Efficient Data Dissemination Sub-structures in Wireless Extension Networks

Dongwook Lee, Gayathri Chandrasekaran, Prasun Sinha
{leedon,chandrga,prasun}@cse.ohio-state.edu
Department of Computer Science and Engineering
Ohio State University

Abstract

The need for an easily deployable and quickly reconfigurable Wireless LAN (WLAN) architecture has led to recent research on networks of Access Points with wireless inter-connection. Such networks, termed XNETs, are suitable for setting up indoor, outdoor and temporary WLANs. Prior work on similar architectures has mostly focused on unicast flows. This paper focuses on the construction and maintenance of efficient sub-structures for providing best-effort broadcast services over XNETs. The association of the users to the APs is controlled for optimizing the size of the sub-structure. We prove lower and upper bounds on the hardness of the problem using approximation preserving reductions with other well known problems. We propose and evaluate a distributed approach for computing the sub-structure. Our approach is based on greedy association of users with APs that have a large number of users in their cells and are also close to the current sub-structure. Our evaluation is based on three channel sharing schemes: uni-channel, dual-channel and multi-channel, which differ in the number of channels and the way they are used by the APs and the users. The proposed approach is compared to two other approaches of association based on signal strength and number of hops to the existing sub-structure. We observe that our approach can reduce the number of APs in the sub-structure by up to a factor of 6. This results in a heavy reduction in control and data packet overhead in the sub-structure, leading to higher packet delivery ratio. We also observe that the cost metric based on the number of hops to the sub-structure is easier to compute and maintain, and performs similar to the normalized cost metric for most scenarios.

1 Introduction

The last few years has witnessed a tremendous growth in the WLAN market in homes, enterprises and public hot-spots. Declining costs of access points (APs), costs of WLAN NICs, and support for high bandwidth, has succeeded in enticing the common user. However, the deployment cost of network of APs is often dominated by the cost of laying cables[1] to provide wired connectivity between the APs. To reduce the deployment cost and design easily deployable wireless access networks, researchers have recently started investigating alternate architectures that involve wireless-only connectivity between the APs. The terms wireless back-haul networks and mesh networks [1] are often used to refer to networks of access points with wireless inter-connection. However, research in these areas has primarily focused on providing Internet connectivity to large communities. To emphasize that our work is relevant to indoor as well as outdoor networks, we introduce the new term, XNET, to refer to networks of APs that provide extended coverage using wireless connectivity between APs. Various types of WLANs, such as city-wide WLANs[2], in-building WLANs, and temporary WLANs, can all benefit from the XNET technology.

XNETs have three key advantages over traditional wired networks of APs. First, the *deployment is easy and low cost* due to the absence of wires between the APs. This also makes XNETs easier and faster to deploy. Second, XNETs are *highly reconfigurable* due to the absence of wired connections between the APs. With an expected rapid growth in newly deployed networks, frequent reconfigurations may be necessary to improve network performance and support its further expansion. Third, XNETs are *ideal for rapid deployment of temporary networks* with large cumulative coverage area in indoor as well as outdoor locations.

Due to rapid increase in data rates of new and upcoming WLAN solutions, XNETs have become ever more practical. In multi-hop networks, interference is often the main reason for low throughput. However, the reduced throughput due to such interference may be comparable or even exceed the throughput of the backbone Internet connection. The wireless access speed has grown rapidly from the 2 Mbps 802.11 solution, to 54 Mbps supported by 802.11a and 802.11g. Even higher speeds are offered by proprietary exten-

---

[1] Although Ethernet is inexpensive there is often non-trivial cost of labor, planning, and leasing associated with it especially for large WLAN deployments.

[2] The city of Chaska, Minnesotta provides WLAN coverage in a 15 sq miles area since Oct 2004 (www.chaska.net).

sions such as Texas Instrument Inc's turbo mode in the form of ACX100 chipset that can provide up to 108 Mbps. Availability of such high speed access solutions has moved the bottleneck to the wired backbone. With declining costs of APs that support newer WLAN technologies, and high cost of upgrading existing wired connections, the wired backbone will remain as the bottleneck for the foreseeable future. Thus, multi-hop wireless networking of APs will not effect the observed throughput of the end-users.

