*International Journal of Communication Systems and Network Technologies*
*Vol.4, No.1, 2015*

# An Adaptive Packet Drop Probability Approach Using Fuzzy Logic to Control Congestion in MANET

**Vikas Thakur,**
Department of Electronics & Communication Engineering, MITS
Gwalior, Madhya Pradesh, India
thakur.vikas250@gmail.com

**Laxmi Shrivastava**
Department of Electronics & Communication Engineering, MITS
Gwalior, Madhya Pradesh, India
lselex@yahoo.com

**Sarita Singh Bhadauria**
Department of Electronics & Communication Engineering, MITS
Gwalior, Madhya Pradesh, India
saritamits61@yahoo.co.in

**Abstract-** Mobile ad-hoc network is an infrastructure less network where every node has its own protocols and services for powerful cooperation in the network. Every node has the ability to handle the congestion in its queues during traffic overflow. This was done through drop tail policy where the node drops the incoming packets to its queues during overflow condition. Previous studies showed that early dropping of incoming packet is an effective technique to avoid congestion. Such approach is known as active queue management. In this paper, entirely different approach is taken to minimize the congestion in which drop probabilities are changed according to the change in load value, known as adaptive drop probabilities.

***Keywords****- Qual-Net, mobile ad-hoc network, fuzzy system, network congestion, active queue management.*

## I. Introduction

Mobile ad-hoc network is a type of network without infrastructure where every node can work as a router. Every node has protocols and services to request and provide services to other nodes with the congestion handling capability. Usually congestion control is done through transmission control protocol. This protocol sends congestion signal when the node's queue is full. Some studies[3,8] showed that dropping the packet prior to reaching the maximum queue length is an effective technique to avoid the congestion and to minimize the packet latency, for instance active queue management drops incoming packets before the queue is full in contrast to traditional queue management which starts dropping only when the queue is overflowed.

There is a high network congestion due to high bit error rate(BER) in the wireless channel, increased collision due to the presence of hidden terminals, interference, location dependant connection, uni directional links, frequent path breaks due to mobility of nodes and the inherent fading properties of the wireless channel . This justify the need for high adaptive system with adapting capabilities to high variability and uncertainty for these type of networks. The proposed fuzzy logic based system is such type of technique to overcome the shortcoming in ad-hoc networks. The use of fuzzy logic to the problem of congestion control allows us to specify the relationship between queue parameters and packets dropping probabilities using "if then" type of linguistic rules. The fuzzy logic algorithm would be able to translate or interpolate these rules into a nonlinear mapping. In this paper, the technique is proposed to reduce congestion by changing the packet drop probability, with the change in load value.

## II. Related Work

Random early detection (RED) [8] is mostly used AQM technique. It manages the queue in an active manner by randomly dropping packets with increasing probability as the average queue size increases. RED maintains two thresholds that determines the rate of packet

*DOI-10.18486/ijcsnt.2015.4.1.02*
*ISSN-2053-6283*

drops, a lower threshold (denoted by $min_{th}$) and an upper threshold (denoted by $max_{th}$).

Some of the previous studies showed the difficulties of choosing the RED parameters [10,13]. Other showed that there is no significant benefit to RED over drop tail for the web traffic [4, 5]. These drawbacks are the main reasons to default disable of the RED (or WRED [7]). In most of the available routers today. To overcome these shortcomings, extension of the RED algorithm had been proposed to make it more and/or adaptive, for instance, stabilized RED (SRED) [14], flow RED (FRED) [2], dynamic RED (DRED) [12] etc. The most famous variant is adaptive RED (ARED) algorithm proposed by Floyed et al. [9]. In AFRED, $max_p$ is configured dynamically to keep the average queue size average within a target range.

Many studies used the fuzzy logic system to dynamically calculate the drop probability behaviour of AQM policy. Wang et al. gives adaptive fuzzy based RED (AFRED) algorithm to find the drop probabilities using the current queue size as the only input for the fuzzy system. Some other studies found the drop probability based on fuzzy explicit rate marking (FERM) algorithm using two queue state inputs: current queue size $q_c$ and its rate of change $\Delta q_c$. The FERM was used in [15] for ATM networks, while in [6] it was used for differentiated services networks.

