International Journal of Advances in Computer Networks and Its Security
Volume 6 : Issue 1 [ISSN 2250-3757]
Publication Date : 18 April, 2016

Stable Cluster Maintenance Scheme for Bee-AdHoc-C: An Energy-Efficient and Scalable Multipath Routing Protocol for MANET

Sasmita Mohapatra, Dr.M.Siddappa

***Abstract*—Mobile ad hoc network (MANET) is one of the most important and unique network in wireless network which has brought maximum mobility and scalability. High efficient routing is an important issue in the design of limited energy resource MANETs. Many research work have been conducted by the researchers in the field of routing protocols for MANETs for making it energy efficient as the nodes are with limited resources in terms of battery supplied energy, storage and processing capability. In this paper we have proposed a new technique of routing protocol which utilizes the concept of swarm intelligence in which bee inspired routing is chosen as the ultimate routing protocol for energy efficient MANETs. To make the system more energy efficient we have chosen the clustered based approach as Bee-AdHoc-C. Bee-AdHoc-C is an evolution from Bee-AdHoc which is a bee inspired routing protocol for MANETs. This method provides parallel routing by which it reduces the overhead and improves the scalability of the system. We have also proposed a Stable Cluster Maintenance Scheme which focuses on minimizing the CH changing. By the proposed method the MANET routing can be properly balanced in terms of energy consumption with a stable cluster network. The results are shown for analysis of clustering overhead, cluster member and cluster head change for different speed and pause time.**

***Keywords*— MANET;Energy Efficiency; Swarm Intelligence; Bee-AdHoc-C; Cluster; Improved Clustering Scheme; Routing Overhead)**

I. Introduction

MANET is self-organizing, rapidly deployable which does not require any fixed infrastructure. Mobile nodes self-organize to form a network over radio links [1]. As the nodes in the MANET are battery operated so there are possibilities that some of the nodes may fail for communication in between for which care has been taken to make the MANETs energy efficient [2, 3, 5].

Sasmita Mohapatra , Research Scholar
Dept. of Electronics and Communication Engineering,
Sri Siddhartha Academy of Higher Education, Tumkur, India

Dr.M.Siddappa, Professor and Head of the Department
Dept. of Computer Science and Engineering,
Sri Siddhartha Academy of Higher Education, Tumkur, India

In this purpose the swarm intelligence concept is considered as one of the best way [4]. Swarm intelligence (SI) [6] is the collective behavior of decentralized, self-organized systems, natural or artificial. Bio inspired, Swarm Intelligence approaches are more promising for ad hoc and wireless AdHoc networks due to Locality of interactions, Availability of multiple paths, Self-organizing behaviors, Failure backup. Recent research has proven that both multipath and clustering communication is very efficient routing methods in MANETs. Clustering has been widely used to extend the network lifetime and achieve network scalability.

In this paper we have discussed regarding the architecture and working of the bee-AdHoc network required for energy efficient MANETs. Ultimately we have tried to get the best protocol for energy efficient MANETs as Bee-AdHoc-C with proposed algorithm which minimizes CH changing.

II. Architecture of Bee AdHoc Network

In Bee Ad-Hoc, each MANET node contains at the network layer a software module called hive, which consists of three parts: the packing floor, the entrance, and the dance floor. The structure of the hive is shown in Fig. 1

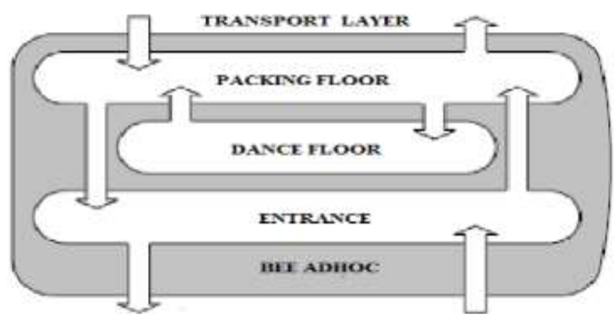

Fig-1: Architecture of Bee AdHoc Network

Packing Floor- Once a data packet arrives from the transport layer, a matching forager for it is looked up on the dance floor. If a forager is found then the data packet is encapsulated in its payload. Otherwise, the data packet is temporary buffered waiting for a returning forager.

Entrance- The entrance is an interface to the lower-level MAC layer. If the packet is a forager and the current node is its destination, then the forager is forwarded to the packing floor; otherwise, it is directly routed to the MAC interface of the next hop node.

