Designing Routing Metrics for Mesh Networks

Yaling Yang
Department of Computer Science
Univ. of Illinois at Urbana-Champaign
Email: yyang8@uiuc.edu

Jun Wang
Nat'l Center for Supercomputing Applications
Univ. of Illinois at Urbana-Champaign
Email: wangj@ncsa.uiuc.edu

Robin Kravets
Department of Computer Science
Univ. of Illinois at Urbana-Champaign
Email: rhk@cs.uiuc.edu

***Abstract*— Designing routing metrics is critical for performance in wireless mesh networks. The unique characteristics of mesh networks, such as static nodes and the shared nature of the wireless medium, invalidate existing solutions from both wired and wireless networks and impose unique requirements on designing routing metrics for mesh networks. In this paper, we focus on identifying these requirements. We first analyze the possible types of routing protocols that can be used and show that proactive hop-by-hop routing protocols are the most appropriate for mesh networks. Then, we examine the requirements for designing routing metrics according to the characteristics of mesh networks and the type of routing protocols used. Finally, we study several existing routing metrics, including hop count, ETX, ETT, WCETT and MIC in terms of their ability to satisfy these requirements. Our simulation results of the performance of these metrics confirm our analysis of these metrics.**

I. Introduction

Mesh networks, motivated by wireless neighborhood networks [1], [2], are composed of static wireless nodes that have ample energy supply. Each of these wireless nodes can be equipped with multiple radios, called a *multi-radio/multi-channel node*, and each of the radios can be configured to a different channel to enhance network capacity. All wireless nodes cooperatively route each other's traffic to the Internet through one or more Internet Transit Access Points (TAPs), which are gateways to the Internet. Nodes may also communicate with each other directly through the mesh network without going through TAPs.

Supporting communication among mesh nodes and TAPs requires the use of routing protocols that must be combined with a routing metric to determine which route among all possible routes between a pair of nodes will be used. The design of effective routing metrics, however, depends on the specific characteristics of the target network. For example, the severe energy constraints of sensor networks demand the design of energy efficient routing, while the mobility of nodes in ad hoc networks demand the design of protocols that can efficiently maintain connectivity. The unique combination of static nodes with the shared nature of the wireless medium in mesh networks also imposes specific requirements for the design of routing metrics. The focus of this paper is to investigate the requirements for designing routing metrics in mesh networks to support high network performance, such as high throughput and low packet delay.

An effective analysis of the requirements for routing metric design in mesh networks must be based on an understanding of two factors: the routing protocols that are used in mesh networks and the characteristics of mesh networks. First, since different routing protocols may impose different costs in terms of message overhead and management complexity, it is important to understand which type of routing protocols are appropriate for mesh networks, so that the design of the routing metrics is compatible with effective routing protocols. Second, the characteristics of mesh networks, such as the static nature of nodes and the shared nature of the wireless medium, also impose challenges for the design of routing metrics. For example, due to the shared nature of the wireless medium, a wireless link in a mesh network does not have dedicated bandwidth since neighboring nodes' transmissions may also contend for the same bandwidth. Therefore, to reflect the quality of a link, an effective routing metric must be able to capture the interference between competing flows. To make things more complicated, since current wireless cards can be configured to different channels, wireless links that are configured to different channels may not interfere with each other even if they are physically located near each other. Effective routing metrics must be able to consider the channel assignments of links to understand the impact of interference on the performance of paths.

In an effort to understand how these challenges impact routing metric design in mesh networks, our work makes the following unique contributions. First, we analyze the performance of different types of routing protocols in mesh networks and show that proactive hop-by-hop routing is the most suitable type of routing protocol. Second, with a focus on proactive hop-by-hop routing protocols, we identify four fundamental requirements for designing routing metrics for mesh networks. These four requirements are: ensuring route stability, good performance for minimum weight paths, existing efficient algorithms to calculate minimum weight paths and ensuring loop-free routing. Third, we show how to use these four requirements to analyze the performance of five existing routing metrics for mesh networks: hop count, ETX [3], [4], ETT [5], WCETT [5] and MIC [6]. Our simulation results confirm our analytical results.

The remainder of the paper is organized as follows. Section II investigates the possible types of routing protocols and whether they are appropriate for mesh networks. Section III introduces the theories regarding the requirements of designing path weight functions for these protocols. Section IV reviews five existing routing metrics in mesh networks and analyzes whether they satisfy these requirements. Section V compares the performance of these routing metrics using simulations. Section VI concludes our work.

II. Routing Protocols for Mesh Networks

Different routing protocols may impose different requirements on the design of their routing metrics. Hence, it is necessary to first understand what routing protocols best fit mesh networks to understand the necessary properties of routing metrics to support effective routing in mesh networks. Depending on when routes are calculated, the possible routing protocols for mesh networks can be divided into two categories: on-demand routing and proactive routing. Based on how packets are routed along the paths, proactive routing can further be divided into two subcategories: source routing and hop-by-hop routing. All of these different routing protocols have different costs in terms of message overhead and management complexity. In this section, we examine the advantages and disadvantages of using these routing protocols in mesh networks and show that hop-by-hop routing is preferable.