In [1], author finds the drop probabilities using the fuzzy proportional derivative controller (FPDC) with two inputs: the error *'e'* and the change of the error *'Δe'*. A conventional fuzzy controller use ($e$, $\Delta e$) as inputs to observe the controlled system response and its parameters. These parameters includes, overshoot, rise time and settle time. These parameters are not only used to evaluate the stability, but the performance of a system as well, and is given in specification.

Li et al. [11] have used the current average queue size 'avg' and 'Δavg' as the input for the fuzzy logic adaptive RED (FLARED) algorithm to adaptively modifying the changes of step size of the parameter $max_p$. This scheme changes only one parameter of ARED algorithm and its drawback is the lack to tune other ARED parameters.

## III. Congestion Control in Ad-Hoc Networks

In ad-hoc networks, congestion is handled by transport layer protocols. For this purpose TCP uses transmission control protocol (TCP) . The objective of this protocol is to set up an end-to-end delivery of data packets, flow control and congestion control. TCP uses window-based flow control mechanism in which sender maintains a variable size window whose size limits the number of packets the sender can send. The destination sends acknowledgement (ACK) for each packet received successfully. When the window gets full, the sender must wait for an ACK before sending a new packet based on a sliding window principle. This waiting time is known as retransmission time out (RTO) period. If the ACK is not received within RTO period, then the sender will assume the packet is lost. The packet is loss due to congestion in the network which will compel the TCP to start the congestion control mechanism.

Mobile ad-hoc networks are dynamic in nature in which network topology changes frequently due to unrestricted mobility of nodes. This will lead to frequent changes in the connectivity of wireless links and hence route reestablishment may be repeated very often and this will take a significant amount of time. The route reestablishment time depends on transmission range of the nodes, distance between source and destination, number of intermediate nodes between the source and destination and node's velocity. If the route reestablishment time exceeds the RTO period of the source node, then it will not receive the ACK and assumes congestion in the network, afterwards the lost packets are retransmitted and the congestion control mechanism is initiated. Congestion in ad-hoc network is illustrated in figure1. The source sends its packet via node A, node A passes those packets node B then to the destination. If the link between source and node A is broken, it starts route reestablishment process and creates a direct link with node B. If this processing time is less than RTO, the source will receive the ACK and send other data packets, or it will resend the previous lost packets.

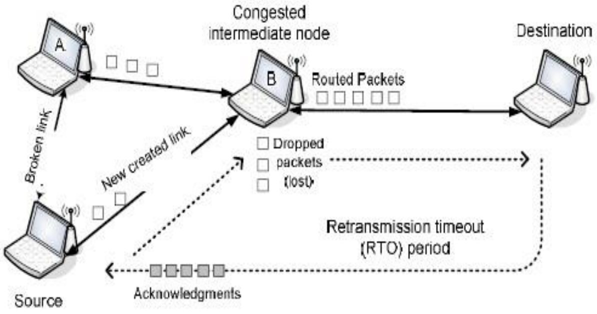


**Fig. 1. Congestion in ad-hoc networks**

## IV. Fuzzy Logic Based Congestion Controller

4.1 Fuzzy logic controller

Fuzzy logic controllers, like expert system can be used to model human decision making behaviours. In fuzzy logic controller input and output relations can be expressed as a set of linguistics rule (If-then rules), to model a particular system. A fuzzy logic controller comprises of four important parts i.e. fuzzifier, defuzzifier, an inference engine and a rule base. Many of the fuzzy control application have an input data which has a crisp value, so a fuzzification is necessary to convert a input crisp data into a suitable set of linguistic value that is needed in the inference engine. Singleton fuzzifier is the general fuzzification method which is used to map the crisp input to a singleton fuzzy set. In the rule base of fuzzy logic controller, a set of fuzzy control rules, which characterize the dynamic behaviour of the system, are defined. The inference engine is used to form inferences as well as draw conclusion from the fuzzy control rules. Figure2 shows the fuzzy logic controller architecture. The output inference engine is sent to defuzzification unit.

