Multicast Routing Algorithms

Uyen Trang Nguyen

Hoang Lan Nguyen

Technical Report CSE-2008-08

October 9, 2008

Department of Computer Science and Engineering

4700 Keele Street Toronto, Ontario M3J 1P3 Canada

Multicast Routing Algorithms: from the Internet to Wireless Networks

Uyen Trang Nguyen, Hoang Lan Nguyen
Department of Computer Science & Engineering
York University
4700 Keele Street, Toronto, Ontario M3J 1P3 Canada
Email: {utn, lan}@cse.yorku.ca

Abstract

Multicast is an efficient form of communication that delivers information from a source to multiple destinations simultaneously. Given a wide range of multicast applications, numerous multicast routing algorithms have been defined to satisfy different requirements of various applications. In addition, given a multicast routing algorithm and metric, the same application may perform differently in different types of networks. Designing an appropriate routing algorithm for a multicast application is crucial to ensure its optimal performance in the target network. It is important to have a broad view of multicast routing algorithms in relation to performance requirements of applications and network characteristics. In this article, we present a survey of multicast routing algorithms in the wireline Internet and wireless multi-hop networks. We also review routing metrics and multicast routing implementation issues, and discuss issues and challenges for future research.

Keywords: multicast, routing, shortest path trees, minimum cost trees, minimum Steiner trees, minimum data overhead trees, wireless multihop networks, mobile ad hoc networks, wireless sensor networks, wireless mesh networks, wireless broadcast advantage.

1 Introduction

Multicast [1] is a form of communication that delivers information from a source to a set of destinations simultaneously in an efficient manner. Important applications of multicast include distribution of financial data, billing records, software, and newspapers; audio/video conferencing; distance education; IP TV; and distributed interactive games. Multicast routing requires the coordination and cooperation of the sources, destinations and routers (intermediate nodes) to relay and deliver data to the destinations in an efficient manner within a finite amount of time.

Different applications have different requirements in terms of quality of service and reliability, and thus different types of routing trees. For example, video-on-demand has stringent delay and delay jitter constraints with some degree of tolerance for packet loss. Distribution of monthly magazine subscriptions, on the other hand, requires 100% delivery reliability but can tolerate some latency.

A routing algorithm typically goes hand in hand with a routing metric. Routing metrics reflect the property of routes/trees and determine if one route/tree should perform better than another. For example, if the link metric is link delay, then the least cost source-to-destination path is one that has the minimum end-to-end latency. If the link metric is the lease cost, then the least cost path is one that incurs the lowest expense.

Beside the requirements of the application, the characteristics of the target network also determine the optimization goal(s) of a multicast routing algorithm. For instance, in wired networks, the major cause of packet loss is congestion; an application requiring high reliability would favor paths with low traffic loads. In wireless multi-hop networks, the main source of packet loss is high error rates of wireless channels. Therefore, the above application needs to take into account the loss rates of the links on a path, and the path length (since the longer a path, the higher the probability a packet is lost or damaged).

One of the open and on-going research problems is to extend high speed IP connectivity to the "last mile" using emerging wireless technologies. Wireless multi-hop networks such as mobile ad hoc networks (MANETs), wireless sensor networks (WSNs) and wireless mesh networks (WMNs) have been considered feasible and attractive solutions because of their low cost, fast deployment, self-organization, self-optimization and fault-tolerance properties. Figure 1 shows an application scenario of these types of networks in cooperation with the Internet. In this scenario, the WMN is connected to the Internet via gateways; one MANET is also connected to the Internet and the other is deployed as a small wireless network at the edge of the WMN; the sensor network collects data and sends to a central server for processing. We thus consider these networks in this paper.

Defining the appropriate routing algorithm and metric for a multicast application is crucial to ensure its desired performance in the target network. It is important to have a broad view of multicast routing algorithms in relation of application requirements and network characteristics. In this paper, we present a survey of multicast routing algorithms in different types of networks, namely the wireline Internet, MANETs, WSNs, and WMNs. We also review routing protocols that use the algorithms and their implementations. We discuss issues and challenges in designing multicast routing algorithms, and open research topics.

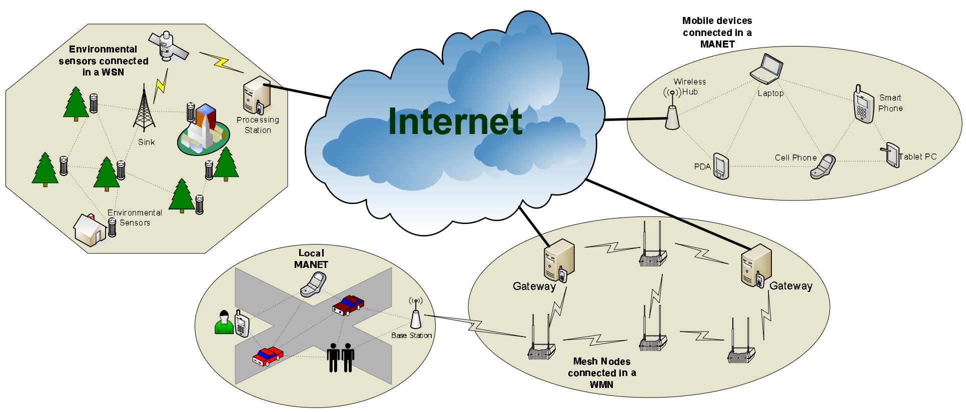

Figure 1: The Internet and "last mile" wireless multi-hop networks

1.1 Classification of Multicast Routing Algorithms

The most popular multicast routing approach is based on shortest (or least cost) source-to-destination paths. A shortest path (or least cost path) from the source to each destinations is computed, whose cost is the sum of the costs of the links along the route. The source-to-destinations paths are then merged to form a multicast tree, which is called a *shortest path tree* (SPT).

The cost of a shortest path is cumulative, either additive or multiplicative. There exist also path costs that are not cumulative. For example, the cost of a path in terms of its available bandwidth is determined by the bandwidth of its most bottlenecked link [2]. Among several paths from a source to a destination, it is desirable to select the path with the maximum available bandwidth. Such a path is called a best path (rather than a shortest or least cost path, since the path cost is not cumulative). The best paths from the source to the destinations form a *best path tree*. SPTs and best path trees are presented in Section 4.

In SPTs and best path trees, the optimization is based on path costs. An alternative optimization goal is to minimize the cost of the entire routing tree (e.g., the sum of the costs of all the links in the tree, the total energy consumption of the nodes in the tree). These are called *minimum cost trees* (MCTs). The tree cost can be either link-based or node-based. When the cost is link-based, we build a tree with the minimum sum of costs of the links (edges). Similarly, when the cost is node-based, we construct a multicast tree such that the sum of costs of the nodes (vertices) is minimum. There exist several types of MCTs, which are discussed in Section 5.

Many real-time applications require a predetermined bound on the end-to-end delay. This requirement has spawned a group of multicast routing trees called *constrained trees*. A SPT or MCT can be constrained by a requirement that the end-to-end delay of any source-to-destination path is less than a predetermined value. Other types of constraints include delay variation, link bandwidth and node degree. Constrained trees are presented in Section 6.

There exist also trees that is a combination of the above types of trees, e.g., optimizing both the tree cost and path lengths. These trees belong to the class of *hybrid trees*, and will be discussed in Section 7.

A summary of different types of multicast routing trees based on various algorithms is given in Figure 2 and Table 1.

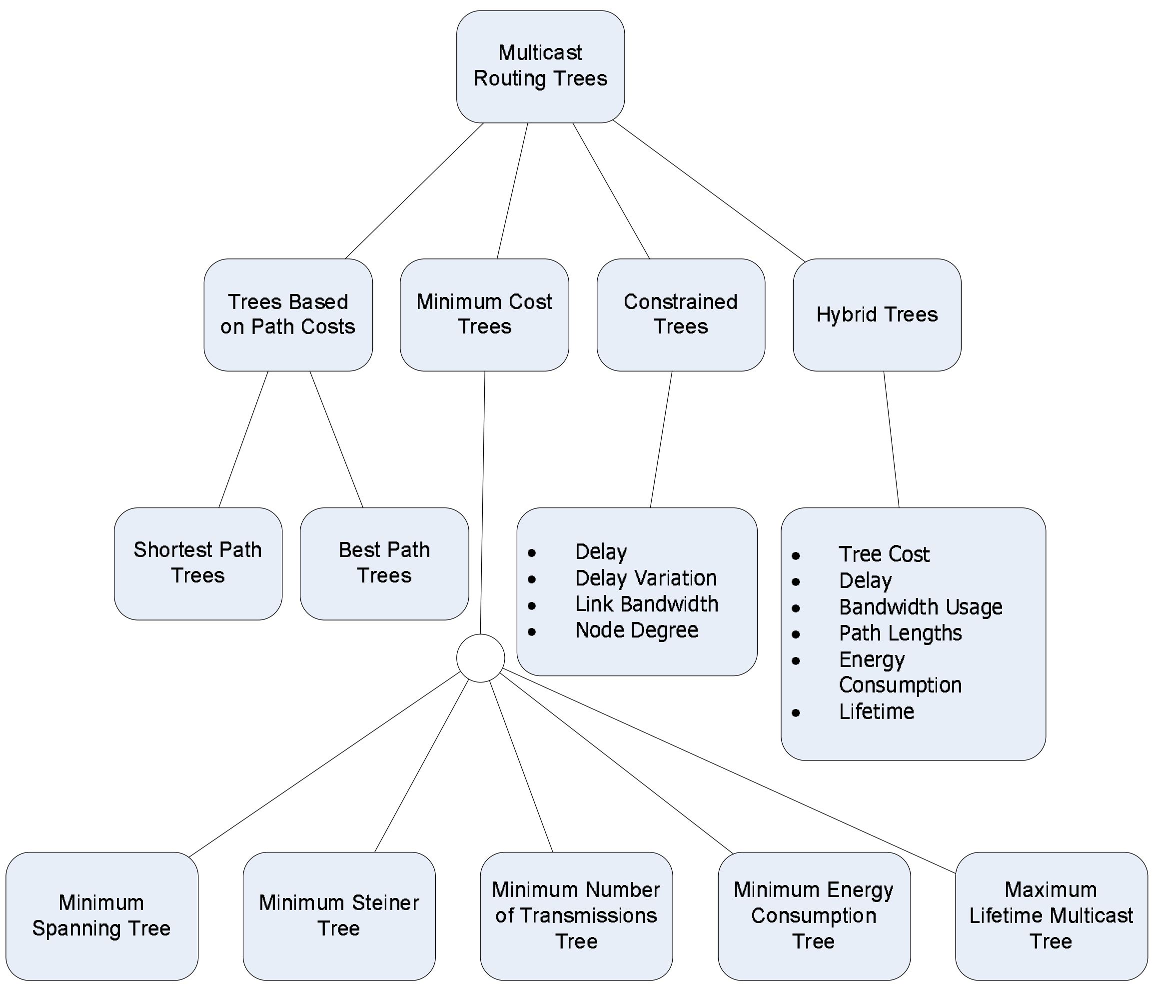

Figure 2: A taxonomy of multicast routing algorithms

Algorithms		Target Networks	References
Trees based on path costs	Shortest path tree (SPT)	All	Dijkstra's [3], Bellman-Ford [4, 5, 6], DVMRP [7] MOSPF [8], ODMRP [9], MAODV [10]
	Best path tree	All	Shacham [11]
Minimum cost trees	Min spanning tree (MSpT)	All	Prim [12], Kruskal [13], Gallager et al. [14]
	Min Steiner tree (MStT)	All	Kou et al. [15], Takahashi et al. [16] Zelikovsky [17], Winter and Smith [18, 19]
	Min number of transmissions	Wireless	Ruiz and Gomez-Skarmeta [20]
	Min energy consumption	Wireless	EWMA [21], S-REMiT [22], G-REMiT [23]
	Maximum lifetime	Wireless	L-REMiT [24]
Constrained trees	Delay-constrained SPT	All	Deering et al. [25], Sun et al. [26]
	Delay-constrained MSpT	All	Salama et al. [27], Chow [28], Kompella et al. [29], Jia [30], Chen et al. [31]
	Delay-constrained MStT	All	Kompella et al. [32, 29], Sriram et al. [33], Feng et al. [34], Kumar et al. [35] Zhu et al. [36], Jia [30], Shaikh and Shin [37]
	Delay variation	All	Rouskas et al. [38, 39]
	Link capacity	All	Jiang [40]
	Node degree	High-speed wired networks (e.g., ATM)	Bauer and Varma [41, 42]
Hybrid trees	Tree cost & delay	All	Bharath-Kuma et al. [43], Chung et al. [44]
	Bandwidth & path length	All	Fujinoki and Christensen [45] Zeng et al. [46], Chou et al. [47]
	Energy & lifetime	Wireless	Wieselthier et al. [48, 49], Banerjee et al. [50]

Table 1: Summary of multicast routing algorithms

1.2 Organization of the Article

The remainder of the paper is structured as follows. Section 2 provides background information on multicast applications, characteristics of different types of networks and multicast routing implementations. In Section 3, we briefly review routing metrics, many of which have been applied to multicast. (Detailed descriptions of the metrics can be found in Appendix A at the end of the article.) We then discuss multicast routing algorithms, namely shortest/best path tree, minimum cost tree, constrained tree and hybrid tree in Section 4, 5, 6 and 7, respectively. In Section 8, we discuss importance issues of multicast routing algorithms and metrics such as performance comparison, implementation issues, and design challenges and open issues. Section 9 concludes the article.

2 Background

This section provides background information to facilitate the discussions in the remainder of the paper. We present a brief overview of multicast applications, characteristics of different types of networks and multicast routing implementations.

2.1 Multicast Applications

Multicast applications can be classified into one-to-many and many-to-many. A one-to-many application consists of a single source and multiple simultaneous receivers. This is one-way communications, with data flowing from the source to the receivers. Commonly seen one-to-many applications include

- Recorded voice/video delivery: lectures, tutorials, presentations, past meetings, or any other type of multimedia coverage of scheduled events. One of the most important emerging applications is IP-TV, which provides television viewing using the Internet Protocol infrastructure [51].
- Automatic ("always-on") media distribution: news headlines, news feeds, stock quotes, weather updates, sports scores, and current traffic conditions.
- File updates and caching: software updates sent to customers; web site contents delivered to a number of web servers for updates or for backup of the contents.
- Delivery of announcements, notifications and reminders in an organization.

A many-to-many application involves two or more sources. The sources can send only, or also act as receivers. Each host may receive data from multiple senders and, in the same session, sends data to the other members. As a result, many-to-many applications require two-way multicast communications, and often involve complex coordination and management issues. Following are typical many-to-many applications:

- Tele-conferencing: online meeting and team work; distance education involving instructors' and students' interactions, Internet chat rooms.
- Distributed parallel processing in which workstations exchange data for parallel computations.
- Online interactive games and distributed multi-object simulations.

- Online auctions. The "auctioneer" starts the bidding by describing the product or service for sale, and regularly multicasts the current bid. Bidders send their bids privately or publicly, i.e., to a unicast or multicast address. The latter case is many-to-many multicast.

There exist also many-to-one multicast applications such as data collection or polling where several participants send data/responses to a central server. However, they are less common and less researched than the other two application categories. In this paper, we consider only one-to-many and many-to-many multicast.

2.2 Characteristics of Different Types of Networks

Internet communications can be characterized by the very large scale of the Internet (e.g., millions of networks) and extreme heterogeneity. Therefore, multicast routing on the Internet has focused on distributed operations for scalability, minimizing the amount of routing information stored in routers, minimizing end-to-end delay, and quality-of-service routing in the presence of heterogeneous networks and hosts.

While the wired Internet has become indispensable in communications, there exist applications where running cables or wires is infeasible or very expensive such as military communications in battlefields; environment monitoring in forests, mountains, water surface, deserts; providing Internet access in remote areas. Wireless ad hoc networks have made these applications possible given their inexpensive, fast deployment and low cost of operating and maintenance. Wireless ad hoc networks can be broadly classified into three types based on their application: mobile ad hoc networks (MANETs), wireless sensor networks (WSNs), and wireless mesh networks (WMNs).

