On the Tree Construction of Multi Hop Wireless Mesh Networks With Evolutionary Algorithms

Zahra Zarei zahra.zarei@gmail.com
Department of Computer Science and engineering
Shiraz University , Shiraz, Iran

Shahram Jafari jafaris@shirazu.ac.ir
Department of Computer Science and engineering
Shiraz University , Shiraz, Iran

Abstract

In this paper, we study the structure of WiMAX mesh networks and the influence of tree structure on the performance of the network. From a given network graph, we search for trees, which fulfill some network, QoS requirements. Since the searching space is very huge, we use genetic algorithm in order to find solution in acceptable time.

We use NetKey representation which is an unbiased representation with high locality, and due to high locality we expect standard genetic operators like n-point cross over and mutation work properly and there is no need for problem specific operators. This encoding belongs to class of weighted encoding family. In contrast to other representation such as characteristics vector encoding which can only indicate whether a link is established or not, weighted encodings use weights for genotype and can thus encode the importance of links. Moreover, by using proper fitness function we can search for any desired QoS constraint in the network.

Keywords: Wireless Mesh Networks, WiMAX, Network Planning, Multihop Networks.

1. INTRODUCTION

Wireless mesh networks have received much attention in recent years due to its low up-front cost, easy network maintenance, robustness, and reliable service coverage [3-5]. Different from traditional wireless networks, WMN is dynamically self-organized and self-configured. In other words, the nodes in the mesh network automatically establish and maintain network connectivity.

In such networks, each mesh node plays both roles of a host and a router. Packets are forwarded in a multihop fashion to and from the gateway (connected to the Internet). It has been shown that the throughput and delay performances in wireless mesh networks are location dependent [6-9]. Mesh networks show great advantages such as good coverage, rapid and cost-efficient deployment, and robustness. Mesh networks are built using various technologies, however the most commonly used are WiFi (based on the IEEE 802.11 family) and WiMAX (based on IEEE 802.16). Using IEEE 802.11 for the wireless backbone leads to dense and suboptimal deployments due to the short transmission range of the standard and, consequently, low aggregate throughput capacity can be obtained. WiMAX, on the other hand, has a transmission range of several kilometers with high data rates.

In WiMAX mesh networks, a MSS (Mesh Subscriber Station) can only have one path to the BS (Base Station) which is through its immediate parent [1]. Thus, MSS nodes are organized in a tree structure rooted at the BS. The tree topology is constructed through temporarily disconnection of some links logically. From the other side, we know for a given network graph, one can construct plenty of spanning trees (see the Kirchhoff's Matrix Tree Theorem [2]). In each resulted tree, nodes fan-out and the tree's depth affected the performance of the WiMAX network markedly. In this paper, we investigate this phenomenon and find desired tree topologies, which satisfy intended delay and throughput trade-off. We first obtain per-node throughput and delay of each node using the model proposed in [9]. In this model, we can obtain throughput and delay for

each node independent of tree structure. Using the obtained results, and, employing a proper scheduling algorithm, we search for the best tree topologies. For this purpose, we use an evolutionary algorithm in order to find trees, which satisfy some QoS requirements. The proposed algorithm converges fast while it finds good enough answers. Due to huge searching space, classical searching algorithms lead to unacceptable searching time and thus are not practical.
The rest of this paper is organized as follows: in section 2, we discuss the related work. In section 3, we define model assumptions. In section 4, we present the genetic algorithm for tree construction. In section 5, we show the results. Finally, the paper is concluded in section 6.

