Multicast Algorithms for Multi-Channel Wireless Mesh Networks

Guokai Zeng, Bo Wang, Yong Ding, Li Xiao, Matt Mutka
Department of Computer Science and Engineering
Michigan State University
Email: {zengguok, wangbo1, dingyong, lxiao, mutka}@cse.msu.edu

Abstract— Multicast is a key technology that provides efficient data communication among a set of nodes for wireless multi-hop networks. In sensor networks and MANETs, multicast algorithms are designed to be energy efficient and to achieve optimal route discovery among mobile nodes, respectively. However, in wireless mesh networks, which are required to provide high quality service to end users as the "last-mile" of the Internet, throughput maximization conflicting with scarce bandwidth has the paramount priority. We propose a Level Channel Assignment (LCA) algorithm and a Multi-Channel Multicast (MCM) algorithm to optimize throughput for multi-channel and multi-interface mesh networks. The algorithms first build a multicast structure by minimizing the number of relay nodes and hop count distances between the source and destinations, and use dedicated channel assignment strategies to improve the network capacity by reducing interference. We also illustrate that the use of partially overlapping channels can further improve the throughput. Simulations show that our algorithms greatly out-perform the single-channel multicast algorithm. We observe that MCM achieves better throughput and shorter delay while LCA can be realized in distributed manner.

keywords: Wireless Mesh Networks, Multicast, Multi-Interface, Channel Assignment.

I. INTRODUCTION

The Wireless Mesh Network (WMN) is an emerging paradigm for the next-generation wireless Internet. In such networks, most of the nodes are either stationary or minimally mobile and do not have power constraints. Compared with their single-hop counterpart, wireless LANs, the WMNs are self-organized with the nodes automatically establishing ad hoc networks and maintaining their connectivity, which provides more reliability as well as larger coverage, and reduces equipment cost. Being used on the last mile for extending or enhancing Internet connectivity, commercial deployments of WMNs are already in the works, such as MIT Roofnet [8] and Seattle Wireless [10].

Mesh networks are characterized by the use of multiple channels and multiple interfaces to improve the system throughput. Recent research has focused on how to assign channels to different wireless interfaces in unicast routing to improve system throughput in WMNs. However, the multicast communication, which intends to transmit the packets from the source to a set of nodes, draws less attention in the literature of mesh networks. We believe that efficient multicast, which cannot be readily achieved through combined unicast

or simplified broadcast, is essential to wireless mesh networks and is worthy of thorough investigation. It is often necessary for a portion of end users to retrieve data packets from the Internet. For example, a large number of users may like to watch the FIFA World Cup or some other field pickup on the Internet. The gateway that helps to connect the mesh network with the Internet can effectively multicast the data packets to those users.

Efficient multicast protocols in WMNs cannot be achieved by adopting or slightly modifying the multicast protocols for other types of multi-hop wireless networks. Unlike mobile ad hoc networks or wireless sensor networks, route recovery or energy efficiency is not the major concern for mesh networks due to the limited mobility and the rechargeable characteristic of mesh nodes. Moreover, supporting the potential major applications such as Video On Demand poses a big challenge for the limited bandwidth of WMNs. Thus, it is necessary to design an effective multicast algorithm for mesh networks.

Traditional multicast protocols for wireless networks assume that each node is equipped with one interface, while the mesh networks provide the nodes with multiple interfaces that can be used to improve the throughput substantially. However, channel assignment is subject to the number of available channels and interfaces, the network topology, the communication requests, and other factors. Especially, the interference cannot be completely eliminated due to the limited number of available channels. Inappropriate channel assignment strategy will result in throughput reduction due to the multi-channel hidden terminal problem [26], disconnection of the topology [20], or unfair bandwidth allocation to various users [27].

In this paper, we aim to design a multicast protocol for mesh networks that has the following characteristics: i) it improves the system throughput by allowing simultaneous close-by transmissions with multi-channel and multi-interface, and ii) it assigns all the available channels to the interfaces instead of just the non-overlapping channels.

We propose a Level Channel Assignment (LCA) algorithm and a Multi-Channel Multicast (MCM) algorithm to optimize throughput for multi-channel and multi-interface mesh networks. The algorithms first build a multicast structure by minimizing the number of relay nodes and hop count distances between the source and destinations, and use dedicated channel

assignment strategies to improve the network capacity by reducing the interferences.

Our design builds a new multicast backbone - "tree mesh", which partitions the mesh routers into different levels based on the Breadth First Search (BFS), and then heuristically assigns channels to different interfaces. Tree-based multicast is well established in wireless networks for its data forwarding efficiency over other types of approaches at the expense of low robustness. However, unlike MANETs, WMNs are normally considered stationary and always put the throughput maximization as the first priority. Thus, tree-based multicast is suitable for WMNs since the topology change is not a major concern in WMNs.

We also illustrate that the use of partially overlapping channels can further improve the throughput. Simulations show that our algorithms greatly outperform the single-channel multicast algorithm. We observe that MCM achieves better throughput and shorter delay while LCA can be implemented in a distributed manner.

The rest of this paper is organized as follows. Section II describes the system model and the design consideration. Section III proposes an intuitive algorithm, the LCA algorithm, which is easy to implement but has drawbacks. Section IV introduces the MCM algorithm to build a more efficient multicast structure, which is followed by the description of how to assign channels on it. Several companion mechanisms for our protocol are presented in Section V. Section VI presents simulation results. Section VII surveys the related work, and the last section concludes this paper.

II. SYSTEM MODEL

We start from the underlying network model by introducing some basic terminology and the partial channel conflict phenomena, which is followed by the design considerations for the multicast algorithms in WMNs.

A. Basics

Mesh networks are composed of three types of nodes: gateways (access points), mesh routers, and mesh clients. Gateways enable the integration of WMNs with various other networks, including the Internet. As dedicated devices providing stable high throughput for mesh clients, the mesh routers have minimal mobility and form the mesh backbone. In order to further improve the flexibility and capacity of WMNs, the mesh routers are often equipped with multiple wireless interfaces. As a result, two transmissions of two nearby pairs can be simultaneously scheduled if non-overlapping channels are assigned. Mesh clients are usually end users, such as laptops and PDAs, which access the Internet through the mesh routers, so that the mesh clients are usually within one-hop of the mesh routers. Since the multicast packets are always relayed among the mesh backbone, we only consider how to transmit the packets to multiple mesh routers; then packets will be forwarded one more hop to the corresponding mesh clients that desire to receive the packets.

