1. Introduction

Wireless Mesh Networks (WMNs) have drawn significant attention in the recent past because of their advantages and applications over other wireless technologies. Mesh routers and Mesh clients are the building blocks of wireless mesh network. Mesh routers form the back bone of the network and have limited mobility. The wireless mesh network can be integrated to other wireless network technologies like Wi-Fi, 802.15, 802.16 and sensor networks through mesh routers. Some of the mesh routers can act as Gateways for connecting to the external world. The WMN is a type of Mobile Adhoc Network (MANET) with mesh connectivity property among its elements. The WMN uses multi-hop mechanism for data transfer from source node to destination node in the multi-hop environment.

The main attractive feature of WMNs is that they are self configured and self organized. Once the WMN is established it automatically forms an ad-hoc connection among the nodes with mesh connectivity. The clients can forward and receive data through multi-hop communication which is the key characteristic of Wireless Mesh Network. The Wireless Mesh Network router has additional capability like establishing mesh connectivity that makes network to be self organized even in case of node failures. WMNs have advantages such as low cost, easy maintenance, reliable coverage, robustness when compared to other ad hoc networks. WMNs deliver wireless services for several applications in campus, metropolitan areas, local, and personal.

In peer to peer networks they are no centralized servers. Each peer acts as both client and server. Without the requirement of any centralized controller or server,

* Author for correspondence
peers can request and reply for resources which are available on the network and this is in contrast to the traditional client server system. Peer-to-Peer Network is an overlay network that is formed at the application layer which is independent of physical network topology. Such overlay networks are helpful for indexing peers and resources. The main traffic in the peer to peer networks is file sharing and content discovery.

P2P systems are classified into two different classes:
- Structured P2P systems ex: Chord, Pastry, tapestry, CAN etc.,
- Unstructured P2P systems ex: Napster, Gnutella, Freenet etc.,

Structured peer-to-peer overlay networks are widely used for building large-scale distributed network applications such as file sharing. It is of interest to build those distributed application over wireless mesh networks. Unfortunately, conventional structured P2P overlays are not suitable for direct deployment in a mobile and wireless environment.

Peer to Peer applications is rather low in multi-hop environment such as Mobile Adhoc Networks and Wireless Mesh Networks due to mobility of nodes, characteristics of wireless links, multi-hop forwarding. The various challenges in deploying Peer to Peer Networks in a multi-hop networks are: Bandwidth constraints, Peer to Peer overlay maintenance, Network Resiliency, Routing stretch, query propagation etc.

2. Methodology

2.1 Structured P2P algorithm-Chord protocol over Wireless Mesh Networks

Chord Protocol is a structured P2P algorithm that is used to distribute resources among all participating nodes. It is based on Distributed Hash Table (DHT). In Chord Protocol Secure Hash Algorithm-1 (SHA-1) is used to generate keys and value pairs. Hashing an IP address of a participating node generates a unique identifier called node ID. Similarly hashing resource i.e. file name or data that is to be distributed in the network generates a unique identifier called resource ID. The identifiers of both resource that is to be distributed and IP address of a participating node are of same length. The idea is to distribute resources over all participating nodes. The resource is distributed to a node whose identifier is greater than or equal to resource identifier.

For example consider a chord network as shown in the Figure 1. The node identifiers are labeled N1, N2......N64. The resource identifiers are labeled K1, K2 .... K64. The key K54 has to store on node N54 in the network if it is available. Since node with identifier 54 is not available in the network the K54 is stored on the node with its identifier greater than key identifier 54. As a result the K54 is stored on the node with identifier N56 since N54 is not available in the network. Similarly if a node leaves the network it assigns it keys to its next successor on the chord ring. Each peer maintains a routing table called finger table to forward queries.

The routing table of a peer consists of log N successors where N is the number of peers in the overlay. The successors of a node routing table is calculated as Node ID + 2i-1 where i = 0, 1, 2,...log N – 1. The chord advantages are simple, scalable, load balance, decentralization.

We have deployed chord protocol over wireless mesh network. The simulations are done for two scenarios: static nodes and dynamic nodes. The QoS parameters like No. of hops for look up, average query response time, average path length are compared for both scenarios. From the simulations it is observed that the performance of the chord protocol degrades in terms of average query response time, average path length etc., when nodes are moving as compared with static nodes

The mobility of the nodes has significant affect on the performance of chord protocol. Maintaining a stable DHT in a wireless mesh network is challenging due to frequent node mobility, link quality etc.