All these networks share the following common characteristics:

- Nodes communicate with each other by forming a multihop radio network and maintaining connectivity in a decentralized manner.
- There is no infrastructure. The control of the network is distributed among the nodes.
- Wireless channels typically have less bandwidth than wired links and higher error rates due to the effects of radio communication, such as noise, fading, and interference.
- The partial mesh topology enables multi-path routing capability to protect against node and link failures.
- To effectively support ad hoc deployment and operations, self-configuration, self-organization and self-healing are among important properties of these networks.
- Wireless ad hoc networks are quick and easy to deploy at low costs, compared with infrastructured networks.

On the other hand, MANETs, WSNs and WMNs are different in several aspects, mainly due to their application.

MANETs target survivable, efficient, dynamic communication for emergency/rescue operations, disaster relief efforts, and military networks. These networks have short lifetime due to their temporary, ad hoc applications. A MANET is an autonomous collection of mobile users that communicate over relatively bandwidth constrained wireless links. Since the nodes are mobile, the network topology may change rapidly and unpredictably over time. The network is decentralized, where all network activity including discovering the topology and delivering messages must be executed by the nodes themselves, i.e., routing functionality will be incorporated into mobile nodes.

Regardless of the application, MANETs need efficient distributed algorithms to determine network organization, link scheduling, and routing. Another challenge is mobile devices having limited power supply or battery capacity.

A WSN consists of a number of sensors spread across a geographical area. Each sensor has wireless communication capability and some level of intelligence for signal processing and networking of the data. Typical applications include detecting and characterizing chemical, biological and radiological levels; monitoring environmental changes in plains, forests and oceans; monitoring vehicle traffic in city or on highways; and security surveillance. Similarly to MANETs, WSNs require low energy use and network self-organization. Depending on the application, nodes in WSNs can be stationary or mobile (e.g., sensors on the ocean surface, robotic sensors in military operations). Unlike the other types of wireless ad hoc networks, WSNs are envisioned to be deployed in very large scales, in the range of tens of thousands of nodes; scalability is thus a critical issue.

Wireless mesh networking [52, 53] is motivated by the need to extend high-speed IP connectivity to the "last mile" with low investment overhead and fast deployment. Other applications of WMNs include low cost Internet access provisioning in remote areas, shopping malls, airports; municipal and metropolitan networking; ad hoc networking for emergency and disaster recovery; health and medical systems in hospitals; and information services in public transportation systems. Nodes in a WMN are wireless routers interconnected to form a wireless backbone. Unlike nodes in MANETs or WSNs, these wireless routers are typically stationary, and not constrained in terms of power supply, storage or computing capability. Users in WMNs can be stationary hosts or mobile devices connected to wireless routers. A WMN is connected to the Internet via one or more gateways (Figure 1).

Until recently, research on wireless ad hoc networks considers mostly networks with a single channel. The theoretical upper limit of per node throughput capacity in such networks is limited by $O(1/\sqrt{n})$, where n is the number of nodes in the network [54]. The theoretical *achievable* throughput is even lower, estimated as $\theta(1/\sqrt{n \log n})$ in a random ad hoc network with ideal global scheduling and routing [54]. It has also been shown through experiments that on a string topology using carrier sense multiple access with collision avoidance (CSMA/CA) medium access control (MAC) such as IEEE 802.11, the throughput degrades approximately to $1/n$ of the raw channel bandwidth [55]. The above results indicate that the throughput capacity becomes unacceptable low as the network size increases. Several factors contribute to such a rapid degradation of throughput such as the behavior of the MAC protocols, greediness of the initial nodes and subsequent flow starvation of the latter hops. However, the single most important factor is the exposed terminal problem, worsened by the use of a single-radio single-channel network. One approach to enhance the aggregate network throughput is to use systems with multiple channels and multiple radios per node [56, 57, 58, 59, 60].

Figure 3 illustrates the multi-channel multi-radio (MCMR) model. The network has n channels, which may either overlap, such that a channel partially shares its spectrum with the adjacent channels, or may be completely non-overlapping. For example, IEEE 802.11b/g networks have 11 channels, but at most three are non-overlapping (e.g., channels 1, 6 and 11). A host in a MCMR network has m radios (interfaces), and typically $1 < m < n$ (e.g., $m = 3$, $n = 11$). A MCMR node can transmit on one channel and receive on another at the same time using two different radios. As a result, a MCMR wireless network at least doubles the throughput, since each node is now in full-duplex mode, being able to transmit and receive simultaneously. In return, MCMR networks require efficient algorithms for *channel assignment* [57, 61, 58, 59, 62], the task of determining which channel a link should use for data transmission in order to minimize interference for maximum throughput.

The use of multiple radios and multiple channels in wireless networks changes significantly the

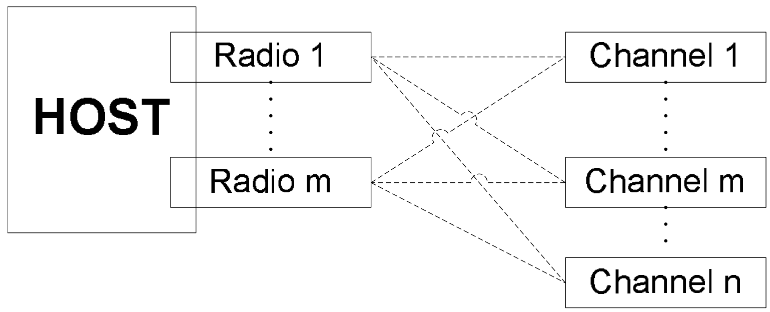

Figure 3: Multi-radio multi-channel model

way routing algorithms and metrics are defined, for both unicast and multicast. Besides measures such as hop count, link bandwidth, and link error rate, the routing algorithms and metrics must consider factors resulting from MCMR operations such as channel switching cost (delay) as well as inter-radio, intra-flow and inter-flow interferences [56, 60, 63].

2.3 Multicast Routing Implementations

Given a multicast routing algorithm, a routing protocol specifies the actions to be taken and messages to be exchanged in order to build a tree based on the algorithm. There are three major issues to be addressed by a multicast protocol: (1) route construction, which specifies how a multicast tree is built, i.e., how routes are established; (2) maintenance of the routing tree, which indicates how the tree is updated upon changes such as members joining/leaving and network topology updates due to node mobility or node/link failures; and (3) data forwarding.

2.3.1 Route Construction

There are two main route construction approaches: table-driven (proactive) and on demand (reactive).

In a table-driven routing protocol, every node maintains consistent, up-to-date routing information to every other node in the network in the form of routing tables. The tables respond to changes in network topology by propagating updates throughout the network in order to maintain a consistent network view.

On-demand protocols, in contrast, do not maintain the information of the entire network topology, but only information of active routes. When a node requires a route to a destination, it initiates a route discovery process within the network. This process is completed once a route is found, or all possible routes have been examined.

Route computations in table-driven protocols are faster than in on-demand protocols thanks to the availability of network topology information at the routers. However, table-driven routing protocols are more expensive to operate than on-demand protocols when network topology or conditions change frequently. The routers need to exchange routing information periodically in order to maintain an up-to-date view of the network topology/conditions at all times.

Table-driven routing is thus suitable for networks with relatively stable topology and stable network conditions that do not change the routing metric(s) often. The majority of Internet routing protocols belongs to this category due to its static topology (e.g., MOSPF [8], DVMRP [7])

On-demand routing better responds to dynamic network conditions or frequent topology changes because a path is computed only when it is needed using the current network information. On-demand multicast routing protocols are commonly seen in MANETs because of their highly dynamic topologies (e.g., ODMRP [9], CAMP [64], MAODV [10]).

There exist also hybrid routing protocols that try to combine the best features of table-driven and on-demand routing. An example is the Multicast Zone Routing (MZR) protocol [65]. In this protocol, nodes within a certain distance from a given node v are said to be within the routing zone of v. Routing within this zone is table-driven. For nodes that are located beyond this zone, an on-demand approach is used.

2.3.2 Maintenance of Routing Trees

Routing trees need to be updated as results of events such as members joining/leaving, network topology changes due to node mobility or node/link failures, and changes of link cost (e.g., available bandwidth, loss rate) or node cost (e.g., residual battery).

There are two mechanisms for updating routing paths/trees: soft state and hard state. In soft state protocols, control packets are flooded *periodically* to refresh the routes. Examples of soft state protocols are DVMRP and ODMRP. Hard state protocols, in contrast, update routes only when triggered by a change in the topology, network condition or multicast membership. For instance, in the ADMR protocol [66] for MANETs, absence of data packets or keep-alive control packets within a multiple of the inter-packet time is considered as an indication of link/node failures. (The inter-packet time is the interval during which a new data/keep-alive packet is expected to arrive.) Only then does the protocol initiate the route repair procedure. Other hard states protocols include MOSPF and BEMRP [67].

Soft state protocols require more control overhead than hard state protocols because the periodic route update procedure is executed even when there are no changes. This may consume network bandwidth and other resources unnecessarily. However, routes in soft state protocols are updated in a more timely manner (when the update frequency is set properly). Hard state protocols suffer from higher latency due to confirmation of the triggering condition (e.g., absence of data/control packets within a multiple of the inter-packet time).

2.3.3 Data Forwarding

For multicast routing in the Internet, a multicast-able router maintains tuples [S, G, L] where S is the source of the multicast group, G is the group ID (a class-D IP address), and L is the list of outgoing interfaces. Multicast data packets belonging to source S in group G are duplicated and queued at the interfaces specified by the list of outgoing interfaces L for transmission to the next node(s).

In wireless multi-hop networks, the transmission of a multicast data packet from a given node to any number of its neighbors can be done with a single data transmission thanks to the broadcast medium (i.e., the *wireless broadcast advantage*). Therefore, in many routing protocols designed for wireless networks (e.g., ODMRP [9], CAMP [64], FGMP [68]), the routers maintain tuples [S, G, b] where b is a boolean variable. If b is true, the router is a *forwarding node* of the multicast group, so it multicasts (broadcasts) the packet to its neighbors. Otherwise, the router discards the packet.

In the example multicast tree shown in Figure 4, nodes A, B and C are forwarding nodes, while X, Y and Z are destinations. A destination can also be a forwarding node, e.g., node Y in this example.

3 Notations and Overview of Routing Metrics

In this section we briefly review commonly seen routing metrics, which can be broadly classified into link, node and path metrics. Detailed descriptions of the metrics are given in Appendix A. These

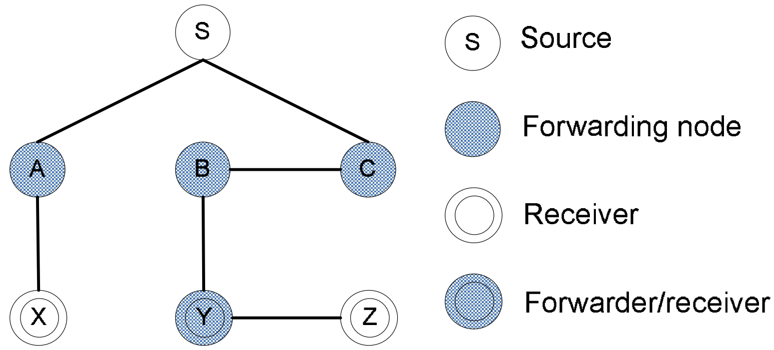

Figure 4: An example multicast routing tree

metrics were first proposed for unicast routing, but many of them have been applied to multicast routing.

We first introduce the notations used to define the routing metrics as well as multicast routing algorithms in later sections.

3.1 Notations

In this article, we consider only undirected graphs. A networks is represented by an undirected graph $G = (V, E)$ where V is the set of vertices and E is the set of edges. The vertices correspond to the routers in a network, and the edges, to the links connected the routers. A link can be denoted by its two end vertices, e.g., link (u, v), where $u \in V$, $v \in V$ and $(u, v) \in E$. To simplify the presentation, we also use a single symbol l to denote a link, with the understanding that l connects two adjacent vertices in V and $l \in E$.

Given a node v, a node u is a neighbor of v if there exist a link connecting u and v, i.e., $(u, v) \in E$. In wireless networks where there are no physical links, u is a neighbor of v if u is located within the transmission range of v. To make the graphs undirected, we assume that u and v have the same transmission range, so that v is also within the transmission range of u.

A path P from a source s to destination d can be considered as a set of concatenated links running from s to d. We use C_P to denote the cost of a path P and w_l, the cost/weight of a link l.

3.2 Link Costs

We list the most commonly used link costs, many of which have been applied to multicast routing. The detailed descriptions of the metrics are given in Appendix A.1.

- Hop count: each link has a cost of one.
- Delay: the delay taken to deliver a packet from one end of a link (transmitter) to the other end of the link (receiver).
- Available bandwidth/capacity: the amount of data that can be transferred over a link over a given period of time.

While the above metrics are applicable to any type of network, the following metrics are intended for use in wireless multi-hop networks. They reflect the characteristics of wireless transmissions and wireless channels (e.g., high error rate, limited bandwidth), and the operations and overhead of the medium access control protocols (e.g., IEEE 802.11 CSMA/CA with RTS/CTS/DATA/ACK exchange).

- Link error rate [69]: a measure of link quality.
- Packet delivery ratio: the number of packets correctly received by the receiver divided by the number of packets sent by the transmitter.
- Expected transmission count (ETX) [70, 71]: the number of required transmissions (including retransmissions) for sending a data packet over a link.
- Expected transmission time (ETT) [56, 72]: defined based on the ETX metric: $ETT = ETX \times S/B$, where S is the packet size and B is the link bandwidth.
- Medium time metric (MTM) [73]: the time required to transmit a data packet over a link, including packet transmission time, medium access control overheads (contention, back-off time, transmission time of Request To Send (RTS), Clear To Send (CTS) and acknowledgment (ACK) messages) and retransmissions (if any).
- Signal strength: an indication of link quality; also an estimate of the distance between the transmitter and the receiver.
- Geographical distance: the distance between the transmitter and the receiver.

3.3 Node Costs

While wired networks typically use link-based costs, routing in wireless multi-hop networks uses both link- and node-based costs. Node metrics were created in response to special properties of wireless networks such as the wireless broadcast advantage and power-limited devices. Following are commonly used node costs.

- Node count: each node has a cost of one.
- Energy consumption (of a forwarding node): the energy required to receive then transmit a packet.
- Residual battery capacity: a measure of the remaining lifetime of a node.
- Residual lifetime: similar to residual battery capacity, but measured in term of number of packets a node can receive and transmit before its battery drains.

Refer to Appendix A.2 for more information about the above node costs.

3.4 Path Costs

In shortest (least cost) path algorithms, the route selection is based on the costs of source-to-destination paths, which are calculated using link costs. There are three ways to calculate paths costs depending on the link cost and the optimization goal: additive, multiplicative, and concave.

3.4.1 Additive Path Cost

The cost C_P of a path P is the sum of the costs of the links on the path.

$$C_P = \sum_{\forall l \in P} w_l$$

For example, the end-to-end delay of a path is the sum of the delay values of all the links on the path. A shortest path is one that has the minimum delay among those considered.

3.4.2 Multiplicative Path Cost

The cost C_P of a path P is the product of the costs of the links along the path.

$$C_P = \prod_{\forall l \in P} w_l$$

As an example, if the link cost is the packet delivery ratio over the link then the path cost is multiplicative and is the product of the link costs along the path.

3.4.3 Concave Path Cost

We say the path cost C_P is concave if

$$C_P = \min_{\forall l \in P} w_l$$

For example, the available bandwidth of a path is determined by the most *bottlenecked* link on the path, i.e., the link that has the lowest available bandwidth.

Given a source s and a destination d, we would like to select a path with the maximum available bandwidth, which is called *maximum bandwidth path* and defined as follows. Let $Q(s,d)$ be the set of all possible paths from s to d. The maximum bandwidth path π from s to d is one that has bandwidth C_π defined as

$$C_\pi = \max_{\forall P \in Q} C_P$$

Wang and Crowcroft showed that the problem of finding maximum bandwidth paths is NP-hard [2]. If there are more than one path from s to d with the same maximum bandwidth then the shortest path among those is chosen. The selected path is called *the shortest widest path*[1].

In this paper, we call paths selected based on a concave function such as the above shortest widest path *best paths* rather than shortest or least cost paths. Furthermore, to simplify the presentation, least cost paths are also referred to as shortest paths.