Both unicast and broadcast services need to be supported in XNETs. The need for supporting efficient broadcast services in the access network is eminent with various emerging applications including real time multi-party conferencing, scientific data visualization, and presentation broadcasting at conferences or during lectures. Broadcast services can also be used to disseminate local news, visitor's information, TV channels, or other multimedia information. *In this paper we focus on the construction and maintenance of efficient sub-structures for providing best-effort broadcast services over XNETs.*

The structure of the network of APs in XNET is akin to the multihop wireless networks. Such networks are known to suffer from increased latency and reduced throughput in the presence of large volumes of traffic. The key reason for such inefficiencies are attributed to interference and hidden terminal problems [2]. These problems become more severe in the presence of broadcast data [3]. For improved performance it is thus critical to optimize the number of transmissions between the APs and from the APs to the users to achieve the desired fidelity of broadcasting.

We propose and evaluate a distributed approach to the problem. Our solution is based on the following two observations. First, controlled dynamic assignment of users to APs can be leveraged to reduce the traffic between the APs. Second, the selected APs can construct an efficient sub-structure to receive data from the APs that are connected to the backbone Internet.

Our contributions in this paper are as follows.

- We formalize the problem of efficient construction of sub-structures for data dissemination over XNETs. We prove lower and upper bounds on the hardness of the problem using approximation preserving reductions with other well known problems.

- We propose a distributed approach for constructing the dissemination sub-structure.

- We evaluate the performance of the proposed approach, and compare it with a signal strength based association approach for various scenarios. Our evaluation also considers three different channel sharing schemes: *Uni-channel:* single channel for all users and APs; *Dual-channel:* one-channel for all users and another for all APs, assuming that APs have dual interfaces for simultaneous communication with users as well as other APs; *Multi-channel:* one-channel for all APs to communicate with each other, and the other channel is used for communicating with users, but neighboring cells use different channels (assumes dual interface APs like in the Dual-channel case).

In the distributed approach, the access points periodically advertise a normalized cost which is computed as the number of hops to the current sub-structure divided by the number of users. Users greedily select the AP in range that has the lowest cost, thus attempting to optimize the number of additional links in the AP-net (the sub-network consisting of all the APs) that are added for serving the newly covered users.

We have implemented our solutions in the *ns2*[4] simulator. In our simulations we compare our normalized cost approach with two other distributed approaches that are based on: Hop-count and signal strength based association. Our approach reduces the number of APs in the sub-structure by up to a factor of 6. It incurs very low control message overhead and creates very low data load in the sub-structure, leading to higher packet delivery ratio. This results in higher packet delivery ratio for our approach. We observe that when there is traffic in the network, the multi-channel configuration typically performs the best but the other two configurations have higher packet delivery ratio due to multiple coverage. We also observe that the cost metric based on the number of hops of the AP from the current tree is easier to compute and maintain, and performs similar to our normalized cost metric for most scenarios.

The rest of the paper is organized as follows. Section 2 defines the problem, the notations, and the terminology used in the paper. The hardness of the problem of computing the sub-structure is captured in Section 3. The metric to optimize for computing the sub-structure is discussed in Section 4. The distributed approach is described in detail in Section 5. Section 6 presents a detailed evaluation of our approach and comparison with other approaches using simulations. Section 7 summarizes relevant related work. In Section 8, we present a discussion of some important extensions and facets of the problem that we did not address in this paper. Finally, Section 9 concludes the paper.

2 Terminology and Problem Definition

We define here some graph theoretic notations and terms used in the rest of the paper. Following the notation used in [5], we use $V(G)$ to denote the set of nodes and E(G) to denote the set of edges for a graph G. For a node v, $N(v)$ represents the set of neighbors of v, and $N[v]$ represents the set of neighbors including v itself. Hence $N[v] = N(v) \cup \{v\}$.

Dominating Set (DS): A set of nodes $S \subseteq V$, such that all nodes not in S have an edge connecting them to a node in S. The nodes in S are called dominators.

Connected Dominating Set (CDS): A DS $S \subseteq V$, such that the induced subgraph of S in G, denoted by G[S] is connected.

Minimum CDS (MCDS): The smallest CDS.

Edge Weighted MCDS (EW-MCDS): Here the objective function is the total weight of the links in the tree connecting the dominating set nodes.