2. RELATED WORKS

Designing mesh networks and specially WiMAX has been studied in some works.
In [4] the authors proposed an algorithm that selects a parent for each MSS, which maximize throughput capacity. The object is to select links that have highest data rates among the set of all possible paths between an MSS and the BS. In [9] the authors model and analyze the location-dependent throughput and delay in WMN. They analyze the packet arrival and the packet departure rate for the forwarding queues at each node, and based on the analytical model, they proposed two network design strategies to provide fair resource sharing and minimize the end-to-end delay in WMN. In [10] they investigate the throughput capacity of a WiMAX mesh tree, and they try to balance the impact of the depth of the tree with its fan-out.The approaches for node placements in WMN are depict in figure 1.
In this paper, we want to optimize WiMAX topology such that it meets some QoS constraints, especially throughput and delay, so we assume that all nodes in the network are in the transmission range and we do not consider the coverage provisioning in our work.
Using Genetic Algorithm (GA) in the field of network planning is widely investigated. Generally, the application is to optimize some network performance like capacity, topology and routing. In most of the works, the topology is graph, which has not the limitation of trees. Tree topologies have limitations dealing with GA operators that we will address in this paper.
The representation that widely used for coding trees in GA is Prufer numbers or Prufer sequence [11]. This encoding only represents trees, and each Prufer number represents exactly one tree. These interesting characteristics make it a good option, but it also has an important problem, low locality, which reduce the GA performance dramatically. In [12] the authors investigate the properties of Prufer numbers and show that it is a poor representation for trees.

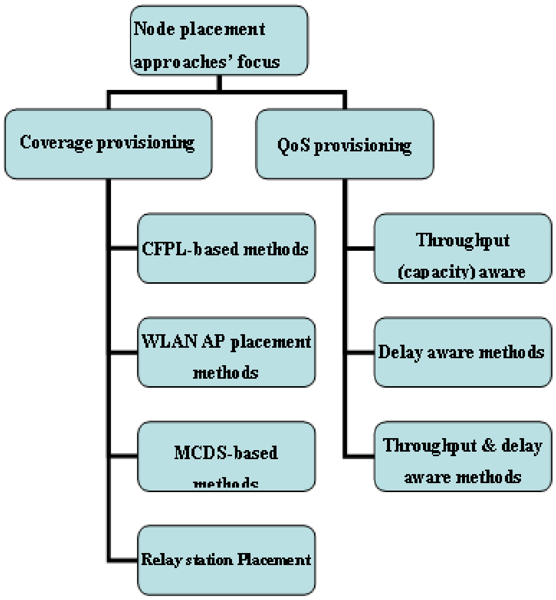


FIGURE 1: Node placement approaches

3. ANALYTICAL MODEL

The network that we consider consists of N Mesh Subscriber Station (MSS) nodes and one Base Station (BS). Packets are forwarded in multihop fashion to or from the BS. We consider unidirectional traffic, i.e., the traffic that only goes from the MSS to the BS.
Each MSS_i has two queues, Q_r for relay packets and Q_s for packets originating from the node itself. The forwarding rules at each node is described as follows: [9]

1- If Q_r is empty, it sends one packet from Q_s.(Q_s always has packets to be transmitting)
2- If Q_r is not empty, it sends a packet from Q_r with a probability of q_i or a packet from Q_s with a probability of $1-q_i$.

q_i is forwarding probability for node i.

3.1 Throughput and Delay Analysis

We define per node throughput as the number of packets originating from node i successfully received by the BS. It can be obtained by counting the packets, which are received successfully by the BS without being blocked in any intermediate nodes. Thus, we need to calculate the blocking probability at each intermediate MSS. Based on queuing theory analysis, it is given as:

$$P_i^b = \begin{cases} \dfrac{[1-\rho_i]\rho_i^K}{1-\rho_i^{K+1}} & ,if\ \rho_i \neq 1 \\ \dfrac{1}{K+1} & ,if\ \rho_i = 1 \end{cases} \qquad (1)$$

Where ρ_i is the traffic intensity given above. Thus, we can state the throughput of node i, as below:

$$T_i = \begin{cases} \sigma_i^s & ,if\ |n_i^u| = 0 \\ \sigma_i^s . \prod_{i \in n_i^u} [1-P_i^b] & ,if\ |n_i^u| > 0 \end{cases} \qquad (2)$$

