering
(a) Linear (b) Fan-out and (c) Fan-in

Types of layering:

1) **Linear:** In a linear layer all operations are intercepted by the top-level vnode, A, and A passes it to a single vnode below it.
2) **Fan-out:** In fan-out, all operations go through the top-level vnode, A, but there are two or more lower-level vnode.
3) **Fan-in:** In fan in, some operations go through the top-level vnode, A, before going to the lower-level vnode, B, but some operations can bypass A and directly access B.

VNODE level encrypted file system can be implemented on modern operating systems without having to change the rest of the system.

5. Comparison of Available Cryptographic File System

The following table shows the comparison of available Cryptographic File Systems discussed in **Section 2** with respect to design goals discussed in **Section 3**.

Table1: Comparison of available Cryptographic File Systems

File System	RW	MP	ACP	IP	DE	AT	KM	PSM	PO	CA	Implementation
CFS	√				√	√			√	√	MSC
TCFS	√				√	√	√		√		MSC
CryptFS	√	√			√	√			√		VNODE
encFS	√				√	√	√		√	√	FUSE
EncryptFS	√				√	√	√				MSC

Read/ Write Control **(RW)**, Modification (delete, rename) Protect **(MP)**, Anti-copy Protect **(ACP)**, Integrity Protect **(IP)**, Data Encrypt **(DE)**, Application Transparent **(AT)**, Key Management **(KM)**, Protection of Sensitive Meta-Data **(PSM)**, Portability **(PO)**, Concurrent Access **(CA)**.

6. Conclusion

This paper puts light on available cryptographic file systems and how much it satisfies the design goals. In general, user-level extensions are easy to implement, but their performance is not as good as kernel extensions because that former involve data copies between the user-level and kernel-level, as well as additional context switches. As shown in table1 no cryptographic file system satisfies all the design goals. We are trying to implement a robust and reliable cryptographic file system which satisfies the maximum design goals.

References

[1] William Stallings. "Operating Systems: Internals and Design Principles", Prentice Hall, Fifth Edition

[2] Michael Palmer, "Guide to Operating System Security", CENGAGE Learning, Second Edition.

[3] Ion Badulescu and Erez Zadok. "A Stackable File system Interface for Linux", LinuxExpo Conference Proceedings in 1990

[4] M. Blaze. A Cryptographic Filesystem for UNIX. In Proceedings of the First ACM Conference on Computer and Communications Security, pp. 9-16, November 1993

[5] G. Cattaneo and G. Persiano. Design and Implementation of a Transparent Cryptographic Filesystern for Unix. Unpublished Technical Report, July 1997.

[6] Erez Zadok, Ion Badulescu and Alex Shender. "CryptFS: A Stackable Vnode Level Encryption File System", Technical Report CUCS-021-98, June 1998

[7] Fauzan Saeed, Mustafa Rashid, Integrating Classical Encryption with Modern Technique, Vol. 10 No. 5 pp. 280-285, May 2010

[8] Meyers, R.K.; Desoky, A.H. "An Implementation of the Blowfish Cryptosystem," Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE International Symposium on , vol., no., pp.346-351, 16-19 Dec. 2008

[9] Erez Zadok, Rakesh Iyer, Nikolai Joukov etc. On Incremental File System Development. ACM Transactions on Storage, Vol.2, No. 3, August 2006: 1-33.

[10] "EncryptFS: A Versatile Cryptographic File System for Linux", http://pompone.cs.ucsb.edu/~wei/EncryptFS.pdf

[11] Valient Gough. "EncFS", Libre Software Meeting, France, 2005. http://www.arg0.net/encfsintro

[12] "FUSE operations", http://www.soe.ucsc.edu/~aneeman/FUSE_how-to.html

[13] "Introduction to FUSE and Working of FUSE", http://fuse.sourceforge.net/

[14] "Implementation of Simple File System Using FUSE", http://fuse.sourceforge.net/helloworld .html

[15] "FUSE Documentation", http://www.prism.uvsq.fr

Author Biographies

Shrinivas Karwa received B.E. degree from Sinhgad College of Engineering, Pune in 2008 and currently pursuing M.Tech from Veermat Jijabai Technological Institute, Mumbai. His current area of research includes Network Security, Cryptography, and Operating system. He is RedHat Certified Engineer (RHCE).

