Enhancing K-Means Algorithm with Initial Cluster Centers Derived from Data Partitioning along the Data Axis with PCA

Adnan Alrabea, A. V. Senthilkumar, Hasan Al-Shalabi, and Ahmad Bader

***Abstract*—Representing the data by smaller amount of clusters necessarily loses certain fine details, but achieves simplification. The most commonly used efficient clustering technique is k-means clustering. The better results of K-Means clustering can be achieved after computing more than one times. In this paper, a new approach is proposed for computing the initial centroids for K-means. This paper uses the first principal component generated using Principal Component Analysis (PCA) for initializing the centroid for K-Means clustering. Initially, the principal components in the dataset are gathered using PCA. From the obtained components, the first principal component is used for initializing the cluster centroid. As a result developed technique helps in decreasing the clustering time at the same time, the clustering accuracy is better for the proposed technique when compared to the existing technique.**

***Index Terms*—K-means clustering, initial centroid, computational time, PCA, clustering time.**

I. Introduction

Cluster analysis divides data into groups (clusters) that are meaningful, useful, or both. If meaningful groups are the goal, then the clusters should capture the natural structure of the data. In some cases, however, cluster analysis is only a useful starting point for other purposes, such as data summarization. Whether for understanding or utility, cluster analysis has long played an important role in a wide variety of fields: psychology and other social sciences, biology, statistics, pattern recognition, information retrieval, machine learning, and data mining. There have been many applications of cluster analysis to practical problems. Some specific examples are presented in this chapter, organized by whether the purpose of the clustering is understanding or utility. Finding nearest neighbors can require computing the pair wise distance between all points. Often clusters and their cluster prototypes can be found much more efficiently. If objects are relatively close to the prototype of their cluster, then the prototypes can be used to reduce the number of distance computations that are necessary to find the nearest neighbors of an object. Intuitively, if two cluster prototypes are far apart, then the objects in the corresponding clusters cannot be nearest neighbors of each other. Consequently, to find an object's nearest neighbors, it is only necessary to compute the distance to objects in nearby clusters, where the nearness of two clusters is measured by the distance between their prototypes.

Cluster analysis groups data objects based only on information found in the data that describes the objects and their relationships. The goal is that the objects within a group be similar (or related) to one another and different from (or unrelated to) the objects in other groups. The greater the similarity (or homogeneity) within a group and the greater the difference between groups, the better or more distinct the clustering will be. In many applications, the notion of a cluster is not well defined.

Manuscript received November 12, 2012; revised March 5, 2013.

Adnan Alrabea is with Albalqa Applied University Jordan (email: adnan_alrabea@yahoo.com).

A.V. Senthil Kumar is with the Department of MCAt, Hindusthan College of Arts and Science, Coimbatore, India (email: avsenthilkumar@yahoo.com).

Hasan Alshalabi is with Al-Hussein Bin Talal University Maan, Jordan (email: hmfnamYahoo.com).

Ahmad Bader is with the American Academy of Cosmetic Surgery Hospital, Dubai (email: ahmad_baderjo@yahoo.com).

II. The K-Means Algorithm

One of the most popular clustering methods is k-means clustering algorithm. It generates k points as initial centroids arbitrarily, where k is a user specified parameter. Each point is then assigned to the cluster with the closest centroid [1], [2]. Then the centroid of each cluster is updated by taking the mean of the data points of each cluster. Some data points may move from one cluster to other cluster. Again new centroids are calculated and assign the data points to the suitable clusters. The assignment is repeated and update the centroids, until convergence criteria is met i.e., no point changes clusters, or equivalently, until the centroids remain the same. In this algorithm mostly Euclidean distance is used to find distance between data points and centroids [3]. Pseudo code for the k-means clustering algorithm is described in Algorithm.

