Traffic Engineering and Optimization Routing for VoIP Traffic in Wireless Mesh Networks

Houda ZARHOUNI, Ghizlane ORHANOU, Said EL HAJJI, Redouane BENAINI

***Abstract*— This paper proposes a traffic engineering model to provide a better quality of service for VoIP in Wireless Mesh Networks (WMNs). We have developed an efficient algorithm in order to find the optimal path by combining the search for feasible routes with an optimization model to enhance the performance of VoIP over WMNs. Our goal of the joint optimization and the computation of feasible route is to minimize the network cost. The selection path use the independent set which are widely used in graph coloring problem to minimize interference. The idea is to build the independent set for the node Source (S) and Destination (D); also we make an intersection between those sets. Moreover, we formulate an optimization model which determines the optimal set of these criteria: hop count, link criticality and load balancing and we present initial performance results.**

***Index Terms*—** *QoS Routing; Wireless Mesh Network;* Traffic Engineering*; Optimization; VoIP*

I. Introduction

Wireless Mesh Networks [1] have emerged recently as a promising technology for next-generation wireless networking to provide better services. Because of their advantages over other wireless networks, WMNs are undergoing rapid progress and inspiring numerous applications. However, supporting delay for sensitive realtime applications such as VoIP over wireless mesh networks represent a real challenge.

VoIP over Wireless LAN (WLAN) is known many problems, due to the loose nature of wireless network, such as: providing QoS sensitive VoIP traffic in presence of the best effort TCP data traffic, packet loss due to channel interference, high overhead of the protocol 802.11/IP/UDP/RTP for each VoIP packet with 20 bytes payload. The above problems become even more severe when supporting VoIP over multihop mesh networks.

Houda ZARHOUNI, Laboratory of mathematics, computer science & Applications, University of Mohamed V Agdal, Faculty of Science Rabat, Morocco, (email: h.zarhouni@gmail.com)
Ghizlane ORHANOU, Laboratory of mathematics, computer science & Applications, University of Mohamed V Agdal, Faculty of Science Rabat, Morocco, (email: ghizlane.orhanou@gmail.com)
Said EL HAJJI, Laboratory of mathematics, computer science & Applications, University of Mohamed V Agdal, Faculty of Science Rabat, Morocco, (email: elhajji@fsr.ac.ma)
Redouane BENAINI, University of Mohamed V Agdal, Faculty of Science Rabat, Morocco, (email: redouane.benaini@gmail.com)

The initial study on the performance of real-time applications over 802.11 was presented by authors in ref. [2]. Study focusing specifically on VoIP over 802.11 in [3] considered the delay and loss characteristics under PCF and DCF is provided. Another recent work on VoIP over WLAN in [4] present analytical studies on the number of calls that can be supported in a single hop WLAN.

The study reports are increased the payload per frame increases the number of supported calls. The work in [5] addresses a more challenging problem of determining the capacity of a call along the path in multi-hop mesh network.

For multihop wireless networks, several modifications on-demand routing protocols have been proposed to support QoS for real-time applications [6]. In spirit of these techniques propose or modify an on-demand routing protocol to support QoS. Furthermore, the above on-demand protocols require exchange of multihop messages to find the route and result in significant call set up time.

The work presented in this paper aims to provide a quality of service of VoIP by using Traffic Engineering concept and routing constraint in wireless mesh network. In fact, the objective of routing constraint is *to reduce the blocking probability* by satisfying a maximum of requests for VoIP calls, *to reduce costs* by choosing a path with a minimum number of hops and distribute the load evenly in the network; this is called *load balancing*.

We develop an efficient algorithm for routing to find a feasible route. For that, we use the independent set in the graph to find the route that connected the source to a destination. This algorithm determines all feasible paths in the network. The search of route using independent set results on several possible routes. On the other hand, we define an optimization cost function according to criteria: hop count, links criticality and the network load in order to find the optimal feasible path.

We propose an integrating solution based on the combination of load balancing, RCS (Routing using Calls Statistics), and hop count. Our goal is to show the importance of the joint consideration of the corresponding criteria.

Finally, we introduce a new parameter for the load metric element *(threshold)* that will control load balancing influence in the overall cost function by limiting its undesirable effects under light load.

II. TRAFFIC ENGINEERING FOR WIRELESS MESH NETWORK

In this section, we define the different objectives of traffic engineering mechanisms for wireless Mesh networks.

1. *Reducing blocking probability*

One goal of traffic engineering is to reduce the blocking probability, ensuring that a maximal number of requests are accepted in the network; hence it maximizes operator revenues and enhances client satisfaction.