To simplify the system model, we consider the network as a graph $G = (V, E)$, where V represents the set of gateways and mesh routers, and E represents the physical links among neighboring nodes (the node refers to the mesh router or the gateway in the subsequent sections). We assume that each node has the same fixed communication range, that is, if node u can transmit directly to node v (and vice versa), there is a link (u, v) in E.

The number of available channels is limited in the current network protocols. In addition, each node is able to be equipped with k ($k \geq 2$) Network Interface Cards (NICs), any of which can be tuned to any available channel. Multi-channel and multi-interface characteristics enable more concurrent transmissions. When one NIC is transmitting or receiving packets on one channel, another NIC on the same node is able to undertake transmission on another different channel at the same time. The value of k usually equals to 2, 3, or 4 due to economical reasons. In this paper, we only consider the situation that each node has 2 interfaces.

B. Measuring Partial Overlap

To improve the throughput of WMNs, many studies have been conducted on how to assign orthogonal channels to adjacent wireless links to minimize interference. It is known that $802.11b/g$ and $802.11a$ provide 3 and 12 non-overlapping channels, respectively. Although $802.11a$ provides more non-overlapping channels than $802.11b/g$, it has several drawbacks. Because $802.11a$ works on a higher frequency spectrum (5GHz) than $802.11b/g$ (2GHz), it is more difficult to penetrate walls and other obstructions, and thus $802.11a$ has a shorter range. In addition, 5GHz belongs to the regulated frequency spectrum, which makes $802.11a$ more expensive. As a result, $802.11b/g$ is more commonly used.

Previous channel assignment algorithms for $802.11b/g$ only use three non-overlapping channels: 1, 6, and 11. In these studies, a binary interference model is usually assumed, that is, if two links are within interference range of each other, they will interfere with each other if they are on the same channel, and otherwise not. However, the interference can be further reduced by using the partially overlapping channels too, that is, by using any channel from 1 to 11 in the channel assignment.

Through experiments, we observe that the interference between two links depends on both their physical distance and channel separation [17]. Unlike the traditional interference model, the interference range is no longer a constant. Instead, it varies with the channel separation. Let I_c be the interference range of two links with channel separation c. That means, when the channel separation of two links is c, they will interfere with each other if their distance is less than I_c, and otherwise not. For example, $I_0 = 2R$, which means the same
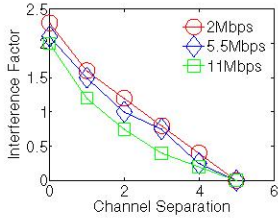
Fig. 1. Interference Factor vs Channel Separation

(a) Network Topology   (b) Multicast Tree   (c) Channel Assignment   (d) Tree Mesh

Fig. 2. An Example for LCA and Tree Mesh

channel can be used on two links without any interference only when they are over twice the transmission range away. In [17][18], experiments have been done to measure the interference between two wireless AP-Client links with different distances and channel separations.

Definition 1: Interference Factor is defined as the radio of the interference range by the transmission range. We use δ_t to represent the Interference Factor when the channel separation of two links is t.

We performed similar experiments to measure the interference factor between two wireless peer-to-peer links. We used 4 laptops with Netgear WAG511 PC Cards, each two of which construct a separate wireless link. We evaluated the interference between two links by comparing the total throughput when both links are active and the sum of each link's throughput when the other link is turned down. We found a similar trend in our experiments, that is, interference range decreases with the increase of channel separation. The results of these experiments are shown in Fig. 1. Therefore, if we can fully utilize these partially overlapping channels, we can further decrease the total interference in the network, and thus improve the network throughput.

C. Design Consideration

Routing protocols do exist to offer efficient multicasting service for conventional multi-hop wireless networks, such as MANETs and wireless sensor networks. Since the nodes become increasingly mobile or the networks only have scarce resources such as power constraints and limited computing ability, most previous work pays much attention to energy efficiency and how to build the multicast structure without knowing the global topology. As a result, the multicast structure should be distributedly constructed, energy efficient, and should take care of the topology change as well as group member management, which may conflict with maximizing the throughput of the network to some extent.

However, since mesh networks are deployed to provide last-mile Internet access for enterprises or communities, the throughput and the network capacity are the major concerns. Deployed at the fixed locations, mesh routers have limited mobility. Furthermore, they are computationally powerful and do not rely on battery power compared with their counterparts in
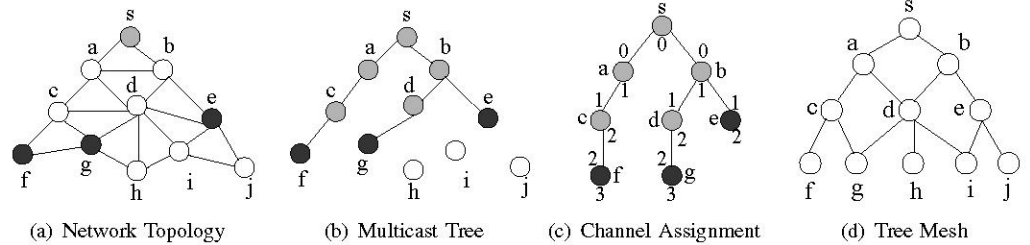
MANETs or sensor networks, which helps to achieve sufficient network capacity to meet the requirement of applications such as audio or video sharing among end users. Thus, we need to create a multicast structure that aims to deliver the packets rapidly to the multi-receivers (multi-receivers are defined as the multicast group members except for the source node) without worrying about the energy consumption and topology changes.

Moreover, equipping the mesh routers with more than one wireless interface could further improve the network capacity. The assignment of channels to interfaces on the multicast structure is also essential to throughput optimization. Inappropriate channel allocation will lead to topology disconnection and exacerbation of multi-channel hidden terminal problems, which reduces the system throughput. Therefore, both efficient multicast structure and effective channel assignment play important roles in mesh network multicast.

III. LEVEL CHANNEL ASSIGNMENT ALGORITHM

A common method for multicast is to build a multicast tree, where the source node is usually the access point in this paper. We first propose the Level Channel Assignment (LCA) algorithm, which can be achieved by the following steps.

First, the nodes obtain their level information. The breadth first search (BFS) is used to traverse the whole network. Based on the BFS, all the nodes are partitioned into different levels according to the hop count distances between the source and the nodes.