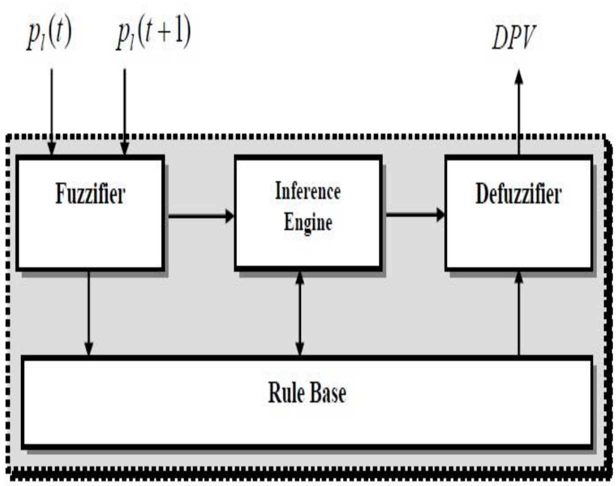


**Fig. 2. Fuzzy logic controller architecture**

Defuzzification is a process of mapping from a space of fuzzy control actions into a space of crisp control actions. Suppose the fuzzy logic controller has $n$ input variables i.e. $x_1, x_2, ----, x_n$. Input vector $X$ is defined as $(x_1, x_2----, x_n)^T$ and suppose rule base consists of $K$ rules with the following general form:

Rule 1: If $X = (A_{11}, A_{12}, ----, A_{1n})$ then $y$ is $B_1$

Rule 2: If $X = (A_{11}, A_{12}, ----, A_{1n})$ then $y$ is $B_2$

Rule k: If $X = (A_{11}, A_{12}, ----, A_{1n})$ then $y$ is $B_k$

Where in the j<sup>th</sup> rule $A_{ij}$ and $B_j$ are fuzzy sets of linguistics variables $x_1, x_2----, x_n$ and $y$ respectively. The output f(X) of this fuzzy controller, product inference engine and centre-average defuzzifier can be calculated as.

$$f(X) = \frac{\sum_{j=1}^{K} y_0^j \prod_{i=1}^{n} \mu_i^j(x_i)}{\sum_{j=1}^{K} \prod_{i=1}^{n} \mu_i^j(x_i)} \qquad (1)$$

Where $y_0^j$ is the centre value of output fuzzy set $b_j$. in the $j^{th}$ rule. $\mu(X)$ is the membership function for the fuzzy sets. In this paper load is taken as input variables and drop probabilities as output variables.

## V. Effect of Queue Size on Drop Probability

Current queue sizes drastically affect the drop probability. The drop probability $p_d$ can be calculated as:

$$p_d = \frac{2N^2}{(CT_p + q_c)^2} \qquad (2)$$

Where $N$ is the load factor, $C$ is a transmission capacity (in packets/seconds) and $T_p$ is a propagation delay (in sec.). Figure 3 shows the relation between the drop probability and the load for various queue sizes. Hence it is evident that the probability of a packet dropping increases as the load increases. More packets in the queue wait for processing as the load increases. Or when the used space of the queue is high, the drop probability of incoming packets is also high and vice versa. Thus, the following rules are proposed:

R1: If load is low then $p_d$ should be low.

R2: If load is medium then $p_d$ should be high.

R3:If load is high then $p_d$ should be high.

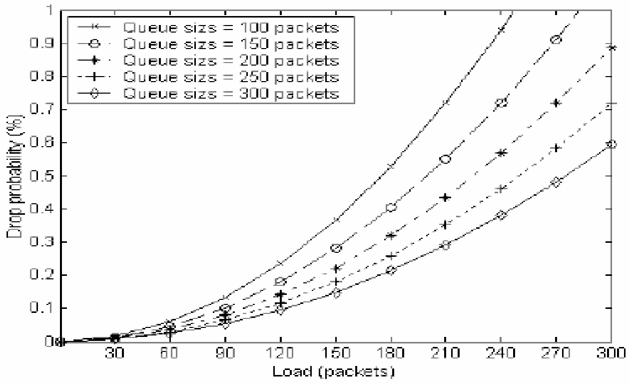
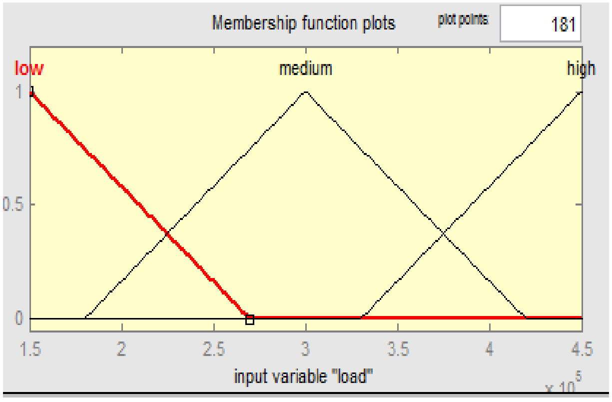