Routing using Calls Statistics (RCS) [5] is an important proposal for VoIP traffic which deals with the reduction of blocking probability. A route computation algorithm for VoIP calls over mesh network in [5] selects path using call statistics to minimize future call rejections. The technique is to find feasible paths that might exist and which should be selected for routing the incoming call. However, a potentially better approach for path selection could be to allow more calls to be supported in future. Such an approach is important to VoIP service providers that are interested in supporting as large a call volume as possible while maintaining call quality. The exact sequence of future call arrivals may be unknown; however, an approach can be designed simply based on long-term call statistics, specified in terms of the probability *p(a, b)* of a mesh node pair *(a, b)* to be the source and destination of a new call. Such statistics may be available to the service providers collected via long term measurements.

A similar idea called *Minimum interference routing algorithm (MIRA)* [7] has been proposed for traffic engineering work in wireline networks. The basic principle behind MIRA is to define a notion of criticality for a given link and select a route that best avoids critical links. For a given source $\mapsto$ and destination *b,* a link is critical if it belongs to the min-cut [8] between *a* and *b*. The level of criticality is determined by *p(a,b).*

A weight is assigned to each link based on link criticality. Weights are defined such that a route computation becomes as simple as finding a shortest path on the weighted graph after the feasibility [5] has been ascertained. All the critical links for a node pair can be found by running the Ford-Fulkerson max-flow algorithm [8] just once. Since we have a wireless medium, any link which interferes with the critical link should also be a critical link, because adding traffic on that link reduces the maximum flow between the node pair as well.

2. *Minimizing network costs*

Static metrics such as hop count have been traditionally incorporated in routing algorithms in order to achieve a minimum network cost objective. Hop count [9] is the most commonly used routing metric in routing protocols such as AODV [10], DSDV [11]. Hop count treats all links in the network to be alike and finds paths with minimum number of hops. It does not consider the difference of transmission rates and packet loss ratios or interference experienced by the links. Hence hop count results in bad performance in terms of rejection ratio in a highly loaded network.

3. *Load balancing*

Load balancing [12] is an important factor for network congestion reduction. The idea is to have some equilibrated load distribution in the network that improves the overall situation. In fact, Load balancing becomes solution key to support Quality of Service, especially in wireless mesh networks. In this paper, we consider the simple way of doing load balancing in traffic engineering by routing over the least loaded links. We should point out that this strategy is a basic form of load balancing and is better qualified as load minimization.

III. OPTIMIZATION

The study of previous methods helped us to identify the blocking probability reduction, the network cost minimization, and the load balancing as relevant criteria for VoIP traffic in wireless mesh network. First, we present an integrating solution based on combination of load balancing, Minhop and Routing using calls statistics. Second, we develop our algorithm to finding feasible path in the network. Finally, we define a cost function to optimize according to criteria: hop count, links criticality and the network load in order to finding optimal feasible path.

1. *Integration solution*

The objective is to reduce the *blocking probability* by satisfying a maximum of requests for VoIP calls, to reduce *costs* by choosing a path with a minimum number of hops and distribute the load evenly in the network by using *load balancing*. The first method is the reduction of costs based on the hop count that is designed to minimize the number of hops and also the path length. The second method is load balancing, that is an important factor for network congestion reduction. The idea is to reduce the load by selecting the least loaded links. The third method is to reduce the blocking probability based on routing using calls statistics (RCS). The objective of RCS is identifying critical links, assignment of weights and calculating the shortest path.

2. *Algorithm*

A mesh network can be modeled as an undirected graph *G = (V, E)* (referred to as the communication graph) where *V* is the set of vertices in the graph, representing the wireless router nodes in the network and *E* is the set of the connections which exist between these nodes.

ISBN: 978-988-19252-1-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

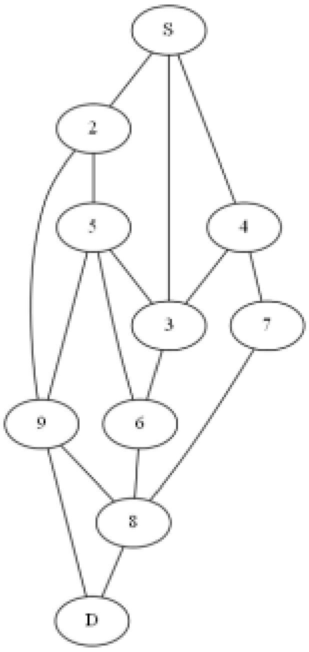


Figure 1. Modeling wireless mesh network as a graph

Algorithm : (G,s,d)