Definition 2: if node a (in level i) and b (in level $i+1$) are within each other's communication range, then a is called the **parent** of b, and b is called the **child** of a.

Second, we build a multicast tree based on the node level information. Initially, the source and all the receivers are included in the tree. Then, for each multi-receiver v, if one of its parents is a tree node, then connect it with that parent, and stop. Otherwise, randomly choose one of its parents, say f_v, as a relay node on the tree, and connect v and f_v. Afterwards, we try to find out the relay node for f_v recursively. This process repeats until all the multi-receivers are included in the multicast tree. Algorithm 1 gives the detail.

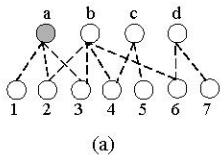Next, the tree nodes decide their channel assignment with

Fig. 3. Relay Node Search Example

Fig. 4. A Multicast Structure Example

Data: M: multi-receivers; s: source node;
Result: T: multicast tree
$V(T) = M \cup \{s\}$; $E(T) = \emptyset$;
for $\forall\ node\ v \in M$ **do**
    $p = v$;
    while *none of p's parents is included in* $V(T)$ **do**
        Randomly select one of p's parents, say f_p.
        $V(T) = V(T) \cup \{f_p\}$;
        $E(T) = E(T) \cup \{(p, f_p)\}$;
        $p = f_p$
    end
    $E(T) = E(T) \cup \{(p, f'_p)\}$ (f'_p is the parent of p, and it is a tree node)
end

Algorithm 1: Multicast Tree Construction for LCA

the level information.

1) The source node (level 0) only uses one interface, which is assigned channel 0. This interface is responsible for sending packets to the tree nodes in level 1.
2) The internal tree node in level i ($i \geq 1$) uses two interfaces: one is assigned channel $i - 1$, which is used to receive packets from the upper level, the other is assigned channel i, which is used to forward packets to tree nodes at level $i + 1$
3) The leaf in the level i ($i \geq 1$) uses two interfaces: one uses channel $i - 1$ to receive the packets from level $i - 1$, the other uses channel i to forward the packets to the mesh clients within its communication range who desire to receive the packets.

One example is shown in Fig. 2, where node s is the source, and e, f, g are the multi-receivers. Initially, $\{s,e,f,g\}$ are included in the multicast tree. At first, since none of g's parents are tree nodes, randomly select one parent d as a tree node and connect g with d. We then choose d's parent b as a tree node and connect d with b. Since b's parent s is a tree node, we connect b with s and stop the process for including g in the multicast tree. Next, we start from the second multi-receiver e. Connect e with its parent b and stop since b is already a tree node. Similarly for the third multi-receiver f, we connect f with c, c with a, and then a with s. Now the tree construction is complete since all the receivers are connected to the tree. The constructed multicast tree is shown in Fig. 2(b).

We can see that in the tree, level 0 = $\{s\}$, level 1 = $\{a, b\}$, level 2 = $\{c, d, e\}$ and level 3 = $\{f, g\}$. Thus, we get the channel assignment in Fig. 2(c), where the number above the node represents the channel for receiving, and the number below the node for sending.

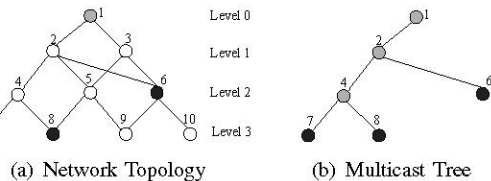The LCA algorithm has two advantages: simple implemen-

tation and throughput improvement. It only needs one BFS of the network at the beginning, and it creates the multicast tree by connecting the multi-receivers with the nearest tree nodes. The tree nodes then can decide the channels by themselves according to the level information, which can be realized distributedly. At the same time, the use of multi-channel reduces the close-by interference and allows more simultaneous transmissions.

However, there is still potential for the LCA algorithm to improve system throughput. Firstly, LCA cannot diminish the interference among the same levels since it uses the same channel at the same level. For example, in Fig. 2(c), since g is in the transmission range of both c and d, there will be interference when c and d use the same channel. Secondly, when the number of available channels is more than that of the levels, some channels will not be utilized, which is a waste of channel diversity. Thirdly, the channel assignment does not take the overlap property of the two adjacent channels into account. As we know, $\forall i$, channel i and channel $i + 1$ are adjacent in frequency, so they partially interfere with each other. Thus, the channel i for level i still has some inference effect with the channel $i + 1$ for level $i + 1$.

IV. MULTI-CHANNEL MULTICAST ALGORITHM

To further improve the system throughput, we propose a Multi-Channel Multicast algorithm (MCM) to minimize the number of relay nodes and the hop count distances between the source and the destinations, and further reduce the interference by exploiting all the partially overlapping channels instead of just the orthogonal channels.

A. Multicast Structure Construction

Following the design constraint of WMNs, we aim for a multicast protocol for WMNs, which includes two primary procedures. The first is to build an effective multicast structure, which is detailed in this subsection, and the second tries to allocate channels for minimizing interference in the next subsection.

Broadcast Structure. Some previous work treats broadcast and multicast in a different way. Actually, when all the nodes are multi-receivers, the multicast problem becomes the broadcast problem. We can say that broadcast is a special case of multicast. In order to focus on the basic idea of MCM, we first consider the situation that all the nodes are the multi-receivers.

We then detail how to trim those unnecessary branches based on the broadcast structure when the multi-receivers are only a portion of the nodes. The broadcast structure in the mesh network is built by the following steps.

The **first step** is realized by BFS, which is similar with the LCA algorithm. After the BFS traversal, all the nodes are divided into different levels. We then delete the edges between any two nodes of the same level, with which we get the elementary communication structure — "tree mesh". Fig. 2(a) and Fig. 2(d) give an example of the original network topology and its responding tree mesh. We use BFS to build the tree mesh for the following reasons:

1) With the hop count distance increasing between the sender and the receiver, the intra-flow contention exacerbates. Moreover, shorter hop count distance means shorter transmission delay. Minimizing the delay is also important in WMNs, thus we build a shallow tree by BFS, which reduces the total hop count distances from the source to the receivers.
2) BFS guarantees that if two nodes are not at the same level or the adjacent levels, they are at least two hops away. Hence, when considering channel assignment, the two nodes may use the same channel since they are very likely not to interfere with each other.
3) The time complexity of BFS is $O(|V| + |E|)$, whose cost is much less than other broadcast or multicast tree construction algorithms.

