

On the Complexity of Bandwidth Constrained Routing in Wireless Mesh Networks

(December 2008)

Bahador Bakhshi

Computer Engineering Department,
Amirkabir University of Technology,
Tehran, Iran
bbakhshi@aut.ac.ir

Siavash Khorsandi

Computer Engineering Department,
Amirkabir University of Technology,
Tehran, Iran
khorsand@aut.ac.ir

Abstract

We investigate the bandwidth constrained routing problem in wireless mesh networks. Whereas the bandwidth constrained routing problem is solvable in polynomial time in wired networks, it is complicated by the intra-flow interference in wireless networks. We prove that the problem is NP-Complete using a general interference model. We propose a heuristic routing algorithm, ADFS, which is an augmented version of DFS algorithm. We also develop an Integer Linear Programming model of the problem which is used to evaluate performance of algorithms in term of finding feasible path. Our extensive simulation results show that the ADFS algorithm outperforms other basic routing algorithms and can find a feasible path for 93% of requests in the worst case, if such path exists.

1. Introduction

QoS routing received considerable attention in the last decade. The research community has extensively studied the problem in wired networks. Many in-depth analysis of the complexity of the problem and approximation algorithms for the problem are available in literature [1]. In the recent years, the need for QoS routing algorithms in wireless networks has increased primarily due to increasing popularity of multimedia applications in the wireless networks.

In this paper, we focus on QoS support in terms of bandwidth allocation because it is known that by controlling bandwidth allocation, delay and jitter can also be controlled [2]. To allocate the required bandwidth, a path is needed from the source node to the destination node that has sufficient resources. Finding the path is known as the *bandwidth constrained routing*

problem, and the found path is named *feasible path*. The problem is easily solved in wired networks but it is challenging in wireless networks due to the shared nature of wireless channel. Interference models are used to take the shared nature of the media into account. The complexity of the bandwidth constrained routing problem is directly related to the interference model. Previous works used very limited interference models and have relied on the distributed flooding-based routing protocols to find the path [3], [4], [5].

In recent years, introducing the wireless mesh networks (WMN), that have a static multi-hop wireless infrastructure, intrigues researchers to use the link state routing protocols [6]. Using the protocols, each node has a complete view of the network graph and can use centralized algorithms to find paths. Whereas the complexity and efficiency of the algorithms is the major issue, there is very limited attention on designing efficient centralized bandwidth constrained routing algorithms and analyzing the complexity of them in literature. In Section 4, we will prove that the bandwidth constrained routing in wireless networks is NP-Complete. Thus, finding feasible path for a given flow is a challenging open problem. An efficient algorithm for the problem should have great ability to find existing feasible paths in reasonable time. To our best knowledge, none of the previous works provided analysis or evaluation of the ability of their proposed algorithms to find existent feasible paths.

In this paper we study the bandwidth constrained routing problem using a general interference model, design an efficient heuristic algorithm and evaluate its ability to find existing feasible paths. Specifically, our contributions with this paper include:

- Analysis of the complexity of the bandwidth constrained routing problem in wireless networks and proof of its NP-Completeness in Section 4.

- Highlighting the difference between the building blocks of bandwidth constrained routing in wired and wireless networks and developing appropriate design considerations in Sections 3 and 5.
- Developing a CAC algorithm and a heuristic routing algorithm based on the design considerations in Section 5.
- Developing an exact Integer Linear Programming (ILP) model of the problem, as the reference point, to evaluate performance of algorithms in term of their ability to find existing feasible paths in Section 5.

It should be noted that whereas efficient utilization of network resources is one of the main consideration in routing algorithms, designing an optimal bandwidth constrained routing algorithm is very difficult. Finding the optimal paths is an NP-Hard optimization problem because its underlying decision problem, finding feasible paths without any optimization, itself is NP-Complete. The optimization problem is not discussed here; we only focus on finding feasible path in this paper.

The remaining sections are structured as follows. We discuss related works in Section 2. The problem is explained in details in Section 3. Our proof of NP-Completeness of the Bandwidth Constrained Routing in Wireless Networks is presented in Section 4. We develop our CAC and routing algorithms in Section in 5. Simulation results are presented in Section 6. We present brief conclusions in Section 7.

2. Related Work

Several solutions have been suggested to achieve end-to-end QoS, particularly in term of required bandwidth, in multi-hop contention based wireless networks which is also the focus of this paper. The problem has been mainly studied in the context of wireless ad-hoc networks [2], [3], [4], [7], [8], [9]. Most works focused on the dynamic nature of ad-hoc networks and proposed flooding based routing algorithms to deal with the problem [3], [4], [5]. Several mechanisms were proposed to reduce the overhead of the flooding [4], [7], [10]. Some authors considered the mobility in ad-hoc networks and proposed several algorithms to predict the route break and rerouting [11], [12].

Interference and its impact on QoS routing were not modeled properly in the previous works. Some of them did not take account of interference at all [7]. In [3], Xue *et al.* considered only the half-duplex operation of radio transceivers. Zhu *et al.* in [9] used the node exclusive interference model. In the most recent and complete work, [2], Yang proposed a few mechanisms to find all nodes in the carrier sense range of a given

node but still relied on flooding to find a path and furthermore, she did not provide any analysis of the complexity of the routing algorithm.