We have considered a location aware chord protocol that takes node locality into consideration for constructing chord. This close coupling allows minimizing the overhead in DHT communication, because nodes mainly
talk to physically close nodes, thus requiring fewer transmissions. In addition to location of nodes, link quality, end to end delay, packet delivery ratio etc., are considered to improve the performance of chord. Since it becomes a multi objective function, meta heuristic algorithms such as Particle Swarm Optimization (PSO) and FireFly Algorithms (FF) etc., are generally employed to solve these problems.

2.2 Particle Swarm Optimization

Particle Swarm Optimization algorithm is a meta-heuristic algorithm derived from the concepts of bird flocks, fish schooling etc. The problem here is to improve the QoS parameters of a P2P based Wireless Mesh Networks like PDR, End to End delay, Network load etc., The fitness function is formed in terms of link quality, end to end delay, query response time. The optimization here is to minimize this fitness function.

The objective is to find the best possible set of nodes with good link quality to hold hash table. The fitness function employed here is given by:

$$\min f(x) = \left[ d_{ijn}Q_{rt} \cdot \frac{L_{ij}}{e + 1} \right]$$

Where $d_{ijn}$ is the normalized end to end delay, $Q_{rt}$ is the normalized query response time, $L_{ij}$ is the link quality between node i and j, $L_{m}$ is the maximum link quality, PDR is the Packet Delivery Ratio.

The PSO algorithm starts with swarm of particles with random positions in the d-dimensional work space. The random positions mean random possible set of nodes in the problem space. Let the positions of these particles is represented with a vector $p = \{X_1, X_2, \ldots, X_d\}$ and initial velocity vector $u = \{u_1, u_2, \ldots, u_d\}$. On every iteration step the positions and velocities of these particles are updated that will take towards optimum results. Each particle $X_1, X_2, \ldots, X_d$ represents a possible set nodes from source to destination. For each particle the fitness is calculated by using equation 1. The particle with best fitness is called Gbest and individual best as Pbset.

PSO is summarized as a pseudo code in Figure 2.

The working of PSO algorithm for chord based Wireless Mesh Network is given below.

1. The problem search space starts with initial possible set of nodes from source to destination called particles. The particles are given by $P = \{X1, X2 \ldots \ldots Xd\}$.
2. The algorithm parameters such as $c_1, c_2, w$, Positions and velocities of particles are initialized.
3. The fitness is calculated for each particle based on link quality, End to End delay, Packet Delivery Ratio and Query Response Time.
4. The particle with best fitness value is assigned as Pbset and global best as Gbest.
5. The position and velocities of particles are updated according to the following equations:

$$v_i(t+1) = \omega v_i(t) + c_1 r_1 (Pbest - x_i(t)) + c_2 r_2 (Gbest - x_i(t))$$

$$x_i(t+1) = x_i(t) + v_i(t+1)$$

Here $c_1$ and $c_2$ represent acceleration coefficients; and $r_1$ and $r_2$ represent random numbers in between 0 & 1. Where ‘i’ varies from 1 to d and ‘t’ is the iteration number.

6. Based on updated positions and velocities, new particles are generated. The fitness of new particles is calculated.
7. For every particle, compare particle’s fitness value with Pbset. If the current value is better than Pbset value, then set this Pbset. Similarly compare the current particles with Gbest with previous Gbest. If this is value is better than previous then set this as Gbest.
8. The steps from 2 to 6 are repeated till the maximum no. of iterations.

Figure 2. Particle Swarm Optimization for Chord.

The best possible set of nodes with good link quality, low end to end delay, high Packet Delivery Ratio, less query response time are obtained at the end of iterations.

3. FireFly Algorithm

The QoS parameters in a P2P based Wireless Mesh Network are a non deterministic polynomial hard problem hence an optimization problem. Natured inspired Metaheuristic algorithms such as Particle Swarm Optimization, Firefly algorithm, Ant Optimization,
bat algorithm etc., have drawn lot of attention in the recent past to solve these optimization problems. Any optimization problem consists of an objective function. The objective is to find the optimized parameters that minimize or maximize this function. Here the objective is to have low end to end delay, high packet delivery ratio, reduction of network load, low average query response time in a P2P based Wireless mesh Network.