We represent the connectivity between the users and the access points using a graph $G = (V, E)$, where V (same as $V(G)$) is the set of nodes (users and access points) and E (same as $E(G)$) is the set of edges. E consists of edges connecting users to access points in range, and between access points that are in range of each other. E does not include user-to-user edges as we do not consider ad-hoc communication between the users. V can be partitioned into the set of users, V_u, and the set of access points, V_a. We assume that one of the APs, called the main AP (MAP), has a connection to the backbone Internet and acts as a gateway to the rest of the APs.

The problem is to find the smallest tree connecting the MAP to a subset of APs such that all users will have coverage from some AP in the computed tree. We name this problem the Restricted Coverage Problem (RCP).

The analysis, and algorithms presented in this paper can be easily extended to the generalized version where multiple APs have backbone connection. As the bandwidth in the backbone is typically much larger that the bandwidth on the wireless channel, we can transform the problem with a multiple MAPs to a problem with a single MAP by *fusing* the nodes corresponding to the MAPs. For purpose of simplicity, in the rest of the paper we assume that there is only a single MAP.

3 The Hardness of the RCP problem

We prove a lower bound and an upper bound for the RCP problem using approximation preserving reductions.

Theorem 1: [Lower bound I] RCP is at least as difficult as MCDS.
Proof: We present a reduction from an arbitrary instance of the MCDS problem to an instance of the RCP problem. Let $G = (V, E)$ be the graph in which an MCDS needs to be computed.

We construct a new graph G' by using two copies of G, say G_1 and G_2. See Figure 1 for an illustration of the reduction. In the illustration, the dotted edges and the white circles represent new edges and nodes added as part of the reduction. We then remove all the edges in G_2. For each edge $(u, v) \in E$, we draw an edge from u_1 to v_2 and another one from v_1 to u_2 (the subscripts correspond to the subscripts of the copies of G). We also draw edges between corresponding vertices in the two sets, i.e., between u_1 and u_2 for all $u \in V$. We pick an arbitrary node say w_1 in G_1, We add a new node x_1 in G_1 and add edges between x_1 and all members of $N[w_1]$.

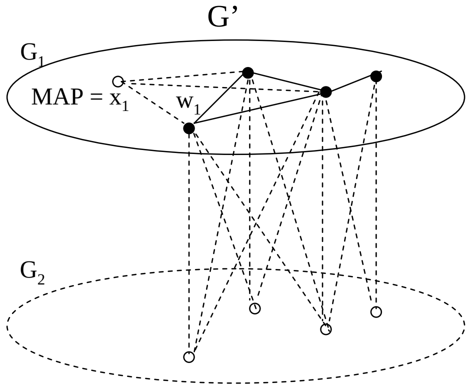

Figure 1: Reduction from MCDS to RCP

We claim that the solution to RCP(G') with $V_a = V(G1)$, $V_u = V(G_2)$ and MAP $= x_1$ gives a solution to MCDS(G). The solution to RCP(G') includes the new vertex x_1. Observe that any solution to MCDS(G) when mapped to G_1, will include at least one neighbor of x_1. So, by deleting x_1 from the solution of RCP(G') we get the solution of MCDS(G). Similarly any solution to MCDS(G) can be augmented with x_1 to get a solution for RCP(G'). ☐

Theorem 2: [Upper bound II] RCP is at most as difficult as the EW-MCDS problem.
Proof: Consider an instance of the RCP problem. Add a new node x_2 with the MAP as its neighbor. Edges with weight of ∞ are added from the MAP to all nodes in V_u to which it does not already have an edge. All edges incident on vertices in V_u are marked with a weight of ∞.

Solution to EW-MCDS(G') will avoid nodes in V_u due to the high cost of edges to nodes in V_u. The MAP will be included in the solution to EW-MCDS(G') due to the node x_2 that is hanging off it. In G', the MAP has edges to all nodes in V_a. ☐

The above reductions bound the hardness of the RCP problem. The MCDS problem represents the lower bound, which is NP-hard problem [6]. To the best of our knowledge, the best approximation factor for a polynomial time algorithm for the MCDS problem [6] is $H(\Delta) + 2$, where H is the harmonic function and Δ is the maximum degree of the graph. However, the EW-MCDS problem is known to be hard to approximate,