Anand Bisen received B.E. degree from Laxmi Narain College of Technology, Bhopal in 2009 and currently pursuing M.Tech from Veermat Jijabai Technological Institute, Mumbai. His current research includes Network Security and Databases.

Dr. B. B. Meshram is Professor and Head of Computer Technology Department at VJTI, Mumbai. He received bachelor, master's and doctoral Degree in computer Engineering. He has participated in more than 16 refresher courses to meet the needs of current technology. He has chaired more than 10 AICTE STTP programs. He has received the appreciation at Manchester and Cardip University, UK. He has contributed more than 75 research paper at National and International Journal. His current research interest is Databases, Data Warehouse and Network Security.

Comparison of Two Proactive Protocols: OLSR and TBRPF using the RNS (Relay Node Set) Framework

V.Umadevi Chezhian
Lecturer, College of Business and Economics, Asmara, State of Eritrea
yazh1999@gmail.com

Kaliyaperumal Karthikeyan
Lecturer, Eritrea Institute of Technology, Asmara, State of Eritrea.
kirithicraj@gmail.com

Thanappan Subash
Department of Civil Engineering, Eritrea Institute of Technology, State of Eritrea
Email: thanappansubash@gmail.com

Abstract: Past study of MANET routing protocols focused on designing new protocols, comparing existing protocols, or improving protocols before standard MANET routing protocols are established. Researchers have studied these protocols using simulations of arbitrary networks with certain traffic profiles. Due to the lack of consistent characterization of different MANET protocols, prior simulation experiments are not well designed. Some protocols that perform well, in terms of control overhead or throughput, in some scenarios may have poor performance under other conditions. Therefore, the conclusions based on these simulations cannot be generalized. These efforts can be aided by a framework that can characterize MANET routing protocols. We can describe MANET routing protocols with the RNS framework so that researchers can understand the protocols more easily. This framework characterizes different MANET routing protocols and highlights the internal relationships among different protocols. Quantitative models based on the RNS framework can be used to identify factors that affect control overhead for different MANET routing protocols. The framework allows comparison of routing protocols by analytical models coupled with network parameters and traffic profiles. Possible ideas for improving proposed MANET routing protocols can be found using the RNS framework. The RNS framework and the corresponding quantitative model can aid the design, evaluation, and validation of new MANET routing protocols with emphasis on control overhead. In this chapter, we concentrate on Comparison of Two Proactive Protocols: OLSR and TBRPF using the RNS (Relay Node Set) Framework. Based on the results and assumptions, OLSR usually has larger overhead in the maintenance module than TBRPF.

Key Words: RNS, MANET Protocols, Overhead, OLSR, TBRPF.

0.1 Introduction

The Optimized Link State Routing (OLSR) protocol [1] is a proactive link state routing protocol for MANETs. One key idea is to reduce control overhead by reducing the number of broadcasts as compared with pure flooding mechanisms. The basic concept to support this idea in OLSR is the use of multipoint relays (MPRs) [1], [2]. The latency for OLSR has the highest values from *1* to 10 hops, and generally the highest slope. This indicates that OLSR has difficulty scaling to hop count in this scenario. As a proactive protocol, we would expect OLSR to have lower average latency than a reactive AODV or DSR [3]. OLSR has fairly uniform control overhead, **as** expected from a proactive protocol. **It** trends downwards with sparse networks because there are fewer links to report. But since there are fewer links, route convergence takes longer[3]. The optimized link state protocol (OLSR) [4] utilizes a multicast-like mechanism (called "multipoint relay") to reduce the amount of traffic produced by the periodic topology updates. This has the potential for performing well on smaller ad hoc networks.

OLSR is designed to reduce duplicate retransmission in the same region. The routes are always immediately available when needed due to its proactive nature. Hop by hop routing is used in forwarding packets in OLSR, only nodes selected as MPRs forward control traffic that causes reducing the size of control message and minimizing the overhead from flooding control traffic.[5],[6].

The overall performance of OLSR was very good when mobile nodes movement was changing over varying time. OLSR has high control traffic as compared to TORA as it searches for routes to destination more frequently. Despite the other routing protocols, OLSR protocol showed increase in throughput even when the routing load was increased. We have analyzed that all routing protocol successfully delivers data when subjected to different network stresses and topology changes [7].