Recently, authors in [13] proposed a DFS based routing algorithm for Multi Constrained Path (MCP) problem. While our proposed algorithm is also based on DFS, we explain in Section 5 that the algorithms for the MCP problem are not efficient algorithms for the bandwidth constrained routing.

The most closely related works to this paper are [14] and [15]. Chiu and *et al.* in [14] studied the effect of MAC-layer scheduling policy on the complexity of the bandwidth constrained routing problem. They considered two cases: 1) CSMA MAC and 2-hop interference model and 2) CDMA MAC and 1-hop interference model, they showed the intractability of the problem in the both cases. Instead of proposing a routing algorithm, they design a MAC-layer scheduling to solve the problem in polynomial time. In [15], authors used k-hop interference model and proved the intractability of the problem. They proposed a heuristic routing algorithm but didn't provide an evaluation of the ability of the algorithm to find the existent feasible paths. In this paper, we prove the NP-Completeness of the problem by reducing the MCP problem to it. The proof shows the relationship between these two problems. The MCP problem is a well studied problem in the literature; there are many in-depth analysis of the complexity of the problem and many exact and approximation algorithms for it. We will use the relationship to analysis the complexity of the bandwidth constrained routing problem that it indicates there are special cases of the problem which can be solved in polynomial time. Moreover, the derived relationship can be used to design optimal algorithms for the bandwidth constrained routing. We evaluate our proposed algorithms based on their ability to find the existent path using a developed ILP model as the reference point; furthermore, we investigate the effect of the *interference range to transmission range* ratio, r , on the performance of the algorithms.

3. Bandwidth Constrained Routing

Bandwidth constrained routing problem is a special case of QoS routing problem in which the QoS metric is the *required bandwidth*. A *feasible* path for the problem is a path that allocating the required bandwidth along it does not violate the link/node capacity constraint. If network cannot guaranty the required bandwidth of a flow, the flow is not allowed to enter the network. Call Admission Control (CAC) algorithm decides to accept or reject requests. Usually, routing and CAC algorithms are integrated. If routing algorithm can find a feasible path, the request is accepted, otherwise it is rejected.

Two parameters are needed to find the feasible path: 1) *available bandwidth* of each link/node and 2) *bandwidth consumption* of the request at each link/node. There is a significant difference between wired and wireless networks in term of obtaining the parameters. In wired networks, each node knows the available bandwidth of its links and the bandwidth consumption of a request at each link is equal to its required bandwidth. But in wireless networks, there is a distinction between bandwidth request and bandwidth consumption because of *interference*. Due to the interference, all nodes in an interference region share the bandwidth of the *region*, and hence, the available bandwidth is defined per region; consequently, a particular flow passing through a node would consume bandwidth on all the nodes in the interference region of the node. Besides, each node belongs to multiple overlapping interference regions. This complicates the *computation* of the available bandwidth. We will formulate the problem in the following.

Suppose that C_u is the channel capacity at node u and b_u is the aggregate transmitted bandwidth by node u . I_u is a set of nodes that are in the interference region of node u . $v \in I_u$ if and only if u and v cannot transmit simultaneously. We assume that all nodes are *static*, have same transmission range TR and same interference range IR , and $IR = r.TR$. We also assume that the network is *single radio* and *single channel*. The available bandwidth of each node is computed using following definitions.

Definition 1. Available Node Bandwidth (ANB) of node u is $ANB(u) = \max\{0, C_u - \sum_{v \in I_u} b_v\}$.

Definition 2. Available Area Bandwidth (AAB) of node u is $AAB(u) = \min_{v \in I_u} ANB(v)$.

Please note that if $C_u < \sum_{v \in I_u} b_v$ then $ANB(u) = 0$. In this case, not only no new flow can be admitted in u , but also the bandwidth requirement of some flows passing through the interference region of node u is violated. This case could happen if there is not any CAC mechanism in network and any flow is allowed to enter the network. This case must be avoided in bandwidth constrained routing; therefore, the CAC and routing algorithms must satisfy the constraint $C_u \geq \sum_{v \in I_u} b_v, \forall u \in V$.

To clarify the definitions, consider Fig. 1. Interference ranges of nodes a , c , e , and g are specified by the dashed circles. Fig. 1 also shows 3 single hop flows f_1 , f_2 , and f_3 . Assume that the $C_u = 10\text{Mbps}$ and the required bandwidth of flows f_1 , f_2 and f_3 are 1Mbps, 1Mbps, and 5Mbps respectively.

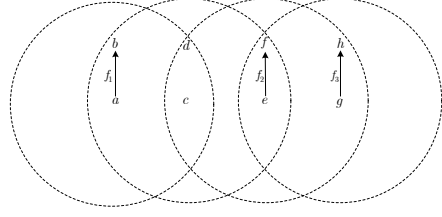


Figure 1. Example of AAB computation

Using the definitions, we have $b_a = 1, b_e = 1, b_g = 5$, $ANB(a) = 10 - b_a = 9$, $ANB(c) = 10 - b_a - b_e = 8$, $ANB(e) = 10 - b_e - b_g = 4$, $ANB(g) = 4$, $AAB(a) = 8$, $AAB(c) = AAB(e) = AAB(g) = 4$.