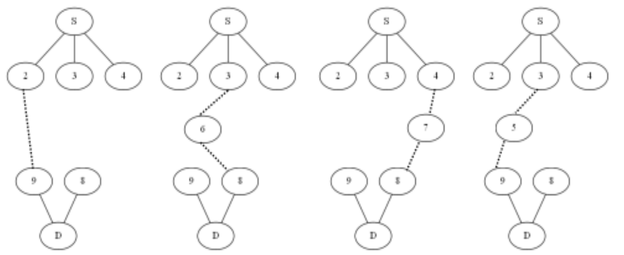
Input: 1. Given G(*V*, *E*)
2. Node source *S*, Node Source *D*

Output: feasible shortest paths P_1, P_2, .. P_i, .. from *S* to *D* in G

2. Find the independent set *A(s)* and *A(d)* where
   A(s)={(s,x_1) ; (s,x_2) ; …}
   A(d)={(d,y_1) ; (d,y_2) ; …}
3. **For each** $x_i \in$ A(s)
   Find A(x_i)
4. **For each** $y_j \in$ A(d)
   Find A(y_j)
5. **if** $\exists$ z $\in$ A(x_i) $\cap$ A(y_j)
6. Our Shortest Feasible Path
   P= { A(x_i) , A(y_j)}
   P'={s, x, z, y_j, d}
7. **else**:
   For each $x_i \in$ A(x_{ih}) and **for each** $y_j \in$ A(y_{jk})
8. find A(x_{ih}) and A(y_{jk})
10. **if** $\exists$ z $\in$ A(x_{ih}) $\cap$ A(y_{jk})
   Step (6)
11. **else**:
   Repeat:
   Step (8)
   Until find z $\in$ A(x_{ih}) $\cap$ A(y_{jk})
12. Our Shortest Feasible Path *P* from s to d in G
   Convert P= {A(x_i), A(x_{ih}), A(y_{jk}), A(y_j)}
   to *P'*={s, x_i, x_{ih}, z, y_{jk}, y_j, d}

Return *P'*

***Algorithm**:* Our goal is to search all possible feasible paths by using the independent sets. An independent set [13] is a set of vertices in a graph G, such that there is no two vertices among the graph ones that are adjacent. We use the independent set in the graph to find the route that connected the source to a destination. This process of determination of the independent set is repeated until forming a path that connects the source to the destination. First, we find independent set for node source S, noted A(S) = {(s,x_1); (s,x_2); …} and respectively for node destination D, noted A(D)={(d,y_1); (d,y_2); …}. Then find for each node x_i connected with a node source S, the independent set A(x_i)={(x_i,x_{i1}) ; (x_i,x_{i2}) ; …} and respectively find for each node y_i connected with a node destination D, the independent set A(y_i)={(y_i,y_{j1}) ; (y_i,y_{j2}) ; …}. Then we search for a node z $\in$ A(x_i) $\cap$ A(y_j), where z form edge with one vertex x_i and also form edge with one vertex y_j. The path P= A(x_i) $\cap$ A(y_j) represents a feasible shortest path. Thus, if we don't find z satisfying A(x_i) $\cap$ A(y_j), (i.e there is no node z forming an edge with one of node x_i and one of node y_j) we construct the independent set i.e we determine A(x_{i1}), A(x_{i2}), … and respectively A(yj1), A(yj2), … then we establish the intersection A(xi1) $\cap$ A(yj1), A(xi1) $\cap$ A(yj2), A(xi2) $\cap$ A(yj1), A(xi2) $\cap$ A(yj2), … While those intersections is null, we repeat this processes as necessary until we finding z that form the path from source S to Destination D. At result the path is *P={s, x_i, … , x_{ih}, z, y_{jk} , … , y_j , d}*



(a) *(b)* *(c)* *(d)*

Figure 2. ***(a):*** P_1= {A(2), A(9)}= {s, 2, 9, d}; ***(b):*** P_2= {A(3), A(8)}= {s, 3, 6, 8, d}; ***(c):*** P_3= {A(4), A(8)}= {s, 4, 7, 8, d}; ***(d):*** P_4= {A(3), A(9)} = {s, 3, 5, 9, d}

Example

Find an optimal path from node source S to destination node D:
Determinate the independent set of A(s) and A(d)
A(s)= {(s,2) ; (s,3) ; (s,4)} and A(d)={(8,d) ; (9,d)}
There is no z$\notin$ A(s) $\cap$ A(d). For this, we find the independent set for all nodes {2,3,4} of A(s) and respectively for {8,9} of A(d)