The Topology Broadcast Based on Reverse-Path Forwarding (TBRPF) protocol is another link state, proactive routing protocol for MANETs [8]. Each router running TBRPF computes a source tree to all reachable destinations based on partial topology information stored locally. The source tree is also known as the shortest path tree. To reduce overhead, routers in TBRPF only broadcast part of their source tree to neighbors. The partial source tree is called the reportable tree. The main idea of sharing reportable trees with neighbors comes from the Partial Tree-Sharing Protocol (PTSP) described in [9]. Basically, in the local copy of network topology, a link cost is equal to the actual value if this link is in the shortest path tree. Otherwise, the cost is equal to or greater than the real value. The procedure to generate a reportable tree at a router is as follows. Links that are in this router's shortest path tree are checked. If such a link is estimated to be in the neighbors' shortest path trees, it is added to the reportable tree. Note that the estimated results may not be correct, but they do include the correct link costs. TBRPF is said to work better in dense networks [8].

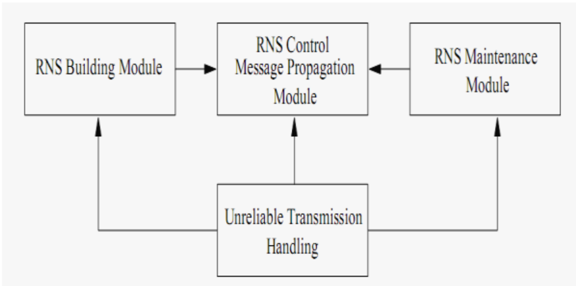


Fig. 1: Four Modules in RNS Framework

RNS Frame work has four modules, according to the four-module RNS framework, the total control overhead for a MANET routing protocol is formed by four overhead components: (i) the overhead to build or rebuild the RNS, (ii) the overhead to maintain the RNS, (iii) the overhead to propagate control messages in the RNS, and (iv) the overhead to handle unreliable transmissions.

Overhead is the number of packets generated by the routing protocols during the simulation, formally speaking it is:

$$overhead = \sum_{i=1}^{n} overhead_i \quad \text{..........(1)}$$

Where overhead *i* is the control packets number generated by node i. The generation of an important overhead will decrease the protocols performance. Although control packets are essential to ensure protocols functioning, their number should be as less as possible [10].

0.2 Analysis of OLSR with the RNS Framework

OLSR is a typical proactive routing protocol. OLSR uses periodic "hello" messages to exchange neighbor lists between neighboring nodes. An MPR node set is a small subset of neighboring nodes that covers all of the center node's two-hop neighbors and may rebroadcast any control message generated or forwarded by that center node. Information about MPR sets is also sent to neighbors via the "hello" messages. All nodes generate their own MPR selector (MPRS) sets. The MPRS set for a node is the set of neighboring nodes that select this node as a member of their MPR sets. Only nodes with non-empty MPRS sets broadcast control messages containing their MPRS sets. Generally, a node re-broadcasts a first-received control message sent by its neighbor if and only if this neighbor selects it as one of the neighbor's MPR nodes. Note that in the propagation procedure, if a node i is in the MPR node set for another node, say node j, and node i already received the broadcast control message originated by a certain initiator from a third node, say node k, before it receives this message from node j, node i keeps silent. In other words, node i is not included in the RNS, which reflects the first-seen rule for OLSR. The procedure described above is the RNS building procedure in OLSR. Therefore, the RNSs in OLSR are associated with certain source nodes and multiple copies of RNSs associated with different initiators can co-exist at any given time.

The sizes of different RNS sets may not be the same. The reason that these RNS sets may have different sizes is due to the distributed selection algorithm of MPR nodes, which is performed by neighbor nodes and may not be consistent among neighboring nodes. Each node rebroadcasts control messages sent by nodes that are in its MPRS set which have not been seen before. Therefore, information about all MPRS sets can be propagated to all nodes in the network with a small number of retransmissions. The total overhead in OLSR is shown in Equation (2).

$$Overhead = O_{construction} + O_{maintenance} + O_{propagation}$$

$$O_{construction} = \sum_{i=1}^{Nhell0} \sum_{i=1}^{N} P_{i.j.hello}$$

$$O_{maintenance} = \sum_{i=1}^{Nm} \sum_{j=1}^{Ni.adjust} (P_{i.j.MPRS} \times S_{i.j.RNS})$$

$$O_{propagation} = \sum_{i=1}^{Nupdate} \sum_{j=1}^{Ni.MPRS} (P_{i.j.MPRS} \times S_{i.j.RNS}) \quad \ldots\ldots\ldots (2)$$