The Euclidean distance between two multi-dimensional data points $X = (x_1, x_2, x_3 \ldots x_m)$ and $Y = (y_1, y_2, y_3 \ldots y_m)$ is described as follows:

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_m - y_m)^2} \qquad (1)$$

Although k-means has the great advantage of being easy to implement, it has some drawbacks. The quality of the final clustering results of the k-means algorithm highly depends on the arbitrary selection of the initial centroids. In the original k-means algorithm, the initial centroids are chosen randomly and hence different clusters are obtained for different runs for the same input data [4]. Moreover, the k-means algorithm is computationally very expensive [4].

DOI: 10.7763/JACN.2013.V1.28

III. Initial Cluster Centers Deriving from Data Partitioning

The algorithm follows a novel approach that performs data partitioning along the data axis with the highest variance. The approach has been used successfully for color quantization [5], [6]. The data partitioning tries to divide data space into small cells or clusters where intercluster distances are large as possible and intracluster distances are small as possible.

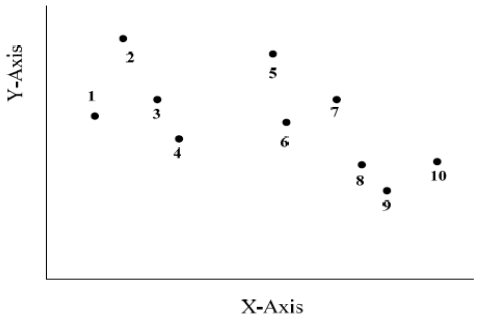

Fig. 1. Diagram of ten data points in 2D, sorted by its X value, with an ordering number for each data point.

For instance, consider Fig. 1. Suppose ten data points in 2D data space are given.

The goal is to partition the ten data points in Fig. 1 into two disjoint cells where the sum of total clustering errors of the two cells is minimal, see Fig. 2. Suppose a cutting plane perpendicular to X-axis will be used to partition the data. Let $\overline{c}_1$ and $\overline{c}_2$ be the first cell and the second cell respectively and $\overline{c}_1$ and $\overline{c}_2$ be the cell centroids of the first cell and the second cell, respectively. The total clustering error of the first cell is thus computed by:

$$\sum_{c_i \in c_1} d\ (c_i, \overline{c_1}) \tag{2}$$

and the total clustering error of the second cell is thus computed by:

$$\sum_{c_i \in c_2} d\ (c_i, \overline{c_2}) \tag{3}$$

where c_i is the ith data in a cell. As a result, the sum of total clustering errors of both cells is minimal (as shown in Fig. 2).

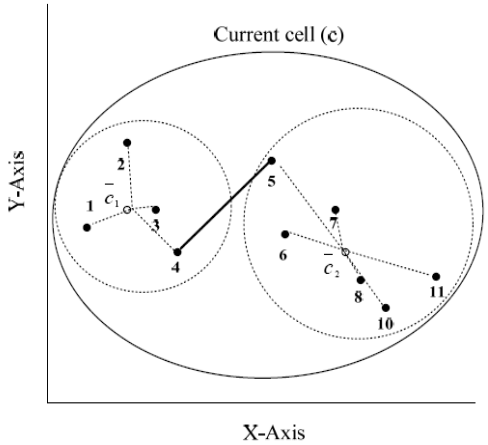

Fig. 2. Diagram of partitioning a cell of ten data points into two smaller cells, a solid line represents the intercluster distance and dash lines represent the intracluster distance.

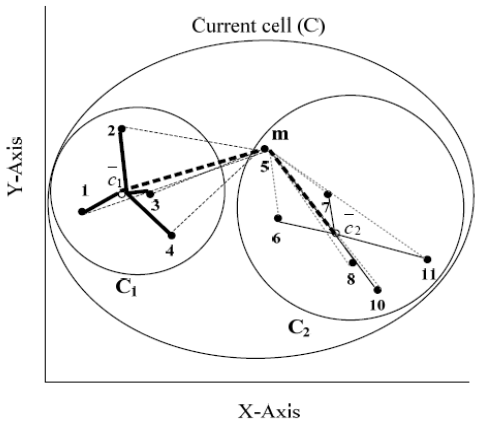

Fig. 3. Illustration of partitioning the ten data points into two smaller cells using m as a partitioning point. A solid line in the square represents the distance between the cell centroid and a data in cell, a dash line represents the distance between m and data in each cell and a solid dash line represents the distance between m and the data centroid in each cell.