Dance Floor- The dance floor is the heart of the hive because it maintains the routing information in the form of foragers. The dance floor is populated with routing information by means of a mechanism reminiscent of the waggle dance recruitment in natural bee hives.

In this paper we have considered Bee AdHoc-C protocol which is having certain advantages over the Bee AdHoc network discussed till now. This is mainly inspired by the cluster based working principle used by Bee-Sensor C [8] protocol used mainly for WSNs. It has some important features as-Bee-AdHoc-C is based on bee inspired mechanism. Bee-AdHoc-C adopts a dynamic clustering scheme to provide parallel data transmission near the event. Bee-AdHoc-C adopts an enhanced multipath construction method to achieve the energy consumption balance. Bee-AdHoc-C takes the multi cluster scenario into consideration based on the conclusion of our mathematical analysis.

III. Architecture And Working Of Bee-AdHoc-C

Bee-AdHoc-C is an event driven and on-demand multipath routing protocol for MANETs. As shown in Figure 2, Bee-AdHoc-C is mainly divided into three phases: cluster formation, multipath construction, and data transmission.

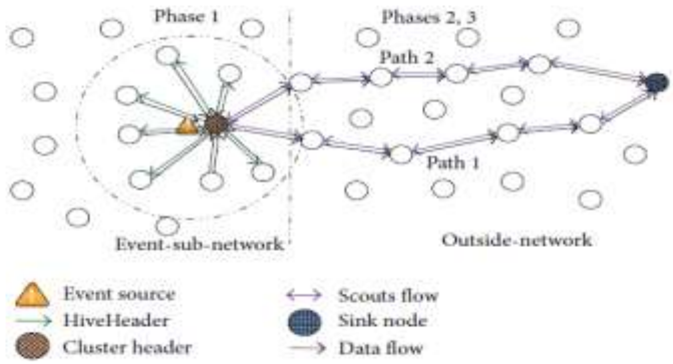

Fig-2: Workflow of Bee-AdHoc-C Network

First Phase:

Cluster Formation and deciding the Cluster Head: When an event occurs in the network Bee-AdHoc-C adopts an event based dynamic clustering algorithm. Bee-AdHoc-C adds a new agent called Hive Header into bee-hive for each AdHoc Network. The major responsibility of Hive Header is to claim that the node wants to be a cluster header (CH) node in an event area. This cluster formation is done with few steps as:

Step1: *An event occurs*
Step2: *Event is detected by nearby nodes*
Step3: *Received signal strength of the nodes calculated*
Step4: *RSS i ≥ RSS Threshold then node is accepted as a member of the cluster and cluster is formed known as event sub network*

Once the cluster is formed few steps are taken to decide the cluster head as:

Step1: *Once the node detects event it waits for a time* T_i
Step2: *After* T_i *time it broadcasts its Hive Header agent in the event sub network*
Step3: *If Hive Header of node i receives Hive Header of node j then its own Hive Header will be replaced by that of node j*
Step4: *The node which sends the Hive Header first becomes the Cluster Head and the other nodes remain as cluster members in the event sub network The waiting time* T_i *is calculated using*

$$T_i = t\left(\frac{\alpha - (Rradio\ Range - d_i)}{\beta \times E_{r_i}}\right)$$

Where t is a constant time to control the value of T_i and can be adjusted appropriately based on specific network conditions. Er_i is the current residual energy of node i. Radio Range is the maximum radio range of node. d_i is the distance from event source to node i,.α and β are two user-defined constants which can be adjusted so that different values of T_i can differ approximately by t.

Second Phase:

Improvements to Bee-AdHoc by Multipath Construction: After the formation of cluster and selection of CH the main work of CH is to send the datas to the required sink. This is also done with some steps as follows:

Step 1: *CH searches for foragers*
Step 2: *Foragers found?*
Step 3: *If step 2 is true CH tries to form multipath to sink node*
Step 4: *Data sent to destination*
Step 5: *If step 2 is not true then CH tries to find the path by foragers*
Step 6: *No. of hops are counted for the scout by H > Hmax?*
Step 7: *If step 6 is true Self destruction of forward scout happens to limit overhead*
Step 8: *If step 6 is not true CH sends a scout outside the cluster network to search destination node*
Step 9: *Intermediate node counts the reward point for the Scout by Ris = Eis/His*
Step 10: *Intermediate node fills the cache table by the reward point*
Step 11: *Cache updated if new reward point > old reward point*
Step 12: *Forward scouts dropped after cache is updated*
Step 13: *Scout reaches sink?*
Step 14: *Scout is destroyed*
Step 15: *Forward scout converted to back ward scout*
Step 16: *Scout received by a node 2nd time?*
Step 17: *If step 16 is true Scout dropped*
Step 18: *Scout is sent to ward CH according to hop information in the cache*