The definition of $ANB(u)$ is based on the *sufficient condition* but **not** the necessary condition for flow vector feasibility. To see why, consider computing $ANB(c)$. Nodes a and e are not in interference region of each other and can transmit simultaneously. If they schedule the transmissions at a same time, their transmissions are overlapped and $ANB(c) = 10 - \max\{b_a, b_e\} = 9\text{Mbps}$. The necessary and sufficient conditions for feasibility of a flow vector were derived in [16]. But unfortunately, checking the conditions needs finding maximal cliques in the interference graph, which is an NP-Complete problem; furthermore, “the conditions only guaranty the *existence* of a feasible schedule In fact, distributed scheduling mechanisms like 802.11b are seen to be quite far from the optimally feasible schedule.” [17]. Due to the shortcoming of the necessary condition and for sake of simplicity, we will use the sufficient condition in this paper. But it should be noted that the proposed routing algorithm is independent of the available bandwidth model and any other model can easily be integrated with it.

Nodes on the path of a flow also interfere with each other; the interference is called *intra-flow interference* [6]. To computing the *bandwidth consumption* of a flow we need to consider the intra-flow interference. It causes that a flow consumes bandwidth of an interference region multiple times. The exact value of the consumption depends on the *routing*. To clarify the problem, consider Fig. 2. The transmission and interference ranges of nodes c and i are shown by solid and dashed circles respectively. Assume that $AAB(u) = 10\text{Mbps}$. There is a request to find a path from a to f and the required bandwidth is 2Mbps. If path $p = \langle a, b, c, d, e, f \rangle$ is selected, the flow consumes 10Mbps in the interference region of node c because five nodes of the path are in the interference region of node c and each consumes 2Mbps to transmit the flow.

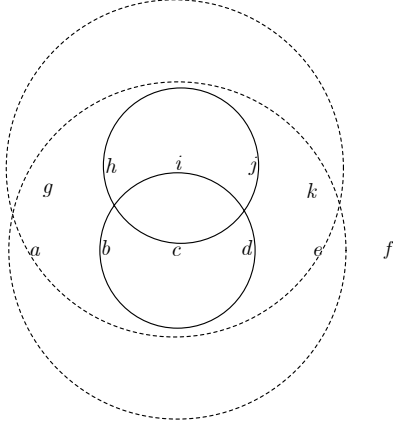


Figure 2. Dependency between routing and bandwidth consumption

Therefore, $AAB(a) = AAB(b) = AAB(c) = AAB(d) = AAB(e) = ANB(c) = 0$. The path is a feasible path because $AAB(u) \geq 0 \quad \forall u$. If the path $p = \langle a, g, h, i, j, k, e, f \rangle$ is selected, the flow should consume 14Mbps in the interference region of node i to meet the requirement, which is not possible. Therefore, the path is not a feasible path.

In wired networks, to find a feasible path for a new request requires bandwidth b , it is sufficient to construct the *feasible residual network* by pruning all links that their available bandwidth is less than b . The constructed network has an important property: *every path* from the source to the destination in the feasible residual network is a *feasible path*. In wireless networks the concept of feasible residual network is not straightforward. If we prune network for the new request by removing every node u if $AAB(u) < b$, we get a *residual network* that each path from the source to the destination in the network is **not** a *feasible path*. For example, pruning network in Fig. 2 does not change it because $AAB(u) > 2\text{Mbps} \quad \forall u$. But the path $p = \langle a, g, h, i, j, k, e, f \rangle$ is not a feasible path in the network as explained before. In the following, we prove that not only building the feasible residual network but also *finding a feasible path* for bandwidth constrained routing problem in wireless networks is NP-Complete problem.

4. Complexity Analysis

In this section we explore the bandwidth constrained routing in wireless networks problem more formally and prove its NP-Completeness.

Networks is modeled by digraph $G = (V, E)$, where the V is the set of n vertices and E is a set of m edges. $\forall u \in V$ corresponds to a node in the network. $(u, v) \in E$ if and only if u and v are in

transmission range of each other. The bandwidth constrained routing problem is defined as following:

Problem: Bandwidth Constrained Routing in Wireless Networks (BCRWN).

Instance: $G = (V, E)$, $I_u, C_u \quad \forall u \in V$, nodes s, t and bandwidth requirement b .

Question: Is there $p = \langle s = u_1, u_2, \dots, u_k = t \rangle$ such that $\sum_{v \in p \cap I_u} b \leq C_u \quad \forall u \in V$?

Please note that we don't apply any restriction on I_u and C_u . Theorem 1 shows the complexity of the BCRWN problem.

Theorem 1. Bandwidth Constrained Routing in Wireless Networks problem is NP-Complete.

Proof. It is easy to see that the BCRWN \in NP. It is known that Multi Constrained Path (MCP) problem in wired networks is NP-complete [18]. We transform the MCP problem to the BCRWN problem. The MCP problem is defined as following:

Problem: Multi Constrained Path in Wired Networks (MCP).

Instance: $G = (V, E)$, weights $W_{uv} = [w_{uv}^1, \dots, w_{uv}^q]$ $\forall (u, v) \in E$, constraints $B = [b^1, \dots, b^q]$ and nodes s, t .

Question: Is there $p = \langle s = u_1, u_2, \dots, u_k = t \rangle$ such that $\sum_{(u,v) \in p} w_{uv}^i \leq b^i \quad 1 \leq i \leq q$?