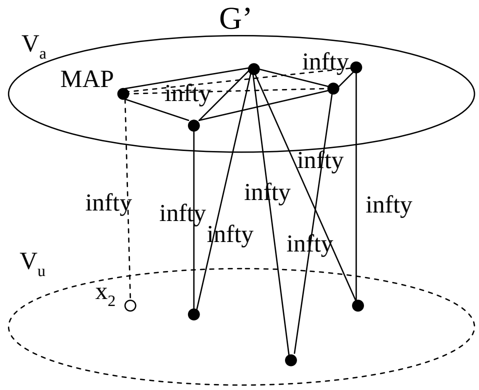

Figure 2: Reduction from RCP to EW-MCDS

for which no non-trivial approximation algorithms are known [6, 7]. The results presented above are a first step towards bounding the hardness of the RCP problem. With discovery of tighter bounds, and equivalent problems, it will be easier to design approximation algorithms. Such algorithms can then be executed centrally for small XNETs or distributedly for larger XNETs.

4 Optimization Metric for the Sub-structure

In this section, we present details of the three channel configurations and precisely derive the metric to optimize for constructing the sub-structure.

Let us assume that there are n APs selected in the solution to RCP. The MAP is one of those APs. Let the number of leaves in the tree joining them be n_l, and the number of APs with associated users in their cells be n_u. Note that $n_u \geq n_l$ as all APs that are leaves in the tree, must have associated users in their cells. There are two choices for communication between the APs: unicast or local broadcast. However, the communication on the last hop from the AP to its associated users is assumed to be a local broadcast transmission. Using unicast on the last hop for distributing broadcast data may increase reliability but at the cost of higher bandwidth requirement.

The bottleneck in this architecture is the channel in the AP-net. In order to precisely derive the criteria to optimize, we first compute the number of transmissions that occur in the AP-net channel. This reflects the amount of load on the AP-net that directly impacts the broadcast throughput. Note that for reliable communication there may be recovery traffic and retransmissions which are ignored in this discussion.

We analyze the three different channel configuration cases separately. The analysis is summarized in Table 1

- *Uni-channel:* All users and APs are communicating in the same channel. In the case of unicast in the AP-net, there will be a transmission on all links on the tree. In addition, all APs with associated users in their cells will require a total of n_u additional transmissions. So the total number of transmissions in the channel is $n - 1 + n_u$. For broadcast transmissions in the AP-net, the total number of transmissions will be n.

	AP-AP unicast	AP-AP broadcast
Uni-channel	$n - 1 + n_u$	n
Dual-channel	$n - 1$	$n - n_l$
Multi-channel	$n - 1$	$n - n_l$

Table 1: Optimization function

- *Dual-channel:* This configuration assumes that APs are equipped with dual interfaces. The primary interface is dedicated for communication to other APs. The secondary interface is configured to a secondary channel for communication with users in its cell. All users have a single interface configured to the same secondary channel. Since transmissions to the users are in a different channel, we only need to count the transmissions within the AP-net. For AP-AP unicast communication, there will be $n - 1$ transmissions over the links of the tree and for broadcast communication, there will be $n - n_l$ transmissions by the internal (non-leaf) nodes of the tree.

- *Multi-channel:* This is like the bi-channel scenario, but nearby interfering cells are assumed to have different secondary channels. The number of transmissions in the AP-net channel is also same.

For the case of unicast transmissions in the AP-net, reducing the total number of nodes in the tree n is critical. In addition, for the uni-channel scenario, the number of APs that have associated users, n_u needs to be minimized too.

Broadcast transmissions in the AP-net require optimizing n for the uni-channel scenario, but for the other two scenarios, the number of internal nodes in the tree or $n - n_l$ is the criteria to optimize.

Reliability in the AP-net is extremely critical for high delivery ratio of broadcast data. We therefore do not consider the case of broadcast transmissions in the AP-net. We note that it is possible to enhance broadcast transmissions in the AP-net with recovery and reliability mechanisms. In the rest of the paper we assume that the transmissions in the AP-net is unicast and the transmissions from the AP to the users are broadcast. Although $n - 1$ and $n - 1 + n_u$ are two different optimization functions, we note that n is larger than n_u, and assume that optimizing n will also optimize n_u. For simplicity of the protocol design, in the rest of the paper we assume n to be the only function to optimize for all the three channel configurations. We therefore propose a single distributed solution for the three configurations.