The partition could be done using a cutting plane that passes through m. Thus

$$d(c_i, \overline{c}_1) \le d(c_i, c_m) + d(\overline{c}_1, c_m) \tag{4}$$

$$d(c_i, \overline{c}_2) \le d(c_i, c_m) + d(\overline{c}_2, c_m) \tag{5}$$

(as shown in Fig. 3). Thus

$$\sum_{c_i \in c_1} d(c_i, \overline{c}_1) \le \sum_{c_i \in c_1} d(c_i, c_m) + d(\overline{c}_1, c_m).|c_1| \tag{6}$$

$$\sum_{c_i \in c_1} d(c_i, \overline{c}_2) \le \sum_{c_i \in c_1} d(c_i, c_m) + d(\overline{c}_2, c_m).|c_2| \tag{7}$$

m is called as the partitioning data point where $|C_1|$ and $|C_2|$ are the numbers of data points in cluster C_1 and C_2 respectively. The total clustering error of the first cell can be minimized by reducing the total discrepancies between all data in first cell to m, which is computed by:

$$\sum_{c_i \in c1} d(c_i, c_m) \tag{8}$$

The same argument is also true for the second cell. The total clustering error of the second cell can be minimized by reducing the total discrepancies between all data in second cell to m, which is computed by:

$$\sum_{c_i \in c_2} d(c_i, c_m) \tag{9}$$

where $d(c_i, c_m)$ is the distance between m and each data in each cell. Therefore the problem to minimize the sum of total clustering errors of both cells can be transformed into the problem to minimize the sum of total clustering error of all data in the two cells to *m*.

The relationship between the total clustering error and the clustering point may is illustrated in Fig. 4, where the horizontal-axis represents the partitioning point that runs from 1 to *n* where n is the total number of data points and the vertical-axis represents the total clustering error. When *m*=0, the total clustering error of the second cell equals to the total clustering error of all data points while the total clustering error of the first cell is zero.

On the other hand, when *m* = *n*, the total clustering error of the first cell equals to the total clustering error of all data points, while the total clustering error of the second cell is zero.

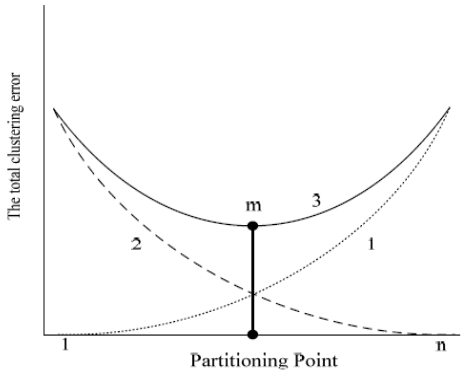

Fig. 4. Graphs depict the total clustering error, lines 1 and 2 represent the total clustering error of the first cell and second cell, respectively, Line 3 represents a summation of the total clustering errors of the first and the second cells

A parabola curve shown in Fig. 4 represents a summation of the total clustering error of the first cell and the second cell, represented by the dash line 2. Note that the lowest point of the parabola curve is the optimal clustering point (m). At this point, the summation of the total clustering error of the first cell and the second cell are minimum.

Since time complexity of finding the optimal point m is O(n2), the distances between adjacent data[7], [8] is used along the X-axis to find the approximated point of n but with time of O(n).

Let $D_j = d(c_j, c_{j+1})^2$ be the squared Euclidean distance of adjacent data points along the X-axis. If (I) is in the first cell then

$$d(c_i, c_m) \le \sum_{j=1}^{m} D_j \tag{10}$$

On the one hand, if i is in the second cell then $d(c_i, c_m) \le \sum_{j=m}^{m} D_j$ (as shown in Fig. 5).