For an arbitrary instance of the MCP problem, we construct graph $G' = (V', E')$ as follows. Set $r = IR/TR$ to an arbitrary large value. Add $x_1 \dots x_q$ nodes to V' and place them in space such that their interference regions do **not** overlap. For $\forall v \in V$ add v' to V' , place it near the interference region of node x_1 but outside the region. For $\forall (u, v) \in E$ and $1 \leq i \leq q$ add $u_{uv,1}^i, u_{uv,2}^i, \dots, u_{uv,w_{uv}^i}^i$ to V' and place them in the interference region of node x_i such that create a chain, $\langle u_{uv,1}^i, u_{uv,2}^i, \dots, u_{uv,w_{uv}^i}^i \rangle$. Connect $u_{uv,w_{uv}^i}^i$ to $u_{uv,1}^{i+1}$ $1 \leq i < q$, u' to $u_{uv,1}^1$ and $u_{uv,w_{uv}^q}^q$ to v' by adding as many as required nodes u'' . For $\forall v \in V' - \{x_1, \dots, x_q\}$ set $C_v = \infty$ and for $\forall x \in \{x_1, \dots, x_q\}$ set $C_x = b^i$.

A sample network and corresponding transformed network are shown in Fig. 3. In the figure, label of each link is the W and for sake of simplicity; we show some nodes u'' and interconnection between them by the dashed lines.

It is not hard to see that G' can be constructed from G, W and B in polynomial time. Suppose two paths as follows. $p = \langle s = u_1, u_2, \dots, u_k = t \rangle$ and

$$\begin{aligned}
p' = & \langle s' = u'_1, u_{u_1 u_2, 1}^1, \dots, u_{u_1 u_2, w_{u_1 u_2}^1}, u'', \dots, u_{u_1 u_2, 1}^2, \dots, \\
& u_{u_1 u_2, w_{u_1 u_2}^2}, u'', \dots, u_{u_1 u_2, 1}^q, \dots, u_{u_1 u_2, w_{u_1 u_2}^q}, \dots, \\
u'_2, & u_{u_2 u_3, 1}^1, \dots, u_{u_2 u_3, w_{u_2 u_3}^1}, u'', \dots, u_{u_2 u_3, 1}^2, \dots, \\
& \dots, u_{u_2 u_3, w_{u_2 u_3}^2}, u'', \dots, u_{u_2 u_3, 1}^q, \dots, u_{u_2 u_3, w_{u_2 u_3}^q}, \dots, \\
u'_{k-1}, & u_{u_{k-1} u_k, 1}^1, \dots, u_{u_{k-1} u_k, w_{u_{k-1} u_k}^1}, u'', \dots, u_{u_{k-1} u_k, 1}^2, \dots, \\
& u_{u_{k-1} u_k, w_{u_{k-1} u_k}^2}, u'', \dots, u_{u_{k-1} u_k, 1}^q, \dots, u_{u_{k-1} u_k, w_{u_{k-1} u_k}^q}, \dots, \\
u'_k & = t' \rangle.
\end{aligned}$$

We claim that p' is a feasible path for the request from s' to t' with bandwidth requirement 1 if and only if p is a feasible path from s to t subject to the constraints B . According to the feasibility of p'

$$\sum_{v \in p' \cap I_{x_i}} b \leq C_{x_i} \quad \forall x_i \in \{x_1, \dots, x_q\}$$

$$\left\{ \begin{array}{l} p' \cap I_{x_i} = \\ \quad \{ \langle u_{u_j u_{j+1}, 1}^i, \dots, u_{u_j u_{j+1}, w_{u_j u_{j+1}}^i} \rangle, 1 \leq j \leq k-1 \}, \\ b = 1, \\ C_{x_i} = b^i \end{array} \right.$$

$$\rightarrow \sum_{v \in p' \cap I_{x_i}} b = \sum_{j=1}^{k-1} \left(\sum_{u \in \langle u_{u_j u_{j+1}, 1}^i, \dots, u_{u_j u_{j+1}, w_{u_j u_{j+1}}^i} \rangle} 1 \right) =$$

$$\sum_{j=1}^{k-1} w_{u_j u_{j+1}}^i = \sum_{(u_j, u_{j+1}) \in p} w_{u_j u_{j+1}}^i \leq b^i.$$

Last inequality indicates that p is a feasible path from s to t in G . Similarly, from the feasibility of p we have $\sum_{v \in p' \cap I_{x_i}} b \leq C_{x_i} \quad \forall x_i \in \{x_1, \dots, x_q\}$; therefore, p' is a feasible path. Other node $v' \in V'$ does not violate the feasibility of p' because $C_{v'} = \infty$. ■

Two special cases of the BCRWN are solvable in polynomial time. First, if $r = 0$, then there is not any interference between nodes. In this case, the problem is reduced to bandwidth constrained routing in wired networks. Second, if $r = \infty$, all nodes are in a same interference region, $AAB(u) = AAB(v) \quad \forall u, v \in V$. In this case, allocating bandwidth b along the path p consumes $b \cdot |p|$ at each node, where $|p|$ is the length of path p . In this case, the BCRWN problem can be solved by the minimum hop count (MHC) algorithm. In fact, if MHC fails to find a feasible path,

there is not any feasible path because each other path is more bandwidth consuming than the paths examined by MHC.