5 Distributed Approach

We seek to design a distributed solution for computing the sub-structure in the wireless backbone.We use the normalized cost$definedinsection$5.1 as a metric for the user to choose between the various access points it can hear. Since our approach uses normalized cost based association for computing the dissemination structure, we call it NCADS. We have proposed a four state solution for dealing with the mobility of the users in the network and have also considered the case where the access point can only support a limited number of users.

The goal of the distributed protocol is threefold:

- The computed sub-structure must minimize the number of selected APs.
- The number of overhead packets generated by our approach should be minimized.
- The dissemination sub-structure must adapt quickly for mobile users.

We use a greedy approach motivated by a greedy solution to the MCDS problem [5]. Note that the greedy solution to the MCDS problem can not be directly used as we have only proven that RCP is at least as hard as MCDS. Our protocol requires each user to select an AP to associate with, among all the APs that it can hear. We use the following observations to optimize the total number of selected APs:

- The APs that are already a part of the dissemination sub-structure should be given preference by users looking for APs to associate with. This will reduce the number of overhead messages in constructing the sub-structure.
- The APs that are in range of a large number of users have a higher potential for serving a large number of users. So APs with more users in range must be preferred.

We have split our discussion into five subsections - the metric computation, the association strategy based on the computed metric, the sub-structure construction, handling mobility of the users and the scenario with limited number of users per AP.

5.1 Metric computation

Based on the above discussion, we derive the following cost metric - normalized cost. Normalized cost is defined as the cost incurred per user for the addition of an access point to the tree and is represented as

$$\text{Normalized Cost of AP} = \frac{H}{N}$$

where H is Number of hops from AP to the existing tree and N is the Number of users in range of AP. H represents the number of intermediate APs that need to join the sub-structure to connect the AP with the already existing tree using the shortest path between them. N represents the potential number of users that are benefiting or that may benefit should this AP join the tree.

Each access point computes its Normalized cost and sends out beacons to the users advertising its normalized cost. Among all users in range, each user selects the AP with the least Normalized cost.

We have compared the normalized cost based metric with two other metrics as part of performance evaluation. They are as follows:

- Signal strength : Among the APs that are in the range of the user, the one with the strongest signal is chosen for association by the user.
- Hop Count: This metric is closer to our Normalized cost metric. The users select the AP which has the least number of Hops to the existing tree. It is hard for APs to keep the value of N up to date s users report to APs only while scanning channels for association. Therefore, this metric is easier to compute and maintain than the normalized cost.

In chapter 6, we compare our metric with the above two commonly used metrics and establish that Normalized cost metric performs better than the signal strength and the Hop Count Metrics.

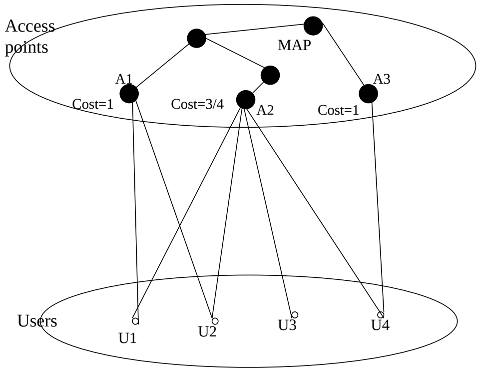

Figure 3: Metrics evaluation

Assume that the current tree only includes the MAP. To illustrate the significance of the Normalized cost metric, lets consider the network shown in Figure3 having 6 APs and 4 users. We compare the number of APs chosen when each of the three metrics - signal strength, Hop count and Normalized cost are considered by the users to choose from the set of access points it can hear. When Signal strength is considered, U_1 would choose A_1, U_2 and U_3 would choose A_2 and U_4 would choose A_3. Hence, all the three access points need to be a part of the tree for this system to serve the users. When the Hop count is used as a metric, U_1 chooses A_1 because A_1 has lesser cost(2) compared to A_2(3). Similarly U_2 chooses A_2, U_4 chooses A_3 and U_3 chooses A_2 because that is the only access point it can hear. Hence, again, 3 access points need to be a part of the tree for this scheme to work. Lets consider the Normalized cost based scheme where the cost of A_1 is $2/2 = 1$, A2 is $3/4$ and A_3 is $1/1 = 1$. Thus, all users select A_2 for association and hence only one AP needs to join the tree. This would greatly reduce the amount of traffic in the AP network and would help improve the throughput of the network.