A. On-demand Routing

Originally proposed for ad hoc networks, on-demand or reactive routing protocols (e.g., DSR [7], AODV [8], MCR [9], LBAR [10], and DLAR [11]) only create a route between a pair of source and destination nodes when the source node actually needs to send packets to the destination. Network-wide flooding is usually used to discover routes whey they are needed. For ad hoc networks, since there are frequent link breaks caused by the mobility of nodes, flooding-based route discovery provides high network connectivity and relatively low message overhead compared to proactive routing protocols. However, in mesh networks, links usually have much longer expected lifetimes due to the static nature of nodes. Since the frequency of link breaks is much lower than the frequency of flow arrivals in mesh networks, flooding-based route discovery is both redundant and very expensive in terms of control message overhead. Therefore, on-demand routing protocols are generally not scalable or appropriate for mesh networks.

B. Proactive Routing

In proactive routing protocols, each node maintains one or more tables containing routing information to every other node in the network. All nodes update these tables to maintain a consistent and up-to-date view of the network. When the network topology changes, the nodes propagate update messages throughout the network to maintain consistent and up-to-date routing information about the whole network. These routing protocols differ in the method by which packets are forwarded along routes.

1) Source Routing: Source routing, such as LQSR [5], imposes minimal burden on relaying nodes since the source node calculates the route for a flow and puts the entire path of the flow in the packet headers. Intermediate nodes only need to relay packets based on the paths in the packet headers. However, considering that the packet size in mesh networks is usually very small to cope with the high bit error rate of wireless channels, putting the entire path in the packet header imposes expensive message overhead.

2) Hop-by-hop Routing: In hop-by-hop routing, every node maintains a routing table that indicates the next hops for the routes to all other nodes in the network. For a packet to reach its destination, it only needs to carry the destination address. Intermediate nodes forward the packet along its path based only on the destination address. Due to its simple forwarding scheme and low message overhead, hop-by-hop routing is dominant in wired networks. Similar reasons also make hop-by-hop routing the most preferable for mesh networks. However, despite its benefits, hop-by-hop routing requires careful design of its routing metrics to ensure loop-free packet forwarding. In Section III, the detailed requirements for designing routing metrics for different routing protocols will be discussed. Due to the fact that hop-by-hop routing is most suitable for mesh networks, the requirements for designing routing metrics for hop-by-hop routing will be especially emphasized.

III. Requirements for Routing Metrics

To ensure good performance, routing metrics must satisfy four requirements. First, the routing metrics must not cause frequent route changes to ensure the stability of the network. Second, the routing metrics must capture the characteristics of mesh networks to ensure that minimum weight paths have good performance. Third, the routing metrics must ensure that minimum weight paths can be found by efficient algorithms with polynomial complexity. Finally, the routing metrics must ensure that forwarding loops are not formed by routing protocols. In this section, we introduce the theories regarding these four requirements.

A. Route Stability

Unstable path weights can be very harmful to the performance of any network. Frequent changes can create a high volume of route update messages. They can also disrupt normal network operations since routing protocols may not converge under frequent route updates.

The stability of path weights is determined by the type of path characteristics that are captured by the routing metrics, which can be either load-sensitive or topology-dependent. *Load-sensitive metrics* assign a weight to a route based on the traffic load on the route. Some examples of load-sensitive metrics are Degree of Nodal Activity [10], Interface Switching Cost [9] and Number of Congested Nodes [11]. Under load-sensitive metrics, the weight of a route may change frequently as flows arrive and depart. On the other hand, *topology-dependent metrics* assign a weight to a path based on the topological properties of the path, such as the hop count and link capacity of the path. Therefore, topological-dependent metrics are generally more stable, especially for static networks where the topology does not change frequently.

Load-sensitive and topology-dependent metrics are best used with different types of routing protocols, since routing protocols have different levels of tolerance of path weight instability. On-demand routing protocols can usually be designed to ignore frequent changes in path weights. In many on-demand routing protocols, such as DSR [7], the route for a flow is only searched for at the flow arrival and does not get updated as long as the route still exists. Therefore, metric changes usually do not trigger flows to change their routes and hence the stability of the network is not affected by frequently changing routing metrics. For this reason, load-sensitive routing metrics are suitable to be used with on-demand routing protocols in mesh networks.

However, in proactive routing protocols or on-demand routing protocols that may update the route of a flow during the lifetime of the flow (e.g., AODV [8]), small changes in routing metrics may cause route updates that affect the paths of many flows. Hence, these load-sensitive metrics have a high risk of creating network instability if traffic variations on the paths are large and irregular. Experiments conducted in wired networks have already demonstrated such effects [12], which have prevented the deployment of load-sensitive routing metrics in wired networks [13]. For mesh networks, we believe that this problem may be even worse. Since the Internet has a very large number of users, multiplexing smooths out traffic variations and reduces the number of route changes. Mesh networks, however, have a much smaller scale. Hence, link traffic variations may be large and irregular, making it very difficult to use load-sensitive routing metrics with proactive routing protocols while maintaining the stability of the network. On the other hand, since topology-dependent routing metrics are more stable, they can be used with both on-demand and proactive routing protocols.

Since load-sensitive routing metrics can only be used with on-demand routing, which has high message overhead, load-sensitive routing metrics are unsuitable for mesh networks. On the other hand, topology-dependent routing metrics can be used with both on-demand and proactive routing protocols, and so are preferable for mesh networks.

B. Good Performance for Minimum Weight Paths

For all of the routing protocols discussed in Section II, the goal is to route packets through minimum weight paths in terms of certain routing metrics. To ensure that the resources of mesh networks are utilized efficiently, the minimum weight paths selected by these routing protocols must have good performance in terms of high throughput and low packet delay. To achieve this, the routing metrics must be able to capture the characteristics of mesh networks that impact the performance of paths.