Path costs using node metrics can be computed using the above three functions in a similar manner.

3.4.4 Path Costs Specific to Wireless Ad-hoc Networks

Given a link/node metric, the cost of a path consisting of several links/nodes can be computed by an additive, multiplicative or concave function presented above. More complex path costs, which are combinations of several metrics, have appeared recently to support routing in wireless ad hoc networks. One example is the Success Probability Product (SPP) proposed by Banerjee et al. [69] (see Appendix A.3.1), which aims at minimizing the energy consumption of a path in wireless multi-hop networks.

Networks with multiple channels and multiple radios have further motivated the creation of several more complex path metrics, which take into account one or more of the following factors: intra-flow interference, inter-flow interference, and channel switching costs. These path metrics include Weighted Cumulative Expected Transmission Time (WCETT) [56], Multi-Channel Routing Metric (MCR) [60], and Metric of Interference and Channel-Switching (MIC) [63]. These metrics assume that a channel assignment algorithm [57, 58, 59, 61, 62] has been applied to the network. The routing protocols will then select the best path based on the given channel assignment result and the routing metric.

[1] This is not to be confused with *widest shortest paths* defined by Guerin et al. [74]. In this path cost definition, the minimum hop count metric is applied first to the find the shortest path between s and d. If there exist several paths with the same minimum hop count, then the path with the maximum available bandwidth is selected.

Having reviewed the routing metrics, we now discuss various algorithms used for constructing multicast routing trees, namely trees based on path costs, minimum cost trees, constrained trees and hybrid trees.

4 Trees Based on Path Costs

In these trees, the shortest or best path from the source to a destination is computed independently of the other source-to-destination paths. All the paths are then merged to form a routing tree.

4.1 Shortest Path Trees

SPT algorithms minimize the cost of the path from the sender to each receiver. In shortest path trees, the path cost can be either additive or multiplicative. Additive path costs are based on link costs such as hop count, delay, geographical distance, ETX, ETT, and MTM, while multiplicative path costs use metrics such as link error rate and packet delivery ratio.

Minimum hop count is the most commonly used path cost for Internet routing because it is simple to compute and minimizes the distance between a source and a destination. Short distances promise low end-to-end delay, which is a desired property of many real-life multicast applications. As a result, the majority of the multicast routing protocols used in the Internet today are based on this metric (e.g., MOSPF [8], DVMRP [7]).

In wireless multi-hop networks, the delay and throughput are influenced by additional factors such as link error rate and medium access contention time. This results in more complex metrics such as ETX, ETT and MTM (Section A.1), which have been applied to both unicast routing [70, 56, 73] and SPT multicast algorithms [75, 76, 77].

The two most commonly used algorithms for computing SPTs are due to Bellman-Ford [4] and Dijkstra [3]. To compute a SPT, we apply the point-to-point shortest path algorithm (Bellman-Ford or Dijkstra's) repeatedly, once for each sender-receiver pair. This is an advantage of SPT algorithms: it is easy to support dynamic members' joins and leaves because every source-to-destination path is computed independently of the others. Another advantage of SPTs is that both Bellman-Ford and Dijkstra's run in polynomial time. In real life, MOSPF [8] uses the Dijkstra's shortest path algorithm and DVMRP [7] is a distributed implementation of the Bellman-Ford shortest path algorithm.

Note that SPTs are not only built by Bellman-Ford or Dijkstra's algorithms (or their variances), but also by several on-demand routing protocols. Implementations of Bellman-Ford or Dijkstra's algorithms are table-driven, which are expensive in wireless networks where channel quality fluctuates often or node mobility results in frequent topology changes (e.g., MANETs). On-demand protocols such as ODMRP [9], CAMP [64], and Multicast Ad hoc Distance Vector (MAODV) [10] have been shown to be more efficient in such environments. Although these protocols do not use Bellman-Ford or Dijkstra's, the routing trees are in fact some form of SPT. For example, in ODMRP, a source s periodically broadcasts Join-Query messages in the network to build/update routes. When a destination d receives the first Join-Query in a broadcast cycle, it sends back to the source a Join-Reply message, which follows the path in the opposite direction of the one traversed by the Join-Query. The path taken by that Join-Query is thus selected (assuming that the source receives the Join-Reply correctly). The established path between s and d can be considered as a path with the shortest delay: there exist several paths between s and d, but d replies only to the first Join-Query it receives and ignores the query messages that traversed the other paths.

4.2 Best Path Trees

One example of best path trees is *maximum bandwidth tree* studied by Shacham [11]. In his algorithm, the best path from the source to a destination is the maximum bandwidth path described in Section 3.4.3. A maximum bandwidth tree is formed by merging the maximum bandwidth paths from the source to the destinations.

Finding the maximum bandwidth path from a source to a destination is an NP-complete problem [2]. Consequently, the problem of computing maximum bandwidth trees is also NP-complete. Shacham proposed a heuristic that works similarly to the Dijkstra's algorithm to compute the (approximate) maximum bandwidth path from the source to each destination of a multicast group [11]. When there are more than one maximum bandwidth path from the source to a destination, the path with the shortest length (i.e., the shortest widest path) is chosen.

5 Minimum Cost Trees

The goal of a MCT algorithm is to minimizes the cost of the entire tree. Two types of algorithms are studied for traditional wired networks: minimum spanning tree and minimum Steiner tree. With the emergence and rapid development of wireless multihop networks, the definition of the tree cost has been revised to reflect the wireless broadcast advantage and/or network-specific properties such as bandwidth or energy constraints.

A routing tree that exploits the wireless broadcast advantage serves two purposes: to minimize the bandwidth consumption (by minimizing the total number of transmissions per packet) and/or energy consumption of nodes in the network. Minimizing bandwidth consumption is critical to achieve high throughput in any type of wireless multi-hop network because of their bandwidth-limited channels (in comparison with their wired counterparts). Minimizing energy consumption is important in networks whose nodes have limited power supply such as MANETs and sensor networks.

There are two types of energy-oriented multicast routing algorithms: one aims at minimizing the overall energy consumption of the multicast tree [78, 79]; the other, maximizing the lifetime of the whole multicast tree [80]. A minimum energy consumption multicast tree may result in rapid depletion of energy at a few forwarding nodes, possibly leading to network partition and interruption of the multicast service. Therefore, the latter approach focuses on maximizing the lifetime of the entire multicast tree (which is determined by the lifetime of the most short-lived forwarding node in the tree).

In this section, we review the algorithms used for minimizing the tree cost, namely minimum spanning tree, minimum Steiner tree, minimum number of transmissions tree, minimum energy consumption tree and maximum lifetime tree.

The following notations will be used to define different MCTs. Assume that $G = (V, E)$ is an undirected, connected and weighted graph with V and E being the set of vertices (nodes) and edges (links) respectively. Each link $l \in E$ is associated with a cost w_l (link costs are presented in Section A.1). Assume that the multicast group has one source s and a set of destinations $D \subseteq V$. Let $t(V', E')$ be a tree that connects the source to the destinations, which is a connected subgraph $G' = (V', E')$ with no cycles, where $V' \subseteq V$ and $E' \subseteq E$. There may exist several of such trees. Let T be the set of possible trees. Let C_t be the cost of a tree t, where the cost function will be defined later for each type of MCT. A minimum cost tree T_μ is one that is defined as

$$C_{T_\mu} = \min_{\forall t \in T} C_t$$

5.1 Minimum Spanning Trees

The objective of a minimum spanning tree algorithm is to compute a tree covering *all nodes* in the network such that the overall cost of the tree is minimal.

Given a graph $G = (V, E)$, a *spanning tree* is a subgraph of G that contains all the vertices V of G and has no cycles. The cost of a spanning tree $t(V', E')$ is the sum of the costs of the edges in that spanning tree, where $V' = V$ and $E' \subseteq E$.

$$C_t = \sum_{\forall l \in E'} w_l$$

A minimum spanning tree T_s is a spanning tree with cost less than or equal to the cost of every other spanning tree.

$$C_{T_s} = \min_{\forall t \in T} C_t$$

The two commonly used algorithms for computing minimum spanning trees are due to Prim [12] and Kruskal [13], which run in polynomial times. Gallager et al. suggested a distributed implementation of the minimum spanning tree algorithm [14].

Minimum spanning trees are more commonly used for broadcasting than multicasting because such a tree connects *all* the nodes in the network. Nevertheless, minimum spanning trees have been used to solve a few multicast routing problems [21, 48, 50, 22]. For instance, Wieselthier et al. use minimum spanning trees to build energy-efficient multicast trees [48]. The idea is to build a broadcast tree and then prune branches to accommodate only the multicast members (see Section 5.4).

5.2 Minimum Steiner Trees

In the Steiner tree problem, given a graph $G(V, E)$, and a set $M \subseteq V$ of required nodes (which can be considered as multicast group members), we want to find a minimum cost tree connecting all nodes in M. The nodes in $V \setminus M$ can be used to connect the required nodes if needed, and are called *Steiner points*. The cost of a Steiner tree $t(V', E')$ is the sum of the costs of the edges in the tree, where $V' \subseteq V$ and $E' \subseteq E$.

$$C_t = \sum_{\forall l \in E'} w_l$$

Note that the Steiner tree cost is written in the same way as the spanning tree cost shown above. The only difference is that $V' \subseteq V$ in a Steiner tree while $V' = V$ in a spanning tree.

A minimum Steiner tree is a Steiner tree with cost less than or equal to the cost of every other Steiner tree.

The minimum Steiner tree problem is NP-complete. Several heuristics have thus been proposed to compute approximate Steiner trees such as the 2-approximation heuristic proposed by Kou *et al.* [15], another 2-approximation heuristic by Takahashi and Matsuyama [16], and the 11/6-approximation algorithm by Zelikovsky [17]. The algorithm by Kou *et al.* [15] is among the most well-known heuristics. The size of the approximate tree is guaranteed to be at most twice the size of the optimal Steiner tree (hence the term "2-approximation"). In practice, this heuristic can give much better results than the claimed guarantee, usually achieving 5% of the optimal tree size for a large number of realistic instances [81]. Winter and Smith studied the performance of several polynomial heuristics [19]. A survey of Steiner tree exponentiation enumeration algorithms is given in [18].

In the Steiner tree problem, given a set $M \subseteq V$ of required nodes, we want to find a minimum cost tree connecting all nodes in M. In practice, it is usually considered that there is only one source node for the whole multicast session. In this case, the set of required nodes is defined as the union of the source and destinations for the purpose of tree cost computation. This is also the most common setup considered in the problem of finding Steiner tree heuristics. For instance, a very simple 2-approximation heuristic proposed by Takahashi and Matsuyama [16] operates similarly to Dijkstra's algorithm. At the beginning, the tree consists of only the source node. Among the destinations currently unconnected, the algorithm searches for a destination d that is closest to the current tree t, and adds to t the shortest path leading to d. The procedure repeats until all destinations are added to the tree.

5.3 Minimum Number of Transmissions Trees

Ruiz and Gomez-Skarmeta explored the problem of multicast routing in wireless multi-hop networks in which nodes are static, e.g., WMNs [20]. The authors re-define the cost of a MCT by applying the *wireless broadcast advantage*: in a broadcast medium, the transmission of a multicast data packet from a given node to any number of its neighbors can be done with a single data transmission. Thus, a minimum cost tree is one which issues a minimum number of transmissions. In other words, the tree contains a minimum number of *multicast forwarding nodes*, rather than having a minimum total edge cost as defined for traditional minimum Steiner trees (MSTs)

The cost C_t of a multicast tree $t(V', E') \in T$ is defined as the number of transmissions required to deliver a packet from the source to all the destinations. (We do not consider retransmissions, and a forwarding node can be a destination itself.) Then $C_t = 1 + |F_t|$, where $F_t \subset V'$ is the set of relay (forwarding) nodes in the multicast tree t. That is, a multicast packet is transmitted first by the source, then by the relay nodes. A minimum number of transmissions tree T_f is a tree with cost less than or equal to the cost of every other tree in set T.

$$T_f = \min_{\forall t \in T} \{1 + |F_t|\} = \min_{\forall t \in T} |F_t|$$

In general, a tree with a minimum edge cost (Steiner tree) may not be one with a minimum number of transmissions (see [20] for an example). Ruiz and Gomez-Skarmeta demonstrated that the problem of computing Minimum Number of transmissions Trees (MNTs) is NP-complete and proposed enhanced heuristics to approximate such trees [20].

Unlike minimum spanning trees and minimum Steiner trees which use link costs, the MNT metric is based on node cost (of one). It exploits the wireless broadcast advantage to minimize

- the bandwidth consumption of the multicast tree by minimizing the number of transmissions per packet
- the total energy consumption of the multicast session, if we assume that all relay nodes require the same amount of energy to receive and forward a packet.

If we assume that the nodes in the network have different levels of energy consumption for a packet reception and transmission, then a different tree cost needs to be defined for the purpose of minimizing the energy consumption in the tree.

5.4 Minimum Energy Consumption Trees

Given a multicast tree $t(V', E')$, let $F_t \subset V'$ be the set of relay nodes in t. The total energy consumption per packet of the multicast tree t is

$$C_t = \sum_{\forall v \in F_t} e_v,$$

where e_v is the energy consumed by node v to receive then transmit a data packet (assuming that all packets are of the same size).

Given a set T of all possible multicast trees connecting the source to the destinations, a minimum energy consumption tree is one that requires that least amount of total energy consumption per packet among the trees in set T.

Optimizing this cost is proven to be NP-hard [78, 79]. Heuristics for computing approximate trees include Embedded Wireless Multicast Advantage (EWMA) [21] and Refining Energy efficiency of Multicast Trees (REMiT) [22, 23] algorithms. EWMA supports both centralized and distributed implementations. REMiT is a family of distributed algorithms, among which S-REMiT [22] and G-REMiT [23] are used for minimizing energy consumption of source-based and group-based multicast trees, respectively, whereas L-REMiT [24] tries to maximize the lifetime of source-based multicast trees (see Section 5.5).

Note that if the energy consumption per transmission and reception of all nodes in the network is the same then the above cost is reduced to minimizing the number of forwarding nodes in the multicast tree (the MNT metric).

5.5 Maximum Lifetime Multicast Trees

In maximum lifetime multicast trees, the multicast tree lifetime can be defined as the time duration starting from the beginning of the multicast service until the first forwarding node in the tree fails due to battery energy exhaustion. In other words, the lifetime of a multicast tree is considered to be the lifetime of the most short lived forwarding node, since the termination of a forwarding node will disrupt the data flow to one or more destinations[2].

Let b_v be the residual battery capacity of a node v, and e_v be the energy require to receive and transmit a packet. The lifetime of node v is defined as $\lfloor b_v/e_v \rfloor$, which is the number of packets v can relay before its battery runs out. Given a multicast tree $t(V', E')$ and its set of forwarding nodes $F_t \subset V'$, the lifetime L_t of t is thus

$$L_t = \min_{\forall v \in F_t} \lfloor \frac{b_v}{e_v} \rfloor$$

The tree cost in this case is a concave function, while the costs of the trees presented in the previous sections are additive (i.e., sums of link/node costs).

The objective of the routing algorithm is then to *maximize* the lifetime of the routing tree, i.e., to find a tree t with the maximum L_t value. MCTs, however, require a minimum value. To conform to the definition of minimum cost tree, we can define the cost C_t of a tree t as $C_t = 1/L_t$. A maximum lifetime multicast tree T_λ is one with the cost defined as follows:

$$C_{T_\lambda} = \min_{\forall t \in T} C_t = \min_{\forall t \in T} \{1/L_t\}$$

This problem is also known to be NP-complete [80]. L-REMiT is a heuristic that aims at maximizing the lifetime of source-based multicast trees [24].

The lifetime cost is a important factor for several critical applications/networks such as battlefield ad hoc networks or sensor networks.

[2]If the multicast routing structure is a mesh as produced by protocols such as ODMRP or CAMP instead of a tree, there may be alternative paths between a source and a destination after the main route is disrupted due to a node failure. However, the protocols do not guarantee that such alternative paths are always available. Hence, the lifetime of the most short lived forwarding node in the tree/mesh is still a concern.