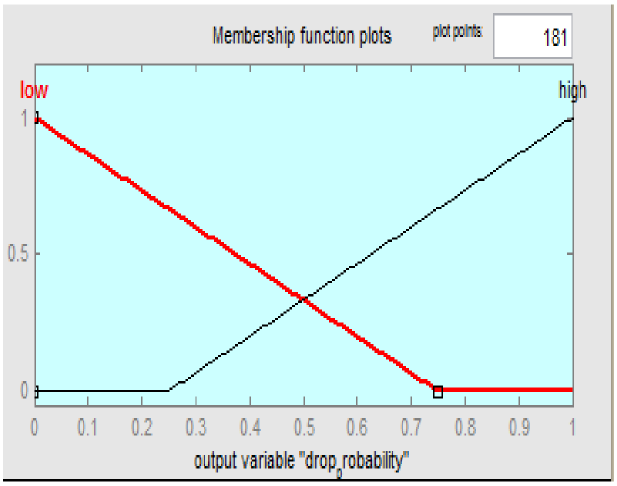
**Fig. 3 Drop probabilities for coming load**

### 6. Membership Function for Fuzzy Variables

The MFs we propose to use for the fuzzy input (queue size) and the fuzzy output ($p_d$) are illustrated in Figure 4.



(a) Membership Functions used for the input variables.



(b) Membership function used for output variables

**Fig. 4 Membership functions used for the fuzzy variables.**

As shown in the above figure the input variable (load) have a range from 150000 bytes to 450000 bytes and is divided into three regions low, medium and high respectively. Values which are chosen for simulation are tabulated below.

Table 1: Input/output variables obtained by Membership Functions

| **Input Load(bytes)** | **Output Packet Drop Probability** |
|---|---|
| 180000 | 0.22 |
| 208000 | 0.40 |
| 255000 | 0.66 |

## VII. Simulation Parameters

Simulation of the proposed ADP (adaptive drop probability) was done in Qual-net 6.1 simulator Proposed scenario is shown below.

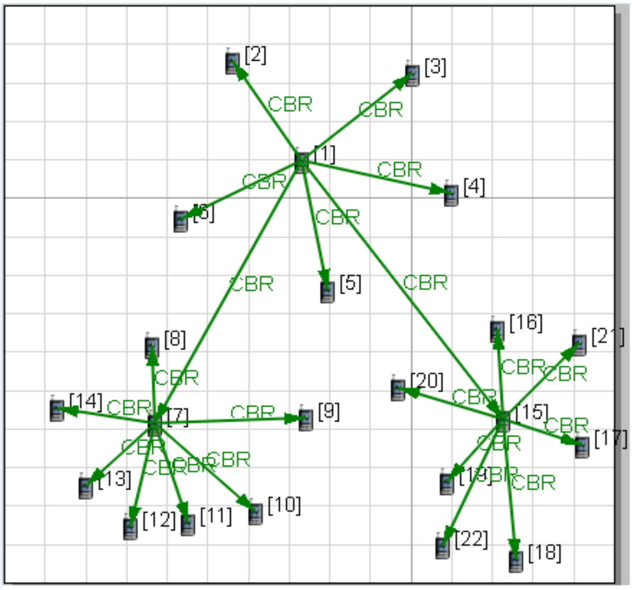


**Fig. 5 Simulation set up**

In this scenario following parameters are taken

Table 2: Simulation parameters

| **Parameters** | **Values** |
|---|---|
| Number of nodes | 22 |
| Connection | CBR |
| Simulation time | 400 seconds |
| Simulation area | 1500×1500 |
| Routing Protocol | AODV |
| Mobility | none |
| Battery Model | Linear |
| Queue Type | FIFO |
| Node type | Default |
| Packet Size | 512 bytes |
| Data rate | 2 Mbps |

## VIII. Simulation Results

In the results following metrics are considered

1. Packet drop ratio- It is the ratio of total no of packets send minus total no of packet received to the total no of packet send.