The first of these characteristics is path length. Since each hop introduces extra delay and potentially more packet loss, a longer path usually increases the end-to-end delay and reduces the throughput of a flow. Therefore, a routing metric should increase the weight of a path when the path's length increases.

The second characteristic is link capacity. Unlike a wired link, whose capacity is independent of the physical distance between the link's end points, the maximum transmission rate between two neighboring wireless nodes (i.e., the link capacity between the two nodes) is directly related to the physical distance between the two nodes. In general, as the distance between two nodes increases, the channel quality degrades. Since current wireless cards can adapt their transmission rates according to channel quality by changing their modulation schemes, the link capacity is reduced as the distance between the nodes increases. Therefore, although the effect of path length seems to favor paths with smaller hop count, the relationship between distance and link capacity counteracts this effect by favoring paths with larger hop count but higher link capacities. Hence, when designing routing metrics, a trade-off must be found between these two trends.

The third characteristic is packet loss ratios. Different wireless links may have different packet loss ratios. A node may need to retransmit a packet multiple times on a link with a high packet loss ratio, which affects both the throughput and the delay of any flow that goes through the link. Hence, a routing metric must capture the packet loss ratios to ensure good performance for the minimum weight path.

The fourth characteristic is interference. Different from wired links that have dedicated bandwidth, the bandwidth of a wireless link is shared between neighboring nodes. A flow through wireless links not only consumes the bandwidth of the nodes along its path, it also contends for bandwidth with the nodes that are in the neighboring area of its path. Such *inter-flow interference* can result in bandwidth starvation for some nodes since these nodes may always experience busy channels. To prevent such starvation, a routing metric must help routing protocols choose paths that can balance not only the traffic load along the path of a flow, but also reduce the inter-flow interference imposed in the entire neighboring area. For instance, in Figure 1, an effective routing metric should give path $A \rightarrow B \rightarrow C$ a lower weight than path $A \rightarrow D \rightarrow C$, since path $A \rightarrow B \rightarrow C$ has much less inter-flow interference than path $A \rightarrow D \rightarrow C$.

Besides inter-flow interference, nodes on the path of the same flow may also compete with each other for channel bandwidth. Such *intra-flow interference* increases the bandwidth consumption of the flow at each of the nodes along the path and causes the throughput of the flow to degrade sharply and the delay at each hop to increase dramatically as the hop count of the flow increases. Therefore, the potential of increased congestion levels due to such intra-flow interference must be considered when designing a routing metric for mesh networks. For example, as shown in Figure 2, an interference-aware metric should give path $A \rightarrow B \rightarrow C$ a higher weight than path $A \rightarrow D \rightarrow C$, since the reuse of channel 1 on $A \rightarrow B \rightarrow C$ creates much more intra-flow interference than that in path $A \rightarrow D \rightarrow C$. In summary, to find minimum weight paths with good performance, routing metrics must capture both intra-flow and inter-flow interference.

It is non-trivial to capture interference using routing metrics

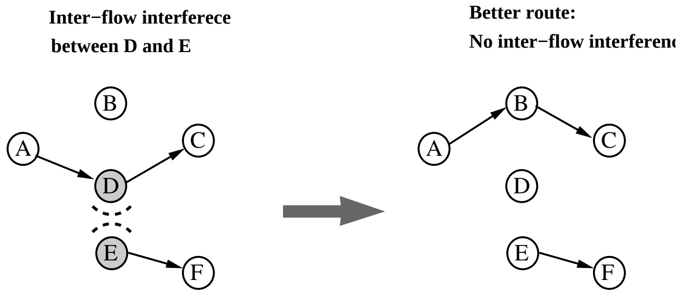


Fig. 1. An example for inter-flow interference.

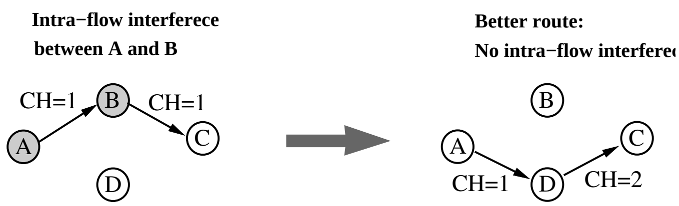


Fig. 2. An example for intra-flow interference. CH represents the channel assignment of a link.

since both the channel used by a link (the channel assignment of the link) and the capacity of the link are related to the amount of intra-flow and inter-flow interference that the link may impose on its neighborhood. In terms of the impact of link capacity, a packet that is transmitted over a 1Mbps link consumes more channel time at its neighbors than if it is transmitted over a 10Mbps link, hence resulting in more intra-flow and inter-flow interference. The channel assignment may impact the interference level since neighboring nodes may use different channels or radio technologies so that they do not interfere with each other. The IEEE 802.11b/g standards and the IEEE 802.11a standard provide 3 and 12 non-overlapped frequency channels respectively and the IEEE 802.11b/g and the IEEE 802.11a operate on different frequency bands (2.4Ghz and 5Ghz, respectively). Hence, both the diversity of channel assignments and the link capacity need to be captured when the routing metrics considers the interference of a path.