5. Feasible Path Algorithms

An optimal QoS routing algorithm has two major functionalities [20]:

- Finding feasible path that satisfy the constraints
- Achieving efficient utilization of network resources

Since the former functionality is very easy in the bandwidth constrained routing in *wired* network, many algorithms have been proposed for the latter functionality, including Minimum Hop Count (MHC) Widest Shortest Path (WSP) and Shortest Width Path (SWP) algorithms. But it is not the case for the MCP in wired network. As mentioned before, it is NP-Complete problem; therefore, the former functionality itself is a very difficult problem. While there are optimization versions of the MCP problem, many proposed algorithms for the MCP problem only took care about the former functionality and did not consider the latter. Fairly complete list of the algorithms can be found in [1]. Since the analysis in Section 4 showed the NP-Completeness of the BCRWN problem and its close relation to the MCP problem, we also *do not* consider the second functionality. Designing an optimal bandwidth constrained routing algorithm is out of scope of this paper; it is an interesting open research problem.

None of the proposed algorithms for the bandwidth constrained routing and the MCP problems in wired networks can be used directly and efficiently for the BCRWN problem.

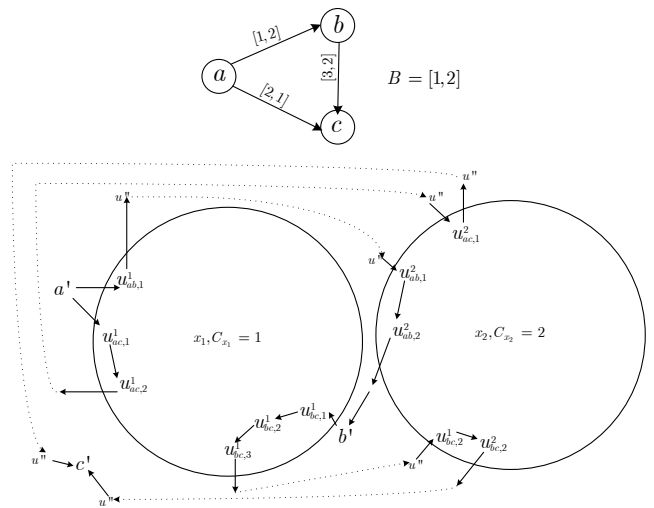


Figure 3. Transformation from MCP problem to BCRWN problem

All algorithms for the bandwidth constrained routing in wired networks including WSP and SWP assume that after network pruning, every path is feasible, which is not the case in wireless networks. Moreover, the concept of the width of path is not straightforward in wireless network; we will clarify its complexity in following subsection. Whereas there is close relationship between the MCP problem in wired networks and the BCRWN problem, algorithms for the MCP problem are not *efficient* algorithms for the BCRWN, because the complexity of the algorithms depends on the number of constraints which is $O(|E|)$ in the BCRWN.

Despite the fact that the algorithms can not be used directly for the BCRWN, the ideas and mechanisms in the algorithms are useful. For example, MHC algorithm selects the minimum bandwidth consuming path that leads to more feasibility probability of the path. Width Path (WP) selects a path that has maximum available bandwidth that it also leads to more feasibility probability.

5.1. Design Considerations

We focus on 1) definition of the *width* of a path, 2) the *feasibility* of a path, and 3) backtracking capability of routing algorithm.

We consider two definitions of the path's width: 1) Width of path p before allocating bandwidth along the path, $BBAW(p)$, and 2) Width of path p after allocating bandwidth b on the path, $ABAW_b(p)$. In wired networks, $ABAW_b(p) = BBAW(p) - b$. Computing the $ABAW_b(p)$ in wireless networks is not as easy as the wired networks, we use the following definition to obtain it

Definition 3. Available Area Bandwidth of node u after allocating bandwidth b along the path p is $AAB_{p,b}(u) = \min_{v \in I_u} ANB_{p,b}(v)$. where $ANB_{p,b}(v) = ANB(v) - |\{u \mid (u, w) \in p\} \cap I_v| \cdot b$. According to the definitions, we have $ABAW_b(p) = \min_{u \in p} AAB_{p,b}(u)$.

To satisfy the feasibility of a path, CAC must be performed at each node while the path is being constructed. It is done as follows. At each node v , bandwidth b is allocated along the partial path $p = \langle s = u_1 \dots u_k = v \rangle$. If $ABAW_b(p) \geq 0$, node v is accepted as a member of the path. It is easy to see that this CAC algorithm works correctly because if $v = t$, the $ABAW_b(p) \geq 0$ means that $AAB_{p,b}(u) \geq 0, \forall u \in p$; thus, $ANB_{p,b}(v) \geq 0, \forall v \in I_u, \forall u \in p$. Since allocating bandwidth along the path p does not affect the available bandwidth node $v \notin I_u, \forall u \in p, AAB(u) \geq 0 \forall u \in V$ which means that the path p is feasible.