In the **second step**, we identify the minimal number of relay nodes that form the broadcast tree. In the tree mesh, one node could have more than one parent. The purpose of this step is to identify the only parent (we call it a relay node here) for a node who has more than one parent so that the number of relay nodes is minimal.

A top-down approach, i.e., from level 0, level 1 to the lowest level, is used to identify the relay nodes. Suppose we have discovered the relay nodes in level 0, level 1... level $i - 1$, now we study how to find the relay nodes in level i. We see that the fewer relay nodes will result in less traffic flows in the network, which means less local interference. Thus, our objective is to identify the minimal number of relay nodes in level i that can communicate with all the nodes in level $i + 1$.

Definition 3: Given a tree mesh T_G, $T_G(i)$ is a subgraph of T_G and consists of only the nodes at level i and level $i + 1$ of T_G. This subgraph $T_G(i)$ is called $(i, i + 1)$ **subtree mesh**. In addition, the set S_i consisting of the nodes from level i is called the **upper node set** of $T_G(i)$, and the set S_j consisting of the nodes from level $i + 1$ is called the **lower node set**.

We can see that identifying the minimal number of relay nodes at level i is equivalent to selecting the minimal number of nodes at upper node set of $T_G(i)$ that can cover all the nodes of the lower node set. In fact, it is a variation of the set-cover problem, which has been proved NP-complete. We devise an approximation algorithm, which is detailed in Algorithm 2.

Data: $T_G(i)$: $(i, i + 1)$ subtree mesh; S_i: nodes in level i; S_j: nodes in level $i + 1$
Result: R: the set of the relay nodes in level i
$R = \emptyset$;
while $S_j \mathrel{!=} \emptyset$ **do**
    In $T_G(i)$, compute the number of parents of each node in S_j, and compute the number of children of each node in S_i;
    Find $v_{i1}, v_{i2}, ...$ in S_j with the minimal number of parents;
    Among the parents of $v_{i1}, v_{i2}, ...$, find t_f with the maximal number of children;
    $R = R \cup \{t_f\}$;
    $S_i = S_i - \{t_f\}$;
    The children of t_f record t_f as their relay node;
    $S_j = S_j - \{$the children of $t_f\}$;
end
Algorithm 2: Relay Node Search in Level i Algorithm

1) Some parents are considered as relay candidates if one of their children has the minimal number of parents.
2) Among the relay candidates, we choose one node that has the maximal number of children. The reason is that given the fixed number of nodes at the level $i + 1$, the more children a relay node can forward packets to, the fewer relay nodes we will need at level i.
3) We remove the relay node and its children, and repeat the above process until all the nodes at level $i + 1$ are removed.

We use a simple example to further explain this algorithm. Fig. 3(a) gives a $(i, i + 1)$ subtree mesh, from which we can compute the number of parents of each node in level $i + 1$. The nodes 1, 5, and 7 have the minimal number of parents (1 parent), and their parents are nodes a, c, and d. The number of children of nodes a, c, and d are 3, 2, and 2, respectively. Since node a has the maximal number of children, a is chosen as a relay node.

We then remove node a and its child nodes 1, 2, and 3 from the subtree mesh. In the new subtree mesh, which is shown in Fig. 3(b), the node 5 and 7 have the minimal number of parents, and their parents are nodes c and d. We randomly choose node c as one relay node since c and d both have 2 children.

Afterwards, we remove c and its children, then get the subtree mesh shown in Fig. 3(c). Similarly, with the process above, we select node d as a relay node. After removal of node d and its children, the level $i + 1$ is empty, thus the algorithm stops. Finally, nodes a, c, and d are chosen as relay nodes at level i, which is shown in Fig. 3(d).

Algorithm 2 is superior to the Greedy Set Cover algorithm [4] by introducing step 1. We observe that if a node has just one parent, the parent has to be selected as a relay node, while that greedy algorithm recursively selects the node with the maximal number of children in the remained graph. For the above example, that greedy algorithm will select a, b, c, and d as relay nodes.

Multicast Structure. The broadcast structure mentioned above contains some unnecessary branches if the destinations do not involve all the nodes. Instead, we propose to construct

a "slim" structure by using the MCM Tree Construction algorithm described in Algorithm 3.

```
Data: T: tree mesh of the network
Result: T': multicast tree
Use BFS to partition nodes into different levels;
for ∀ node v ∈ V(T) do
    c[v]=true if and only if v is a multi-receiver or the source.
end
for l = LevelNum − 1; l >= 1; l = l − 1 do
    Sᵢ = {node vᵢ|vᵢ belongs to level l − 1};
    Sⱼ = {node vⱼ|vⱼ belongs to level l and c[vⱼ] = true};
    while Sⱼ! = ∅ do
        Find vᵢ₁, vᵢ₂, .... in Sⱼ with the minimal number of parents;
        Among the parents of vᵢ₁, vᵢ₂, ...., find node tⱼ with the maximal
        number of children;
        c[tⱼ] = true;
        Sᵢ = Sᵢ − {tⱼ};
        The children of tⱼ record tⱼ as their relay node;
        Sⱼ = Sⱼ − {the children of tⱼ};
    end
end
V(T') = ∅; E(T') = ∅;
for ∀ node v ∈ V(T) do
    V(T') = V(T') ∪ {v} if and only if c[v] = true;
    edge e = (v, v's relay parent);
    E(T') = E(T') ∪ {e};
end
```
Algorithm 3: MCM Tree Construction Algorithm

The goal of the algorithm is to discover the minimal number of relay nodes to construct a multicast tree. The search process starts from the bottom to the top. We use a boolean variable – "c[v]" for any node v to represent that v is either a multi-receiver or a relay node if c[v] is true. At each step, we intend to minimize the number of relay nodes at the upper level, which can cover all the multi-receivers and relay nodes at the lower level. The process is similar with the broadcast structure, except that we do not require that the relay nodes should cover those non-receiver and non-relay nodes of the lower level.

We use a simple example to illustrate the process. There is a tree mesh in Fig. 4(a), where nodes 6, 7, and 8 are multi-receivers. First, we select node 4 at level 2 because it covers all the multi-receivers at level 3. Next, we select node 2 at level 1, which covers all the multi-receivers and the relay node at level 2. Finally, we get the multicast tree in Fig. 4(b).