C. Efficient Algorithms to Calculate Minimum Weight Paths

All of the routing protocols essentially rely on certain forms of efficient algorithms, such as the Bellman-Ford or Dijkstra's algorithms, to compute the minimum weight paths. (Under the ideal case where there is no packet loss, the flooding-based route discovery in on-demand routing is essentially also a form of the Bellman-Ford algorithm.) Even if a routing metric ensures that its minimum weight paths have good performance, there is no guarantee that a routing protocol can have good performance if there does not exist an efficient algorithm to calculate the minimum weight paths based on the routing metric. The necessary and sufficient condition for the existence of such efficient algorithms is that the routing metrics must have a property called *isotonicity* [14], [15]. If a routing metric is not isotonic, only algorithms with

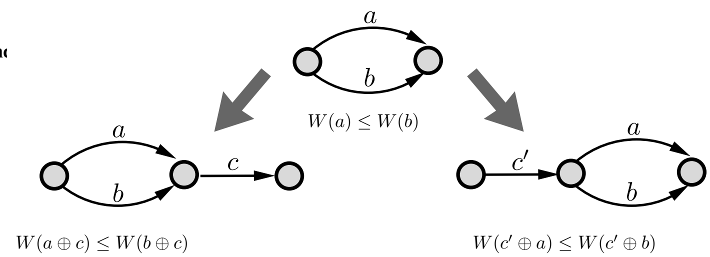


Fig. 3. Example of isotonicity

exponential complexity can calculate minimum weight paths based on this routing metric, which are not tractable even for networks with moderate size. Hence, it is important that routing metrics designed for mesh networks must be isotonic. In the reminder of the section, the definition of isotonicity and its relationship with efficient algorithms for minimum weight paths are explained. In Section III-D, it is further shown that isotonicity is also very important for ensuring loop-free routing in mesh networks.

Briefly speaking, the isotonic property essentially means that a metric should ensure that the order of the weights of two paths are preserved if they are appended or prefixed by a common third path. More precisely, assume that for any path a, its weight is defined by a routing metric, which is a function of a, denoted as $W(a)$. Denoting the concatenation of two paths a and b by $a \oplus b$, the definition of isotonicity is:

Definition 1: A routing metric $W(\cdot)$ is isotonic if $W(a) \leq W(b)$ implies both $W(a \oplus c) \leq W(b \oplus c)$ and $W(c' \oplus a) \leq W(c' \oplus b)$, for all a, b, c, c' (See Figure 3).

Given this definition of isotonicity, the following relationship exists between the isotonicity property and the optimality of the Bellman-Ford and Dijkstra's algorithms as shown by the work of Sobrinho [14], [15]:

Theorem 1: Isotonicity is a sufficient and necessary condition for both the Bellman-Ford and Dijkstra's algorithm to find minimum weight paths.

Theorem 1 implies that if a routing metric is not isotonic, routing protocols based on the Bellman-Ford or Dijkstra's algorithm may not find the minimum weight path between two nodes. The resulting sub-optimal paths may degrade network performance. Therefore, routing metrics must be either isotonic or be able to transfer to some isotonic forms (see Section IV-E for an example) to ensure good network performance.

D. Loop-free Routing

Not only does isotonicity determine whether minimum weight paths can be calculated efficiently, it may also be needed to ensure loop-free routing. As shown by Sobrinho's work [14], a metric must be isotonic to ensure that no routing loops can be formed when hop-by-hop routing is combined with Dijkstra's algorithm:

Theorem 2: If Dijkstra's algorithm is used in hop-by-hop routing, isotonicity is a sufficient and necessary condition for loop-free forwarding.

Theorem 2 reveals the importance of isotonicity for loop-free routing since non-isotonic routing metrics are not usable for link-state routing, which is a widely used hop-by-hop routing protocol based on Dkijkstra's algorithm. This implies that either on-demand routing, source routing or distance-vector routing must be used for non-isotonic routing metrics, since these routing protocols do not require isotonicity to ensure loop-free routing. In source routing, since the source nodes have complete control over the path of flows, using either the Bellman-Ford or Dijkstra's algorithm results in loop-free paths. For on-demand routing and hop-by-hop routing based on the Bellman-Ford algorithm, such as distance-vector routing, routing loops also cannot be created even if routing metrics are not isotonic.

However, as discussed in Section II, on-demand routing and source routing have too high a message overhead to be used in mesh networks. Therefore, the only remaining choice is distance-vector routing. Unfortunately, due to the lack of central management of mesh networks and the unreliable nature of wireless links, it is expected that link breaks or link quality changes in mesh networks will not be rare events. In such environments, distance-vector routing converges much slower than link-state routing and can potentially degrade network stability. Hence, it is a non-trivial drawback that non-isotonic routing metrics cannot be used in link-state routing. Therefore, isotonic routing metrics should be used in mesh networks.

IV. Routing Metrics for Mesh Networks

To satisfy the four requirements of routing metrics discussed in Section III, routing metrics must be isotonic, topology-dependent and must capture the characteristics of mesh networks. In this section, we discuss five routing metrics that have been proposed for mesh networks and whether they satisfy the three required properties. These five routing metrics are: hop count, ETX [3], [4], ETT [5], WCETT [5] and MIC [6]. All five routing metrics are topology-dependent and each routing metric was proposed as an improvement over the previous one.