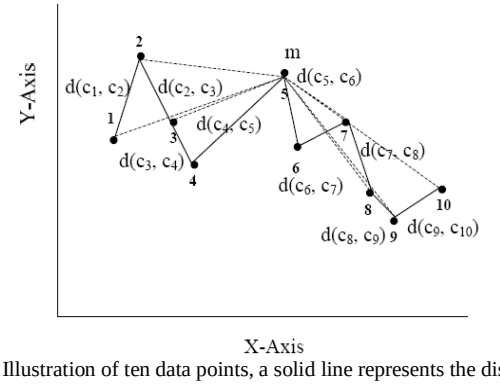

Fig. 5. Illustration of ten data points, a solid line represents the distance between adjacent data along the X-axis and a dash line represents the distance between m and any data point .

The task of approximating the optimal point (m) in 2D is thus replaced by finding m in one-dimensional line as shown in Fig. 6.

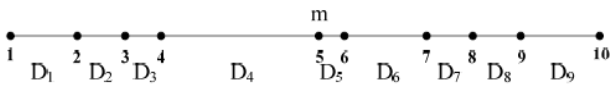

Fig. 6. Illustration of the ten data points on a one-dimensional line and the relevant Dj

The point (*m*) is therefore a centroid on the one dimensional line (as shown in Fig. 6), which yields

$$\sum_{i=1}^{m-1} d(c_i, c_m) \approx \sum_{i=m}^{n} d(c_i, c_m) \tag{11}$$

Let $dsum_i = \sum_{j-1}^{i} D_j$ and a centroidDist can be computed

$$\text{centroidDist} = \sum_{i=1}^{n} dsum_1 \tag{12}$$

It is possible to choose either the X-axis or Y-axis as the principal axis for data partitioning. However, data axis with the highest variance will be chosen as the principal axis for data partitioning. The reason is to make the inter distance between the centers of the two cells as large as possible while the sum of total clustering errors of the two cells are reduced from that of the original cell. To partition the given data into *k* cells, it is started with a cell containing all given data and partition the cell into two cells. Later on the next cell is selected to be partitioned that yields the largest reduction of total clustering errors (or Delta clustering error). This can be defined as Total clustering error of the original cell – the sum of Total clustering errors of the two sub cells of the original cell. This is done so that every time a partition on a cell is performed, the partition will help reduce the sum of total clustering errors for all cells, as much as possible.

The partitioning algorithm [9] can be used now to partition a given set of data into *k* cells. The centers of the cells can then be used as good initial cluster centers for the K-means algorithm. Following are the steps of the initial centroid predicting algorithm.

1) Let cell c contain the entire data set.
2) Sort all data in the cell c in ascending order on each attribute value and links data by a linked list for each attribute.
3) Compute variance of each attribute of cell c. Choose an attribute axis with the highest variance as the principal axis for partitioning.
4) Compute squared Euclidean distances between adjacent data along the data axis with the highest variance $D_j = d(c_j, c_{j+1})^2$ and compute the $dsum_i = \sum_{j-1}^{i} D_j$
5) Compute centroid distance of cell c:

$$\text{CentroidDist} = \frac{\sum_{i=1}^{n} dsum_i}{n}$$

where dsumi is the summation of distances between the adjacent data.

6) Divide cell c into two smaller cells. The partition boundary is the plane perpendicular to the principal axis and passes through a point m whose dsumi approximately equals to centroidDist. The sorted linked lists of cell c are scanned and divided into two for the two smaller cells accordingly
7) Compute Delta clustering error for c as the total clustering error before partition minus total clustering

error of its two sub cells and insert the cell into an empty Max heap with Delta clustering error as a key.

8) Delete a max cell from Max heap and assign it as a current cell.
9) For each of the two sub cells of c, which is not empty, perform step 3 - 7 on the sub cell.
10) Repeat steps 8 - 9. Until the number of cells (Size of heap) reaches K.
11) Use centroids of cells in max heap as the initial cluster centers for K-means clustering

The above presented algorithms for finding the initialization centroids do not provide a better result. Thus an efficient method is proposed for obtaining the initial cluster centroids. The proposed approach is well suited to cluster the gene dataset. So the proposed method is explained on the basis of genes.