Where $|n_i^u|$ is number of nodes in the uplink path from node i to the BS. The average network throughput is obtained as follows:

$$mean(T) = \frac{\sum_{i=1}^{N} T_i}{N} \qquad (3)$$

To drive the delay which a packet from node i encounter, we need to compute the waiting time of a packet in each Q_r. For this we first obtain the steady state queue size of Q_r in node i as follows (using M/M/1/K analysis):

$$L_i^r = \begin{cases} \dfrac{\rho_i}{1-\rho_i} - \dfrac{\rho_i[K\rho_i^K + 1]}{1-\rho^{K+1}} & ,if\ \rho_i \neq 1 \\ \dfrac{K(K-1)}{2(K+1)} & ,if\ \rho_i = 1 \end{cases} \qquad (4)$$

Then we will have the following expression for the waiting time of a packet in Q_r of node i:

$$W_i^r = \frac{1}{\mu_i^r} + \frac{L_i^r}{\lambda_r[1 - P_b^r]} \qquad (5)$$

Now, we can obtain D(i) by summing the waiting times spent in the intermediate nodes and the transmission times(i.e. τ), for traversing the i hopes:

$$D_i = \begin{cases} \tau & ,if\ |n_i^u| = 0 \\ \tau n_i^u . \sum_{i \in n_i^u} W_i^r & ,if\ |n_i^u| > 0 \end{cases} \qquad (6)$$

In addition, the average delay of the network can be found by taking into account the delay of packets, which have been successfully delivered:

$$mean(D) = \frac{\sum_{i=1}^{N} T_i D_i}{\sum_{i=1}^{N} T_i} \qquad (7)$$

4. PROPOSED GENETIC ALGORITHM

Given the number of nodes which exist in a WiMAX network; we intend to obtain a tree topology that satisfies some QoS metrics. For this purpose, we use a Genetic Algorithm (GA) approach. GA is one the most powerful approaches for optimizing complicated, multi-objective and large scale problems. Our problem in this paper is also, large scale and constrained. By a slight increasing in the number of nodes the search space grows dramatically and become too complicated to be addressed by conventional problem solving strategies, e.g. Linear Programming.

The proposed Genetic Algorithm is as followed:

Algorithm 1. Proposed GA

```
1. Create a specified number of random keys with length $l=n(n-1)/2$.
2. For each individual produced in step 1, generate it's permutation sequence and it's
   corresponding spanning tree.
3. i=1
4. Do{
     a. At iteration i, evaluate fitness of each individuals.
     b. Select top 10 percent individuals from $P_i$ and transfer them directly to population
        $P_{i+1}$ , apply selection mechanism as defined in 4-2 to choose individuals.
     c. Apply cross over to generate offsprings.
     d. Apply mutation operator.
     e. Generate the next population $P_{i+1}$.
     f. Increment i.
5. } While a solution is not found or total number of generation is not reached or the function
   tolerance is less than threshold.
```

Details of our algorithm are as follows:

4.1 Network Representation

We use NetKey representation which is simple and convenient to implement and characteristics of this representation is studied in [13].

The NetKey encoding belongs to the class of weighted encodings. In contrast to other representations such as CV encoding (Davis et al., 1993) which can only indicate whether a link is established or not, these encodings use weights for the genotypes and can thus encode the importance of the links Also, an additional construction algorithm is necessary which constructs a valid tree from the genotypic weights.

For coding a tree, we first generate a random key sequence with length l=n(n-1)/2. Each element of this sequence which is between zero and one, describes the importance of the corresponding link (link ordering is like CV encoding).

The permutation r^s corresponding to a key sequence r of length l is the permutation δ such that δ_r is decreasing. These definitions say that the positions of the keys in the key sequence r, are ordered according to the values of the keys in descending order.

After generating a random key sequence and a permutation, we should construct a valid tree from that permutation. When constructing the tree, the positions of the key sequence r, are interpreted in the same way as for the CV.