A. Hop Count

Hop count is the most commonly used routing metric in existing routing protocols such as DSR [7], AODV [8], DSDV [16] and GSR [17]. It reflects the effects of path lengths on the performance of flows. Since a hop count metric is isotonic, efficient algorithms can find loop-free paths with minimum hop count. However, hop count does not consider the differences of the transmission rates and packet loss ratios between different wireless links, or the interference in the network. Hence, using a hop count metric may not result in good performance.

B. Expected Transmission Count (ETX)

ETX, proposed by De Couto et al. [3], [4], is defined as the expected number of MAC layer transmissions that is needed for successfully delivering a packet through a wireless link. The weight of a path is defined as the summation of the ETX's of all links along the path. Since both long paths and lossy paths have large weights under ETX, the ETX metric captures the effects of both packet loss ratios and path length. In addition, ETX is also an isotonic routing metric, which guarantees easy calculation of minimum weight paths and loop-free routing under all routing protocols. However, the drawbacks of ETX is that it does not consider interference or the fact that different links may have different transmission rates.

C. Expected Transmission Time (ETT)

The ETT routing metric, proposed by Draves et al. [5], improves ETX by considering the differences in link transmission rates. The ETT of a link l is defined as the expected MAC layer duration for a successful transmission of a packet at link l. The weight of a path p is simply the summation of the ETT's of the links on the path. The relationship between the ETT of a link l and ETX can be expressed as:

$$ETT_l = ETX_l \frac{s}{b_l}, \tag{1}$$

where b_l is the transmission rate of link l and s is the packet size. Essentially, by introducing b_l into the weight of a path, the ETT metric captures the impact of link capacity on the performance of the path. Similar to ETX, ETT is also isotonic. However, the remaining drawback of ETT is that it still does not fully capture the intra-flow and inter-flow interference in the network. For example, ETT may choose a path that only uses one channel, even though a path with more diversified channels has less intra-flow interference and hence higher throughput.

D. Weighted Cumulative ETT (WCETT)

To reduce intra-flow interference, WCETT [5] was proposed by Draves et al. [5] to reduce the number of nodes on the path of a flow that transmit on the same channel. For a path p, WCETT is defined as:

$$WCETT(p) = (1 - \beta) \sum_{\text{link } l \in p} ETT_l + \beta \max_{1 \leq j \leq k} X_j, \tag{2}$$

where β is a tunable parameter subject to $0 \leq \beta \leq 1$. X_j is the number of times channel j is used along path p and captures the intra-flow interference. The $\max_{1 \leq j \leq k} X_j$ component in Equation (2) counts the maximum number of times that the same channel appears along a path. It captures the intra-flow interference of a path since it essentially gives low weights to paths that have more diversified channel assignments on their links and hence lower intra-flow interference.

WCETT has two limitations. The first limitation is that it does not explicitly consider the effects of inter-flow interference, although it does capture intra-flow interference. Therefore, WCETT may route flows to dense areas where congestion is more likely and may even result in starvation of some nodes due to congestion.

Besides the lack of consideration of inter-flow interference, WCETT has another unique limitation: there is no efficient algorithm that can calculate the minimum weight path based on WCETT since it is not isotonic. Figure 4 depicts a simple topology that shows that WCETT is not isotonic. In this figure,

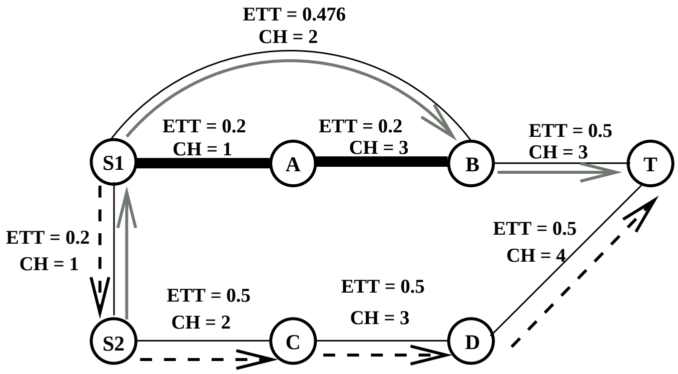


Fig. 4. Example topology, where Dijkstra's algorithm based on WCETT does not find minimum weight paths and hop-by hop routing based on WCETT creates forwarding loops.

two numbers are associated with each link, the ETT and the channel assignment (CH), respectively.

Assuming β in the definition of WCETT (see Equation (2)) is set to 0.5, the minimum weight path from S_1 to T should be $S_1 \to B \to T$. However, due to the non-isotonic property of WCETT, when node S_1 uses Dijkstra's algorithm to calculate its path to node T, node S_1 incorrectly chooses $S_1 \to S_2 \to C \to D \to T$ as the minimum weight path, indicated as the dotted arrows in Figure 4. This is because when running Dijkstra's algorithm at node S_1, the minimum weight path from node S_1 to node B is found to be $S_1 \to A \to B$, since $S_1 \to A \to B$ has the same $\max_{1 \le j \le k} X_j$ but a smaller aggregated ETT than the direct link $S_1 \to B$. $S_1 \to B$, hence, is eliminated from Dijkstra's algorithm's future consideration, although $S_1 \to A \to B \to T$ has a larger weight than $S_1 \to B \to T$. This incorrect early discard of $S_1 \to B$ causes Dijkstra's algorithm to fail to find the minimum weight path $S_1 \to B \to T$ from node S_1 to T.