IV. Principle Component Analysis

The developed method uses Principal Component Analysis (PCA) for initializing the cluster centroid. Principal Components Analysis is a method that reduces data dimensionality by performing a covariance analysis between factors. As such, it is suitable for data sets in multiple dimensions, such as a large experiment in gene expression.Principal Component Analysis is playing a vital role in data mining, and it has been also utilized in various fields. PCA involves the process in which a data space is transformed into a feature space, which has a reduced dimension. Consider that {xt} where t = 1, 2 . . . , N are stochastic n dimensional input data records with mean (μ). It is defined by the following Equation:

$$\mu = \tfrac{1}{N}\sum_{t=1}^{N} x_t \tag{13}$$

The covariance matrix of xt is defined by

$$c = \tfrac{1}{N}\sum_{t=1}^{N}(x_t - \mu).(x_t - \mu)^T \tag{14}$$

PCA solves the following Eigen value problem of covariance matrix C:

$$cv_i = \lambda_i v_i \tag{15}$$

where λ_i (i = 1, 2, ..., *n*) are the Eigen values and v_i (*i* =1, 2, ..., *n*) are the corresponding eigenvectors.

To denote data records with low dimensional vectors, m eigenvectors (called principal directions) equivalent to those m largest Eigen values ($m < n$) are need to be calculated. The variance of the projections of the input data onto the principal direction is greater than that of any other directions.

$$\text{Let } \phi = [v_1, v_2, ..., v_m], \Lambda = diag[\lambda_1, \lambda_2, ..., \lambda_m] \tag{16}$$

Then

$$c\phi = \phi\Lambda \tag{17}$$

The parameter v represents to the approximation precision of the m largest eigenvectors so that the following relation holds.

$$\frac{\sum_{i=1}^{m}\lambda_i}{\sum_{i=1}^{n}\lambda_i} \geq v \tag{18}$$

Based on (10) and (11) the number of eigenvectors can be selected and given a precision parameter v, the low dimensional feature vector of a new input data x is determined by

$$x_f = \phi^T x \tag{19}$$

After gathering all the principal components in the entire data, the first principal component axis is used to initializing the cluster centroid. Next, the K-Means clustering algorithm is applied to the dataset with initial cluster centroid as the first principal component axis generated using PCA.

The developed semi-unsupervised gene selection method is experimented using the following data sets:

- Lymphoma
- Wine
- Iris
- Glass
- Leukemia

V. Comparison with Results

First, the number of iterations required for various techniques are compared. Table I represents the comparison of number of iterations required for various techniques with different dataset. From the table, it can be observed that the proposed clustering results in lesser number of iteration when compared to K-Means and modified K-Means techniques.

TABLE I: Comparison of Number of Iterations Required for the Proposed and Existing Technique for Various Different Datasets

Dataset	Iterations		
	K-Means	Modified K-Means	Modified K-Means with PCA
Wine	7	5	5
Iris	10	11	8
Glass	13	5	5
Leukemia	10	2	2
Lymphoma	10	8	7

Next, the cluster distance resulted for various techniques are compared. Table II represents the comparison of resulted cluster distance for various techniques with different dataset. From the table, it can be observed that the proposed clustering results in maximum cluster distance when compared to K-Means and modified K-Means techniques.

TABLE II: Comparison of Cluster Distance Resulted for the Proposed and Existing Technique for Various Different Datasets

Dataset	Cluster Distance		
	K-Means	Modified K-Means	Modified K-Means with PCA
Wine	2.36936	3.124	4.7082
Iris	75.4294	85.625	114.26
Glass	9.213	11.01	12.2154
Leukemia	365.366	400.235	443.3769
Lymphoma	1288.05	1352.21	1626.75

Next, the elapsed time clustering using various techniques are compared. Table III represents the comparison of resulted elapsed time for various techniques with different dataset. From the table, it can be observed that the developed clustering results in lesser time for clustering when compared to K-Means and modified K-Means techniques.