For A(s):
A(2)= {(2,5) ; (2,9)}
A(3)= {(3,4) ; (3,5) ; (3,6}
A(4)= {(4,3) ; (4,7)}

For A(d):
A(8)= {(6,8) ; (7,8)}
A(9)= {(2,9) ; (5,9)}

Find intersection between sets (A(2), A(3), A(4)) and (A(8), A(9))
The feasible paths are:
- 2 $\in$ A(2) $\cap$ A(9) then P_1= {A(2), A(9)}= {s, 2, 9, d}
- 6 $\in$ A(3) $\cap$ A(8) then P_2= {A(3), A(8)} = {s, 3, 6, 8, d}
- 7 $\in$ A(4) $\cap$ A(8) then P_3= {A(4), A(8)} = {s, 4, 7, 8, d}
- 5 $\in$ A(3) $\cap$ A(9) then P_4= {A(3), A(9)} = {s, 3, 5, 9, d}

3. simulation model

The search of route using independent set can result on several possible routes; we are trying to select the optimal path by optimizing a cost function based on criteria such as path length, number of hops, critical links and network load:

Cost (e) = f (hop count, critical links, network load, etc)

A path in *G* is denoted by *P* if it links the source node *S* to destination node *D*. A Quality of Service routing algorithms are those which calculate the path *P*, among the *k* paths, optimizing one or more constraints of QoS.

Model:

- *G = (V, E)* is a graph.
- Foreach request of bandwidth *b* between nodes *S, D*
  Given the capacity *cap(e)* and the load *load(e)* on each edge $e \in E$.
  Determine the optimal set of binary variables *x(e)* and *y(e)*, that:

Minimize : $\sum_{e \in E} cost(e) \times [x(e) + y(e)]$,

Subject to: $[x(e) + y(e)] \times [load(e) + b] \leq cap(e)$, *for* $e \in E$

With:

$$cost(e) = \begin{cases} n\_hops & \text{hop count} \\ load(e) / cap(e) & \text{load balancing} \\ criticality(e) & \text{RCS (Routing using Call Statistics)} \end{cases}$$

- **Results:** Optimal path

Combining Algorithm 1 with our Model optimization; we can derive an efficient algorithm for selecting an optimal shortest feasible path as described in Algorithm 2.

Algorithm 2:

Input: Given G(*V*, *E*)
Node source *S*, Node Source *D*
Output: Optimal Shortest Feasible Path
Step1: Applied the *algorithm 1* for searching feasible paths in
the graph G.
Set $P_1, P_2, \ldots, P_i \ldots$ the output of the *algorithm 1*.
Step2: Applied the *Model 1* for each path $P_1, P_2, \ldots, P_i \ldots$
Return: Optimal Shortest Feasible Path P_i

IV. Performance Evalution

We discuss the simulation of integrating solution that is based on the combination of load balancing, RCS (Routing using of Calls Statistics) and MinHop. Our goal is to show the importance of the joint consideration of the corresponding criteria. Thus, the main challenge is to define the optimal weighting (T1, T2, T3) for each element in the integrating cost function given by (Eq. 1). First, the weight associated with MinHop should be increased to emphasize its good performance under light load. For instance, according to (Eq. 2), T1 is inversely proportional to the total network load. We can see that T1 is predominant under light load and starts to decrease as the total network load increases to reach the total network capacity. Second, RCS should really get involved when links criticality is changing (links are getting rapidly loaded). We choose T2 (Eq. 2) to be proportional to the network load. Third, we introduce in (Eq. 3) a new parameter for the load metric element that will control load balancing influence in the overall cost function by limiting its undesirable effects under light load. Moreover, constants a, b and c are used in order to scale the numeric values to a comparable range.

$$cost(e) = T_1 + T_2 \times criticality(e) + T_3\, load(e) \qquad (1)$$

$$T_1 = a \times total\_cap / total\_load; \quad T_2 = b \times total\_load / total\_cap; \quad T_3 = c \qquad (2)$$

$$load(e) = \begin{cases} load(e) / cap(e) & load(e) > threshold \\ 0 & otherwise \end{cases} \qquad (3)$$

Here, we present the results of our joint model optimization and routing algorithm. The evaluation requires a large number of nodes. Indeed, we evaluate routing strategies using call statistics (RCS), shortest feasible path (SFP) and max residual feasible path (MRFP). Graph in *Figure 3* represents the percentage of calls rejected, given as a function of average number of calls. We compare our algorithm for searching the optimal path (*Algorithm 2*) to those obtained from previous works in [5]. *Figure 3* shows the percentage of calls rejected for each routing schema in a uniform topology. We remark that *Algorithm 2* drops a few numbers of calls, which is similar to RCS routing, in a heavily skewed traffic pattern.