6 Constrained Trees

Due to the very large scale of the Internet, the distance between a source and a destination may be very long, and the end-to-end delay may become unacceptable for many applications such as conferencing, virtual classrooms, and video-on-demand. This problem has spawned a new type of routing algorithms that enforce a constraint on the end-to-end delay. That is, the end-to-end delay from a source to each destination must be less than some pre-determined threshold. If such a path cannot be found, the new connection request is denied and the user is notified. Delay-constrained trees support a form of QoS routing. There exist algorithms that incorporate delay constraints into SPTs [25, 26], minimum spanning trees [27, 28] and MSTs [32, 29, 33, 34, 35, 36, 30].

Besides delay, other types of constraints have also been considered, e.g., delay variation[38, 39], link bandwidth [40] and outgoing degrees of forwarding nodes [41].

6.1 Delay-constrained Trees

A delay constraint can be imposed on any type of tree mentioned earlier. In practice, it has been applied to shortest path trees, minimum spanning trees and Steiner trees.

6.1.1 Delay-constrained Shortest Path Trees

The SPT problem can be solved in polynomial times using the Bellman-Ford or Dijkstra's algorithm. However, finding a SPT with a delay constraint is a NP-hard problem [82]. Deering et al. [25] and Sun et al. [26] provide heuristics for efficient implementations of delay-constrained SPTs.

6.1.2 Delay-constrained Minimum Spanning Trees

The minimum spanning tree problem can be solved in polynomial time by Kruskal's or Prim's algorithm [83]. Nonetheless, similarly to the SPT problem, the minimum spanning tree problem becomes NP-hard when delay constraints are applied to the resulting paths in the routing tree [27].

Salama et al. proposed a simple heuristic, which resembles Prim's algorithm, to give an approximate solution to the problem [27]. Chow also suggests a heuristic which combines different routes into one single routing tree [28]. A comparison of several algorithms for the delay-constrained minimum spanning tree problem can be found in [84].

6.1.3 Delay-constrained Steiner Trees

Delay-constrained Steiner tree is the most well studied type of constrained trees. This is also an NP-complete problem. There exist numerous heuristics that build approximate Steiner trees with delay constraints [32, 29, 33, 34, 35, 36, 30].

Detailed summaries of delay-constrained Steiner tree algorithms can be found in the surveys by Oliveira and Pardalos [85] and Wang and Hou [86].

6.2 Constraints on delay variation

Let $d_1, d_2, \ldots, d_n$ be the end-to-end delay between a source and n destinations of a multicast group, respectively. Let $d_{max} = \max\{d_1, d_2, \ldots, d_n\}$, $d_{min} = \min\{d_1, d_2, \ldots, d_n\}$, and $\delta = d_{max} - d_{min}$ be the *delay variation* of the multicast tree.

In some applications, it is desirable that all participants receive the same information at about the same time so that no one could gain an advantage over some others by having the information sufficiently earlier. Thus the delay variation δ must stay within a specific range. Haberman, Rouskas and Baldine propose algorithms for solving the problem of Steiner tree with delay and delay variation constraints [38, 39].

6.3 Constraints on link capacity

Link capacity is also used as a constraint in the Steiner tree problem. Jiang solved this problem for the application of video conferencing [40]. A video conference with N participants requires N separate Steiner trees, from each participant to the others. Jiang proposed heuristics to build such Steiner trees, taking into account the link bandwidths and constraint on the bandwidth.

6.4 Degree-constrained Steiner trees

In some kinds of high speed networks, the speed requirements may impose a limit on the number of interfaces a switch/router can have. For instance, in asynchronous transfer mode (ATM) networks, the number of outgoing links of a switch may be limited to a fixed value [87]. As a result, multicast routing algorithms in these networks must consider the node degree constraint while building multicast trees.

Bauer and Varma [41, 42] studied the Steiner tree problem in combination with the restriction in the number of adjacent nodes. They show that their proposed heuristics for degree-constrained Steiner tree give solutions very close to the optimal solution to the general Steiner tree problem in a number of sample instances.

7 Hybrid Trees

In hybrid trees, two (or possibly more) metrics are optimized at the same time, e.g.,

- optimizing both the tree cost and source-to-destination path cost (delay) [43, 32];
- minimizing the bandwidth consumption of the multicast session at the same time as minimizing the tree cost or source-to-destination path lengths [45];
- minimizing the energy consumption of the multicast tree at the same time as maximizing the tree lifetime [48, 49].

7.1 Optimizing Tree Cost and Delay

Chung et al. proposed heuristics that try to optimize the cost of the routing tree (Steiner tree) and the maximum end-to-end delay at the same time [44].

Bharath-Kuma and Jaffe [43] study algorithms to optimize both the tree cost and source-to-destination path lengths. If we consider path length as an indicator of end-to-end delay (at least in wired networks), then these algorithms are considered to optimize both the tree cost and delay.

Note that the resulting end-to-end delays in these algorithms should be treated as a "best-effort" service, because although the tree is optimized with respect to end-to-end delay, some path can still have delay greater than a predetermined value Δ. This is different from delay-constrained trees, which contain only paths with delay equal or smaller than Δ. (It may happen that we cannot find a path from the source s to some destination d in delay-constrained tree algorithms, because all possible paths from s to d have delays greater than Δ.)

7.2 Optimizing Bandwidth Consumption and Path Length

Assuming that each link in a *wired network* consumes a unit of bandwidth and has a cost of one, a minimum Steiner tree is one that consumes the least amount of bandwidth, because it contains the least number of links (edges) among all possible trees. That is, all source-to-destinations paths share as few links as possible in a minimum Steiner tree. However, source-to-destination paths in a MST are longer than those in a SPT. On the other hand, a SPT is not bandwidth-efficient, because it uses more links than a MST.

Fujinoki and Christensen propose an algorithm that try to optimize bandwidth consumption and path length at the same time [45]. In other words, it optimizes both the tree cost and the path lengths, assuming that the link cost is one and each link consumes a unit of bandwidth. In wired networks, the less number of links in the tree, the less bandwidth it consumes[3]. In the first phase, the algorithm builds a shortest path from the source to the new destination. The path length is the number of hops between the source and the destination. In the second phase, the path is modified so that the new path shares as many links as possible with the existing tree.

Unlike in wired networks, bandwidth consumption in wireless multi-hop networks can be minimized by taking advantage of the broadcast nature of wireless transmission, i.e., by minimizing the required number of transmissions per packet, or equivalently, the number of forwarding nodes, as discussed in Section 5.3. However, source-to-destination paths in a MNT is typically longer than that in a SPT. This has an impact on the loss rate, end-to-end delay and delay jitter [88, 89]. Therefore, algorithms have been proposed that attempt to construct shortest paths while minimizing the number of forwarding nodes in the multicast tree [46, 47]. For example, the protocol by Zeng et al. [46] first constructs shortest paths from the source to *all* the nodes in the network using breadth first search (BFS) algorithm [83]. Based on this BFS tree, the multicast tree is built gradually. First a set of nodes M consists of only the source and the destinations. Nodes from the BFS tree are then selected as relay nodes and added to M. A non-relay node that can reach the most number of destinations and relay nodes currently in M is chosen and added to M. This continues until all paths to the destinations are established. By maximizing the number of multicast neighbors a forwarding node can reach, the protocol minimizes the number of forwarding nodes in the tree.

Considering the above algorithms, we observe that algorithms for wired networks minimize bandwidth consumption by minimizing the number of links in the tree, while algorithms for wireless networks minimize the number of (relay) nodes. That is, a wireless-based algorithm would try to re-use *nodes* that is already in the tree for future paths to new destinations, while an algorithm targeted for wired networks would re-use existing *links* in the tree.

7.3 Optimizing Energy Consumption and Tree Lifetime

Both goals are important in MANETs and sensor networks, whose nodes have limited battery capacity. Wieselthier et al. propose a family of algorithms that optimize both energy consumption and tree lifetime. Their Broadcast Incremental Power (BIP) algorithm constructs source-based broadcast trees [48, 49]. Starting from on a BIP tree, the Multicast Incremental Power (MIP) algorithm builds a source-based multicast tree by pruning unused branches [48]. The BIP and MIP algorithms incorporate the initial and residual battery levels of nodes into the tree cost computation for the purpose of prolonging the lifetime of a multicast tree.

Banerjee et al. extend the BIP and MIP algorithms to support reliable broadcast and multicast by incorporating link error rates into the node cost [50].

[3] We can say that the algorithm by Fujinoki and Christensen [45] and those studied by Bharath-Kuma and Jaffe [43] (Section 7.1) try to solve the same problem (i.e., optimizing both path length and the tree cost) using different approaches.

8 Discussions

Having reviewed existing multicast routing approaches and algorithms, we now discuss important performance, design and implementation issues. In particular, we compare shortest path tree and minimum cost tree, the two most popular multicast routing approaches, with respect to their performance and implementations. We discuss many-to-many multicast, and design issues for multicast routing in wireless multi-hop networks.

8.1 Shortest Path Trees vs Minimum Cost Trees

We compare the performance of SPTs and MCTs, and consider the issues of handling dynamic membership and distributed implementations.

8.1.1 Performance

The majority of the multicast routing protocols used in the Internet today are based on shortest path trees with the hop count metric (e.g., MOSPF, DVMRP), because they are easy to implement and they provide minimum delay from the source to each destination, which is a desirable property for most real-life multicast applications. The end-to-end delay metric is a very important performance parameter for very large scale systems such as the Internet.

On the other hand, MCTs are needed for many specific applications or networks to minimize bandwidth usage [90] or energy consumption [48, 50, 21, 22]. However, the distance between the source and a destination in a MCT is typically longer than that in a SPT, by definition of SPTs. Therefore, MCT algorithms are usually combined with a delay constraint, or both the tree cost and end-to-end delay are optimized at the same time (Section 6).

The impact of path length is more prominent in wireless multi-hop networks using contention-based MAC such as CSMA/CA than in wired networks. In general, the longer the path length,

- the higher the probability a packet will be lost or damaged, hence the higher the packet loss rate;
- the longer end-to-end delay;
- the higher the delay jitter [88, 89].

Another important routing metric in wireless networks is the number of forwarding nodes in a multicast tree. The higher the number of forwarding nodes in a multicast tree, the more traffic it generates in the network, causing more congestion, channel contention and packet collisions. This negatively impacts the performance of both the multicast session and other flows in the network. Compared with MCTs such as minimum Steiner trees (Section 5.2) and minimum number of transmission trees (Section 5.3), SPTs have the advantage of shorter path lengths, but the disadvantage of higher numbers of forwarding nodes. Nguyen and Xu [88, 89] explore how these two factors, path length versus number of forwarders, affect the other performance metrics such as packet delivery ratios, end-to-end delay, delay jitter and throughput. Experimental results [88, 89] show that SPTs perform significantly better than MCTs in terms of PDR, throughput, end-to-end delay and delay jitter. However, when the group size is large and the multicast sending rate is high, SPTs cause more packet losses to other flows in the network than MCTs, due to more forwarding nodes used by SPTs.

Future research is needed to compare the performance of SPTs and MCTs in multi-channel multi-radio networks using effective channel assignment algorithms and routing metrics that take advantage of multiple channels and multiple radios.

8.1.2 Handling Dynamic Membership

It is much easier to support dynamic joins and leaves using SPTs than MCTs, because in a SPT each source-to-destination path is established independently of the other paths in the tree. In a MCT, a node joining or leaving the multicast session may require the whole tree to be re-computed in order to maintain the cost optimality (or the new tree would no longer be optimal). It would be very expensive to compute a new tree for every membership change. This would also cause disruption of the multicast service and higher delay.

Researchers in the topic of multicast routing have proposed a number of methods for solving the problem of updating MCTs efficiently upon membership changes. One approach is to cache previously computed trees for future use. Kheong et al. [91] proposed an algorithm that maintain caches of pre-computed multicast trees from previous groups. The cache can be used to quickly compute a path for a new member. The proposed algorithm retrieves data from the path cache and searches for similarities between the previous and the current multicast groups. It then builds a source-to-destination path using parts of the paths stored in the cache.

A more popular approach is to perform multiple additions and/or deletions before re-computing the entire tree. An index is defined and monitored, which indicates the quality factor of the multicast tree as members join/leave. When the quality factor (index) reduces to a specific threshold, this triggers a tree re-computation. Algorithms using this approach include those by Sriram et al [33], Narvaez et al. [92], Bauer and Varma [93], and Imase and Waxman [94].

There exist also MCT algorithms that do not recompute the whole tree as node joins and leaves. Instead, the algorithms try to keep the tree cost low by re-using links (nodes) already in the tree for new destinations. Such an algorithm was proposed by Feng and Yum [34], which operates in a similar manner to Prim's algorithm for minimum spanning tree. The algorithm maintains a priority queue Q containing the already connected destinations. Given a new destination D, the algorithm uses a parameter k to determine how to compute the path from D to the current tree. It computes k minimum delay paths, from the current destination D to each of the smallest k elements in the priority queue. The path with the smallest delay among these paths is then chosen and added to the current tree. There is a trade-off when setting the value of k: when k increases, more minimum delay paths need to computed, but will result in better solutions.

8.1.3 Distributed Implementations

It is easy to implement SPTs in a distributed manner using Dijkstra's algorithm or distributed versions of Bellman-Ford [5, 6]. Therefore, in practice, SPT is the more commonly used type of tree for large-scale networks such as the Internet.

It is more difficult to compute MCTs in a distributed manner. Nevertheless, MCTs play an important role in several applications and networks, especially wireless multi-hop networks. These networks require minimum bandwidth consumption to achieve high throughput. MANETs and sensor networks require minimum energy consumption and prolonged lifetime. With respect to performance, MCTs incur less data overhead than SPTs for large groups in wireless multi-hop networks [88]. Therefore, it is important to devise distributed implementations of MCT algorithms to improve their scalability and time complexity. Distributed strategies of several MCT algorithms have been proposed, such as distributed minimum Steiner tree algorithm by Bauer [95], and distributed energy-efficient tree algorithms for wireless networks by Wieselthier et al. [49], Cagalj et al. [21], and Wang et al. [22, 23, 24].

Similarly, distributed implementations of delay-constrained trees have also been researched extensively. Example algorithms include distributed delay-constrained minimum Steiner tree algorithms by Kompella et al. [29], Jia [30], and Shaikh and Shin [37]; and distributed delay-constrained minimum spanning tree algorithms by Kompella et al. [29], Chen et al [31], and Jia [30];

8.2 Many-to-many Multicast

SPTs by definition are per sender. Therefore, for many-to-many multicast, separate trees need to be computed, one for each sender. When there are m senders in a multicast session, m separate trees are computed. If the multicast group is large, it requires a large amount of storage to store the routing information of the whole session at routers. Therefore, shared trees (also called core-based trees) were proposed to solve this problem [96, 97, 98].

8.2.1 Shared Trees

In shared tree protocols, a single tree is shared by all the sources within the multicast group and is rooted at a node referred to as the *core* node. The sources send their data to the core, which then multicasts the data to the receivers in the tree rooted at itself.

In a scheme like DVMRP or MOSPF which uses source-based SPTs, given a multicast group with m sources, a router may need to maintain as many as m entries of the form (S_i, G) where

- $i = 1, \ldots, m$;
- S_i is the ith source;
- G is the multicast group ID, a class-D IP address.

On the other hand, a router using a shared tree protocol needs to maintain only a single entry of the form $(*, G)$, irrespective of the number of sources, because there is only one tree rooted at the core.

Examples of real-life implementations of shared trees include Protocol Independent Multicast (PIM) [98] and Core-Based Tree (CBT) [97]. PIM introduces the notion of a rendez-vous point (RP), which acts as a meeting place of the receivers and the senders. The receivers explicitly join a tree rooted at the RP. A source will then send data to the RP which will then relay (by multicasting) the data to the receivers in the shared RP-tree. Similarly, CBT sets up a single shared bidirectional tree connecting the senders and receivers. It should be emphasized that the reason for using shared trees in the Internet (as opposed to per-source trees) is to reduce the amount of routing information to be stored in routers rather than to minimize the overall tree cost.

In the context of implementation, core-based trees are used in MANETs as well, but for a different purpose. In several mesh-based protocols such ODMRP [9] and FGMP [68], all sources of a multicast group periodically flood control packets in the network to create and maintain the routing mesh. This incurs very high traffic overhead when the number of sources is large. The Core-Assisted Mesh Protocol (CAMP) [64] tries to overcome this problem by using a core node in a routing mesh. Only the core node floods control packets periodically, significantly reducing the control overhead.