B. Channel Assignment

The tree node discovery in the previous subsection allows each multi-receiver to connect with the gateway through the minimal hop count distance. In this section, we discuss how to assign channels to the interfaces of the tree nodes by proposing two allocation algorithms: ascending channel allocation, and heuristic channel assignment.

Ascending Channel Allocation. As assumed in Section II, each node has two interfaces. Specially, the interface that a node uses to receive packets from its relay node at the upper layer, termed Receive-Interface (RI), is disjoint from the interface the node uses to forward packets to its children, called Send-Interface (SI). In order to guarantee that the relay node can communicate with its children, each node's RI is associated with the SI of its relay node, i.e., they should be

assigned the same channel. Ascending Channel Allocation is proposed to assign channels and described in Algorithm 4.

```
Data: {0, 1, ...C − 1}: available orthogonal channel set; T': multicast tree;
Result: Channel assignment for interfaces
The source uses channel 0 for its SI;
Its children use channel 0 for their RIs;
A = 0;
for l = 1; l ≤ LevelNum-2; l++ do
    for ∀ relay node u at level l do
        A = (A + 1) mod C;
        u uses channel A for its SI;
        u's children use channel A for their RIs;
    end
end
```
Algorithm 4: Ascending Channel Allocation Algorithm

The basic idea of the algorithm is straightforward: from top to down in the tree, the channels are assigned to the interfaces in the ascending order until the maximum channel number is reached, then start from channel 0 again. Although simple, this approach avoids the situation where the same channel is assigned to two nearby links that interfere with each other. We use a simple case to illustrate this algorithm in Fig. 5, where the the number of the orthogonal channels is 3. Note that the number above the node represents the channel number used for its RI, while the number below the node represents the channel number for its SI.

In the algorithm we only use limited orthogonal channels. As we know, 802.11b protocol provides 14 channels, 5 MHz apart in frequency. However, to be totally orthogonal, the frequency should be at least 30 MHz, so 802.11b can offer only 3 non-overlapping channels. Thus, although the Ascending Channel Allocation is easy to implement, its performance is still constrained by the limited number of orthogonal channels. Fortunately, as mentioned in Section II, network throughput can be further improved by exploiting all the partially overlapping channels.

Heuristic Channel Assignment. In fact, we can utilize all the channels instead of just orthogonal channels. In Section II, we observed that the interference range decreases with the increase of the channel separation. Intuitively, the channel assignment should make a large channel separation for two wireless links if the physical distance between them is short. We aim to minimize the sum of the interference area of all the transmissions.

We use $IR(u_v)$ to indicate the interference range of sender u of one link with respect to sender v of another link. According to the experiment we performed in Section II, under the condition that all the nodes have the same transmission range R, $IR(u_v) = R * \delta_{|i_u - i_v|}$. Here, u and v use channel i_u and i_v for their SIs respectively, and δ_t is the Interference Factor. When allocating a channel for relay node u, the channel assignment should take a channel that minimizes the sum of the square of the IRs between u and u's neighboring relay nodes, that is, minimize $\sum_{v \in N(u)} IR^2(u_v)$, where $N(u)$ represents the set of the neighboring relay nodes of u. Since $\sum_{v \in N(u)} IR^2(u_v) = \sum_{v \in N(u)} (R * \delta_{|i_u - i_v|})^2$, we just need to
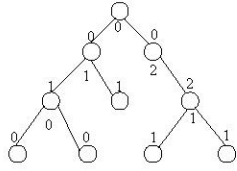
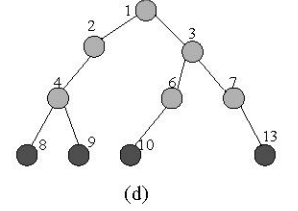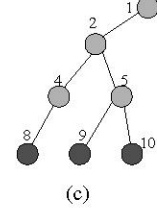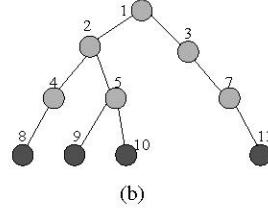Fig. 5. Ascending Channel Allocation Example

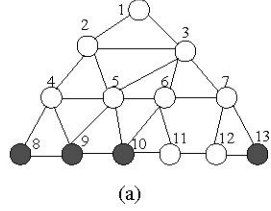(a)   (b)   (c)   (d)

Fig. 6. Repair Mechanism Example

minimize $\sum_{v \in N(u)} \delta^2_{|i_u - i_v|}$. Based on this consideration, we propose the Heuristic Channel Assignment in Algorithm 5.

Data: CH: available channel set; T': multicast tree;
Result: Channel assignment for interfaces
The source uses channel 0 for its SI;
Its children use channel 0 for their RIs;
for $l = 1$; $l \leq LevelNum-2$; l++ **do**
    for $\forall$ *relay node* u *at level* l **do**
        S(u) = {u's neighboring relay nodes that have been assigned channels for their SIs }
        Choose channel $i \in CH$ that minimizes $\sum_{v \in S(u)} \delta^2_{|i_u - i_v|}$
        u uses channel i for its SI;
        u's children use channel i for their RIs;
    end
end

Algorithm 5: Heuristic Channel Assignment Algorithm

V. FURTHER DISCUSSION ON MCM ALGORITHM

In this section, we discuss some companion mechanisms to further improve the MCM algorithm including the failure recovery mechanism and the node join mechanism.

Failure Recovery. Usually, the mesh routers work reliably, but node failure still could happen for various reasons. When a tree node fails, nodes in its subtree lose their connectivity to the root. Our mechanism will reorganize the multicast tree to bypass the failed node and restore the connectivity.

At first, we can safely assume that any node is able to detect the failure of its neighbor quickly since the nodes periodically send "hello" messages to their neighbors. If a node does not receive a hello message from one neighbor for a period of time, it considers the neighbor to have crashed. There are two cases that apply to the node failure: the collapsed node is a leaf or a relay node.

For the first case, if the collapsed node v is a leaf, we propose two approaches. One is just to leave it alone, since the leaf is not responsible for forwarding packets to any other tree nodes. (That means that the mesh clients within the communication range of the failed node restore the connectivity to the network, which is beyond the scope of this paper.) This approach is simple, but the parent of the failure node will continue to receive packets even if it is not a multi-receiver.