If CAC does not accept a node as a member of the path, routing algorithm cannot go further from that node. It must *backtrack* and select another node. DFS is a standard graph search algorithm that has built-in backtracking capability. We augment DFS for the BCRWN problem. First, CAC checks the feasibility of the partial path $p = \langle s = u_1 \dots u_k = v \rangle$ before selecting node v . Second, we sort neighbors of node v according to $AAB_{p,b}(u)$, search continues from the neighbor that has more $AAB_{p,b}(u)$. Third, we visit node u multiple times, each time for a new path from source to u . The reason is as follows. Suppose that in the first visit of node u , we reach to the node by path p_1 and there is not any path p_3 from u to the destination that path $p = \langle p_1, p_3 \rangle$ is a feasible path. It is possible that there is another path p_2 from source to u that path $p = \langle p_2, p_3 \rangle$ is a feasible path; this path can be found if revisiting of nodes is allowed, contrary to the standard DFS algorithm. While revisiting of nodes increases the probability of finding feasible paths, it explores all paths in network and leads to an exponential running time if it is not controlled. We control it using the *path domination* concept. If we reach node u using path p_1 and there is not a feasible path from u to destination, we backtrack to the last node in p_1 . But before backtracking, we set the state of node u as unvisited and record $AAB_{p_1,b}(u)$ as its *dominated bandwidth*. If we again reach the node u along another path p_2 , we do not select the node u if $AAB_{p_2,b}(u) < AAB_{p_1,b}(u)$ because with high probability, there won't be any feasible path from node u to destination.

Please note that whereas it is possible to create some pathological topologies that the path domination heuristic cannot effectively control the revisiting process, such topologies are very rarely exist in real networks. Moreover, ADFS is not an optimal exact algorithm; therefore, it may reject a flow while there is a feasible path for it. In spite of these possible inefficiencies, our simulation results show that ADFS has near optimal performance and reasonable running time in practice.

5.2. Proposed Algorithm

We apply three algorithms for the BCRWN problem. First algorithm is Feasible Minimum Hop Count (FMHC). The algorithm is an integration of the standard minimum hop count algorithm and our CAC algorithm. Second algorithm, FBAWP, is an integration of the widest path and CAC algorithms. In the algorithm, width of path is defined as $BBAW$. Third algorithm, ADFS, is the augmented version of DFS algorithm. Pseudo code of the recursive implementation of ADFS is shown in Fig. 4.

ALGORITHM: ADFS	
Input:	$G = (V, E), I_v$ and $C_v \forall v \in V, s, t$, and b
Output:	$p = \langle s = v_1, v_2, \dots, v_k = t \rangle$
1.	Prune network: Remove node u if $AAB(u) < b$
2.	Initialize
3.	Visit (NIL, s, t, b)
ALGORITHM: Visit	
Input:	$(pred, u, t, b)$
Output:	$p = \langle s = v_1, v_2, \dots, v_k = t \rangle$
1.	if $u = t$ then
2.	Path is found, $t.pred \leftarrow pred$
3.	else
4.	Allocate bandwidth b along path $p = \langle s \dots u \rangle$
5.	if $CAC(u) = \text{ACCEPT}$ then
6.	$u.state \leftarrow \text{VISITED}$
7.	Sort $N(u) = \{v \mid (u, v) \in E\}$ according to $AAB_{p,b}(v)$
8.	while $N(u)$ is not empty do
9.	$v = \text{Extract}(N(u))$
10.	if ($v.state = \text{UNVISITED}$ and $AAB_{p,b}(v) > v.DominatedBandwidth$ and $AAB_{p,b}(v) > b$) then
11.	if ($p = \text{Visit}(u, v, t, b) \neq \text{NIL}$) then
12.	Path is found: $u.pred \leftarrow pred$
13.	Return p
14.	/* No path exists from u to t */
15.	$u.DominatedBandwidth \leftarrow AAB_{p,b}(u)$
16.	$u.state \leftarrow \text{UNVISITED}$
17.	return NIL

Figure 4. ADFS Algorithm

5.3 Integer Linear Programming Model

To evaluate the performance of the proposed algorithms, we develop an ILP model of the problem. The model provides us the exact solution of an instance of the BCRWN problem. If solving the model indicates that we cannot accept the given request, we ensure that there is no feasible path for the request. The model is as follows.

$$\text{maximize } a \quad (1)$$

Subject to:

$$\sum_{(u,v)} x_{uv} - \sum_{(v,u)} x_{vu} = 0, \quad \forall u \in V \setminus \{s, t\} \quad (2)$$

$$\begin{aligned} \sum_{(s,v)} x_{sv} - \sum_{(v,s)} x_{vs} &= a \\ \sum_{(t,v)} x_{tv} - \sum_{(v,t)} x_{vt} &= -a \end{aligned} \quad (3)$$

$$\sum_{\{(u,v) \mid u \in I_w\}} b \cdot x_{uv} \leq C_w \quad \forall w \in V \quad (4)$$

$$a, x_{uv} \in \{0, 1\} \quad (5)$$

In the model, x_{uv} is the routing variable, $x_{uv} = 1$ if and only if $(u, v) \in p$. Restricting it to $\{0, 1\}$ prevents flow splitting and multipath routing. a is the CAC variable, $a = 1$ if and only if there is a feasible path. Objective function (1) aims to accept the request. Equation (2) is the flow conservation constraint. Equation (3) controls amount of flow that enters to (leaves from) network; flow enters to network if and only if it is accepted, $a = 1$. The path feasibility constraint is (4).

It should be note that the ILP model is the optimal model of our objective in this paper. As mentioned before, similar to many algorithms for the MCP problem [1], we only consider finding a feasible path and don't seek an optimal routing algorithm to optimize some parameters of the network. The model also only maximize the a , which means accepting the flow by finding a feasible path, and doesn't optimize anything else.