TABLE III: COMPARISON OF REQUIRED TIME FOR THE PROPOSED AND EXISTING TECHNIQUE FOR VARIOUS DIFFERENT DATASETS

Dataset	Elapsed Time		
	K-Means	Modified K-Means	Modified K-Means with PCA
Wine	0.703	0.25	0.195
Iris	0.719	0.485	0.438
Glass	0.437	0.297	0.215
Leukemia	0.313	0.219	0.136
Lymphoma	0.453	0.312	0.232

Next, the classification accuracy using various techniques is compared. Fig. 6 represents the comparison of resulted accuracy for various techniques with different dataset. From the table, it can be observed that the proposed clustering results in better clustering accuracy when compared to K-Means and modified K-Means techniques.

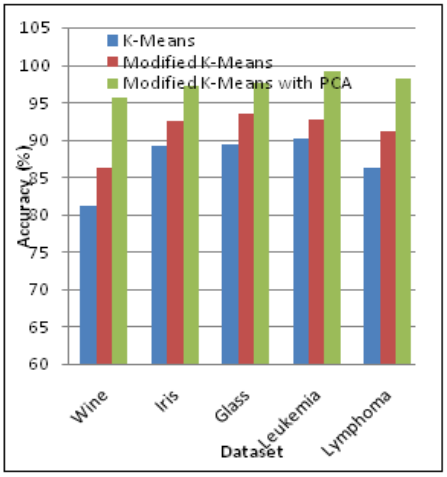

Fig. 6. Comparison of Classification Accuracy for the Proposed and Existing Technique for Various Different Datasets

VI. CONCLUSION

In this paper, we proposed a new framework for uses the first principal component generated using Principal Component Analysis (PCA) for initializing the centroid for K-Means clustering. First principal component is used for initializing the cluster centroid. Principal Components Analysis is mainly used in this paper because it reduces data dimensionality by performing a covariance analysis between factors. As such, it is suitable for data sets in multiple dimensions, it can be observed that the developed technique results in lesser number of iteration which in turn reduces the clustering time. When cluster distance is considered, the developed clustering technique results in maximum cluster distance which indicates that the proposed technique produces better accuracy for clustering. Considering all these results, the developed clustering results in better clustering result when compared to the other existing techniques. This is satisfied for all the considered dataset.

REFERENCES

[1] H. Liu, H. Motoda, R. Setiono, and Z. Zhao, "Feature selection: an ever evolving frontier in data mining," *JMLR: Workshop and Conference Proceedings*, the Fourth Workshop on Feature Selection in Data Mining, vol. 10, pp. 4-13.

[2] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Computing Surveys*, vol. 31, no. 3, September 1999.

[3] P. S. Bradley and M. U. Fayyad, "Refining initial pointsfor K-means clustering," in *Proc. the 15th International Conference on Machine Learning (ICML98)*, J. Shavlik, Ed. pp. 91- 99, Morgan Kaufmann, San Francisco, 1998.

[4] M. Yedla, S. R. Pathakota, and T. M. Srinivasa, "Enhancing K-means Clustering Algorithm with Improved Initial Center," *International Journal of Computer Science and Information Technologies*, vol. 1, issue 2, pp. 121-125, 2010.

[5] J. M. Bland, "Cluster randomised trials in the medical literature: two bibliometric surveys," *BMC Medical Research Methodology*, 2004.

[6] D. T Pham, S. S. Dimov, and C. D. Nguyen, "Selection of K in K-means clustering," in *Proc. IMechE*, vol. 219, Part C: *J. Mechanical Engineering Science*, pp. 103-119, 2005.

[7] K. Arai and A. R. Barakbah, "Hierarchical K-means: An algorithm for Centroids initialization for k-means," *Reports of the Faculty of Science and Engineering*, Department of Information Science and Electrical Engineering Politechnique in Surabaya, Faculty of Science and Engineering, Saga University, vol. 36, no.1, pp. 25-31, 2007.

[8] L. Wei, "A Parallel Algorithm for Gene Expressing Data Biclustering," *Journal of computers*, vol. 3, no. 10, October 2008.