If a link-state protocol based on WCETT is used, this incorrect minimum weight path between S_1 and T can cause forwarding loops. When node S_2 calculates its path to T, Dijkstra's algorithm correctly indicates that $S_2 \to S_1 \to B \to T$ is the minimum weight path, depicted as the shadowed arrows in Figure 4. Since S_1 has the incorrect minimum weight path, any packets destined to T are forwarded by S_1 to S_2. S_2 immediately forwards the packets back to S_1 again. Hence, a forwarding loop is formed between S_1 and S_2.

Similar to Dijkstra's algorithm, routing protocols based on the Bellman-Ford algorithm (e.g., distance-vector routing) may not find optimal paths based on WCETT either. Using the same example in Figure 4, since node B's minimum distance to node S_1 is the weight of $B \to A \to S_1$, node B only tells its neighbors about the weight of this path. Hence, node T does not have a chance to check the weight of $T \to B \to S_1$, which is the correct minimum weight path. Therefore, node T incorrectly sets its distance to S_1 as the weight of $T \to D \to C \to S_2 \to S_1$ and forwards any packets for S_1 to node D.

Because of WCETT's lack of isotonicity, there is no efficient algorithm with polynomial complexity to calculate minimum weight paths. In addition, the non-isotonicity of WCETT makes it unusable for link-state routing. To ensure loop-free routing, WCETT can only be used in on-demand routing, source routing such as LQSR [5] or distance-vector routing. This limitation is non-trivial since on-demand routing, source routing and distance-vector routing all have significant drawbacks compared to link-state routing (See Section III-D).

E. Metric of Interference and Channel-switching (MIC)

The MIC metric, proposed in our previous work [6], improves WCETT by solving its problems of non-isotonicity and the inability to capture inter-flow interference. The MIC metric of a path p is defined as follows:

$$MIC(p) = \frac{1}{N \times \min(ETT)} \sum_{\text{link } l \in p} IRU_l + \sum_{\text{node } i \in p} CSC_i, \tag{3}$$

where N is the total number of nodes in the network and $\min(ETT)$ is the smallest ETT in the network, which can be estimated based on the lowest transmission rate of the wireless cards. The two components of MIC, IRU (*Interference-aware Resource Usage*) and CSC (*Channel Switching Cost*), are defined as:

$$IRU_l = ETT_l \times N_l, \tag{4}$$

$$CSC_i = \begin{cases} w_1 & \text{if } CH(prev(i)) \neq CH(i) \\ w_2 & \text{if } CH(prev(i)) = CH(i), \end{cases} \tag{5}$$

$$0 \le w_1 < w_2, \tag{6}$$

where N_l is the set of neighbors that the transmission on link l interferes with, $CH(i)$ represents the channel assigned for node i's transmission and $prev(i)$ represents the previous hop of node i along the path p.

Essentially, the physical meaning of the IRU_l component is the aggregated channel time of neighboring nodes that transmissions on link l consumes. It captures the inter-flow interference since it favors a path that consumes less channel times at its neighboring nodes. The CSC part of MIC represents the intra-flow interference since it gives paths with consecutive links using the same channel higher weights than paths that alternate their channel assignments, essentially favoring paths with more diversified channel assignments.

It is worth noting that MIC is not isotonic if it is used directly as shown by the example in Figure 5. In the example, assuming that link a has a slightly smaller IRU than link b, the weights of paths a and b satisfy: $MIC(a) < MIC(b)$. However, adding link c to path a introduces a higher cost than adding link c to path b due to the reuse of channel 1 on path $a \oplus c$ ($\oplus$ means concatenation of two paths). Hence, $MIC(a \oplus c) > MIC(b \oplus c)$. Therefore, based on the definition of isotonicity, MIC is not an isotonic path weight function if used directly in real networks.

However, although MIC itself is not an isotonic metric, in our technical report, we have shown that it is possible to introduce virtual nodes [6], which are images of real nodes, into the network and decompose MIC into isotonic link weight assignments on virtual links between these virtual nodes. The

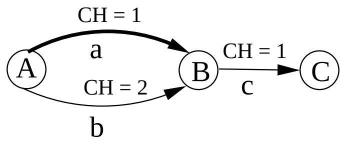


Fig. 5. Non-isotonicity of MIC

decomposition of MIC is based on the fact that the non-isotonic behavior of MIC is caused by the different increments of path weights due to the addition of a link on a path. Whether a cost increment will be different by adding a link is only related to the channel assignment of the previous link on the path. Since the possible assignments of channels for the precedent link are limited, by introducing several virtual nodes to represent these possible channel assignments, MIC can be translated into isotonic weight assignments to the links between these virtual nodes. Due to this isotonic form of MIC on the virtual network, efficient algorithms then can be used to find loop-free minimum weight paths based on MIC.

For WCETT, however, there is no known scheme that can find a isotonic form for WCETT. This is because WCETT's non-isotonicity is caused by the dependence of its $\max X_j$ component (Equation (2)) on the channel assignments of multiple links. Essentially, the weight increment of adding a link c to a path p depends on how many times each channel has appeared in path p. As the length of p increases, the combination of channel assignments can become infinite. Hence, it is impossible to introduce virtual nodes to represent all channel assignment states. Therefore, WCETT cannot be transformed into isotonic form and no efficient algorithms can be used to find loop-free minimum weight paths for WCETT.