5.2 Metric based channel selection

Both active and passive scanning can be used to associate the users with the access point.

Active scanning *Active scanning is a process in which the wireless node cycles through all the channels and sends a "Probe Request" to all APs within its range and waits for a "Probe ACK" from these Access points within a time period. Among the stations that it heard from, the user selects the one with the least normalized cost and sends an association update message to it. The AP then adds an entry in its user table for the user it recently got associated with.*

Passive scanning *Passive scanning is a process which is initiated by the Access point. Here, the user cycles through all the channels and listens to beacons from APs that are within its range in that channel. After scanning all the channels, the station associates itself to the access point having the least normalized cost.*

When the network starts up, the APs get to know about the number of users in its range only if the user reports to it. Hence, the users start active scanning. On reception of the probe messages from the users, APs record the number of users in its range and sends out beacons advertising its Normalized cost. The computation of the Normalized cost requires *H*, which is learnt using a DSDV like approach in the AP-net. The user selects the AP with the least Normalized cost and sends "Association message" to the selected AP. The AP updates its associated user list on hearing the "Association message". The association is then maintained by periodic "Association update" messages.

5.3 Sub-structure computation

When an AP is newly selected to join the sub-structure, it sends out a "Join" message to the closest node which is already a part of the sub-structure and joins the closest node through the shortest path once it receives a "Join ACK". Similarly, if it wants to get disconnected from the sub-structure, it sends out "Prune" message to its parent node and gets disconnected on the reception of "Prune ACK" from its parent.

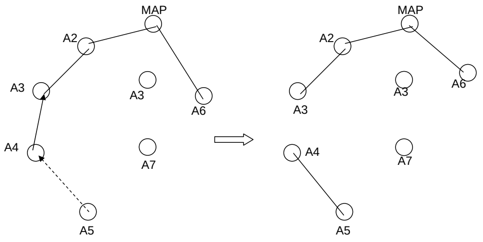

Figure 4: Motivation for the four state protocol

However this simple scheme can result in the formation of a forest (collection of trees). Consider the Figure 4 where the node A_4 initiates a "Prune" message and is waiting for a "Prune ACK" message from its parent node to get disconnected from the tree. While A_4 is waiting, assume that A_5 sends out a "Join" message to A_4. Since A_4 is already a part of the Dominating set and has not got disconnected from the tree yet, it adds A_5 to it. Eventually, when its parent node serves A_4's prune request which was raised before A_5 asked for a "Join", A_4 would get disconnected form the substructure and there would be two disconnected trees. To avoid the formation of disconnected trees during the tree construction stage, we propose a four state solution to this problem and ensure that the tree remains connected in all possible situations.

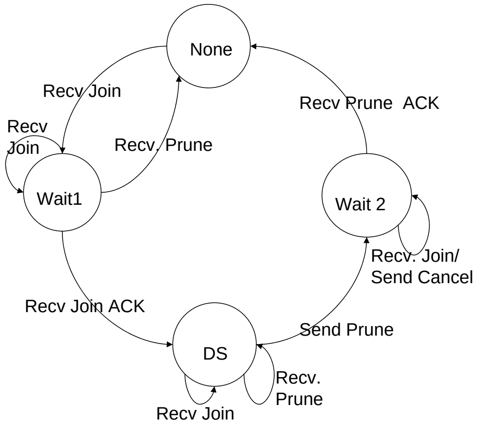

Figure 5: 4 state protocol

As shown in Figure 5, each AP can be in one of these four states - None, Wait1, DS(Dominating Set) and Wait2. When an AP which is in "None" state receives a "Join" request, it forwards the request to the closest node in sub-structure and moves to "Wait1" state. In this state, the access point can receive additional "Join" messages, in which case the new AP that requested for a "Join" is added to its list of APs which have already requested for a "Join". A "Prune" message can be initiated from the AP that initiated the "Join". This message causes the AP to return back to None state. At the reception of the "Join ACK", the AP moves to DS state and forwards the "Join ACK" message to all the nodes in its list which requested for a "Join". At the DS state, A "Join" request would be served immediately with a "Join ACK" because the AP being requested is already a part of the dominating set. A "Prune" request would move the AP's state from DS to Wait2 if the node that requested for a prune is its only child. If the AP has other children, the AP would remain in DS state and reply back with a "Prune ACK" to the AP that requested a prune. In the Wait2 state, the AP can receive only "Join" messages because there are no children connected to the AP which can ask for being pruned. The "Join" message would be canceled because serving that would lead to a disconnected graph which is incorrect. On reception of the "Prune ACK" from its parent, the AP moves back to the "None" state. All these ACK messages have a timeout and if the node that requested for a "Join" or "Prune" does not receive an ACK within the specified time, it retransmits the message.