A critical issue in CBT algorithms is how to select the core so as to optimize the performance of the multicast session. Ideally, the core should be the closest node to any other node in the network (in anticipation of new members joining in the future). That is, the core of a graph $G(V, E)$ is a node c which minimizes $\max_{v \in V} d(c, v)$, where $d(c, v)$ is the distance between c and v and usually measured by hop count. Optimal core placement is an NP-hard problem [97], and heuristics have been proposed for efficient implementations [97, 98, 99]. Calvert et al. present an experiment-based comparison of different methods used for computing the optimal core of a routing tree [100].

It should be noted that in actual implementations of shared trees, the tree rooted at the core is a SPT. In theory, a shared tree could be any type of tree described in Sections 5, 6 and 7.

8.2.2 Shared Trees versus Source-based Trees

For many-to-many multicast, shared trees offer better scalability than source-based trees in several ways:

- For routing on the Internet, which has very large scale, shared trees reduce the amount for routing information to be stored at routers. For each multicast group, the number of entries in the routing table at a router is $O(1)$ (shared tree) versus $O(m)$ (source-based tree), where m is the number of sources in the multicast group.
- The number of forwarding nodes in a shared tree is smaller than that in a source-based tree. This has an impact on the traffic overhead in the network, especially in wireless multi-hop networks. The less number of forwarding nodes, the less bandwidth the routing tree consumes.
- Shared trees reduce the traffic overhead of control packets needed to maintain the routing tree in on-demand routing protocols. This is an important issue in networks where the topology changes frequently such as MANETs. For instance, CAMP uses a shared tree to reduce the amount of control packets that would otherwise be flooded periodically by all the sources in order to update the routing structure, as done in ODMRP or FGMP.

Compared with per-source trees, shared-tree algorithms produce source-to-destination paths longer than necessary. Longer paths typically lead to longer end-to-end delay: Wall proved that the bound on maximum delay of an optimal shared tree is two times the shortest-path delay [96]. In wireless environments, the longer the path length, the higher the probability that a packet will be lost or damaged. This performance issue requires careful placement of the core in the network [97, 98, 99]. For real-time applications where low end-to-end delay is critical, source-based trees are the better routing structure.

Shared trees also suffer from the *traffic concentration problem*: the traffic from all the sources of a group is carried by only one tree. Thus when the source rates are high (e.g., in a video conference) source-based trees perform better than shared trees in wired networks thanks to efficient traffic distribution among paths in several trees [99, 98]. (This performance gain of source-based trees may not be necessarily true in contention-based wireless multi-hop networks, because a higher number of forwarding nodes from multiple source-based trees may incur more contention and packet collisions in these networks than a single shared tree.) An analysis of the trade-offs between shared trees and source-based trees for routing in the Internet can be found in [101, 98].

On the other hand, the core of a shared tree is a traffic bottleneck, which is a serious issue in wireless networks using contention-based MAC such as CSMA/CA. The contention at the core among packets from the sources is high under high multicast loads, which may result in unacceptable performance when there is a large number of sources. In Figure 5, data packets from nodes v_1, v_2 and v_3 must contend for the medium to reach the core C. Furthermore, multicast packets from the core interfere with (unicast) packets from the sources arriving at the one-hop neighbors of the core. In the above example, multicast packets from the core C interfere with the receptions of unicast packets from the sources at nodes v_1, v_2 and v_3. Future work is needed to quantify the performance differences between shared trees and per-source trees in wireless multi-hop networks.

In networks with energy-constrained nodes, the battery of the core may be depleted more quickly than those of the other relay nodes in the tree due to its intensive tasks. With respect to fault tolerance, the core is a critical point of failure; routing protocols thus should make provision for the case of core failure by using secondary or backup cores.

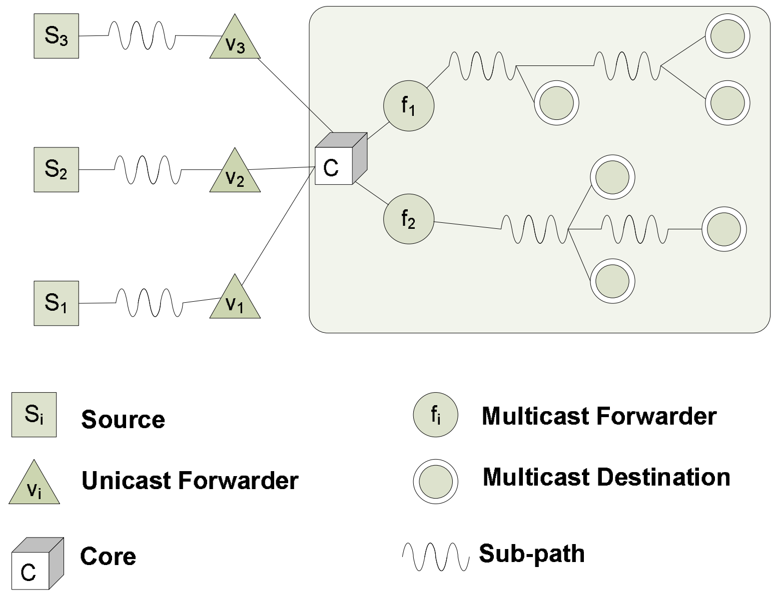

Figure 5: A shared tree with core C and sources S_1, S_2, and S_3

8.3 Design Issues for Wireless Ad hoc Networks

The characteristics of wireless ad hoc networks are radically different from those of wired networks, e.g., high bit error rates, highly fluctuating and unpredictable channel conditions, location-dependent contention, and dynamics and constraints of mobile devices. Therefore, multicast routing algorithms have been adapted to the characteristics of wireless ad hoc networks. For instance, frequent topology changes in MANETs requires on-demand implementations and mesh routing structures such as those in ODMRP, CAMP, and FGMP. Multicast routing in MANETs and WSNs are also concerned with minimizing energy consumption and/or maximizing the session lifetime. SPT algorithms in WMNs use more complex metrics such as ETT, ETX, and SPP, instead of the simple hop count metrics used by Internet multicast routing. These metrics require information from the physical and/or MAC layers to enhance the performance of the routing protocol at the network layer. This design technique is called *cross-layer optimization*. Multi-channel multi-radio (MCMR) systems have also been researched extensively to further improve network throughput and scalability.

Algorithms minimizing bandwidth consumption and energy consumption were presented in Sections 5 and 6. We now briefly discuss the issues of mesh routing structures, cross-layer optimization and MCMR operations in wireless networks.

8.3.1 Mesh Routing Structures

The traditional multicast tree structure does not perform well in MANETs because of the dynamic nature of the topology. In a routing tree, every multicast packet traverses each node and each link in the tree only once. Therefore, a tree in MANETs could easily break due to the mobility of multicast nodes. Furthermore, re-constructing and maintaining the tree can incur substantial control traffic overhead due to the updates and maintenance of global routing information.

Mesh routing structures were proposed to provide more reliable routing in MANETs. In routing meshes, there is usually more than one path between a source-destination pair thanks to the wireless broadcast advantage. Figure 6 shows an example of such a routing mesh. There exist several paths between source S and a destination. For instance, possible paths between S and D_3 include $S-Z-W-D_3$, $S-X-Z-W-D_3$, $S-X-Y-W-D_3$ and $S-Z-X-Y-W-D_3$. Mesh-based protocols are thus more robust than tree-based protocols due to the availability of multiple paths between the multicast source and a destination.

Note however that the mesh structure may result in routing loops. Mesh-based routing protocols thus must consider this problem. For example, in ODMRP, a forwarding node keeps track of the packets its has received and forwarded using packet sequence numbers. If it receives the same packet more than once, the duplicates are discarded to avoid routing loops.

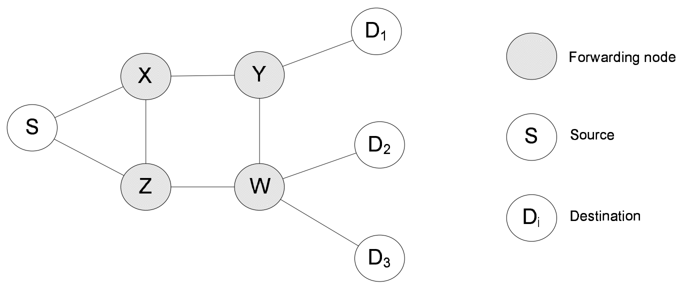

Figure 6: An example routing mesh. The links are not physical links but rather wireless connections.

8.3.2 Cross-layer Optimization in Wireless Networks

In wireless networks, there are many interactions among the transport, routing, MAC and physical layer protocols. Cross-layer design can improve the network performance and scalability. For example, the transmission power and rate at the physical layer will influence MAC throughput and routing decisions. The link selection in the routing layer will affect the MAC layer contention level. In addition, the MAC protocol can adapt the back-off window size according to the end-to-end delay information provided by the transport layer. The above example interactions show that design of routing algorithms should take advantage of and can benefit from cross-layer optimization.

An example is the cross-layer (unicast) routing algorithm called mesh routing strategy introduced by Lannone and Fdida [102]. The goal of the algorithm is to find high throughput paths with reduced interference and increased reliability by optimally controlling transmission power. It is observed that the more power used, the lower the packet error rate, but the higher the interference. The algorithm searches the optimal trade-off by setting an optimal transmission power level that minimizes the distance from the ideal optimum.

Other advances in wireless communications such as multi-rate, multi-radio, multi-channel, smart antenna technologies further encourage and enable cross-layer design in high-performance scalable wireless multi-hop networks such as WMNs[4].

It should be noted that the IEEE 802.11 standard currently does not provide much information about the MAC and PHY layers to the higher layers. The only available information about channel quality is the received signal strength at the PHY layer, whose values are allowed to be vendor-dependent. However, IEEE 801.11k standards are being prepared, which will enable higher layers to obtain far more detailed information about channel conditions from the MAC and PHY

[4] Nevertheless, cross-layer design should be used with caution because it may result in the loss of design abstraction, incompatibility with existing protocols in the traditional protocol stack, unforeseen impact on the whole system, and difficulty in management [103]. Some cross-layer design principles have been suggested to avoid these potential problems [103].

layers. Available measurements will include a standardized signal strength measurement as well as a "neighbor report" that includes information on neighboring nodes that have been detected. The information would help create better routing algorithms for high throughput and QoS routing.

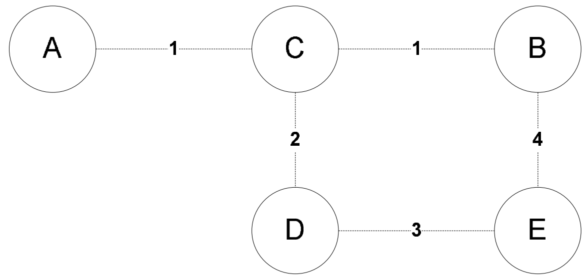

Figure 7: An example of the close relationship between routing and channel assignment

8.3.3 Channel Assignment and Routing in MCMR Networks

Designing routing algorithms and metrics is closely related to the channel assignment (CA) problem in MCMR networks. The following example illustrates that close relationship. In this example, there are two paths between a source A and destination B: $A-C-B$ and $A-C-D-E-B$ (see Figure 7). If the objective is to minimize the end-to-end delay and we assume that the traffic load is low enough so that the delay of transmitting a packet on a link is approximately close to the transmission time, and that the bandwidth of all the links are the same, then path $A-C-B$ should be selected because its end-to-end delay is $2s/b$, while the end-to-end delay of the other path is $4s/b$, where s is the packet size and b is the link bandwidth. On the other hand, if we wish to maximize the end-to-end throughput, then the better path of the two is $A-C-D-E-B$. All links on this path use a different channel and thus can be active simultaneously, resulting in an end-to-end throughput of b. In contrast, the end-to-end throughput of path $A-C-B$ is only $b/2$ since the two links use the same channel, and hence cannot be utilized at the same time.

The above example illustrates the close relationship between routing and CA. Thus the two problems should be treated jointly in order to maximize performance. This is the approach used in the routing/CA algorithm by Alicherry et al. [58]. The authors formulated the CA, routing and link scheduling problems jointly, resulting in a mixed integer linear programming model. An approximation algorithm was proposed that maximizes the network throughput under a fairness constraint. Mohsenian et al. [104] use a similar method, formulating the CA problem as a non-linear program and proposing an algorithm that computes a log-quadratic formula representing the solution. The goal of the algorithm is to maximize the aggregate utility across all sources. Both algorithms allow channel reassignment when the traffic loads vary.

The joint approach, however, is generally complex to solve and difficult to obtain optimal solutions. Therefore, the more commonly seen approach is to do CA first then routing second, or vice versa. If CA is done first, the routing algorithm/metric can estimate intra-flow and inter-flow interference based on the CA result, and is thus called *interference-aware* routing [59]. The WCETT, MCR and MIC path metrics presented in Section A.3.2 are interference-aware routing metrics.

There exist numerous CA algorithms [59, 105, 106, 107, 108] for the CA-then-routing approach. These algorithms are designed for unicast communications and not readily applied to multicast. Consider a 4×4 grid network shown in Figure 8 and the CA resulting from the CA algorithm by Kyanusar et al. [60]. The example illustrates a multicast group with source S and three destinations D_1, D_2 and D_3. Obviously S cannot multicast a data packet to all three destinations using one transmission, because links (S, D_1) and (S, D_2) use channel 1 while link (S, D_3) use channel 2. In this case, S needs to transmit two copies of the packet (either sequentially or in parallel), one on channel 1 and the other on channel 2. This example also shows that routing algorithms designed specific for multicast are needed to re-use the existing CA algorithms, e.g., those in [57, 61, 59, 62, 58, 105, 106, 107, 108]. Also, interference-aware routing metrics such as WCETT, MCR and MIC cannot not be applied to multicast without modifications.

Another weakness of the CA-then-routing approach is that the CA algorithms assume uniform load distribution across links. In real networks, some of the mesh routers serve as gateways to the Internet; the traffic to and from these gateways may be much higher than the traffic in other parts of the network. Hence the computed CA may not be optimal in a real network.

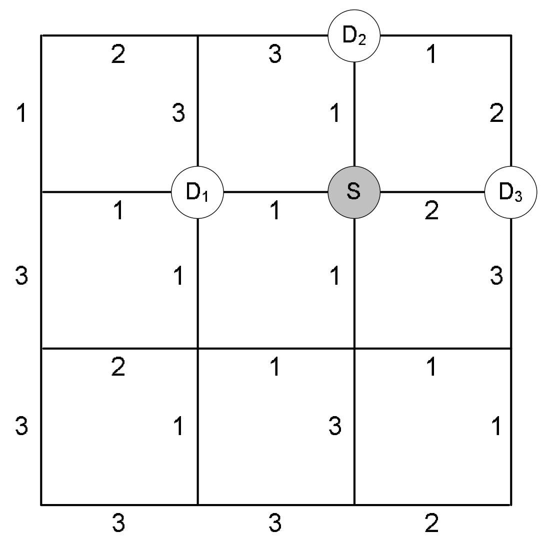

Figure 8: A network and an example channel assignment result (adapted from [60]). There are three available channels and each node has two radios. Each link is labeled with the assigned channel.

An alternative approach is to perform routing before CA. In this case, the traffic load on each link is determined by the choice of the routing protocol and associated routing metric. The CA should then take into account the link traffic loads to minimize interference, and is thus called *load-aware* channel assignment [57]. The most popular algorithms using this approach were proposed by Raniwala et al. [57, 61]. Given a source and destination, the algorithms first attempt to construct a route and then perform CA based on a pre-determined criterion. Since link traffic loads vary over time, these two functions, routing and CA, are performed periodically one after another until the system is stable. Wu et al. [109] suggested a distributed CA and routing scheme that operates between the MAC and network (routing) layers. A node may initiate a channel reassignment if the channel utilization of its corresponding interface is higher than a predetermined threshold.