V. Evaluation

Due to constraints of hardware cost, in some mesh networks, each node may have only one radio interface and these radio interfaces must be configured to the same channel to ensure the connectivity of the network. In other mesh networks, however, it may be affordable to equip each node with multiple radio interfaces so that these radio interfaces can be configured to different channels to reduce both intra-flow and inter-flow interference. Hence, to understand the performance of different routing metrics under different network configurations, our evaluation includes two parts. In the first part, we consider mesh networks where each node has only one radio interface and all the radio interfaces are configured to the same channel. Since it is impossible to use channel switching to reduce intra-flow interference in such networks, the routing metrics' ability to capture other network characteristics, such as inter-flow interference, is the major factor that affects the metrics' performances in this part of the evaluation. The metrics that are compared in this part include hop count, ETT and MIC. WCETT is not included since WCETT is aimed to be used only in multi-channel environments. In the second part of our evaluation, we consider networks where each node have multiple radio interfaces that are configured to different channels. In such networks, since channel switching may reduce intra-flow interference, the ability to capture intra-flow interference affects the performance of the routing metrics. The metrics that are compared in this part include hop count, ETT, MIC and WCETT.

The performance of the routing metrics are compared in terms of the total network throughput, the average end-to-end packet delay and the maximum channel utilization. While the physical meaning of total network throughput and packet delay are obvious, the channel utilization at a node, which is the fraction of channel busy time at a the node, indicates the channel congestion level at the node. Hence, the maximum channel utilization among all nodes implies routing metrics' ability to avoid creating hot spots.

All of our simulations are performed in the NS2 simulator [18]. The topologies of simulations are randomly generated. Since we expect that most of the traffic in a real mesh network will be traffic to/from the wired network, in our simulations, all flows are destined to the Internet through one to four TAPs. The sources of the flows are randomly located in the mesh network. All flows are CBR flows with 512 Byte packets. Since WCETT is not isotonic, distance-vector routing is used to ensure that there are no routing loops. The evaluation of all protocols is based on the performance of the system after the routing tables have stabilized. The transmission range is 250m while the carrier-sensing range is 550m. The transmission rates between neighboring nodes are related to the distance between the nodes as shown in Table I. Both the w_2 in MIC and β in WCETT are set to 0.5.

A. Single Channel Environments

In the first set of simulations, we randomly generated six $1500m \times 1500m$ networks with 160 nodes, 15 flows and 4 TAPs. Figures 6(a), 6(b) and 6(c) show the maximum channel utilization among all nodes, the total network throughput and the average end-to-end packet delay, respectively. The MIC metric has the best performance in terms of the lowest maximum channel utilization, the highest total network throughput and the smallest average end-to-end packet delay since it satisfies all of the requirements of routing metric design. ETT's performance is worse than MIC since it does not capture the interference between nodes. Hop count has the worst performance since it captures the fewest characteristics of mesh networks.

B. Multi-Channel Environment

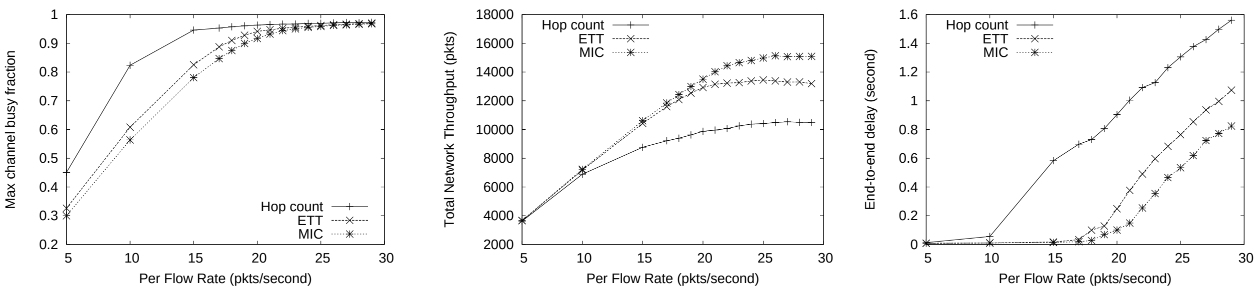
In the second set of simulations, every node has two radios and each radio can be configured to one of three channels. We randomly generate ten $1000m \times 1000m$ networks, each with 100 nodes, 20 flows and 1 TAP. Compared to the first set of simulations, the capacity of the network is increased by having multiple channels. Therefore, we use a higher node density, a larger number of flows and a smaller number of TAPs to increase network load.

Figures 7(a), 7(b) and 7(c) show the maximum channel utilization among nodes, the total network throughput and

TABLE I

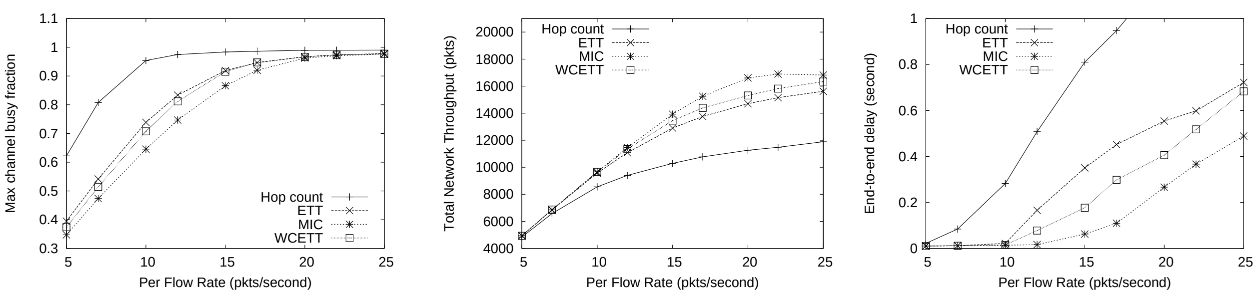
DISTANCE/RATE RELATIONSHIPS

Distance(m)	25	50	75	100	125	150	175	200	225	250	>250
Rate(Mbps)	54	48	36	24	18	12	9	6	2	1	0