5.4 Handling user mobility

When a user moves in or out of the range of an AP, the APs update their user table and send out beacons periodically informing the modified cost to the users in range. The users then use the metric to choose between the APs.

In the uni-channel and dual channel cases, each user sends regular "Association update" message to the AP it has been associated with and waits for an "Association Update" response from the AP within a given time. If it times out, the user determines that it has moved out of the range of the associated AP and hence starts active scanning by sending out "probe" messages. The new APs in the user's range recalculate their number of users they currently support and start advertising the new cost.

In a multi channel case, if a mobile user loses its association with the AP and receives no acknowledgment for its association update message, the user hops to different channels, sends out probe messages and waits for a response from the APs in its range within the time out period. Among the APs that it heard from, the user selects the AP with the least cost, and shifts to that channel and associates itself to the corresponding AP.

5.5 Limited users per AP

Our discussion so far has assumed that an unlimited number of users can associate with an AP. But in reality, the number of users per AP is often bounded. A typical limit is 32 users for most 802.11 based APs. Our protocol can be easily extended to support limited users. A simple extension involves a flag in each beacon message. The flag is set only when the AP is already associated with the maximum number of users allowed. If a user decides to associate with an AP which is already serving its maximum allowable number of users, the AP would reply back to the user's "Join" message with a NACK, forcing the user to associate with another AP.

Another way will be to advertise higher costs when APs start to get saturated. These approaches have an impact on the number of users that get starved (rejected by all neighboring APs as they are saturated).

6 Performance Analysis

In this section we present a thorough comparison of the NCADS protocol using simulations in the Network Simulator ns2 [4]. We compare its performance with two other association strategies: based on signal strength and based on hop count metric (see Section 5.1 for their description). The metrics of evaluation are: the number of control messages to compute and maintain the broadcast sub-tree, the number of APs in the subtree, the number of data packets transmitted in the subtree and the number of unique data packets received by the users. Our study is mostly based on the multi-channel configuration. The highlights of our evaluation for the five components of our study are as follows:

1. *Impact of user density:* We observe that NCADS has the lowest number of dominating APs and the lowest number of control messages. For the case of 50 users, the number of dominating APs in NCADS is 33% lower than signal strength based association, and 10% lower than hop count based association. The improvement is higher for higher density of users.

2. *Density of APs:* When the separation between APs in the grid becomes 50 m or larger, on an average NCADS performs 5.6% and 45.03% better than the hop count based algorithm and the signal strength based algorithm in terms of the number of APs in the sub-structure. The number of dominating APs chosen based on NCADS is one-fourth of the number of APs chosen by the signal strength metric, for an inter-AP separation of 30 m.

3. *Delivery ratio of packets at the users:* NCADS performs 21% and 16% better than the signal strength algorithm and the hop count algorithm, respectively, at a packet transmission rate of 100 packets/sec (800 Kbps).

4. *Three channel configurations: single-channel, dual-channel, and multi-channel:* When there is traffic in the network, the multi-channel configuration typically performs the best but the other two configurations have higher packet delivery ratio due to multiple coverage.

5. *Speed of users:* NCADS has lowest number of dominating APs at all speeds. In terms of the number of APs, NCADS performs 30.74% and 8.56% better than the signal strength algorithm and the hop count algorithm, respectively, at a maximum user speed of 15 m/sec.

For our simulations we use a grid topology of 10x10 APs. The distance between neighboring APs, D, is 80m and radio propagation range of AP, R, is 100m, unless mentioned otherwise. We use a single MAP in our simulation. Users are uniformly distributed in the area and move randomly according to the random way-point model. We used a maximum speed of 8 m/s with 2 sec pause time for the random way-point model for all experiments, unless mentioned otherwise. The users associate with APs using active scanning. The unicast data rate is 11 Mbps and the broadcast data rate is set to 1 Mbps.