The routing-then-CA approach considers both interference and link loads in the computation, leading to load balancing over multiple channels. It adapts better to the network dynamics than the CA-then-routing approach. However, it is very difficult to estimate the dynamic traffic loads, and expensive to update and maintain this information. Routing-then-CA algorithms may experience non-convergent behaviors, for example, as the result of simultaneous discoveries of an underutilized channel by several nodes [57]. Furthermore, reassignments may lead to excessive topology alterations but offer only marginal improvements; how to provide the best trade-off between system stability and performance is a challenging issue.

Among the three approaches, only the routing-then-CA approach has been considered for multicast routing [46, 110, 111]. In the protocol by Zeng et al.[46], a routing tree is first constructed (as described in Section 7.2), and a CA algorithm is applied to the multicast tree. The protocol, however, considers only interference within the multicast tree. Yin et al. [111] take into account interference from neighboring flows in their CA algorithm. The algorithm depends on the use of the probability that a channel is being busy. The paper did not mention how to compute this probability; furthermore, collecting and maintaining this information for all links in the network would incur high overheads.

High-performance multicast routing in MCMR networks is a challenging problem that requires further research to exploit the advantage of MCMR technology for maximizing network throughput.

8.3.4 Multiple Channels and Loop-free Routing

Using multiple channels and their interference information in MCMR path metrics such as WCETT, MCR and MIC (Appendix A.3.2) may result in routing loops, also called non-isotonicity property [112]. A routing metric is isotonic if it satisfies the following condition. Let $C(P)$ denote the cost of a path P. For any two paths P_a and P_b, if $C(P_a) \leq C(P_b)$, then for any path P_c, it must be that $C(P_a \oplus P_c) \leq C(P_b \oplus P_c)$ if P_c follows P_a and P_b, or $C(P_c \oplus P_a) \leq C(P_c \oplus P_b)$ if P_c precedes P_a and P_b. The $\oplus$ symbol denotes path concatenation.

The non-isotonicity property of the WCETT metric can be illustrated by the example in Figure 9. In this example, all links have ETT = 0.3, except link SA whose ETT value is 0.2. Applying the WCETT path cost given in Appendix A.3.2 with parameter $\beta = 0.5$, we have $C(S-A-C) = 0.5 \times (0.2+0.3+1) = 0.75$ and $C(S-B-C) = 0.5 \times (0.3+0.3+1) = 0.8$. Thus $S-A-C$ is the better path. When we append path $C-D$ to $S-A-C$ and $S-B-C$, the WCETT values of the new paths become: $C(S-A-C-D) = 0.5 \times (0.2+0.3+0.3+2) = 1.4$ and $C(S-B-C-D) = 0.5 \times (0.3+0.3+0.3+1) = 0.95$. Path $S-A-C-D$ now incurs higher cost than the other path, which proves that WCETT is not isotonic.

Yang et al. pointed out the non-isotonicity property of WCETT, MCR and MIC metrics [112], and introduced the concept of virtual links and nodes in their LIBRA routing protocol [113] to make MIC isotonic.

Although the WCETT, MCR and MIC metrics are designed for unicast routing, the above example shows that the isotonicity property of a routing metric must be carefully considered when designing routing algorithms and protocols for multicast in MCMR networks in order to avoid routing loops.

8.4 Summary: Design Issues and Challenges

Designing multicast routing algorithms goes hand in hand with routing metrics, and depends on the network environment as well as the performance requirements of the application.

Network Environments

For Internet routing, SPT is the most commonly deployed algorithm due to its easy implementations and low end-to-end delay. MCTs such as Steiner trees have also been studied extensively for purposes such as minimizing bandwidth usage. Because of the very large scale of the Internet, research on multicast routing algorithms focuses on bounded delay or optimized delay to meet the

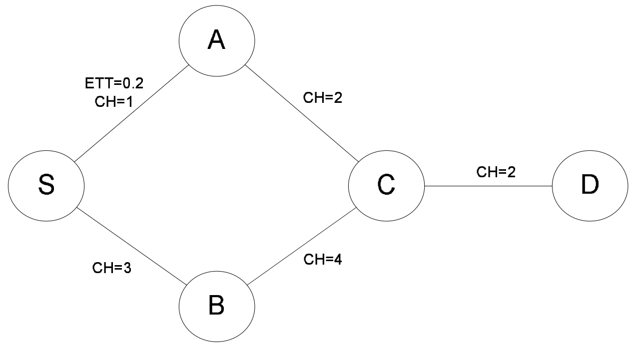

Figure 9: Non-isotonicity property of path metric WCETT. All links have ETT = 0.3, except link SA which has ETT = 0.2. Each link is associated with a channel number given by a CA algorithm.

requirements of many applications. Other issues related to multicast routing algorithms for the Internet include:

- distributed operations for improved time complexity and scalability;
- minimal storage and processing at routers to accommodate large groups of thousands to hundreds of thousands of members;
- inter-domain operations in the presence of heterogeneous networks and computers;
- QoS routing.

Wireless multi-hop networks have smaller scales, but more error-prone channels, more limited bandwidth, hidden/exposed terminal problems and location-dependent contention. Although high throughput, low delay and distributed operations remain important goals, the focus of multicast routing algorithms and metrics has shifted to the following issues:

- minimizing bandwidth consumption (e.g., by minimizing the number of forwarding nodes);
- minimizing energy consumption, especially in MANETs and WSNs;
- cross-layer optimization for maximum throughput and QoS routing;
- multi-channel multi-radio (MCMR) operations for improved throughput and scalability.

Application Requirements

With respect to performance, multicast applications can be broadly classified into three categories based on reliability and end-to-end requirements. At one end of the spectrum are interactive real-time applications (e.g., tele-conferencing, IP TV, video on demand) whose latency requirements are very stringent, typically on the order of 100 ms. However, these applications can tolerate some loss because of the inherent redundancy in audio and video data. At the other end of the spectrum are reliable multicast applications (e.g., distributions of documents, newsletters and online magazines) which require 100% reliability. End-to-end delay is not as critical to these applications as to interactive real-time applications. There exist also applications that fall in between these two extremes: their latency requirements are less stringent than those in real-time applications, and their reliability requirements are not as rigorous as that of reliable multicast applications.

The performance requirements of the application determines the multicast routing algorithm. For example, for the first category, SPT algorithms with path metrics that minimize end-to-end delay (or with bounded delay) are most suitable. For the second category[5], paths with high reliability (e.g., low traffic loads, low error rates) should be selected.

Routing Metrics

In addition to the multicast routing algorithms (e.g., SPT or MCT), the performance of a multicast group also depends on the chosen routing metric. For instance, in wireless networks, metrics such as ETT or ETX provide better performance than the hop count metric when used with SPT algorithms [75]. Therefore, routing metrics must be taken into account in the design of a multicast routing algorithm for a specific application.

The design objectives of such a metric include the following: loop-free routing, route stability, quick path establishment and high throughput. Furthermore, the routing metric should allow for efficient resource usage (minimal control and data overhead) and route maintenance. For example, a metric that requires global knowledge of network topology and conditions is not efficient in terms of storage usage and route maintenance.

Challenges

Designing multicast routing algorithms faces the following challenges:

- scalable and efficient distributed operations;
- handling dynamic membership, especially with MCTs;
- mobility management in wireless ad hoc networks;
- QoS routing, in both wired and bandwidth-limited wireless networks;
- network and device heterogeneity;
- inter-domain operations.

Algorithms designed for wireless multi-hop networks further face the following challenges:

- wireless channels with high error rates and low bandwidth;
- hidden and exposed terminal problems;
- location-dependent contention;
- inter-radio interference when multiple radios are available;
- inter- and intra-flow interference when multiple channels are used;
- energy-constrained devices;
- efficient and seamless interactions with the wireline Internet.

[5] In addition, a reliable multicast protocol is required for these applications to recover lost/damaged data.

9 Conclusion

We review existing multicast routing algorithms for different types of networks and application requirements. For the Internet, the major goals are distributed operations, heterogeneity, minimizing the amount of state stored in routers, minimizing delay, and QoS routing. The properties of wireless channels and transmissions and smaller scales of wireless multi-hop networks have shifted the focus to other issues. In particular, the focus in MANETs is mobility management and fault tolerance in order to cope with frequent link breaks. Energy-efficient operation is another objective in several protocols. WSNs mostly concern with broadcast and many-to-one communications (from nodes to sink). However, a few multicast protocols are proposed with minimization of energy consumption and/or maximizing session lifetime as goals. In WMNs, effective use of multiple channels and multiple radios and cross layer optimization are important to maximize throughput. Other goals include support for QoS routing and efficient and seamless interactions with the Internet.

Between the two most popular multicast routing approaches SPT and MCT, SPT is the more commonly deployed method because it is easy to implement and provide minimum end-to-end delay. However, MCT and its constrained problems are the more extensively researched problems, because they require efficient distributed implementations and dynamic membership handling in addition to more complex performance objectives (e.g., minimizing bandwidth usage or energy consumption).

Although multicast routing for the Internet has been studied extensively, integrating wired and wireless algorithms/protocols remains a challenge. Issues to be considered include efficient distributed operations across wired and wireless networks, especially when QoS is required (e.g., bounded end-to-end delay, guaranteed bandwidth), seamless interactions among protocols, effective mobility management to ensure continuous services in handover/handoff areas, and efficient handling of joins and leaves with minimal latency and sustained tree optimality.

More research is also required to exploit multiple channels, multiple radios and cross layer optimization for multicast in wireless ad hoc networks, especially wireless mesh networks, to maximize network throughput and enhance scalability. Topics to be studied include channel assignment in combination with multicast routing, effective methods for estimating wireless link quality, multicast routing in conjunction with medium access control and transport issues (e.g., flow/congestion control, error control) for maximized performance.

A Appendix: Routing Metrics

In this appendix we discuss in detail the link, node and path metrics listed in Section 3. A comprehensive survey of routing metrics can be found in [63]. We use the notations presented in Section 3.1.

A.1 Definitions of Link Costs

Following are detailed descriptions of the link metrics listed in Section 3.2.

A.1.1 Hop Count

Hop count is a metric that is widely used in both wired and wireless multi-hop networks because of its simplicity. The objective is, for each source-destination pair, to find a route that has the least number of links. The hop count metric is used in many routing protocols such as OLSR [114], DSR [115], DSDV [116], AODV [117].

A.1.2 Delay

Low end-to-end delay is a desirable property in many real-life applications. Moreover, bounded end-to-end delay is crucial in QoS routing for real-time applications such as tele-conferencing, video on demand, distributions of financial data and online games. End-to-end delay of a path is considered to be the sum of the delays of the links along the path.

In practice, people usually measure end-to-end delay in the form of *round trip time*, rather than taking the sum of link delays. Round trip time is the time taken by a packet to travel from the source to the destination and then back to the source.

Although less commonly used then round trip time, link delay can be measured using probe packets carrying timestamps and sent between the two neighbor routers.

A.1.3 Available Bandwidth/Capacity

Available bandwidth indicates the amount of data that can be transferred over a link over a given period of time. It is one of the most popular metrics for routing both in the Internet and wireless multi-hop networks. It is also a commonly used QoS routing parameter: many applications require a specific guarantee of bandwidth in order to operate successfully (e.g., video on-demand).

Bandwidth metrics are used for throughput guarantee, maximization of throughput and load balancing. Methods for measuring link bandwidth include the use of TCP throughput, probe packets in PATHCHAR algorithm [118] and packet pairs [119].

The cost of a path based on link bandwidth is usually determined by the most bottlenecked link on the path (the concave function given in Section 3.4.3). Nevertheless, we can compute path costs additively using link bandwidth as follows [90]. The cost of a path P is defined by

$$C_P = \sum_{\forall l \in P} \frac{1}{B_l},$$

where B_l is the bandwidth of every link l along path P. In this case, the path cost takes into account both the link bandwidths and the path length.

Roy et al. use the packet pair method to compute link bandwidths for multicast routing in WMNs [75].

A.1.4 Link Error Rate

This metric is intended for use in wireless networks [69], because wireless channels are much more error-prone than their wired counterparts. By selecting links with low error rates, we maximize throughput and minimize energy consumption. Retransmissions resulting from lost/damaged packets decrease the throughput and increase energy consumption.

The link error rate is modeled based on the modulation scheme used. For example, if binary phase shift keying (BPSK) is used, the error rate r_l of link l is given by the following formula:

$$r_l = 0.5 \times erfc(\sqrt{\frac{p_r}{p_n \times f}}),$$

where p_r is the received power level, p_n is the noise level, f is the transmission rate, and $erfc(x) = \int_x^\infty e^{-t^2} dt$ is the complementary error function. Link error rate formulas of several other modulation schemes can be found in [69].

Link error rates are used in the Success Probability Product (SPP) metric (Section A.3.1).

Link PDR	Link cost
$90\% - 100\%$	1
$79\% - 90\%$	3
$47\% - 79\%$	8
$0\% - 47\%$	28

Table 2: Converting link PDR to link cost [120]

A.1.5 Packet Delivery Ratio

The packet delivery ratio (PDR) of a link measures the link quality in terms of the number of packets correctly received by the receiver node divided by the number of packets transmitted by the sender node. Similarly to link error rate, the goal of the PDR metric is to maximize throughput, and/or minimize energy consumption by reducing the number of retransmissions.

The PDR of a path can be computed as the product of the PDRs of the links along the path (Section 3.4.2). However, one way to compute the cost of a path P using the link PDRs in practice is as follows [70]:

$$C_P = \sum_{\forall l \in P} \frac{1}{d_l},$$

where d_l denotes the PDR of link l. Yarvis et al. [120] convert link PDRs to link costs as shown in Table 2, and compute the path cost as the sum of the link costs.

A.1.6 Expected Transmission Count (ETX)

The ETX of a link predicts the number of required transmissions (including retransmissions) for sending a data packet over the link [70]. It is defined as

$$ETX = \frac{1}{d_f \times d_r},$$

where d_f and d_r are the PDRs of the links in the forward and backward directions, respectively. As such, the ETX metric assumes a medium access control protocol using acknowledgments such as IEEE 802.11 MAC with RTS/CTS/DATA/ACK exchange. If an ACK is lost, the corresponding DATA packet is considered lost and will be retransmitted. ETX thus needs to take into account the quality of the link in both directions.

The ETX cost of a path is the sum of the ETX values of the links along the path. The path cost thus considers both link quality and the path length.

ETX is used in the OLSR routing protocol [71]. Zhao et al. [76] compared ETX with the hop count metric for multicast routing in WMNs using ODMRP as the routing protocol. Also in the context of multicast routing in WMNs, Roy et al. [75] compared ETX with several other metrics such as hop count, Expected Transmission Time (section A.1.7), link bandwidth measured using the packet pairs method (Section A.1.3), Success Probability Product and multicast ETX (Section A.3.1).

A.1.7 Expected Transmission Time (ETT)

ETT [56] is defined based on the ETX metric:

$$ETT = ETX \times S/B,$$

where S is the packet size and B is the link bandwidth. The ETT metric thus considers both the error rate and the bandwidth of a link. The link bandwidth is measured using the packet pairs method (section A.1.3). The ETT cost of a path is the sum of the ETT values of the links on the path. ETT is used in the Link Quality Source Routing (LQSR) protocol [72].

A.1.8 Medium Time Metric (MTM)

The MTM of a wireless link l is given by the following formula [73]:

$$MTM_l = (S/R_l + H_l)/(1 - r_l)$$

In the above equation, S is the packet size. R_l is the optimal transmission rate for link l at the physical layer (i.e., the link rate), and calculated based on the signal strength of the packet at the receiver node. The weaker the signal strength, the worse the channel quality and thus the lower the transmission rate to be used. H_l is the MAC overhead, which include the back-off and contention time, and transmission time required by RTS, CTS and ACK messages. The link error rate r_l can be obtained from the physical layer using information such as loss rate history, signal level, noise level, and modulation/demodulation technique used. The MTM of link l is essentially the time required to transmit a packet over link l. The MTM of a path is the sum of the MTM values of the links on the path.

The MTM metric was proposed for multirate-aware routing in MANETs [73]. Zhao et al. defined a metric called PARMA [121], which is similar to MTM. Xing and Nguyen used a simplified form of the MTM for rate adaptive multicast routing in MANETs [77].

A.1.9 Signal Strength

Signal strength or signal-to-noise ratio (SNR) is an indication of link quality in wireless environments. It is also an indicator of the distance between the transmitter and the receiver, because the signal power weakens as the distance increases. The Signal Stability-Based Adaptive (SSA) routing protocol [122] uses signal strength as the routing metric.