(a) Maximum channel utilization (b) Total network throughput (c) Average end-to-end packet delay

Fig. 6. $1500m \times 1500m$ 160 node single channel/single radio networks



(a) Maximum channel utilization (b) Total network throughput (c) Average end-to-end packet delay

Fig. 7. $1000m \times 1000m$ 100 node 2-radio/ 3-channel networks

the average end-to-end packet delay, which again confirm that MIC has the best performance in terms low maximum channel utilization, high throughput and low delay. Since WCETT captures intra-flow interference, it has better performance than ETT. Once again, hop count has the worst performance.

VI. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we have presented a comprehensive study on designing routing metrics in mesh networks. We systematically investigate the possible choices of routing protocols for mesh networks and show that hop-by-hop routing is generally preferable. We describe four requirements that routing metrics for mesh networks must satisfy to ensure good network performance. Then, we analyze five existing routing metrics to understand whether they satisfy these four requirements. Finally, we compare the performance of these routing metrics and show that it is necessary to satisfy all of the requirements to achieve the best performance.

Our future work is to investigate the performance of all of the existing metrics in real mesh networks based on actual hardware measurements. We also want to further investigate the design of the MIC metric by studying the trade-offs of setting the w's in Equation (5) and α in Equation (3). We will investigate the appropriate w's for real mesh networks and how α affects the delay and throughput of flows and the overall load on the network. Finally, we will investigate the performance of mesh networks that have both mobile and static nodes.

ACKNOWLEDGMENT

This work was funded in part by a Vodafone-US Foundation Graduate Fellowship and National Science Foundation Grant 0081308. The author would also like to thank Al Harris, Cigdem Sengul, Robin Snader, William Yurcik and Lili Qiu for their help in making this work possible.

REFERENCES

[1] Roger Karrer, Ashutosh Sabharwal, and Edward Knightly, "Enabling Large-scale Wireless Broadband: The Case for TAPs," in *Proceedings of HotNets*, Cambridge, MA, 2003.

[2] Violeta Gambiroza, Bahareh Sadeghi, and Edward Knightly, "End-to-End Performance and Fairness in Multihop Wireless Backhaul Networks," in *ACM Mobicom*, 2004.

[3] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris, "A High-Throughput Path Metric for Multi-Hop Wireless Routing," in *ACM Mobicom*, 2003.

[4] R. Draves, J. Padhye, and B. Zill, "Comparison of routing metrics for static multi-hop wireless networks," in *ACM SIGCOMM*, 2004.

[5] Richard Draves, Jitendra Padhye, and Brian Zill, "Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," in *ACM Mobicom*, 2004.

[6] Yaling Yang, Jun Wang, and Robin Kravets, "Interference-aware Load Balancing for Multihop Wireless Networks," Tech. Rep. UIUCDCS-R-2005-2526, Department of Computer Science, University of Illinois at Urbana-Champaign, 2005.

[7] David B Johnson and David A Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," in *Mobile Computing*. 1996, vol. 353, Kluwer Academic Publishers.

[8] Charles Perkins, "Ad-hoc on-demand distance vector routing," in *MILCOM panel on Ad Hoc Networks*, 1997.

[9] Pradeep Kyasanur and Nitin Vaidya, "Multi-Channel Wireless Networks: Capacity and Protocols," Tech. Rep., University of Illinois at Urbana-Champaign, 2005.

[10] Hossam Hassanein and Audrey Zhou, "Routing with Load Balancing in Wireless Ad hoc Networks," in *ACM MSWiM*, 2001.

[11] Sung-Ju Lee and Mario Gerla, "Dynamic Load-Aware Routing in Ad hoc Networks," in *IEEE ICC*, 2001.

[12] Atul Khanna and John Zinky, "The Revised ARPANET Routing Metric," in *ACM SIGCOMM*, 1989.

[13] E. Anderson and T. Anderson, "On the Stability of Adaptive Routing in the Presence of Congestion Control," in *IEEE INFOCOM*, 2003.

[14] J. L. Sobrinho, "Algebra and algorithms for QoS path computation and hop-by-hop routing in the Inernet," in *IEEE INFOCOM*, 2001.

[15] J. L. Sobrinho, "Network Routing with Path Vector Protocols: Theory and Applications," in *ACM SIGCOMM*, 2003, pp. 49–60.

[16] Charles Perkins and Pravin Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *ACM SIGCOMM*, 1994.

[17] Tsu-Wei Chen and Mario Gerla, "Global State Routing: A New Routing Schemes for Ad-hoc Wireless Networks," in *IEEE ICC*, 1998.

[18] Kevin Fall and Kannan Varadhan, "NS notes and documentation," in *The VINT Project, UC Berkely, LBL, USC/ISI, and Xerox PARC*, 1997.