The second choice is related with v's parent u on the tree. If u has only one child on the tree, then it stops forwarding packets. Moreover, if u is not a multi-receiver, it sends out a message to its neighbors, announcing that it is no longer part of the tree. Then, the parent of u will do the same thing as u

does. The process continues until one ancestor of v has more than one child or it is a multi-receiver. The first choice is easy to implement, and v is able to join the multicast group again if v recovers from failure after a short period of time. The second choice helps to remove those unnecessary branches, which reduces the interference and saves the bandwidth.

For the second case, when v is a relay node, all its children on the tree should check whether they are physical neighbors of some other relay nodes on the tree. (If two nodes are within each other's transmission range, they are called **physical neighbors** even if they are not using the same channel.) If they are, the channels of their RIs will be reassigned as that of the "backup" relay node's SI and re-establish the connectivity with the gateway. If they are not physical neighbors of any relay node, each node will randomly choose one neighbor t at the upper layer, requesting t to be its relay node. If t is the physical neighbor of one relay node, it connects with the relay node by using the same channel, otherwise, t will randomly choose one of its neighbors at the upper layer, asking that node to be a relay node. This process continues until the request arrives at a physical neighbor of any relay node.

We use an example in Fig. 6 to illustrate the repair mechanism. Fig. 6(a) and Fig. 6(b) give the network topology and the responding multicast tree respectively, where nodes 8, 9, 10, and 13 are multi-receivers. If node 13 breaks down, because it is a leaf, we can simply leave it alone. The other choice is that 13 requires its parent 7 to stop forwarding packets. And node 7 realizes that it has one child on the tree, so it also asks its parent 3 to stop forwarding. The resulting multicast tree after failure of node 13 is shown in Fig. 6(c).

If node 5 breaks down, its children begin to look for other connections to the source. Node 9 finds that it can communicate with relay node 4, so it changes the channel on its RI for packet reception from 4. Node 10 cannot communicate with any relay node on the tree, so it randomly slects one of its physical neighbors at the upper level, such as 6, requesting 6 to be its relay node. Node 6 then tries to communicate with any neighboring relay node, and it sets the channel of its RI the same as the SI of relay node 3. Node 6 then chooses a channel that is not used by any of its neighbors for its SI, and node 10 accordingly sets the same channel for its RI for packet reception. The resulting multicast tree after the failure of node 5 is shown in Fig. 6(d).

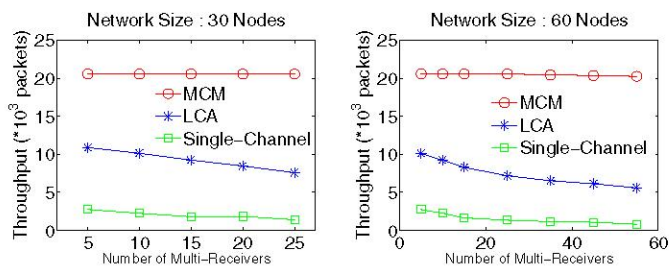***Node Join.*** If a new node wants to join the multicast group,

Fig. 7.   Impact of Network Size

Fig. 8.   Impact of Number of Available Channels

it sends out a request for connecting to a nearby relay node. The request is locally flooded, and the nodes on the path that reaches the nearest relay node will be absorbed into the multicast tree.

VI. SIMULATIONS

We evaluate the MCM algorithm by comparing it with the LCA algorithm and a single channel multicast algorithm through the following metrics.

- **Throughput:** the throughput is the average number of packets each multi-receiver receives during a time unit.
- **Delay:** the delay is the average time it takes for a packet to reach the destination after it leaves the source.

We use an NS2 simulator (version 2.29)[9] to simulate a flat area of 900m by 900m with varying number of randomly positioned wireless router nodes. By extending the NS2 simulator, we configure all nodes to use multiple interfaces/channels with a transmission range of 250m and a carrier sensing range of 550m. We use the default IEEE 802.11 MAC configuration in NS2 which supports multicasting using broadcasting at the base rate 1Mbps.

We evaluate LCA and MCM algorithms in different scenarios. For each scenario, we randomly generate 100 different graphs where the source and the destinations are randomly selected. Traffic is generated by constant bit rate (CBR) sessions. We vary the session rate at some scenarios. The packet size for all traffic is set to be 512 bytes. Except for the last subsection, we use the orthogonal channels in the simulation.

A. Impact of Network Size

We evaluate the throughput in different network sizes by assigning the number of nodes with 30 and 60, and assigning the number of the available channels with 12. For each network size, we vary the number of multi-receivers from 5 to 55. We measure the throughput of the MCM algorithm, the LCA algorithm, and the single-channel algorithm in which only one single channel is used in the multicasting. The results are shown in Fig. 7. We can see that using multi-channel and multi-interface significantly improves the throughput. The reason is that using different channels prevents the channel interference among close-by transmissions. Compared with
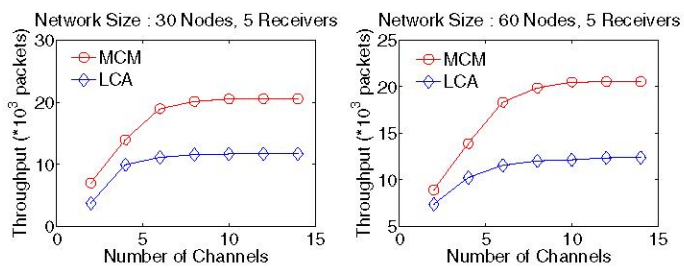
LCA, MCM further improves throughput though they both take advantage of multi-channel and multi-interface. This is because MCM builds a more efficient multicast tree and carefully assigns the channels on the tree; thus, it further reduces the interference.

B. Impact of Number of Available Channels

We vary the number of available channels from 2 to 14 and measure the throughput of MCM and LCA. Fig. 8 shows that MCM enhances more and more throughput than LCA with the increasing of the number of available channels. In addition, we notice that when the number of available channels varies from 2 to 6, both MCM and LCA have great throughput improvement. However, when the number of available channels varies from 7 to 14, MCM has small throughput improvement while LCA almost has no improvement. The explanation of the phenomena on MCM is that when the number of channels increases to a certain extent, it is enough to eliminate almost all the interferences in the network, thus using more channels cannot further improve throughput. At the same time, the number of channels that LCA uses is equal to the tree height, thus some channels are left unused when the number exceeds tree height, although some interferences still exist in the network.