0.3 Analysis of TBRPF with the RNS Framework

TBRPF is also a proactive routing protocol that provides shortest path routing. Each node uses periodic "hello" messages to detect links to its neighbors. Based on the local link state database, each node first builds a shortest path tree to all possible destinations. A node decides whether or not to report links in its shortest path tree to its neighbors by an estimation algorithm based on its local link state database. Information that is shared with a neighbor is considered to be reportable. Basically, a neighbor node is added to a reportable node set if this node has at least one neighbor which is not connected to this neighbor. Links in the shortest path tree are added to a reportable link set if one end point is in the reportable node set or one adjacent link that does not connect to the center node and is included in the reportable link set. Therefore, the reportable link set in the shortest path tree form a reportable tree. Each node broadcasts its reportable tree. This is the RNS building module in TBRPF. For any link, there is a set of nodes that broadcasts that link to neighboring nodes. Therefore, an RNS is built for each link in TBRPF. The control messages sent in TBRPF are reportable trees. Nodes have enough information to build proper shortest path trees based on reportable trees from neighbors.

When the topology changes, the maintenance module uses online computation to update corresponding RNSs and the link state update is propagated to all related nodes in the associated RNSs. Similar to OLSR, TBRPF uses periodic broadcast messages to handle unreliable transmissions. The overhead for TBRPF is presented in Equation (3)

$$Overhead = O_{construction} + O_{maintenance} + O_{propagation}$$

$$O_{construction} = \sum_{i=1}^{Nhell0} \sum_{i=1}^{N} P_{i.j.hello}$$

$$O_{maintenance} = \sum_{I=1}^{Nm} \sum_{j=1}^{Si,RNS} P_{i.j.LS}$$

$$O_{propagation} = \sum_{I=1}^{Nupdate} \sum_{j=1}^{Ei} \sum_{k=1}^{S.i.j.RNS} Pi.j.k.LS \qquad \text{----------(3)}$$

0.4 Comparison of Two Proactive Protocols: OLSR and TBRPF

We discuss using the RNS framework to compare two protocols in this section. There are some schemes proposed in the literature to improve MANET routing protocols in terms of control overhead. Those schemes can be derived from the analytical model based on the RNS framework. For example, Perkins, et al. describes an effort to reduce the range of the RNS built when a route request is sent [11]. Perkins, et al. [11] and Johnson, et al. [12] incorporate routing caches to reduce the propagation range of control messages. It can be seen from the analytical model that these approaches are trying to limit the size of RNS in the RNS construction operation. In other words, reducing the blind broadcast range reduces the control overhead. Moreover, this analytical model can also guide us to improve MANET routing protocols in other ways. Since little research has been done to improve proactive routing protocols in terms of control overhead, the following paragraphs give such an example with OLSR.

OLSR uses "hello" messages not only to detect link connections, but also to exchange MPR information. So the overhead of "hello" messages in OLSR is larger than that of TBRPF. The packet size of $P_{i\,j,MPRS}$ is formed by a header (MAC layer header and an IP header) and a data payload. Equation (4) illustrates the calculation of the packet size. Here, we assume that P_{unit} is the basic unit to describe a four-byte node ID. We can assume $P_{i,j,MPRS}$ equals the size of packet header (MAC and IP headers) plus several basic units to describe the MPRS. The link description packet used in TBRPF, defined as $P_{i,LS,}$, shares one header since a node broadcasts a reportable tree. Each link needs at most two IDs, each of size $P_{unit.}$. Some links with common nodes can lead to smaller packet sizes. This yields the upper bound shown in Equation (4).

$$P_{i.j.MPRS} = P_{header} + (|MPRS| + 1) \times P_{unit}$$

$$P_{i.LS} \le P_{header} / S_{reportable.tree.i} + 2P_{unit} \le P_{header} + 2\,P_{unit}. \qquad \text{……….(4)}$$