2. Total unicast message received- It is defined as the total number of messages received by the receiver.

3. Throughput- Throughput is the rate of successful message deliver over a communication channel.

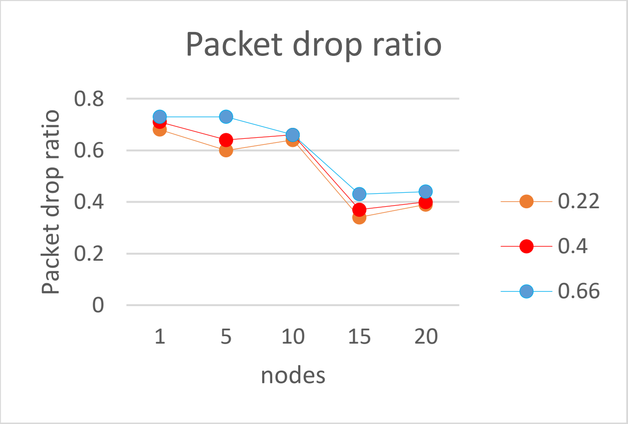


**Fig.6 (a).Simulated packet drop ratio**

The packet drop ratio remains almost constant as the packet drop probabilities increases.

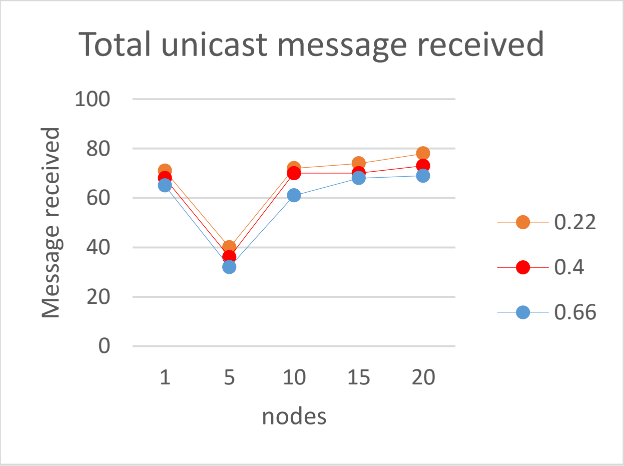


**Fig.6 (b). Total unicast message received**

Total unicast message received remains almost constant as the packet drop ratio increases.

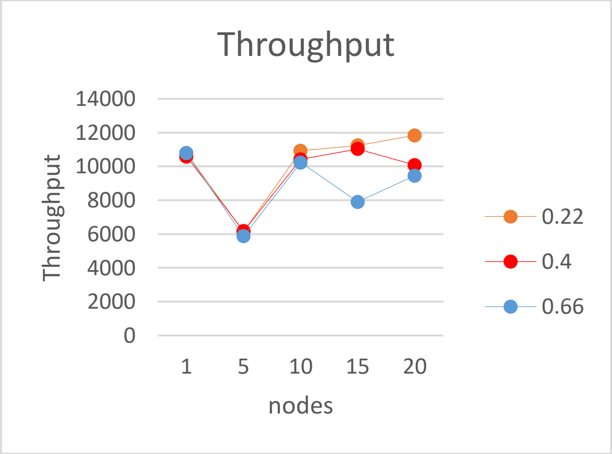


**Fig.6(c).Throughput**

Throughput follows same trend as that of previous metrics i.e. it also remains approximate same as the packet drop probability increases.

## IX. Conclusion

In this paper, a novel technique ADP (adaptive drop probability) based on fuzzy logic has been suggested. As shown in the results there is no significant change in the values of metrics as the drop probability increases with the load. Hence as the load increases the packet drop ratio remains almost unchanged, therefore congestion is reduced .This technique for adaptive drop probability is implemented in wireless ad-hoc network in order to provide effective congestion control by achieving high queue utilization low packet losses in case of increasing drop probabilities.