Third Phase:

Data Transmission: Once the back ward scout comes with hop information the main work of the CH is to transmit the data to the destination sink node through the shortest path. For this the following steps are followed:

Step 1: *Backward Scout arrives at CH with unique path ID*
Step 2: *Scouts recruit foragers using waggle dance*
Step 3: *According to no. of dances quality of path is decided*
Step 4: *Multiple paths are constructed between CH and sink node*
Step 5: *CH sends data to sink using foragers with valid routes*

The challenge in this type of Bee-AdHoc-C routing protocol is to make the network stable and the information should not change by reducing the frequency of CH change as The node in MANET is movable in nature Frequent change in CH increases clustering over head

IV. Cluster Maintenance Schemes

There are many cluster maintenance schemes already developed for MANET. Few of them are discussed here and we have tried to find out the best method.

A. Least Cluster head Change (LCC)

There is a protocol know as Lowest ID node Count (LIC) in [9]. In LIC, a mobile node with lowest ID among its all neighbors of a cluster is elected as CH. When a CH find another mobile node with lower ID than the cluster head with highest ID is forced to handover its charge to the new node and act as cluster member within the same cluster. Similarly in HCC [10] a CH having highest node degree is allowed to maintain the cluster. Because of frequent changes in cluster head by mobility of nodes in the network both LIC and HCC increased cluster overhead. A new clustering scheme was proposed to avoid frequent changes in CH, which is known as Least Cluster head Change (LCC). In LCC clustering algorithms are divided into two phases, in first phase of algorithm cluster formation is done and the second phase cover cluster maintenance. First phase follow LIC method to select a CH for a cluster. Second phase, that is the cluster maintenance phase is event driven and executed only forbelow two conditions-If two cluster head are within the same radio range, then the cluster head with lowest ID will work as cluster head and the mobile node with highest ID will release its role as cluster head to cluster member. If there is no cluster head in a cluster, the new cluster head will be formed according to LIC method.

B. Cluster Based Routing Protocol

Another cluster changing method is as Cluster Based Routing Protocol (CBRP) according to [11]. In this method for changing CH A member of same cluster never challenges to the cluster head to form a new cluster head. If two cluster heads are within the same radio range over an extended period of time then only one of them will act as cluster head second will hand over its role from cluster head to cluster member.

C. Incremental Maintenance Scheme (IMS)

The IMS as proposed in [12] uses lowest ID clustering algorithm, in which the node with lowest ID in its neighbors elected as cluster head. In IMS when two CH are within same radio range, then they will wait till delay period is less than or equal to maximum limit. As delay period reaches up to maximum limit, both cluster head will check whether they are still in same radio range, if both cluster heads are still in same radio range then the cluster head with highest ID will act as cluster head and other will hand over its charge and act as member within the cluster. But in IMS, cluster head change is delayed up to maximum limit time duration, which is calculated by dividing two times transmission range by speed. The CH change in IMS is not possible even if any emergency situation arises and need a CH change.

D.Proposed Stable Cluster Maintenance Scheme (SCMS)

The aim of the proposed SCMS algorithm is to minimize the CH selection cost by delaying the process up to acceptable time period. It will not immediately change the CH as and when two CH comes closer, it will also not delay beyond a time limit which is very long at all. Further, the algorithm computes a priority of cluster head in order to select a new one. The working of the algorithm is described as follows: If two cluster heads are within the same radio range, cluster head change will be delayed till delay timer expired. After delay timer expire if both cluster head are still within the same radio range then priority of each cluster head is calculated. If newly arrived cluster head has high priority over old cluster head then newly arrived cluster head will act as cluster head and other will hand over its charge and act as member of the cluster.If newly arrived cluster head has low priority over old cluster head then SCMS follow IMS method to delay cluster head change.Priority factor of a cluster head is calculated by addition of maximum degree of cluster head and battery life.If there is no cluster head in a cluster, the new cluster head will be formed according to LIC method.Maximum limit is calculated by dividing transmission range two time by speed.The cluster formation in proposed SCMS depends on the priority of both CH which are within the same radio range and method of IMS. The algorithm is given below:

Step 1: *Received Hello from Other_CH*
Step 2: *Delay_timer=delay_period*
Step 3: *Delayed clusterhead change for delay_timer.*
Step 4: *IF [delay_timer expire]*
Step 5: *IF[Is still in the same radio range]*
Step 6: *Old_CH_Priority= Max_degree_CH+Battery_life*

International Journal of Advances in Computer Networks and Its Security
Volume 6 : Issue 1 [ISSN 2250-3757]
Publication Date : 18 April, 2016

New_CH_Priority=Max_degree_CH+Battery_life
Step 7: *IF[New_CH_Priority>Old_CH_Prioritryr]*
Step 8: *GOTO STEP 13*
Step 9: *ELSE IF[delay_time<=Max_limt]*
Step 10: *delay_time=delay_time+hello_interval*
Step 11: *GOTO STEP 3*
Step 12: *ELSE IF[New CH_ID<=Other_CH_ID]*
Step 13: *Send Triggered Hello as CH*
Step 14: *GOTO STEP 20*
Step 15: *END IF*
Step 16: *ELSE*
Step 17: *GOTO STEP 13*
Step 18: *END IF*
Step 19: *Send Trigggered hello as CH.*
Step 20: *END*

V. Results And Analysis

The algorithm is simulated in ns-2. Some of the performance parameters are measured for all the present algorithms and the result has been put in an graphical way and analyzed. Certain parameters are taken for consideration Cluster Head change, Cluster Member change, Cluster Overhead.

Cluster Head Change:

Here cluster head change has been calculated for all the protocols once depending on various pause time where the pause time has been taken from 60 sec. till 240 sec. with a time interval of 60 sec. The graph is done for different values of cluster head change.

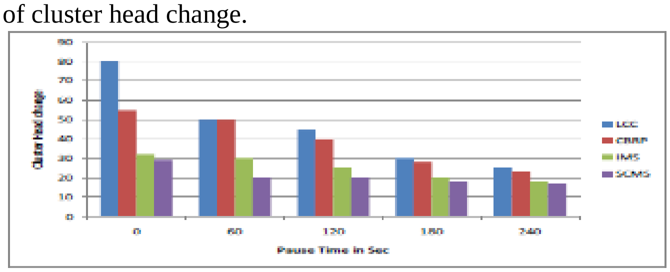

Fig.3. Cluster head change over pause time

Fig.3. shows the cluster head change for varying pause time. From the graph it is seen clearly that as the pause time increases the cluster head change decreases for all the schemes however the performance of the proposed SCMS is the best. Also the cluster head change has been calculated over different MN (Mobile Node) speed and put in the graph. Likewise from the figure-4 it is clear that as speed of mobile nodes increases the cluster head change also increases.

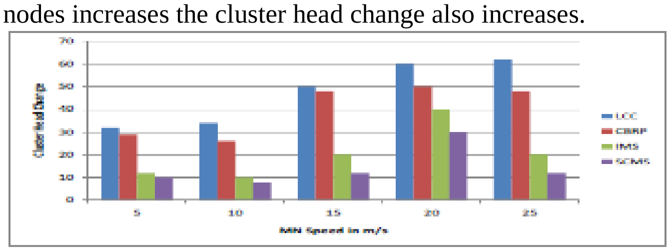

Fig.4. Cluster head change for varying MN speed

Cluster Member Change:

Hear the cluster member change for all the schemes have been considered over various pause time and plotted in a graph.

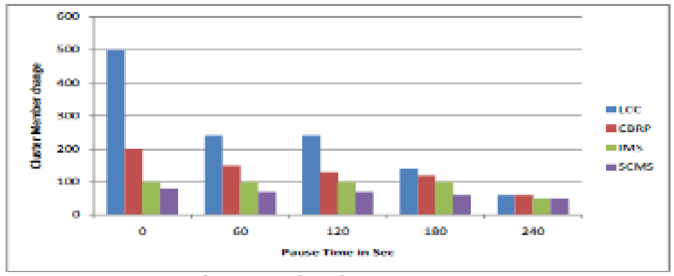

Fig.5.Cluster member change over pause time

From the Fig-5 it is seen that for all the methods as the pause time increases the change in the no. of cluster member decreases. However the change in cluster member number is lowest for the proposed SCMS scheme.