The positions are labeled and each position represents one possible link in the tree. From a Key sequence r of length l, we have a permutation r^s ,of l numbers. Then the tree is constructed from the r^s as follows:

Algorithm 2. Tree Construction

```
1- Let i=0 , T be an empty tree with n nodes, and $r^s$ the permutation of length l=n(n-1)/2 that can be constructed from the key sequence r. All possible links of T are numbered from 1 to l.
2- Let j be the number at the ith position of the $r^s$.
3- If the insertion of the link with number j in T would not create a cycle, then insert the link in the T.
4- Stop, if there are n-1 links in T.
5- Increment I and go to step 2.
```

The construction rule is based on Kruskal’s algorithm (Kruskal 1956) and only consider the weights of the Random Keys vector for building the tree. With this rule, we can construct a unique, valid tree from every possible Random key sequence.

4.2 Selection and Reproduction

Based on trial and error we employ elitism to directly transfer top 10 percent individuals from current population to the next one. The rest of the population is created by crossing and mutating the genes. We use tournament selection with tournament size of four, for selecting the parents for mating pool. The reason why we employ this selection mechanism is to keep a balance between randomness and proportionality in search process.

4.3 Cross Over and Mutation Operator

Since the NetKey encoding have high locality, the standard cross over, mutation operators will work properly, and there is no need for problem specific operators. Because the operators will change the Random key sequence there will be no infeasible solution.

4.4 Fitness Function

With fitness function, we can evaluate the goodness degree of each individual in the population. For any desired QoS, we can have different fitness functions.

A well-known problem in wireless mesh networks is the fairness problem [5-7], i.e., the nodes farther away from the gateway may have a lower throughput than the nodes closer to the gateway. We address this problem here by searching for trees, which have smoothest throughput among its nodes. So in our fist scenario we want to find a tree t such that function f(t)=std(T) is minimized.

In the second scenario, we study the delay-minimization problem based on the analytical model described in 3. Here we want to find a QoS tree, which minimize the average delay in whole network. The fitness function here is f(t)=min(D).

5 NUMERICAL STUDY AND RESULTS

The network we choose for our numerical study is consisting of 18 nodes and one gateway. Based on Cayley(1889), there is exactly n^{n-2} possible tree for a graph with n nodes. This huge search space dare us to use GA, otherwise it will be too complex.

For simplicity we do not consider spatial reuse in the simulation, thus only one node is allowed to send within one time slot. The channel capacity is 75Mbs and the packet size is 1500B, so the length of time slot is set to 1500B/75Mbs=0.16 ms. The buffer size of Q_r is set to 30 for each node. In addition, for each node the channel access probability is assumed to be proportional to its sub tree and, probability for relaying packets is q=0.7 for all nodes. Population size is 50; crossover and mutation rates are adjusted to 0.8 and 0.02 respectively. Total number of GA iterations is set to 200.

The resulted tree topologies for fairness throughput and minimum delay are show in figure 2, 3 and the corresponding per-node throughput and delay for each node is depicted in figure 4(a), (b) respectively. As we can see, in the first scenario, the throughput among nodes is smooth and close to each other as we expected and the resulted tree is also almost balance. We know that for having a fair throughput among nodes especially for those that are far away from gateway we should consider tuning the forwarding probability for nodes,in the leaf nodes q=0 because there is no data to be forwarded but in intermediate nodes the q should be set to a value such that the node forward the childs traffic and also it's packets.