In the RNS maintenance module, the overhead for OLSR, shown in Equation (2), equals the sum of products of $P_{i\,j,MPRS}$ and $S_{i,j,RNS.}$. Generally, if $P_{i\,j,MPRS}$ increases, $S_{i,j,RNS}$ will also increase. Therefore, we can assume that the covariance between these two variables is greater than zero. Now, we have the lower bound for the OLSR overhead for the RNS maintenance module, shown in Equation(5) Based on Equations (4)and (5), the upper bound of the overhead for the RNS maintenance module for TBRPF can be formulated as shown in Equation (5)

$$\begin{aligned} O_{maint,OLSR} &\ge N_m \bullet \overline{N_{adjust}} \bullet \overline{S_{i,j,RNS}} \\ &= [P_{header} + (\overline{|MPRS|} + 1)P_{unit}]N_m\overline{N_{adjust}} \bullet \overline{S_{RNS,OLSR}} \\ O_{maint,TBRPF} &\le N_m \sum_{i=1}^{S_{i,RNS}} (P_{header} + 2P_{unit}) \\ &= N_m(P_{header} + 2P_{unit})\overline{R_{RNS,TBRPF}} \end{aligned} \qquad \text{…..(5)}$$

We used simulation to estimate the parameters for the size of the RNSs for these two protocols. Two nodes can communicate with each other if the distance between them is less than the given maximum radio range. The number of nodes ranged from 2 to 100. OLSR and TBRPF were simulated. The latest OLSR draft states that a node "should select an MPR set such that any two-hop neighbor is covered by at least MPR COVERAGE MPR nodes" [13]. We assume that the minimum MPR set is used in OLSR, i.e., MPR COVERAGE equals 1. We generated 1000 random connected topologies for each set of parameters and obtained the average size of RNSs. Results are shown in Figures 2,3,4 and 5 with radio ranges of 25, Note that we only show the average values in these figures. Since for a given number of nodes in a network, say N, the range for all possible RNS sizes is [0, N]. Therefore, the variance of results can be large. In other words, our comparison results only give an idea on the average performance and which protocol generates smaller RNSs really depends on the actual MANET application it applies to.

According to the simulation results, in most cases when the maximum radio ranges were 25 and 50, the average size of RNSs in OLSR is larger than in TBRPF. Therefore, based on these results and assumptions, OLSR usually has larger overhead in the maintenance module than TBRPF. According to the RNS framework, we can improve the first two modules in OLSR without increasing the overhead in the propagation module.

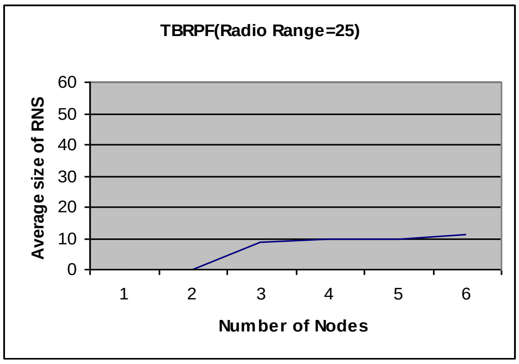


Fig 2: Average size of RNS versus number of nodes (Radio range=25) in TBRPF.

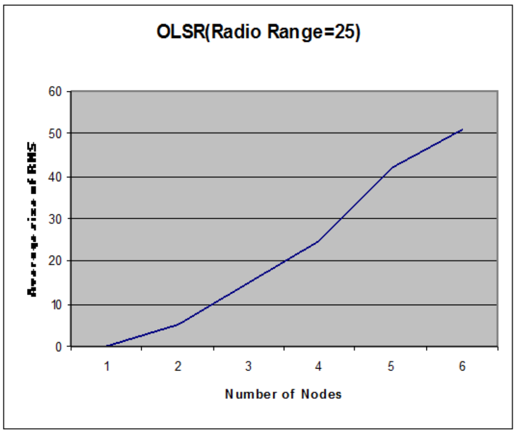


Fig 3: Average size of RNS versus number of nodes (Radio range=25) in OLSR.

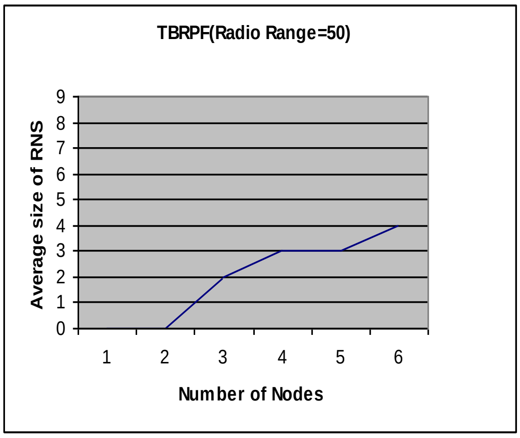


Fig 4: Average size of RNS versus number of nodes (Radio range=50) in TBRPF.