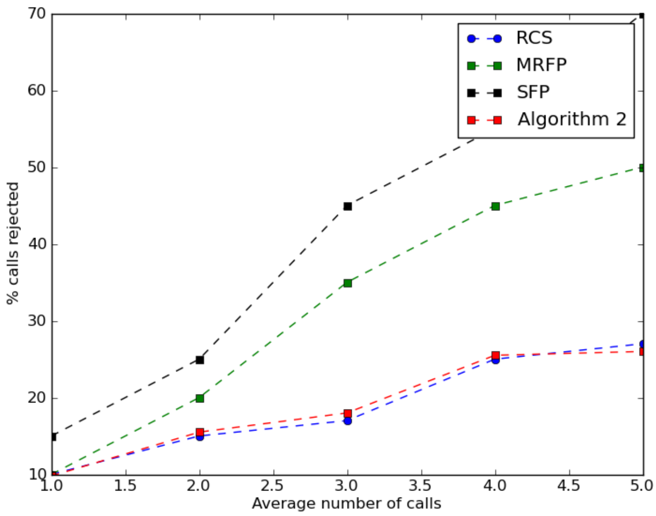


Figure 3. Show the percentage of calls rejected for each routing schema in uniform topology. Algorithm2 drops lesser number of calls, similar to RCS routing, in a heavily skewed traffic pattern

ISBN: 978-988-19252-1-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

V. CONCLUSION

We have presented an approach based on traffic engineering and routing constraint to provide quality of service of VoIP in wireless mesh network. Moreover, because of the wireless interference, looking for a feasible route to accommodate an incoming call can be computationally hard. We have used the independent set to avoid the critical links. We have also optimized a cost function according to routing metrics such as hop count, links criticality and the network load in order to find an optimal path in the network. Our modelling improves performance significantly compared to naive methods.

REFERENCES

[1] I.F. Akyildiz, X. Wang, A survey on wireless mesh networks, IEEE Communication Magazine 2005.
[2] J.-Y. Yeh and C. Chen, "Support of multimedia services with the ieee 802.11 mac protocol," in In Proceedings of ICC 2002.
[3] D. Hole and F. Tobagi, "Capacity of an ieee 802.11b wireless lan supporting voip," in In Proceedings of ICC 2004.
[4] S. Garg and M. Kappes, "Can I add a VoIP call?," in IEEE International Conference on Communications(ICC), (Anchorage, Alaska), 2003.
[5] A. Kashyap, S. Ganguly, S. R. Das, S. Banerjee, «VoIP on Wireless Meshes: Models, Algorithms and Evaluation», IEEE Journal on Selected Areas in Communications, vol. 24, no. 11, pp. 2147–2158, Nov. 2007.
[6] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Measurement-based Models of Delivery and Interference in Static Wireless Networks," in SIGCOMM, 2006.
[7] M. Kodialam and T. V. Lakshman, "Minimum Interference Routing with Applications to MPLS Traffic Engineering," in Proc. of IEEE INFOCOM, 2000.
[8] L. R. Ford and D. R. Fulkerson, "Flows in Networks," Princeton University Press, 1962.
[9] W. Stallings, Data and Computer Communications, Prentice Hall, 1997.
[10] I.D. Chakeres, E.M. Belding-Royer, AODV routing protocol implementation design, in: Proceedings of the International Workshop on Wireless Ad Hoc Networking (WWAN), Tokyo, Japan, 2004.
[11] C. Perkins, Bhagwat, Highly dynamic destination-sequence distancevector routing (DSDV) for mobile computers, in: ACM SIGCOMM Computer Communication Review, 1994, pp. 234–244.
[12] M. Brahma, K. W. Kim, A. Abouaissa, and P. Lorenz. A Load-Balancing and Push-Out Scheme for Supporting QoS in MANETs. Telecommunication Systems, 30(1–3) :161–175, Novembre 2005.
[13] W. Arbaugh, S. Banerjee, A. Mishra, "Weighted Coloring Based Channel Assignment for WLANs," ACM SIGMOBILE Mobile Computing and Communications Review, July 2005, vol. 9, no. 3, pp. 19-31.

ISBN: 978-988-19252-1-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)