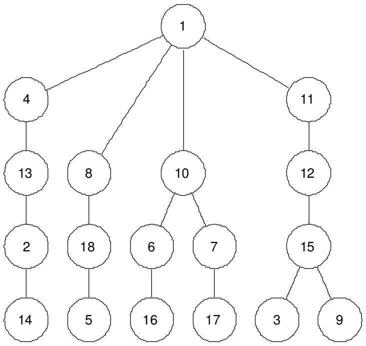


FIGURE 2: Resulted tree topology for fair Throughput

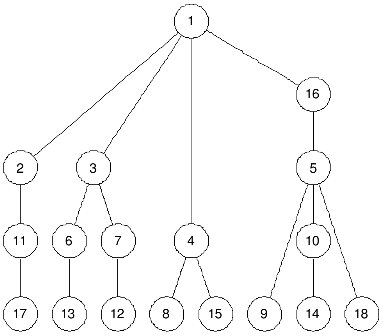


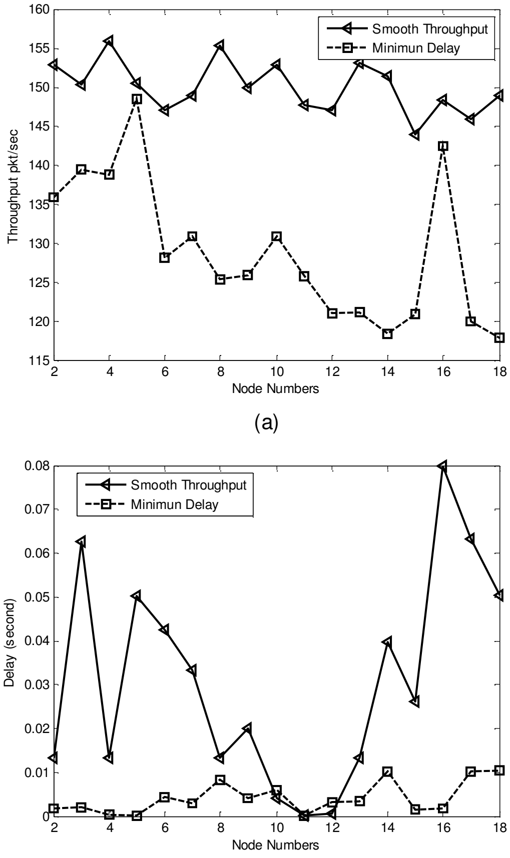
FIGURE 3: Resulted tree topology for minimum delay

In the second scenario the delay is minimized in the cost of decrease in throughput. As we can see in the resulted tree the depth is decreased and the node fan-out increased. Tuning forwarding probability (q) in this scenario is also important, if we set the q so high, the intermediate nodes will always be busy by forwarding the leaf nodes data and their delay may increase or even they may starve.

We also change our scenario a bit to compare our proposed algorithm with [14]. We increase the number of nodes from 5 to 120 with step of 5 like [14] and change our fitness function to gain maximum throughput for each network.

For each network with n nodes we set the population size to $n^{1.5}$ based on the [13] and increase the number of generations based on the size of the problem from 100 to 700.

Figure 5 show the results from [14] and figure 6 show our result. We can see that our algorithm can find a topology with better overall throughput for networks from 5 to 65 nodes and the remain is almost the same.



(a)

(b)

FIGURE 4: Per-node delay and throughput of output tree, (a) per-node throughput, (b) per-node delay

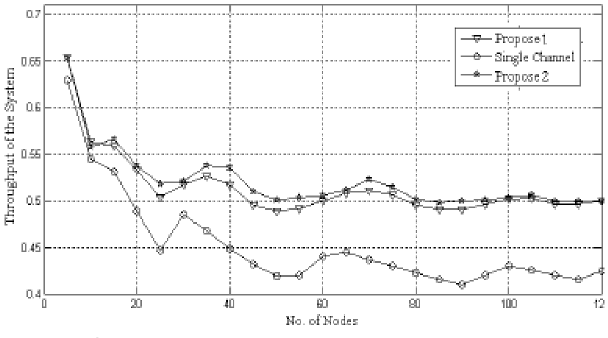


FIGURE 5: Resulted throughput from [14]