## References

1. Al-Frihat J., “Advanced Queue Management Algorithms for Computer Networks,” *Studies in Informatics and Control Journal*, vol. 14, no. 2, pp. 111-116, June 2005.
2. Aweya J., Ouellette M., and Montuno D. Y., “A Control Theoretic Approach to Active Queue Management,” *Computer Networks*, vol. 36, no.2-3, pp. 203-35, July 2001.
3. Braden B., Clark D., Crowcroft J., Davie B.,Deering S., Estrin D., Floyd S., Jacobson V.,Minshall G., Partridge C., Peterson L.,Ramakrishnan K., Shenker S., Wroclawski J., and Zhang L., “Recommendations on Queue Management and Congestion Avoidance in the Internet,” *Request for Comments (RFC) 2309*,April 1998.
4. Brandauer C., Iannaccone G., Diot C., Ziegler T., Fdida S., and May M., “Comparison of Tail Drop and Active Queue Management Performance for Bulk-Data and Web-like Internet Traffic,” *in Proceedings of the 6th IEEE Symposium on Computers and Communications*, Hammamet,July 2001.
5. Christiansen M., Jeffay K., Ott D., and Smith F. D., “Tuning RED for Web Traffic,” *IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp.249-264, June 2001.
6. Chrysostomou C., Pitsillides A., Rossides L., Sekercioglu Y. A., and Polycarpou M., “Congestion Control in Differentiated Services Networks Using Fuzzy-RED,” *IFAC Journal Control Engineering Practice*, vol. 11, no. 10,pp. 1153-1170, October 2003.
7. Cisco Systems, *Weighted Random Early Detection on the Cisco 12000 Series Router*, March 2002.
8. Floyd S. and Jacobson V., “Random Early Detection Gateways for Congestion Avoidance,” *IEEE/ACM Transactions Networking*, vol. 1, no.4, pp. 397-413, August 1993.
9. Floyd S., Gummadi R., and Shenker S., “Adaptive RED: An Algorithm for Increasing the Robustness of RED’s Active Queue Management,” available at *http://www.icir. org/floyd/red.html*, 2005.
10. Iannaccone G., May M., and Diot C., “Aggregate Traffic Performance with Active Queue Management and Drop

from Tail," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 3, July 2001.

11. Li Z., Zhang Z., Addie R., and Clerot F., "Improving the Adaptability of AQM Algorithms to Traffic Load Using Fuzzy Logic," *in Proceedings of the Australian Telecommunications, Networks and Applications Conference (ATNAC'03)*, Melbourne, Australia, December 2003.
12. Lin D. and Morris R., "Dynamics of Random Early Detection," *in Proceedings of ACM SIGCOMM'79*, France, pp. 127-137, 1997.
13. May M., Bolot J., Diot C., and Lyles B., "Reasons not to Deploy RED," *in Proceedings of the 7th International Workshop on Quality of Service (IWQoS'99)*, pp. 260-262, June 1999.
14. Ott T. J., Lakshman T. V., and Wong L., "SRED: Stabilized RED," *in Proceedings of IEEE INFOCOM*, vol. 3, pp. 1346-1355, 1999.
15. Pitsillides A., Sekercioglu Y. A., and Ramamurthy G., "Effective Control of Traffic flow in ATM Networks Using Fuzzy Logic Based Explicit Rate Marking (FERM)," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 2, pp. 209-225, February 1997.
16. Laxmi Shrivastava, SS. Bhadauria, G.S. Tomar, "Influence of Traffic Load on the performance of AODV, DSR and DSDV in MANET", International Journal of Communication Systems and Network Technologies, Vol.1 Issue 1. pp 22-34, Apr 2013.
17. Tripti Sharma, Brijesh Kumar, G.S.Tomar, "FBRA: Fuzzy Based Redundancy Avoidance Protocol in Densely Deployed Wireless Sensor Network",International Journal of Smart Device and Appliance Vol.1, No.1 pp 1-16, Dec 2013.

**Vikas Thakur** obtained his B.E in Electronics and Communication engineering from Sagar Institute of Research and Technology in M.P (India), in 2010. Currently he is pursuing his M.E in Communication Control and Networking from Madhav Institute of Technology and Science M.P (India). His current areas of interest includes effective use of fuzzy logic to control the congestion in the mobile ad-hoc networks.

**Laxmi Shrivastava** obtained her B.Tech from GBPUAT, Pantnagar University, India, in 1993. She obtained her M.Tech and PhD.(ad-hoc networking) from R.G.P.V Bhopal, in 2006 and 2013 respectively. Currently she is associate professor in M.I.T.S, Gwalior.