C. Impact of Transmission Rate

We vary the transmission rate from 50 packets/s to 300 packets/s, and measure the throughput of MCM and LCA. Fig. 9 shows that MCM achieves much better throughput than LCA under different transmission rates. We also observe that the saturated transmission rates for MCM and LCA to achieve nearly the maximal throughput are 225 packets/s and 125 packets/s respectively. The transmission rate exceeding the saturated rate almost does not help to improve the throughput. Since MCM has the higher saturated transmission rate, this means that MCM can take greater advantage of the channel diversity than LCA.

D. Delay Comparison

In this simulation, we evaluate the delay of LCM and MCM by comparing the average time each packet takes to reach the destination. The transmission rate is set to 200 packets/s. Fig. 10 shows that MCM has a much shorter delay than LCA.
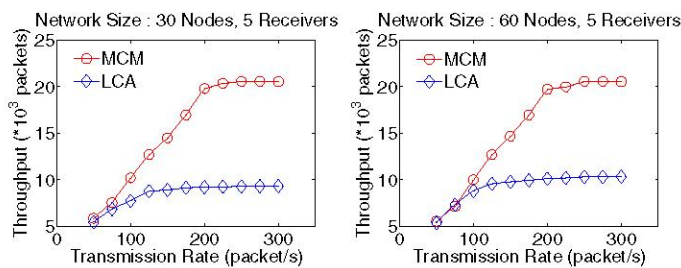
8

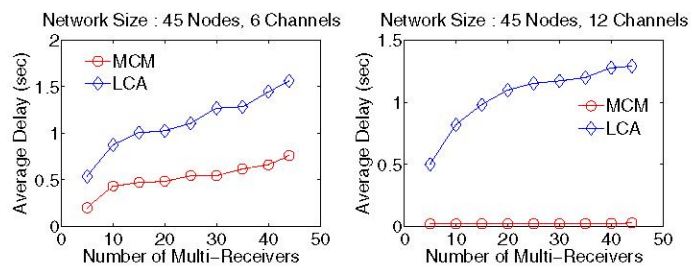Fig. 9.   Impact of Number of Transmission Rate

Fig. 10.   Delay Comparison

We also see that the delay of MCM decreases rapidly when the number of channels increases from 6 to 12, since we choose a fast transmission rate. When the number of available channels is 6, it just diminishes some interferences, and residual interferences still restrain more packet transmissions. The packets have to remain at the buffer of nodes, and delay becomes large. At the same time, when the number of available channels is 12, it can greatly diminish interference, so the more concurrent transmissions take place, and data packets do not need to stay at the buffer for a long time. As a result, when using 12 channels, the delay of MCM decreases to nearly zero.

E. Partially Overlapping Channel Assignment Test

Usually, users are offered a range of frequency spectrum, where the numbers of orthogonal channels and partially overlapping channels are fixed. We choose two frequency spectrums: i) 30MHz, which can offer 2 orthogonal channels and 6 partially overlapping channels, and ii) 60MHz, which can offer 3 orthogonal channels and 12 partially overlapping channels. In the two spectrums, we compare the throughput of the MCM algorithm under the following different channel assignment methods.

1) We only use the orthogonal channels defined in the fixed range of frequency spectrum.
2) We use Ascending Channel Allocation to allocate all the partially overlapping channels in the frequency spectrum to the interfaces.
3) We use Heuristic Channel Assignment to allocate all the partially overlapping channels in the frequency spectrum to the interfaces.

The results in Fig. 11 show that using partially overlapping channels can achieve better throughput than using just orthogonal channels. This is because the orthogonal channels are so scarce that they can not eliminate all the interference, while the partially overlapping channels can further reduce interference. We also observe that the Heuristic Channel Assignment is better than the Ascending Channel Allocation, since it makes a large channel separation for the adjacent wireless links.

VII. Related Work

As a basic data communication mechanism, multicast has been intensively studied in Internet and multi-hop wireless networks. Those proposed multicast algorithms and protocols do not take advantage of multi-channel and multi-interface. At the same time, there exists a large number of studies that aim to address the channel assignment problem of mesh networks, which is characterized by using multi-channel and multi-interface. Thus, we summarize multicast research in MANET, and representative research on multi-channel in wireless network.

The multicast routing protocols in MANET are classified into three categories according to the way multicast routes are created: 1) tree-based, 2) mesh-based, and 3) stateless multicast. In the tree based protocols, the data packets are transmitted from the source to the destinations along the paths on the multicast tree, which helps to minimize the bandwidth cost [11][25][29]. The mesh-based protocols try to create multiple trees among the group members, such that the packets can be delivered to each receiver through multiple paths [5][7][14]. The alternative paths help to increase the protection against the topology alteration. However, both tree-based and mesh-based multicast protocols have to bear the overhead of creating and maintaining the multicast overhead keeping at the intermediate nodes. In order to address this drawback, stateless multicast protocols have been proposed to store the destination list in the packet header, and the packets are self-routed to the destinations based on the geographical information [3][12][16].

Our proposed MCM algorithm inherits the tree-based multicast design, but it differs from previous approaches in the following aspects: i) Instead of power efficiency and route recovery, which are usually the paramount goals of multicast in MANET and sensor networks, MCM aims at maximizing throughput. ii) The multicast tree constructed by MCM is formed by the diverse channels. Based on this, MCM is able to minimize the interference by making use of multi-channel and multi-interface.

In recent years, multi-channel issues in wireless networks have drawn much attention. Several link layer and MAC layer protocols have been proposed to improve the performance of wireless networks [2][15][19][26][28]. These approaches are to find the optimal channel for the current packet transmission for essentially avoiding interference. Such schemes have the key advantage that a single radio is required to support multiple channels. Some researchers aim to derive

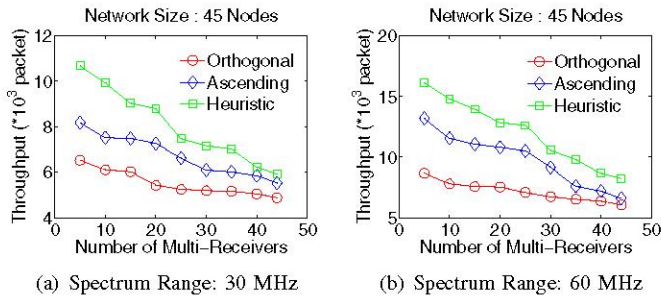|	
(a) Spectrum Range: 30 MHz	(b) Spectrum Range: 60 MHz

Fig. 11. Partially Overlapping Channel Test

the lower bound or upper bound of the capacity in terms of achievable QoS in mesh networks [13][27]. Many studies focus on how to assign channels to nodes in the network, either by the static or the dynamic methods [1][20][21][22][23]. They develop a set of centralized or distributed algorithms for channel assignments by taking the bandwidth cost, efficient routing, and load-balance into account. Most of them believe that static assignment outperforms dynamic assignment due to the channel switching cost and the delay. New metrics have also been proposed for multihop wireless networks with considering the impact of channel interference, which is used to find high throughput paths between sources and destinations [6][24].