6. Simulation Results

In this section, we evaluate the ability of the algorithms to find the feasible path and their running time. To measure the running time, we use the *Visit ratio* metric, $Vr(A)$, which is

$$Vr(A) = \frac{\# \text{ of Visited Nodes by Algorithm } A}{\# \text{ of Nodes in Network}} \quad (6)$$

To measure the path finding ability, we use the *Fail percentage* metric, $Fp(A)$, which is defined as

$$Fp(A) = \frac{ILP's \text{ accept } \# - A's \text{ accept } \#}{ILP's \text{ accept } \#} \times 100. \quad (7)$$

Where, the *ILP's accept #* and *A's accept #* are the number of accepted requests by the ILP model and algorithm A respectively.

Metrics similar to $Fp(A)$ and $Vr(A)$ are commonly used to evaluate the algorithms for the MCP problem, e.g. *success rate* and *normalized running time* were used in [1], [13]. It should be note that the metrics are quite adequate, because the problem is an NP-Complete decision problem; therefore, an efficient exact algorithm for it should be able to solve more instances of the problem than other algorithms, which means less $Fp(A)$, and has reasonable running time, which means less $Vr(A)$.

To compute the metrics, we create a set of requests and apply the algorithms on them. We count 1) the number nodes in network after pruning, 2) the number of the nodes which are visited by each algorithm, 3) the number of requests that each algorithm can accept, and 4) the number of the requests that ILP model accepts. In these simulations, we do **not** allocate bandwidth for the accepted requests because we only evaluate algorithms according to their ability to find the feasible path. We simulate our algorithms in two topologies. First topology is a 10×10 grid. Distance between nodes in a row (column) is 200m. Second topology is a random topology. 100 nodes are uniformly placed in $1500m \times 1500m$ area. In the simulations, the transmission range is 200m and the interference range is varied from 200m to 1000m. To solve the ILP model, we use the open source integer programming package ZIB Optimization Suite [19].

$Vr(A)$ and $Fp(A)$ of the algorithms are shown in Fig. 5 and Fig. 6 for different values of r in the grid and random topologies respectively. In these figures, solid lines are the results of the random network and dashed lines are the results of the grid network. Each point in the Fig. 5 and 6 is average of 10 different experiments. In each experiment we initialize C_u by a random value chosen in $[0,100]$ and create 250 random requests. Source and destination of requests are random nodes in the network and the required bandwidth is randomly chosen in $[1,10]$. In Fig. 5 we also show the $Vr(MHC)$, as a reference point.

The Fig. 5 shows that revisiting of nodes in ADFS does not increase the running time of the algorithm significantly in practice. Please note that when $r \gg 1$, there is no feasible path for most requests. In this case, algorithms terminate immediately after checking few percentages of nodes that leads to reduction of $Vr(A)$ by increasing r .

Fig. 6 shows that ADFS outperforms other algorithms in both topologies. As depicted in the figure, $Fp(ADFS)$ is less than 7% in the worst case. We believe that it is acceptable in many real applications.

As mentioned before, two special cases $r = 0$ and $r = \infty$ are solvable in polynomial time. This statement is justified by the Fig. 6, if r is very small or large respects to network size, $Fp(A) \rightarrow 0$. The reason is as follows: In the proof of the theorem 1, we showed the relation between the MCP and the BCRWN problems; that is, each interference region in the BCRWN is equivalent to a constraint in the MCP. The complexity of the MCP problem depends on the values of the constraints; if the constrains are too loose or too tight, the MCP problem is easy to solve [21] Very small r in the BCRWN is equivalent to large number of the constraints in the MCP problem. Moreover in this case, $w_{ab}^i = 0$ for most (a,b) and i , because there are very

few nodes in each interference region. Therefore, the constraints are too loose; it leads to many feasible paths which one of them can be found easily. When r is large, there are very few interference regions in the BCRWN and they are mostly overlapped. In analogy to the MCP, there are very few constraints. In this case, the values of the constraint are very tight, because many nodes share same interference regions. As indicated in [21], this condition causes that there is no path in most cases; that also simplifies the problem.

In the proof of the theorem 1, we make two assumptions: 1) arbitrary large interference range and 2) very large C_u of some nodes. None of the assumption is completely valid in our realistic simulations; consequently, the problem is not very difficult in practice and heuristic algorithms can achieve acceptable performance.