[9] R. Dash, D. Mishra, A. Kumar Rath, and M. Acharya, "A hybridized K-means clustering approach for high dimensional dataset," *International Journal of Engineering, Science and Technology*, vol. 2, no. 2, pp. 59-66, 2010.

Adnan Ibarahim S. Alrabea received the Dr. Eng. Degree in 2004 from the Electronic and Communication Department, Faculty of Engineering, Donetsk University, Ukraine. He is a visiting Assistant Professor and Assistant dean of Prince Abdullah Bin Ghazi Faculty of Science and Information technology at Al-Balqa Applied University, Assalt, Jordan. His research interests cover: analyzing the various types of analytic and discrete event simulation techniques, performance evaluation of communication networks, application of intelligent techniques in managing computer communication network, and performing comparative studies between various policies and strategies of routing, congestion control, sub netting of computer communication networks. He published 6 articles in various refereed international journals and conferences covering: Computer Networks, Expert Systems, Software Agents, E-learning, Image processing, wireless sensor networks and Pattern Recognition. Also, in the current time, he is too interested in making a lot of scientific research in wireless sensor networks in view point of enhancing its algorithms of congestion control as well as routing protocols.

A. V. Senthil Kumar obtained his BSc Degree (Physics) in 1987, P.G.Diploma in Computer Applications in 1988, MCA in 1991 from Bharathiar University. He obtained his Master of Philosophy in Computer Science from Bharathidasan University, Trichy during 2005 and his Ph.D in Computer Science from Vinayaka Missions University during 2009. To his credit he had industrial experience for five years as System Analyst in a Garment Export Company. Later he took up teaching and attached to CMS College of Science and Commerce, Coimbatore. He has to his credit 3 Book Chapters, 7 papers in International Journals, 2 papers in National Journals, 13 papers in International Conferences, 4 papers in National Conferences, and edited a book in Data Mining (IGI Global, USA) and a book in Mobile Computing (IGI Global, USA). He is an Editor-in-Chief for International Journal titled "International Journal of Data Mining and Emerging Technologies" and "International Journal of Image Processing and Applications". Key Member for India, Machine Intelligence Research Lab (MIR Labs).

He is an Editorial Board Member and Reviewer for various International Journals. He is also a Committee member for various International Conferences. He is a Life member of International Association of Engineers (IAENG), Systems Society of India (SSI), member of The Indian Science Congress Association, member of Internet

Society (ISOC), International Association of Computer Science and Information Technology (IACSIT), Indian Association for Research in Computing Science (IARCS), and committee member for various International Conferences. He has got many for awards from National and International Societies. Also a freelance writer for Tamil Computer (a fortnightly) and PC Friend (monthly).

Hasan Mohammed Al-Shalabi received the Dr. Eng. Degree in 1994 from the National Technical University of Ukraine, Kyiv, Ukraine. He is a visiting Associate Professor and a dean computer engineering and information technology faulty and he is a students affairs deanship in Al-Hussein Bin Talal University Maan, Jordan. His research interests cover performance evaluation of communication networks, application of intelligent techniques in managing computer communication network, and performing comparative studies between various policies and strategies of routing, congestion control, sub netting of computer communication networks.E-learning. He published 19 articles in various refereed international journals and conferences covering: Computer Networks, Expert Systems, Software Agents, E-learning, Image processing, wireless sensor networks and Pattern Recognition. Also, in the current time, he is too interested in making a lot of scientific research in communication networks in view point of enhancing its algorithms of congestion control as well as routing protocols.

Ahmad Fuad Bader received the Dr. Eng. Degree in 2007 from the Electronic and Communication Department, Faculty of Engineering, Donetsk University, Ukraine. He is a Chief Information officer American Academy of Cosmetic Surgery hospital-Dubai. His research interests cover: networks, application of intelligent techniques in managing computer communication network, and performing comparative studies between various policies and strategies of routing, congestion control, Care heath information's. in the current time, he is too interested in making a lot of scientific research in wireless sensor networks in view point of enhancing its algorithms of congestion control as well as routing protocols.