Our channel assignment is different from previous channel assignment approaches at two aspects: i) we assign channels based on the multicast structure so that we can exploit the broadcast property of wireless nodes, and ii) we make full use of the partially overlapping channels instead of just orthogonal channels, which ultimately reduces the interference and improves the throughput.

VIII. CONCLUSION

In this paper, we investigate the multicast algorithm in wireless mesh networks where the throughput and the delay have the paramount priorities. In order to achieve efficient multicast in WMNs, two multicast algorithms, LCA and MCM, are proposed by using multi-channel and multi-interface. An effective multicast structure is constructed to minimize the number of relay nodes and the communication delay. The dedicated channel assignment helps to further minimize the interference as well. Compared with previous multicast approaches, our algorithms are based on the multi-channel and focus on the throughput maximization. The performance evaluation shows that our algorithms outperform the single-channel multicast in terms of throughput and delay, and more efficient multicast structure and subtle channel assignment can further improve throughput and reduce delay.

IX. ACKNOWLEDGEMENTS

REFERENCES

[1] M. Alicherry, R. Bhatia, and L. Li. Joint channel assignment and routing for throughput optimization in multiradio wireless mesh networks. In *MobiCom 05*, 2005.
[2] P. Bahl, R. Chandra, and J. Dunagan. Ssch: slotted seeded channel hopping for capacity improvement in ieee 802.11 ad-hoc wireless networks. In *ACM MobiCom 2004*, 2004.
[3] K. Chen and K. Nahrstedt. Effective location-guided overlay multicast in mobile ad hoc networks. *International Journal of Wireless and Mobile Computing(IJWMC)*, 3, 2005.
[4] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction To Algorithms*. The MIT Press, 2001.
[5] S. Das, B. Manoj, and C. Murthy. A dynamic core based multicast routing protocol. In *MobiHoc 2002*. 2002.
[6] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *MobiCom 04*, 2004.
[7] J. J. Garcia-Luna-Aceves and E. L. Madruga. The core-assisted mesh protocol. *IEEE Journal on Selected Areas in Communications*, 17(8), Aug. 1999.
[8] http://pdos.csail.mit.edu/roofnet/doku.php.
[9] http://www.isi.edu/nsnam/ns/index.html.
[10] http://www.seattlewireless.net.
[11] J. Jetcheva and D. B. Johnson. Adaptive demand-driven multicast routing in multi-hop wireless ad hoc networks. In *MobiHoc 2001*, 2001.
[12] L. Ji and M. S. Corson. Differential destination multicast-a manet multicast routing protocol for small groups. In *IEEE INFOCOM 2001*, pages 1192–1201, Apr. 2001.
[13] M. Kodialam and T. Nandagopal. Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. In *MobiCom 05*, 2005.
[14] S. Lee, M. Gerla, and C. Chiang. On demand multicast routing protocol. In *IEEE WCNC 99*, pages 1313–1317, Aug. 1999.
[15] Y. Liu and E. Knightly. Opportunistic fair scheduling over multiple wireless channels. In *IEEE INFOCOM 03*, 2003.
[16] M. Mauve, H. Fuler, J. Widmer, and T. Lang. Position-based multicast routing for mobile ad hoc networks. In *MobiHoc 2003*, June 2003.
[17] A. Mishra, E. Rozner, S. Banerjee, and W. Arbaugh. Exploiting partially overlapping channels in wireless networks: Turning a peril into an advantage. In *ACM/USENIX Internet Measurement Conference*, 2005.
[18] A. Mishra, V. Shrivastava, and S. Banerjee. Partially overlapped channels not considered harmful. In *SIGMetrics/Performance 06*, 2006.
[19] A. H. M. Rad and V. W. Wong. Joint channel allocation, interface assignment and mac design for multi-channel wireless mesh networks. In *INFOCOM 07*, 2007.
[20] K. Ramachandran, E. M. Belding, K. Almeroth, and M. Buddhiko. Interference-aware channel assignment in multi-radio wireless mesh networks. In *Infocom 06*, 2006.
[21] B. Raman. Channel allocation in 802.11-based mesh networks. In *INFOCOM 06*, 2006.
[22] A. Raniwala and T. cker Chiueh. Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network. In *IEEE INFOCOM 2005*, 2005.
[23] A. Raniwala, K. Gopalan, and T.-c. Chiueh. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks (mc2r). *ACM Mobile Computing and Communications Review*, 8(2):50–65, 2004.
[24] S. Roy, D. Koutsonikolas, S. Das, and Y. C. Hu. Highthroughput multicast routing metrics in wireless mesh networks. In *ICDCS 06*, 2006.
[25] E. Royer and C. Perkins. Multicast operation of the ad-hoc on demand distance vector routing protocol. In *MOBICOM 99*, 1999.
[26] J. So and N. Vaidya. Multi-channel mac for ad hoc networks:handling multi-channel hidden terminals using a single transceiver. In *MobiHoc '04*, 2004.
[27] J. Tang, G. Xue, and W. Zhang. Maximum throughput and fair bandwidth allocation in multi-channel wireless mesh networks. In *INFOCOM 06*, 2006.
[28] A. Tzamaloukas and J. J. Garcia-Luna-Aceves. A receiver-initiated collision-avoidance protocol for multi-channel networks. In *IEEE INFOCOM 2001*, 2001.
[29] C. Wu, Y. Tay, and C. Toh. Ad hoc multicast routing protocol utilizing increasing id-numbers (amris) functional specificastion. In *Internet Draft*, Nov. 1998.