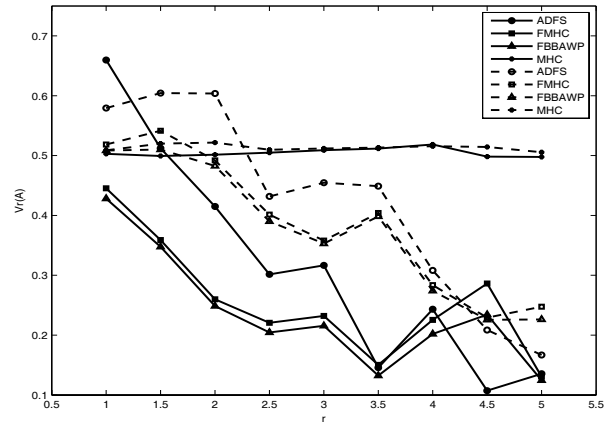


Figure 5. $Vr(A)$ of algorithms in the Random and Grid networks

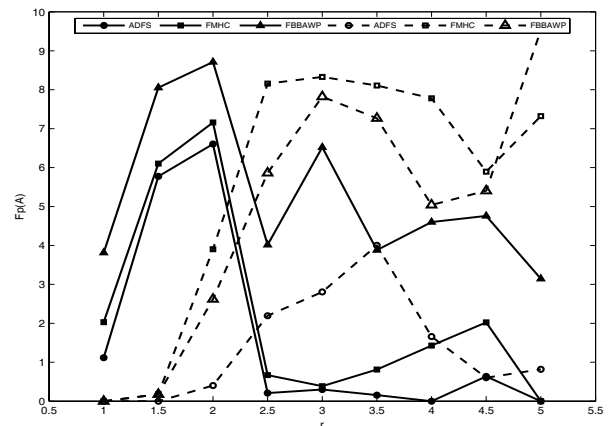


Figure 6. $Fp(A)$ of algorithms in the Random and Grid networks

7. Conclusion

We have studied the bandwidth constrained routing problem in wireless mesh network. Bandwidth constrained routing in wired networks is solvable in polynomial time, but we proved that the problem in wireless networks is NP-Complete by reducing the MCP problem to it. The difficulty arises from the dependency between bandwidth consumption and routing. We highlighted the difference between bandwidth constrained routing in wired and wireless networks in terms of 1) computing the available bandwidth of nodes, 2) computing the bandwidth consumption of flows, 3) the concepts of the residual feasible network, and 4) the definition of the width of path. Based on the differences, we propose a CAC algorithm and a heuristic routing algorithm. Analysis and simulation results indicated that the complexity of problem depends on $r = IR/TR$. If r is too small or too large, the proposed algorithm achieve almost the same performance as the exact ILP model.

- [1] Fernando Antonio Kuipers, "Quality of Service Routing in the Internet: Theory, Complexity and Algorithms," *PhD Thesis*, Delft University, 2004.
- [2] Y. Yang, "Distributed Resource Allocation in Contention-Based Wireless Networks," *PhD Thesis*, University of Illinois at Urbana-Champaign, 2006.
- [3] Q. Xue and A. Ganz, "Ad hoc QoS On-demand Routing (AQOR) in Mobile Ad hoc Networks," *Journal of Parallel and Distributed Computing*, Vol. 63, pp. 154-165, February 2003.
- [4] S. Chen and K. Nahrstedt, "Distributed Quality-of-Service Routing in Ad-Hoc Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 17, pp. 1488-1505, August 1999.
- [5] Vinod Kone, Sudipto Das, Ben Y. Zhao and Haitao Zheng, "QUORUM – QUality Of service RoUting in wireless Mesh networks," *ACM QShine*, 2007.
- [6] Y. Yang, J. Wang, R. Kravets, "Designing Routing Metrics for Mesh Networks," *IEEE WiMesh*, 2005.
- [7] Y. Ge, T. Kunz, and L. Lamont, "Quality of Service Routing in Ad-Hoc Networks Using OLSR," *Hawaii International Conference on System Science*, 2003.
- [8] I. Gerasimov and R. Simon, "A bandwidth-reservation mechanism for on demand ad hoc path finding," *IEEE/SCS Annual Simulation Symposium*, 2002.
- [9] H. Zhu, I. Chlamtac, "Admission control and bandwidth reservation in multi-hop ad hoc networks," *Elsevier Journal on Computer Networks*, Vol. 50, pp. 1653-1647, 2005.
- [10] P. Sinha, R. Sivakumar, and V. Bharghavan, "CEDAR: a Core-Extraction Distributed Ad hoc Routing Algorithm," *IEEE Infocom*, 1999.
- [11] S. Ge, S. K. Das, H. Wu, and C. Qiao, "Trigger-Based Distributed QoS Routing in Mobile Ad Hoc Networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, No. 3, 2002.
- [12] S. H. Shah, K. Nahrstedt, "Predictive location-based QoS routing in mobile ad hoc networks," *IEEE ICC*, 2002.
- [13] Zhenjiang Li · J.J. Garcia-Luna-Aceves, "Finding multi-constrained feasible paths by using depth-first search," *Wireless Networks*, Vol. 13, pp. 323–334, 2007.
- [14] C.-Y. Chiu, Y.-L. Kuo, E. H.-K. Wu, and G.-H. Chen, "Bandwidth-Constrained Routing Problem in Wireless Ad Hoc Networks," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 19, No. 1, pp. 4-14, January 2008.
- [15] L. Georgiadis, P. Jacquet, and B. Mans, "Bandwidth Reservation in Multihop Wireless Networks: Complexity, Mechanisms and Heuristics," *IEEE ICDCSW*, 2004.
- [16] R. Gupta, J. Musacchio, and J. Walrand, "Sufficient Rate Constraints for QoS Flows in Ad-Hoc Networks," *AD HOC NETWORKS*, Vol. 5, Issue 4, pp. 429-443, May 2007.
- [17] Jean Walrand, "Implementation of QoS Routing for Manets," *University of California at Berkeley Technical Report*, 2007.
- [18] Z. Wang and J. Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications," *IEEE Journal of Selected Areas in Communications*, Vol. 14, No. 7, pp. 1228–1234, 1996.
- [19] Zib Optimization Suite v1.1, <http://zibopt.zib.de/ZIBopt.shtml>.
- [20] T. Korkmaz and M. Krunz, "Multi-Constrained Optimal Path Selection," *IEEE Infocom*, 2001.
- [21] F. A. Kuipers and P. Van Mieghem, "Conditions that impact the Complexity of QoS Routing," *IEEE/ACM Transaction on Networking*, Vol. 13, No. 4, pp. 717–730, 2005.