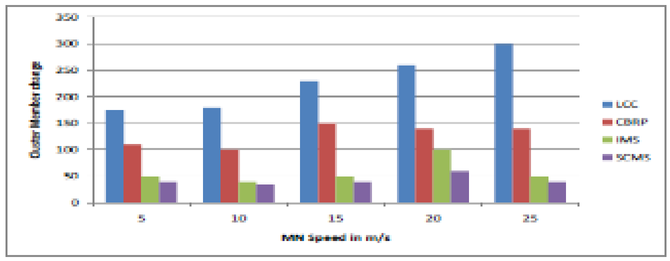

Fig.6. Cluster member change varying MN speed

In Fig. 6 clustering member change with respect to different MN speed is shown. It shows that as speed increases total number of cluster member changes also increases but increment of cluster member changes in SCMS is slower then LCC, CBRP and IMS. SCMS performance is better than the other three schemes as speed of mobile nodes increase.

Cluster Overhead

Finally the cluster overhead is calculated with respect to pause time and different MN speed and plot in graph. From Fig. 7 Clustering overhead of SCMS is low as compare to LCC, CBRP and IMS. In SCMS initially up to pause time 60 cluster overhead reduced very fast and as soon as pause time increases cluster overhead is in reducing order but from Fig. 7 IMS initially decreases cluster head up to 100s pause time and then again it increase cluster overhead. This shows that SCMS is better than LCC, CBRP and IMS.

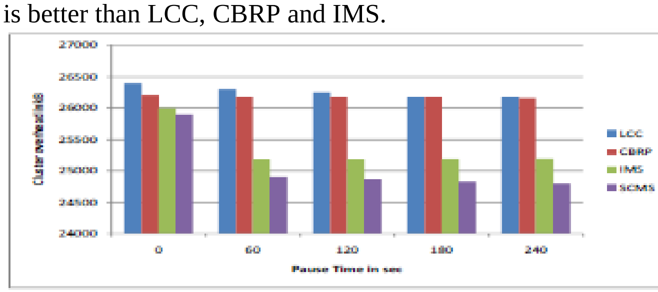

Fig.7. Clustering overhead with pause time

International Journal of Advances in Computer Networks and Its Security
Volume 6 : Issue 1 [ISSN 2250-3757]
Publication Date : 18 April, 2016

As in fig.8 clearly indicate the performance of SCMS is far better in compared to LCC, CBRP and IMS in term of clustering overhead by increasing speed of a mobile node. From the figure it is clear that the clustering overhead is low in SCMS as speed increases. When speed is near 15 clustering overhead is increasing in LCC, CBRP and IMS but SCMS is better compared to other schemes. However the no. of cluster head again decreases for higher value of MN speed.

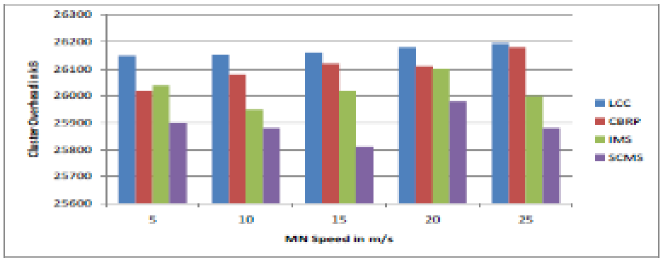

Fig.8.Clustering overhead for varying MN speed

VI. Conclusion And Future Scope

Thus from all the above discussions we conclude that Bee-AdHoc-C is found to be the best protocol for MANETs which is a multipath routing protocol based on dynamic clustering and foraging behavior of bee swarm. According to this paper Stable Cluster Maintenance Scheme is the best method for Clusterded Bee-AdHoc network as it makes the cluster network more stable by less variation of CH. The results clearly show that the performance of SCMS is best compared to LCC, CBRP and IMS in terms of cluster head change, cluster member change and cluster overhead.

References

[1] J.A. Boyan and M.L. Littman., "Packet routing in dynamically changing networks: a re-inforcement learning approach.Advances in Neural Information Processing Systems", 6:671–678, 1993.

[2] L.M. Feeney., "An energy consumption model for performance analysis of routing pro-tocols for mobile ad hoc networks.Mobile Networks and applications" , 6(3):239–249,2001.