A.1.10 Geographical Distance

Geographical distance of a link is the distance between the transmitter and the receiver. In wireless environments, the longer the distance, the worse the quality of the received signal. This metric is mostly used in WSNs where nodes are usually static and the location information is an important parameter. Geographical distance are often measured using GPS devices.

Seada et al. [123] considered the geographical distance metric and the packet delivery ratios in their routing protocol, while Zhang et al. [124] combined the geographical distance with the expected end-to-end delay.

A.2 Definitions of Node Costs

This section provides details of the node metrics listed in Section 3.3.

A.2.1 Node Count

This metric has been used only for multicast in wireless multi-hop networks [20, 76, 88, 89]. The goal of the multicast routing algorithm is to minimize the number of forwarding (relay) nodes in the routing tree (see Section 5.3). This metric exploits the wireless broadcast advantage to minimize the number of transmissions per packet, and thus minimize the bandwidth usage of the multicast tree.

A.2.2 Energy Consumption

Minimizing energy consumption is one of the most important routing objectives in energy-constrained networks such as MANETs and WSNs. The energy consumption of a node (typically a relay node) is defined as the energy required to receive then transmit a packet. The energy consumption of a path is the total energy consumed by all the nodes on the path to deliver a packet from the source to the destination.

The energy-consumption metric is used in many energy-efficient routing protocols such as BIP/MIP [48], RBIP [50], EWMA [21], REMiT [22] and many others for unicast routing [125, 126, 127, 128, 129].

A.2.3 Residual Battery Capacity

Many routing protocols designed for power-constrained networks take into account the amount of remaining battery power in wireless devices [130, 125, 131]. The purpose is to distribute routing workload to nodes that have higher residual battery capacity so as to prolong the lifetime of a path or the whole network.

Singh et al. [125] proposed to use $w_v = 1/\rho_v$ as node cost function, where ρ_v is the current (residual) battery capacity of node v. The path cost is a convex function:

$$C_P = \max_{\forall v \in P} w_v$$

This path cost considers the weakest node on a path. Given several possible paths between a source and a destination, the path with the least cost is selected.

Sheu at al. [130] and Gupta and Das [131] used the *ratio of battery remaining capacity* as the node cost, which is defined as $w_v = \rho_v/\varepsilon_v$, where ε_v is the battery full capacity of node v. The ratios of battery remaining capacity are used in the routing protocol by Gupta and Das [131] as follows: the protocol avoids nodes with less than 10% of their initial battery capacity, uses nodes with 10% to 20% of their initial battery capacity only when needed, and uses the other nodes indiscriminately.

A.2.4 Residual Lifetime

Another way to balance the energy consumption in the network is to look at the residual lifetime of nodes. The residual life time τ_v of a relay node v is defined as the number of packets that the node can receive and transmit before its battery drains:

$$\tau_v = \lfloor \frac{\rho_v}{e_v} \rfloor,$$

where ρ_v is the residual battery capacity of node v, and e_v is the energy v consumes to receive and transmit a packet.

Chang and Tassulias apply this metric to their Maximum Residual Energy Path (MREP) routing algorithm [126].

A.3 Path Costs Specific to Wireless Ad-hoc Networks

This section provides definitions of the path metrics listed in Section 3.4.4, namely Success Probability Product (SPP), Weighted Cumulative Expected Transmission Time (WCETT), Multi-Channel Routing Metric (MCR), and Metric of Interference and Channel-Switching (MIC).

A.3.1 Success Probability Product (SPP)

The SPP path metric [69] minimizes the energy consumption in multi-hop wireless networks. The SPP of a path P is defined as follows:

$$SPP_P = \frac{\sum_{\forall l \in P} e_l}{\prod_{\forall l \in P} 1 - r_l},$$

where e_l is the energy required to deliver a packet over link l and r_l is the link error rate, which accounts for the energy required for retransmissions of lost/damaged packets. The SPP of a path is the total energy required to successfully deliver a packet from the source to the destination.

Dong et al. [132] also use the energy consumption per packet e_l and link error rate r_l to compute source-to-destination path cost. Their path cost C_P is computed somewhat differently from SPP:

$$C_P = \sum_{\forall l \in P} \frac{e_l}{1 - r_l}$$

Roy et al. [75] compared the performance of a simplified form of Dong's path cost called "multicast ETX" with that of SPP (and a few other metrics) for multicast routing in WMNs.

A.3.2 Path Costs in Multi-channel Multi-radio Networks

These metrics are based on the result of a channel assignment algorithm such as those from [59, 105, 106, 107, 108]. Paths are selected using the path metric and the given channel assignment.

WCETT

WCETT is an extension of the ETT metric and used in the multi-radio link-quality source routing (MR-LQSR) protocol [56]. In addition to ETT, it tries to minimize intra-flow interference by penalizing paths that use one channel several times. The WCETT cost of a path P is given by:

$$WCETT_P = (1 - \beta) \sum_{\forall l \in P} ETT_l + \beta \max_{1 \leq j \leq k} X_j,$$

where $0 \leq \beta \leq 1$ is a tunable parameter, k is the total number of channels implemented in the network, and X_j is the number of times channel j is used along path P. By minimizing X_j, we can minimize the intra-flow interference on path P.

MCR

Kyasanur and Vaidya [60] extend the WCETT metric by adding the cost (delay) of switching channels to the path cost, as follows:

$$MCR_P = (1 - \beta) \sum_{\forall l \in P} (ETT_l + SC_l) + \beta \max_{1 \leq j \leq k} X_j$$

In the above equation, SC_l is the cost for switching channel to transfer a packet from link l to the next link when the two links are assigned two different channels. If link l and the next link use the same channel then $SC_l = 0$. Although channel diversity on a path is advantageous to its performance, the cost of switching channels must be considered as this cost is significant [60].

MIC

Yang et al. [63] also consider the channel switching cost in their MIC path metric. Moreover, they incorporate the effect of inter-flow interference into the metric. The MIC of a path P is defined as:

$$MIC_P = \alpha \sum_{\forall l \in P} IRU_l + \sum_{\forall v \in P} CSC_v$$

Variable α is a weighting factor. CSC_v is the cost incurred by a node v to switch from one channel to another. IRU_l is the interference-aware resource usage of link l, which is calculated from the ETT value of l and the number of nodes which may interfere with a packet transmission on l.

Acknowledgment

This work is supported in part by a Discovery Grant from NSERC (Natural Sciences and Engineering Research Council of Canada). We would like to thank Celia Li for her comments and suggestions to improve the article.

References

[1] S. Paul, *Multicast on the Internet and Its Applications.* Kluwer Academic Publishers, 1998.

[2] Z. Wang and J. Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications," *IEEE J. Sel. Areas Commun.*, vol. 14, pp. 1228–1234, Sep. 1996.

[3] E. W. Dijkstra, "A Note on Two Problems in Connexion with Graphs," *Numerical Mathematics*, vol. 1, pp. 269–271, Dec. 1959.

[4] R. Bellman, *Dynamic Programming.* Princeton University Press, 1957.

[5] C. Cheng, R. Riley, S. Kumar, and J. Garcia-Luna-Aceves, "A Loop-free Extended Bellman-Ford Routing Protocol Without Bouncing Effect," *ACM SIGCOMM Computer Communications Review*, vol. 19, pp. 224–236, Sep. 1989.

[6] M. S. Sloman and X. Andriopoulos, "Routing Algorithm for Interconnected Local Area Networks," *Computer Networks and ISDN Systems*, vol. 9, pp. 109–130, 1985.

[7] D. Waitzman, C. Partridge, and S. Deering, "Distance Vector Multicast Routing Protocol," RFC 1075, Nov. 1988.

[8] J. Moy, "Multicast Extensions to OSPF," RFC 1584, Mar. 1994.

[9] S. J. Lee, M. Gerla, and C. C. Chiang, "On-Demand Multicast Routing Protocol," in *Proc. IEEE WCNC'99*, 1999, pp. 1298–1302.

[10] E. M. Royer and C. E. Perkins, "Multicast Operation of the Ad-hoc On-demand Distance Vector Routing Protocol," in *Proc. ACM/IEEE MobiCom'99*, 1999, pp. 207–218.

[11] N. Shacham, "Multipoint Communication by Hierarchically Encoded Data," in *Proc. IEEE InfoCom'92*, 1992, pp. 2107–2114.

[12] R. C. Prim, "Shortest Connection Networks and Some Generalisations," *Bell System Technical Journal*, vol. 36, pp. 1389–1401, 1957.

[13] J. B. Kruskal, "On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem," in *Proc. American Mathematical Society*, 1956, pp. 48–50.

[14] R. G. Gallager, P. A. Humblet, and P. M. Spira, "A Distributed Algorithm for Minimum Spanning Tree," *ACM Transactions on Programming Languages and Systems*, vol. 5, pp. 66–77, 1983.

[15] L. Kou, G. Markowsky, and L. Berman, "A Fast Algorithm for Steiner Trees," *Acta Informatica*, vol. 2, pp. 141–145, Jun. 1981.

[16] H. Takahashi and A. Matsuyama, "An Approximate Solution for the Steiner Problem in Graphs," *Mathematica Japonica*, vol. 24, pp. 573–577, 1980.

[17] A. Zelikovsky, "An 11/6-approximation Algorithm for the Network Steiner Problem," *Algorithmica*, vol. 9, pp. 463–470, May 1993.

[18] P. Winter, "Steiner Problem in Networks: A Survey," *Networks*, vol. 17, pp. 129–167, 1987.

[19] P. Winter and J. M. Smith, "Path-Distance Heuristics for the Steiner Problem in Undirected Networks," *Algorithmica*, vol. 7, pp. 309–327, Dec. 1992.

[20] P. M. Ruiz and A. F. Gomez-Skarmeta, "Approximating Optimal Multicast Trees in Wireless Multihop Networks," in *Proc. IEEE ISCC'05*, 2005, pp. 686–691.

[21] M. Calgalj, J. P. Hubaux, and C. Enz, "Minimum-Energy Broadcast in All-Wireless Networks: NP-Completeness and Distribution Issues," in *Proc. ACM MobiCom'02*, 2002, pp. 172–182.

[22] B. Wang and S. K. S. Gupta, "S-REMiT: An Algorithm for Enhancing Energy-Efficiency of Multicast Trees in Wireless Ad Hoc Network," in *Proc. IEEE GLOBECOM'03*, 2003, pp. 3519–3524.

[23] ——, "G-REMiT: An Algorithm for Building Energy-Efficient Multicast Trees in Wireless Ad Hoc Networks," in *Proc. IEEE NCA'03*, 2003, pp. 265–272.

[24] ——, "On Maximizing Lifetime of Multicast Trees in Wireless Ad Hoc Networks," in *Proc. IEEE ICPP'03*, 2003, pp. 333–340.

[25] S. Deering and D. Cheriton, "Multicast Routing in Datagram Internetworks and Extended LANs," *ACM Transactions on Computer Systems*, vol. 8, pp. 85–110, May 1990.

[26] Q. Sun and H. Langendoerfer, "Efficient Multicast Routing for Delay-Sensitive Applications," in *Proc. PROMS'95*, 1995, pp. 452–458.

[27] H. F. Salama, D. S. Reeves, and Y. Viniotis, "The Delay-Constrained Minimum Spanning Tree Problem," in *Proc. IEEE ISCC'97*, 1997, pp. 699–703.

[28] C. Chow, "On Multicast Path Finding Algorithms," in *Proc. IEEE InfoCom'91*, 1991, pp. 1274–1283.

[29] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos, "Two Distributed Algorithms for the Constrained Steiner Tree Problem," in *Proc. IEEE ICCCN'93*, 1993, pp. 343–349.

[30] X. Jia, "A Distributed Algorithm of Delay-Bounded Multicast Routing for Multimedia Applications in Wide Area Networks," *IEEE/ACM Trans. Netw.*, vol. 6, pp. 828–837, Dec. 1998.

[31] G. Chen, M. Houle, and M. Kou, "The Steiner Problem in Distributed Computing Systems," *Information Science*, vol. 74, pp. 73–96, Oct. 1993.

[32] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos, "Multicast Routing for Multimedia Communication," *IEEE/ACM Trans. Netw.*, vol. 1, pp. 286–292, Jun. 1993.

[33] R. Sriram, G. Manimaran, and C. S. R. Murthy, "Algorithms for Delay-Constrained Low-Cost Multicast Tree Construction," *Computer Communications*, vol. 21, pp. 1693–1706, Dec. 1998.

[34] G. Feng and T. P. Yum, "Efficient Multicast Routing with Delay Constraints," *International Journal of Communication Systems*, vol. 12, pp. 181–195, Jan. 1999.

[35] G. Kumar, N. Narang, and C. Ravikumar, "Efficient Algorithms for Delay Bounded Multicast Tree Generation for Multimedia Applications," in *Proc. HiPC'99*, 1999, pp. 169–173.

[36] Q. Zhu, M. Parsa, and J. J. Garcia-Luna-Aceves, "A Source-Based Algorithm for Delay-Constrained Minimum-Cost Multicasting," in *Proc. IEEE InfoCom'95*, 1995, pp. 377–385.

[37] A. Shaikh and K. G. Shin, "Destination-Driven Routing for Low-Cost Multicast," *IEEE J. Sel. Areas Commun.*, vol. 15, pp. 373–381, Apr. 1997.

[38] B. K. Haberman and G. Rouskas, "Cost, Delay, and Delay Variation Conscious Multicast Routing," NC State Univ., Tech. Rep., 1997.

[39] R. N. Rouskas and I. Baldine, "Multicast Routing with End-to-End Delay and Delay Variation Constraints," *IEEE J. Sel. Areas Commun.*, vol. 15, pp. 346–356, Apr. 1997.

[40] X. Jiang, "Routing Broadband Multicast Streams," *Computer Communications*, vol. 15, pp. 45–51, Jan. 1992.

[41] F. Bauer and A. Verma, "Degree-Constrained Multicasting in Point-to-Point Networks," in *Proc. IEEE InfoCom'95*, 1995, pp. 369–369.

[42] F. Bauer, "Multicast Routing in Point-to-Point Networks Under Constraints," Ph.D. dissertation, University of California, Santa Cruz, 1996.

[43] B. Kadaba and J. M. Jaffe, "Routing to Multiple Destinations in Computer Networks," *IEEE J. Sel. Areas Commun.*, vol. 31, pp. 343–351, Mar. 1983.

[44] S. J. Chung, S. P. Hong, and H. S. Huh, "A Fast Multicast Routing Algorithm for Delay-Sensitive Applications," in *Proc. IEEE GLOBECOM'97*, 1997, pp. 1898–1902.

[45] H. Fujinoki and K. J. Christensen, "The New Shortest Best Path Tree (SBPT) Algorithm for Dynamic Multicast Trees," in *Proc. IEEE LCN'99*, 1999, pp. 204–204.

[46] G. W. Zeng, B. Ding, Y. L. Xiao, and M. Mutka, "Multicast Algorithms for Multi-Channel Wireless Mesh Networks," in *Proc. IEEE ICNP'07*, 2007, pp. 1–10.

[47] X. Zhao, C. T. Chou, J. Guo, and S. Jha, "Protecting Multicast Sessions in Wireless Mesh Networks," in *Proc. IEEE LCN'06*, 2006, pp. 467–474.

[48] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks," in *Proc. IEEE InfoCom'00*, 2000, pp. 585–594.

[49] ——, "Distributed Algorithms for Energy-Efficient Broadcasting in Ad Hoc Networks," in *Proc. IEEE MILCOM'02*, 2002, pp. 820–825.

[50] S. Banerjee, A. Misra, J. Yeo, and A. Agrawala, "Energy-Efficient Broadcast and Multicast Trees for Reliable Wireless Communication," in *Proc. IEEE WCNC'03*, 2003, pp. 660–667.

[51] Y. Xiao, X. Du, J. Zhang, F. Hu, and S. Guizani, "Internet Protocol Television (IPTV): The Killer Application for the Next-Generation Internet," *IEEE Commun. Mag.*, vol. 45, pp. 388–404, Nov. 2007.

[52] I. Akyildiz, X. Wang, and W. Wang, "Wireless Mesh Networks: A Survey," *Elsevier Computer Networks*, vol. 47, pp. 445–487, Mar. 2005.