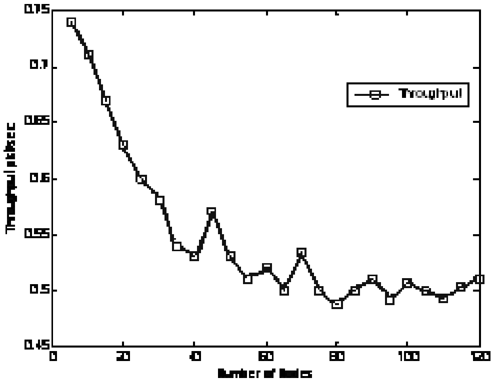


FIGURE 6: Maximum Throughput for different networks

6 CONCLUSION

In this paper, we propose a GA algorithm for finding a QoS tree for WiMAX mesh networks. Buffer size for each MSS node, transmission opportunity, lost packets, traffic load per node and delay in each node is considered. Our proposed GA algorithm is able to find QoS tree topology for given network. Any change in network parameter or nodes or application of network will not affect the algorithm at all. With defining proper fitness function this approach can always find the QoS tree with any desired delay and throughput. The obtained results show that different delay and throughput will lead to different tree's depth and fan out. For future works, node movements can be considered to overcome the limitation of the proposed algorithm, which are for the fixed nodes.

7 REFRENCES

1. IEEE 802.16d-2004,"Draft IEEE Standard for Local and Metropolitan area networks", May 2004.

2. S. Skiena, "Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica",Addison-Wesley,1990.

3. M. J. Lee, J. Zheng, Y.-B. Ko, and D. M. Shrestha, "Emerging standards for wireless mesh technology," *Wireless Commun",* 13(2): 56–63, 2006.

4. Salim Nahle, Luigi Iannone, Benoit Donnet and Naceur Malouch, "On the Construction of a wimax mesh tree", IEEE COMMUNICATIONS LETTERS, 11(12): 967 – 969, 2007

5. I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey," *Comput. Netw.*, 47(4):445–487, 2005.

6. V. Gambiroza, B. Sadeghi, and E. W. Knightly, "End-to-end performance and fairness in multihop wireless backhaul networks," in *Proc. ACM MOBICOM*, Sep. 2004.

7. J. Jun and M. L. Sichitiu, "Fairness and QoS in multihop wireless networks," in *Proc. IEEE VTC*, Oct. 2003.

8. J.-F. Lee, W. Liao, and M.-C. Chen, "An incentive-based fairness mechanism for multi-hop wireless backhaul networks with selfish nodes," *IEEE Trans. Wireless Commun.*, 7(2): 697 - 704,2008.

9. T.Liu, W.Lio , "Location Dependent Throughput and Delay in Wireless Mesh Networks" , IEEE Transactions on Vehicular Technology, 57(2):1188-1198, 2008.

10. S.Nahle, L.Iannone, B.Donnet, and T. Friedman, “Investigating depth fanout trade-off in WiMAX mesh networks”, in Proc. 1st WEIRD workshop, May 2007.

11. H. Prufer. “Neuer Beweis eines Satzes uber Permutationen” Arch.Math. Phys.27(1) : 742-744,1918.

12. J. Gottlieb, G. R. Raidl, B.A. Julstrom, F. Rothlauf ,” Pr¨ufer Numbers: A Poor Representation of Spanning Trees for Evolutionary Search” IEEE Transactions on Evolutionary Computation, 4(2):125–134, 2002.

13. F.Rothluf , D.E. Goldberg and A.Heinzel, “Network Random Keys—A Tree Representation Scheme for Genetic and Evolutionary Algorithms” MIT press journals , 10(1): 75-97,2002.

14. Ali Al-Hemyari, Nor Kamariah Noordin, Alyani Ismail, Sabira Khatun, Yaseen H. Tahir, and Y.A. Qassem “Centralized Scheduling, Routing Tree in WiMAX Mesh Networks”, Innovations in Information Technology, 2008. IIT 2008, Al Ain UAE.