Firefly algorithm is a natured inspired meta heuristic algorithm developed by Xin she yang from the flashing behaviour of fireflies. For simplicity the following assumptions are considered:

• All fireflies are unisex. That is one firefly is attracted towards regardless of its gender.
• Attractiveness between two fireflies is proportional to their brightness. i.e. a less bright firefly will move towards a nearby brighter firefly. If a firefly finds no better brighter firefly nearby it will move randomly in the search space.
• Firefly brightness or intensity is calculated from the objective function.

Initially the search problem starts with an initial number of fireflies in the search space. Let the initial fireflies are considered as \( x_i \), \( i = 1, 2, 3, \ldots, m \). Here initial fireflies mean initial random solutions of the objective function. The brightness or intensity of each firefly is calculated through the objective function as \( I_i \). The firefly with least objective function value has best brightness as the optimization is minimizing objective function. The intensity of a firefly varies with distance as
\[
I = I_0 e^{-\gamma r^2}
\]
where \( I_0 \) is the intensity of a firefly at the source and \( \gamma \) is the absorption coefficient. The attractiveness between two fireflies is given by
\[
\beta = \beta_0 e^{-\gamma r^2}
\]
where \( \beta_0 \) is the attractiveness between two fireflies with distance between them \( r = 0 \).

The lesser brighter firefly will move towards brighter one according to
\[
x_i = x_i + \beta_0 e^{-\gamma r^2} \left( x_j - x_i \right) + \alpha \epsilon_i
\]
where \( \alpha \) being the randomization parameter, \( \epsilon_i \) and is a vector of random numbers drawn from a Gaussian distribution or uniform distribution at time \( t \).

The firefly algorithm is given15.

### FFA Meta-heuristic

**Begin**

- Initialize algorithm parameters
- \( \text{MaxIter: the maximum number of Iterations} \)
- \( \gamma: \text{the light absorption co-efficient} \)
- \( r: \text{the particular distance from the light source} \)
- \( d: \text{the domain space} \)

Define the objective function \( f(x) \), where \( x = (x_1, x_2, \ldots, x_d) \)
- Generate initial population of fireflies \( x_i \) \( i = 1, 2, 3, \ldots, d \)
  - i.e. random solutions of the objective function

Determine the light intensity \( I_i \) of a firefly \( x_i \) via \( f(x_i) \)
- While \( (t < \text{MaxIter}) \)
  - For \( i = 1 \) to \( n \) (all \( n \) fireflies)
    - For \( j = 1 \) to \( n \) (all \( n \) fireflies)
      - If \( I_j > I_i \)
        - Move firefly \( i \) towards \( j \) using equation \( 4 \);
        - End if;
      - Attractiveness varies with distance \( r \) via \( \exp \left[ -\gamma r^2 \right] \);
      - Evaluate new solutions and update light intensity,
    - End for \( j \);
    - End for \( i \);
  - Rank the fireflies and find the current best
- End while;
- Post Process results and visualization;
- End procedure;

### Figure 3. Block diagram of the proposed algorithm.

#### 4. Hybrid Particle Swarm Optimization-FireFly Algorithm (PSO-FF)

This section presents the hybrid of Particle Swarm optimization and Firefly algorithm (PSO-FF) to optimize the QoS parameters of a P2P based Wireless mesh Network. The PSO-FF integrates the advantages of both PSO and FF algorithms. The FF algorithm has the limitation of getting trapped in local minima. Both local and global searches are performed concurrently with hybrid PSO-FF algorithm. The optimum solution of PSO is given as initial step for FF algorithm.

Any optimization algorithm initially takes a random set of solutions and proceeds towards optimum results.
through iteration steps. In the same way, the PSO-FF algorithm starts with random initial population $x_i$. From the initial population pbest and gbest is chosen randomly. The intensity $I_i$ of each firefly is calculated through the fitness function. The firefly with low fitness value has high brightness [16].

The position vectors $x_i$ of the PSO-FF are updated by using equation 5.

$$x_i(t+1) = wx_i(t) + c_1e^{-\gamma x^2}(\text{pbest}_i - x_i(t))$$
$$+ c_2e^{-\gamma x^2}(\text{gbest}_i - x_i(t)) + a\left(y - \frac{1}{2}\right)$$

(5)

The results given below shows that though PSO and FF algorithms obtained nearly similar results but hybrid PSO-FF algorithm obtained improved results as compared with individual.

5. Experimental Results

The performance of proposed algorithm is evaluated in OPNET simulator integrated with MATLAB. The QoS parameters are compared for PSO-FF, PSO and FF when nodes are static and dynamic. The mobile nodes follow random way point.