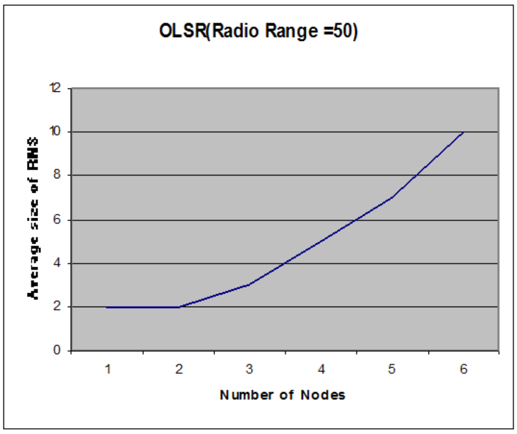


Fig 5: Average size of RNS versus number of nodes (Radio range=50) in OLSR.

0.5 Conclusion:

We presented a framework based on the concept of a relay node set that can characterize MANET routing protocols. We developed an analytical model with the RNS framework for control overhead for MANET routing protocols. Simple examples were used to show how we can compare and, possibly improve routing protocols using the RNS framework. There are some parameters defined in the analytical model that may not be measured directly for a MANET application. This is a limitation of using the RNS framework. One suggested approach is to use simulations or real-time measurements to estimate those values. This suggests a potential research topic for MANET routing protocols in which estimates of environmental parameters, including network and user application profiles, are used to adaptively choose different routing protocols or different sub-functions for one protocol. Here we compared OLSR and TBRPF using RNS. From the result we find out that OLSR usually has larger overhead in the maintenance module than TBRPF.

Reference:

[1] T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L. Viennot, “Optimized Link State Routing Protocol,” Internet Engineering Task Force (IETF) draft, March, 2002. http://www.ietf.org/internet-drafts/draftietf-manet-olsr-06.txt.

[2] A. Qayyum, L. Viennot, And A. Laouiti, “Multipoint Relaying: An Efficient Technique For flooding In Mobile Wireless Networks,” Research Report-3898, Inria, France, 2000.

[3]. Julian Hsu, Sameer Bhatia, Ken Tang Et.Al., “Performance Of Mobile Ad Hoc Networking Routing Protocols In Large Scale Scenarios“, Milcom 2004 - 2004 IEEE Military Communications Conference.

[4] P. Jacquet, P. Muhlethaler, and A. Qayyum, “Optimized link state routing protocol,” *IETF MANET*, Internet Draft, Nov. 1998.

[5]. “Optimized Link State Routing Protocol”, Internet draft, www.irtf.org.

[6]. http://en.wikipedia.org/wiki/OLSR.

[7]. Nadia Qasim, Fatin Said, Hamid Aghvami.,”Mobile Ad Hoc Networking Protocols’ Evaluation through Simulation for Quality of Service”, IAENG International Journal of Computer Science, 36:1, IJCS_36_1_10, February 2009.

[8] R. G. Ogier, F. L. Templin, B. Bellur, and M. G. Lewis, “Topology Broadcast Based on Reverse-Path Forwarding (TBRPF),” Internet Engineering Task Force (IETF) draft, November 2002. http://www.ietf.org/internet-drafts/draft-ietf-manettbrpf-06.txt.

[9] R. G. Ogier, “Efficient Routing Protocols for Packet-Radio Networks Based on Tree Sharing.” in Proc. of the 6th IEEE Intl. Workshop on Mobile Multimedia Communications, pp. 104-113, 1999.

[10]. Djamel et.al., “Ad Hoc Networks Routing Protocols and Mobility”, The International Journal of Information Technology, Vol.3. No.2, April 2006.

[11] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing,” Internet Engineering Task Force (IETF) draft, November 2002. http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-12.txt.

[12] D. B. Johnson, D. A. Maltz, Y. C. Hu, and J. G. Jetcheva, “The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR),” Internet Engineering Task Force (IETF) draft, Febuary 2002.

[13] T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L. Viennot, “Optimized Link State Routing Protocol,” Internet Engineering Task Force (IETF) draft, March, 2002.