[53] Y. Zhang, J. Luo, and H. H. (editors), *Wireless Mesh Networking.* Auerbach Publications, 2007.

[54] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE Trans. Inf. Theory*, vol. 46, pp. 388–404, Mar. 2000.

[55] J. Li, C. Blake, D. S. J. D. Couto, H. I. Lee, and R. Morris, "Capacity of Ad Hoc Wireless Networks," in *Proc. ACM International Conference on Mobile Computing and Networking (MobiCom'01)*, 2001, pp. 61–69.

[56] R. Draves, J. Padhye, and B. Zill, "Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," in *Proc. ACM MobiCom'04*, 2004, pp. 114–128.

[57] A. Raniwala, K. Gopalan, and T. Chiueh, "Centralized Channel Assignment and Routing Algorithms for Multi-Channel Wireless Mesh Networks," in *Proc. ACM SIGMOBILE'04*, 2004, pp. 50–65.

[58] M. Alicherry, R. Bhatia, and L. Li, "Joint Channel Assignment and Routing for Throughput Optimization in Multi-Radio Wireless Mesh Networks," in *Proc. ACM SIGMOBILE'05*, 2005, pp. 58–72.

[59] J. Tang, G. Xue, and W. Zhang, "Interference-aware Topology Control and QoS Routing in Multi-Channel Wireless Mesh Networks," in *Proc. ACM SIGMOBILE'05*, 2005, pp. 68–77.

[60] P. Kyasanur and N. H. Vaidya, "Routing and Link-layer Protocols for Multi-Channel Multi-Interface Ad Hoc Wireless Networks," in *Proc. ACM SIGMOBILE'06*, 2006, pp. 31–43.

[61] A. Raniwala and T. Chiueh, "Architecture and Algorithms for an IEEE 802.11-based Multi-Channel Wireless Mesh Networks," in *Proc. IEEE InfoCom'05*, 2005, pp. 2223–2234.

[62] A. K. Das, H. M. K. Alazemi, R. Vijayakumar, and S. Roy, "Optimization Models for Fixed Channel Assignment in Wireless Mesh Networks with Multiple Radios," in *Proc. IEEE SECON'05*, 2005, pp. 463–474.

[63] R. Baumann, S. Heimlicher, M. Strasser, and A. Weibel, "A Survey on Routing Metrics," Computer Engineering and Networks Labolatory, ETH-Zentrum, Tech. Rep. TIK Report 262, Feb. 2007.

[64] J. J. Garcia-Luna-Aceves and E. L. Madruga, "The Core-Assisted Mesh Protocol," *IEEE J. Sel. Areas Commun.*, vol. 17, pp. 1380–1394, Aug. 1999.

[65] V. Devarapalli, A. A. Selcuk, and D. Sidhu. (2001) MZR: A Multicast Protocol for Mobile Ad Hoc Networks. Internet Draft. [Online]. Available: http://tools.ietf.org/draft/draft-vijay-manet-mzr/draft-vijay-manet-mzr-01.txt

[66] J. G. Jetcheva and D. B. Johnson. (2001) Adaptive Demand-Driven Multicast Routing Protocol (ADMR). Internet Draft. [Online]. Available: http://www3.tools.ietf.org/html/draft-jetcheva-manet-admr-00

[67] T. Ozaki, J. Kim, and T. Suda, "Bandwidth Efficient Multicast Routing Protocol for Ad Hoc Networks," in *Proc. IEEE ICCCN'99*, 1999, p. 1017.

[68] C. C. Chiang, M. Gerla, and L. Zhang, "Forwarding Group Multicasting Protocol for Multi-Hop, Mobile Wireless Networks," *ACM/Baltzer Journal of Cluster Computing: Special Issue on Mobile Computing*, vol. 1, pp. 187–196, Jun. 1998.

[69] S. Banerjee and A. Misra, "Minimum Energy Paths for Reliable Communication in Multi-Hop Wireless Networks," in *Proc. ACM MobiHoc'02*, 2002, pp. 146–156.

[70] D. Couto, D. Aguayo, J. C. Bicket, and R. Morris, "A High-Throughput Path Metric for Multi-Hop Wireless Routing," in *Proc. ACM MobiCom'03*, 2003, pp. 134–146.

[71] T. Lopatic. (2004) OLSRD Link Quality Extensions. [Online]. Available: http://www.olsr.org/docs/README-Link-Quality.html

[72] R. Draves, J. Padhye, and B. Zill, "The Architecture of the Link Quality Source Routing Protocol," Microsoft Research, Tech. Rep., 2004.

[73] B. Awerbuch, D. Holmer, and H. Rubbens, "High Throughput Route Selection in Multi-Rate Ad Hoc Wireless Networks," Johns Hopkins University, Tech. Rep., Jul. 2003.

[74] R. Guerin, A. Orda, and D. Williams. (1996) QoS Routing Mechanisms and OSPF Extensions. Internet Draft. [Online]. Available: http://tools.ietf.org/html/draft-guerin-qos-routing-ospf-00

[75] S. Roy, D. Koutsonokolas, S. Das, and Y. C. Hu, "High-Throughput Multicast Routing Metrics in Wireless Mesh Networks," in *Proc. IEEE ICDCS'06*, 2006, pp. 48–48.

[76] X. Zhao, C. T. Chou, J. Guo, and S. Jha, "A Scheme for Probabilistically Reliable Multicast Routing in Wireless Mesh Networks," in *Proc. IEEE LCN'07*, 2007, pp. 213–214.

[77] X. Xiong and U. T. Nguyen, "Rate-Adaptive Multicast in Mobile Ad-Hoc Networks," in *Proc. IEEE WiMob'05*, 2005, pp. 352–360.

[78] A. E. F. Clementi, P. Crescenzi, P. Penna, G. Rossi, and P. Vocca, "On the Complexity of Computing Minimum Energy Consumption Broadcast Subgraphs," in *Proc. STACS'01*, 2001, pp. 121–131.

[79] F. Li and I. Nikolaidis, "On Minimum-Energy Broadcasting in All-Wireless Networks," in *Proc. IEEE LCN'01*, 2001, pp. 193–193.

[80] S. Singh, C. S. Raghavendra, and J. Stepanek, "Power-Aware Broadcasting in Mobile Ad Hoc Networks," in *Proc. IEEE PIMRC'99*, 1999, pp. 22–31.

[81] M. Doar and I. Leslie, "How Bad is Naive Multicast Routing," in *Proc. IEEE InfoCom'93*, 1993, pp. 82–89.

[82] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: Freeman, 1979.

[83] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. The MIT Press & McGraw Hill, 2001.

[84] H. F. Salama, D. S. Reeves, and Y. Viniotis, "Evaluation of Multicast Routing Algorithms for Real-Time Communication on High-Speed Networks," *IEEE J. Sel. Areas Commun.*, vol. 15, pp. 332–345, Apr. 1997.

[85] C. A. S. Oliveira and P. M. Pardalos, "A Survey of Combinatorial Optimization Problems in Multicast Routing," *Elsevier Computers and Operations Research*, vol. 32, pp. 1953–1981, Aug. 2005.

[86] B. Wang and J. Hou, "Multicast Routing and Its QoS Extension: Problems, Algorithms and Protocols," *IEEE Netw.*, vol. 14, pp. 22–36, Jan. 2000.

[87] W. D. Zhong, Y. Onozato, and J. Kaniyil, "A Copy Network with Shared Buffers for Large-Scale Multicast ATM Switching," *IEEE/ACM Trans. Netw.*, vol. 1, pp. 157–165, Apr. 1993.

[88] U. Nguyen and J. Xu, "Multicast Routing in Wireless Mesh Networks: Minimum Cost Trees or Shortest Path Trees?" *IEEE Commun. Mag.*, vol. 45, pp. 72–77, Nov. 2007.

[89] U. T. Nguyen, "On Multicast Routing in Wireless Mesh Networks," *Elsevier Computer Communications*, vol. 31, pp. 1385–1399, May 2008.

[90] S. Nelakuditi and Z.-L. Zhang, *Localized Quality of Service Routing for the Internet*. Kluwer Academic Publishers, 2003.

[91] C. Kheong, D. Siew, and G. Feng, "Efficient Setup for Multicast Connections Using Tree-Caching," in *Proc. IEEE InfoCom'01*, 2001, pp. 249–258.

[92] P. Narvaez, K. Siu, and H. Tzeng, "New Dynamic SPT Algorithm Based on a Ball-and-String Model," in *Proc. IEEE InfoCom'99*, 1999, pp. 973–981.

[93] F. Bauer and A. Varma, "ARIES: A Rearrangeable Inexpensive Edge-Based On-line Steiner Algorithm," *IEEE J. Sel. Areas Commun.*, vol. 1, pp. 382–397, Apr. 1997.

[94] M. Imase and B. Waxman, "Dynamic Steiner Tree Problem," *SIAM J. Disc. Math*, vol. 4, pp. 369–384, 1991.

[95] F. Bauer and A. Varma, "Distributed Algorithms for Multicast Path Setup in Data Networks," University of California Santa Cruz, Tech. Rep., 1995.

[96] D. Wall, "Mechanisms for Broadcast and Selective Broadcast," Ph.D. dissertation, Stanford University, 1980.

[97] T. Ballardie, P. Francis, and J. Crowcroft, "Core Bared Trees (CBT): An Architecture for Scalable Multicast Routing," in *Proc. ACM SIGCOMM'95*, 1995, pp. 85–95.

[98] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. G. Liu, and L. Wei, "The PIM Architecture for Wide-Area Multicast Routing," *IEEE/ACM Trans. Netw.*, vol. 4, pp. 153–162, Apr. 1996.

[99] ——, "An Architecture for Wide-Area Multicast Routing," *ACM SIGCOMM Computer Communications Review*, vol. 24, pp. 126–135, Oct. 1994.

[100] K. L. Calvert, E. W. Zegura, and M. J. Donahoo, "Core Selection Methods for Multicast Routing," in *Proc. IEEE ICCCN'95*, 1995, pp. 638–642.

[101] L. Wei and D. Estrin, "The Trade-offs of Multicast Trees and Algorithms," in *Proc. IEEE ICCCN'94*, 1994.

[102] L. Lannone and S. Fdida, "MRS: A Simple Cross-layer Heuristic to Improve Throughput Capacity in Wireless Mesh Networks," in *Proc. ACM CoNEXT'05*, 2005, pp. 21–30.

[103] V. Kawadia and P. R. Kumar, "A Cautionary Perspective on Cross Layer Design," *IEEE Wireless Commun. Mag.*, vol. 12, pp. 3–11, Feb. 2005.

[104] A. Mohsenian and V. Wong, "Joint Optimal Channel Assignment and Congestion Control for Multi-Channel Wireless Mesh Networks," in *Proc. IEEE ICC'06*, 2006, pp. 1984–1989.

[105] M. Marina and S. Das, "A topology Control Approach for Utilizing Multiple Channels in Multi-Radio Wireless Mesh Networks," in *Proc. IEEE International Conference on Broadband Networks (Broadnets'05)*, 2005, pp. 381–390.

[106] A. P. Subramanian, H. Gupta, and S. R. Das, "Minimum Interference Channel Assignment in Multi-Radio Wireless Mesh Networks," in *Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON'07)*, 2007, pp. 481–490.

[107] F. Tasaki, H. Tamura, M. Sengoku, and S. Shinoda, "A New Channel Assignment Strategy Towards the Wireless Mesh Networks," in *Proc. APCC'04*, 2004, pp. 71–75.

[108] K. Ramachandran, E. Belding, K. Almeroth, and M. Buddhikot, "Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks," in *Proc. IEEE InfoCom'06*, 2006, pp. 1–12.

[109] H. Wu, F. Yang, K. Tan, J. Chen, Q. Zhang, and Z. Zhang, "Distributed Channel Assignment and Routing in Multi-Radio Multi-Channel Multi-Hop Wireless Networks," *IEEE J. Sel. Areas Commun.*, vol. 24, pp. 1972–1983, Nov. 2006.

[110] H. L. Nguyen and U. T. Nguyen, "Minimum Interference Channel Assignment for Multicast in Multi-Radio Wireless Mesh Networks," in *Proc. IEEE IWCMC'08*, 2008.

[111] Z. Yin, Z. Li, and M. Chen, "A Novel Channel Assignment Algorithm for Multicast in Multi-radio Wireless Mesh Networks," in *Proc. IEEE ISCC'07*, 2007, pp. 283–288.

[112] Y. Yang, J. Wang, and R. Kravets, "Designing Routing Metrics for Mesh Networks," in *Proc. IEEE WiMesh'05*, 2005.

[113] Y. Yang, J. Wang, , and R. Kravets, "Interference-aware Load Balancing for Multihop Wireless Networks," University of Illinois at Urbana-Champaign, Tech. Rep. UIUCDCS-R-2005-2526, 2005.

[114] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," RFC 3626, Oct. 2003.

[115] D. B. Johnson, D. A. Maltz, and J. Broch, *Ad Hoc Networking.* Addison-Wesley, 2001, ch. 5, pp. 139–172.

[116] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," in *Proc. ACM SIGCOMM'94*, 1994, pp. 234–244.

[117] C. E. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561, Jul. 2003.

[118] A. B. Downey, "Using **pathchar** to Estimate Internet Link Characteristics," in *Proc. ACM SIGCOMM'99*, 1999, pp. 241–250.

[119] S. Keshav, "A Control-Theoretic Approach to Flow Control," in *Proc. ACM SIGCOMM'95*, 1995, pp. 188–201.

[120] M. Yarvis, W. Conner, L. Krishnamurthy, J. Chahabra, B. Elliott, and A. Mainwaring, "Real-World Experiences with an Interactive Ad Hoc Sensor Network," in *Proc. International Workshop on Ad Hoc Networking*, 2002, pp. 143–151.

[121] S. Zhao, Z. Wu, A. Acharya, and D. Raychaudhuri, "PARMA: A PHY/MAC Aware Routing Metric for Ad-Hoc Wireless Networks with Multi-Rate Radios," IBM Research Division, Tech. Rep., 2004.

[122] R. Dube, C. Rais, K. Wang, and S. Tripathi, "Signal Stability Based Adaptive Routing (SSA) for Ad Hoc Mobile Networks," in *Proc. IEEE Personal Communication*, 1997, pp. 36–45.

[123] K. Seada, M. Zuniga, A. Helmy, and B. Krishnamachari, "Energy-Efficient Forwarding Strategies for Geographical Routing in Lossy Wireless Sensor Networks," in *Proc. SenSys'04*, 2004, pp. 108–121.

[124] H. Zhang, A. Arora, and P. Sinha, "Learn on the Fly: Data-Driven Link Estimation and Routing in Sensor Network Backbones," in *Proc. IEEE InfoCom'06*, 2006, pp. 1–12.

[125] S. Singh, M. Woo, and C. Raghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks," in *Proc. ACM MobiCom'98*, 1998, pp. 181–190.

[126] J. Chang and L. Tassiulas, "Maximum Lifetime Routing in Wireless Sensor Networks," in *Proc. ATIRP'00*, 2000.

[127] I. Stojmenovic and X. Lin, "Power-Aware Localized Routing in Wireless Networks," in *Proc. IEEE IPDPS'00*, 2000, pp. 371–371.

[128] R. Shah and J. Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks," in *Proc. IEEE WCNC'02*, 2002, pp. 350–355.

[129] A. Michail and A. Ephremides, " Energy-Efficient Routing for Connection-Oriented Traffic in Wireless Ad-Hoc Networks," *Mobile Networks and Applications*, vol. 8, pp. 517–533, Oct. 2003.

[130] J. P. Sheu, C. T. Hu, and C. M. Chao, *The Handbook of Ad Hoc Wireless Networks.* CRC Press, Inc, 2003, pp. 399–413.

[131] N. Gupta and S. R. Das, "Energy-Aware On-demand Routing for Mobile Ad Hoc Networks," in *Proc. IWDC'02*, 2002, pp. 164–173.

[132] Q. Dong, S. Banerjee, M. Adler, and A. Misra, "Minimum Energy Reliable Paths Using Unreliable Wireless Links," in *Proc. ACM MobiHoc'05*, 2005, pp. 449–459.