The simulation parameters considered are shown in Table 1:

| Table 1. Simulation Parameters |
|--------------------------------|
| No. of Nodes                  | 17                          |
| Deployment area               | 4 Km X 4 Km                 |
| Bandwidth                     | 2Mbps                       |
| PSO Parameters                | $c_1 = c_2 = 0.5, w = 1$    |
| FF Parameters                 | $\gamma = 0.2, \alpha = 0.5$|
| Termination criteria          | 100 iterations              |

In this study, the Fig. 5 to 7 explain the results of metrics such as PDR, End to End delay and Network load using Static Network and the Fig. 8 to 10 explain the results of metrics such as PDR, End to End delay and Network load using with different Node mobility.

A. Results for Static Network
6. Conclusion

The performance of structured P2P algorithm—chord is evaluated in a Wireless Mesh Network and simulation results showed that performance degradation due to dynamic nature of nodes. Constructing a stable DHT in a multi-hop environment such as MANETs and WMNs is challenging due to frequent node mobility, link quality, bandwidth considerations etc., To improve the QoS parameters of chord in a wireless mesh network environment, we have integrated chord with natured inspired metaheuristic algorithms such as PSO, FF and a Hybrid PSO-FF. The simulations results showed that PSO-FF obtained better performance as compared with PSO and FF.

7. References

1. Akyildiz Ian F, Wang X, Wang W. Wireless mesh networks: a survey. Computer Networks. 2005; 445–87.
2. Muogilim OE, Loo KK, Comley R. Wireless mesh network security: A traffic engineering management approach. Journal of Network and Computer Applications. 2011; 34(2):478–91.
3. Ahmed I, Mohammed A, Alnuweiri H. On the fairness of resource allocation in wireless mesh networks: a survey. Wireless Networks. 2013; 19 (6):1451–68.
4. Chongang W, Li I.B. Peer-to-Peer Overlay Networks: A Survey. The Hong Kong University of Science and Technology, Hong Kong, 2003.
5. Jaiswal V. Adaptive content replication in Peer to Peer Network. Doctoral dissertation, San Diego State University, 2012.
6. Al Asaad A, Gopalakrishnan S, Victor L. Peer to Peer File sharing over Wireless Mesh Networks. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing. 2009. p. 697–702.
7. Castro M, Kassler A, Chiasserini CF, Casetti C, Korpeoglu I. Peer-to-Peer Overlay in Mobile Ad-hoc Networks. Hand book of Peer-to-Peer Networking. 2009; 1045–80.
8. Stoica I, Morris R, Liben-Nowell D, Karger DR, Kaashoek MF, Dabek F, Balakrishnan H. Chord: a scalable peer-to-peer lookup protocol for internet applications. IEEE/ACM Transactions on Networking. 2003; 11(1):17–32.
9. Canali C, Renda MF, Santi P Evaluating load balancing in peer-to-peer resource sharing algorithms for wireless mesh networks. 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, MASS 2008. 2008 Sep. p. 603–9.
10. Murali Krishna PP, Subramanyam MV, Satyaprasad K. Performance Analysis of Chord Protocol for Peer to Peer Overlay Topology in Wireless Mesh Network. International Journal of Computer Applications. 2013; 65(13).
11. Murali Krishna Krishna PP, Subramanyam MV, Satyaprasad K. Mesh DHT approach for efficient resource sharing in P2P based Wireless Mesh Networks. ARPN Journal of Engineering and Applied Sciences. 2015; 10 (21).
12. Dian P, Mariyam S, Sophiyati S. Particle Swarm Optimization: Technique, System and Challenges. International Journal of Computer Applications. 2011; 14 (1):0975–8887.
13. Pengjun Z, Sanyang L, Guo C. Modified Particle Swarm Optimization for Optimization Problems. Journal of Theoretical and Applied Information Technology. 2012; 46(2):610–4.
14. Yang X-S, He X. Firefly Algorithm: Recent Advances and Applications. Int J Swarm Intelligence. 2003; 1(1):36–50.
15. Farahani SM, Abshouri AA, Nasiri B, Meybodi MR. Some hybrid models to improve firefly algorithm performance. International Journal of Artificial Intelligence. 2012; 8(12):97–117.
16. Padmavathi K, Ramakrishna KS. Hybrid Firefly and Particle Swarm Optimization algorithm for the detection of Bundle Branch Block. International Journal of the Cardiovascular Academy, International Journal of the Cardiovascular Academy. 2015.