[3] L.M. Feeney and M. Nilsson., "Investigating the energy consumption of a wireless net-work interface in an ad hoc networking environment", In Proceedings of IEEE INFO-COM, 2001.

[4] F. Ducatelle, G. Di Caro, and L.M. Gambardella., "Using ant agents to combine reactive and proactive strategies for routing in mobile ad hoc networks", International Journal of Computational Intelligence and Applications , Special Issue on Nature-Inspired Approaches to Networks and Telecommunications, 5(2):169–184, 2005.

[5] S. Russell and P. Norvig., "Artificial Intelligence: A Modern Approach", Prentice Hall, second edition, 2002.

[6] S. Ossowski and A. Garcıa-Serrano., "Social structure in artificial agent societies: Implications for autonomous problem-solving agents", In Intelligent Agents V, Agent Theories, Architectures, and Languages, 5th International Workshop, ATAL '98, Paris, France, July 4-7, 1998, Proceedings, volume 1555 of Lecture Notes in Computer Science,pages 133–148. Springer, 1995.

[7] M. Saleem and M. Farooq., "Beesensor: a bee-inspired power aware routing protocol for wireless sensor networks". In M. Giacobini et al. (Eds.), Lecture Notes in Computer Science, LNCS 4449, pages 81–90. Springer Verlag, 2007.

[8] Xuelian Cai, Yulong Duan, Ying He, Jin Yang, and Changle Li, "Bee-Sensor-C: An Energy-Efficient and Scalable Multipath Routing Protocol for Wireless Sensor Networks", International Journal of Distributed Sensor Networks Volume 2015,pp.224-238, August 2014.

[9] A. Ephremides, J. E. Wieselthier, D. J. Baker, "A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling," in Proc.of IEEE, vol. 75, 1987, pp. 56–73.

[10] M. Gerla, J. T. Tsai, "Multiuser, Mobile, Multimedia Radio Network," Wireless Networks, vol. 1, Oct. 1995, pp. 255–65.

[11] A. Ephremides, J. E. Wieselthier, D. J. Baker, "A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling," in Proc.of IEEE, vol. 75, 1987, pp. 56–73.

[12] Narendra Singh Yadav, Bhaskar P Deosarkar, R. P. Yadav, "A Low Control Overhead Cluster Maintenance Scheme for Mobile ad hoc networks," International Journal of recent trends in Engineering, Vol 1, No. 1, May 2009,pp. 1-9.

About Author (s):

Sasmita Mohapatra received B.E degree in Electronics and Instrumentation from Utkal University, Orissa, India in 2001 and M.Tech degree in Electronics & Communication Engineering from IASE University of Rajasthan, India in 2005. She is currently working toward the Ph.D. degree in Computer Networking in the depaprtment of Electronics & Communication Engineering at Sri Siddhartha Academy of Higher Education, Tumkur, India under the supervision of Dr.M.Siddappa. She is currently working as an Associate Professor in the department of Electronics and Communication Engineering at Sapthagiri College of Engineering, Bangalore, India. Her current research interests include Routing protocols for Energy Efficient MANETs, Bee Ad-Hoc Network etc.

M.Siddappa received B.E and M.Tech degree in Computer Science & Engineering from University of Mysore, Karnataka, India in 1989 and 1993 respectively. He has completed doctoral degree from Dr.MGR Educational Research Institute Chennai under supervision of Dr.A.S.Manjunatha, CEO, Manvish e-Tech Pvt. Ltd., Bangalore in 2010. He worked as project associate in IISc, Bangalore under Dr.M.P Srinivasan and Dr. V.Rajaraman from 1993 – 1995. He has teaching experience of 26 years and research of 10 years. He published 52 Technical Papers in National, International Conference and Journals. He has citation index of 129 till 2015 and h-index of 5 and i10-index of 2 to his credit. He is a member of IEEE and Life member of ISTE. He is working in the field of data structure and algorithms, Artificial Intelligence, Image processing and Computer networking. He worked as Assistant Professor in Department of Computer Science & Engineering from 1996 to 2003 in Sri Siddhartha Institute of Technology, Tumkur. Presently, he is working as Professor and Head, Department of Computer Science & Engineering from 1999 in Sri Siddhartha Institute of Technology, Tumkur. He has visited Louisiana University Baton rouge, California University and Wuhan University Chaina.