In a multi channel scenario, each user has a single wireless interface and each AP has two wireless interfaces: backbone interface and local subnet interface. APs communicate with each other through the backbone interface. The backbone interface of all APs share a single channel. APs communicate with users via local subnet interface. The neighboring APs are configured in such a way that they are on different channels on the local subnet interface. In simulation, we assign 13 channels to the local subnet interfaces of APs. In the single channel scenario, APs and users have one wireless interface and they share a single channel. In a dual channel scenario, the AP has two wireless interfaces and the user has a single wireless interface. Priority queuing is being used to give higher priority to the control packets as compared to the data packets.

6.1 User Density

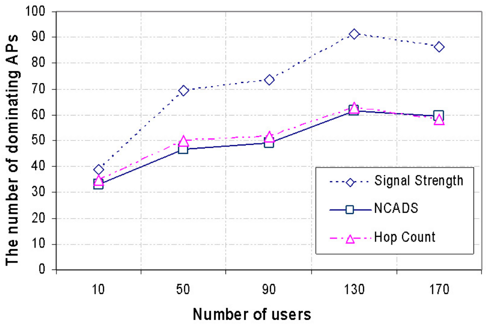

Figure 6: Number of dominating APs with respect to the number of users

Figure 6 shows that the number of dominating APs with respect to the number of users. As the number of users increases, the number of dominating APs also increases. We observe that NCADS has the lowest number of dominating APs. For the case of 50 users, the number of dominating APs is 33% lower than hop count based approach and 6% better than signal strength based approach. Figure 7 shows the number of control messages sent by APs in the broadcast subtree. The control messages includes all the tree management messages such as “Join” and “Prune”. NCADS has the lowest control packet overhead (43.7% better than signal strength algorithm when the number of users is 170), since NCADS has lowest number of APs in the broadcast tree. Figure 8 shows the number of data packet transmissions in the tree. As the number of users increases, the size of the tree increases. This increases the data traffic in the tree. However, NCADS has the lowest number of data packet transmissions. Hence, NCADS generates lowest backbone traffic load.

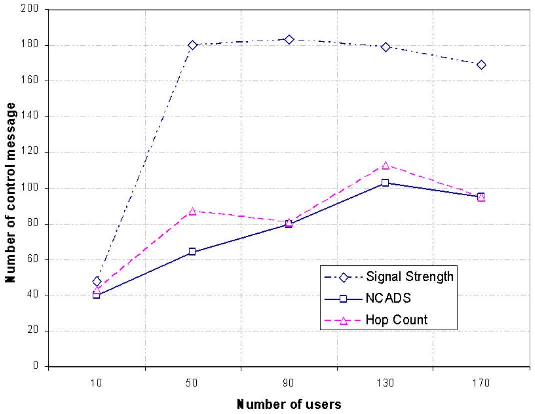

Figure 7: The number of control messages sent by AP with respect to the number of users.

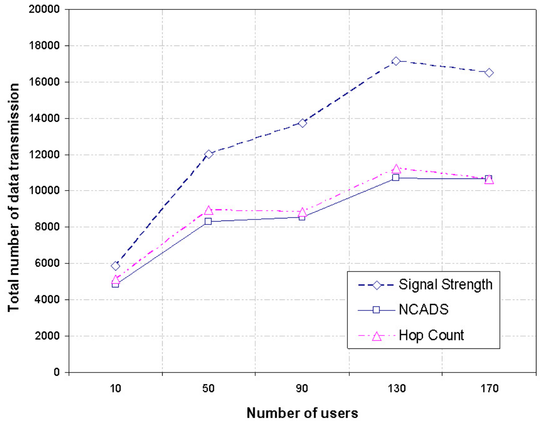

Figure 8: The number of data forwarded on the broadcast subtree with respect to the number of users.

6.2 AP Density

Figure 9 shows the number of dominating APs versus the separation of adjacent APs. As the density of APs decreases, the number of dominating nodes increases. In the denser AP topology (i.e., AP distance is 10m), hop count algorithm has the best performance, since many users are within the range of the MAP. Hence, they can associate to the MAP directly. The NCADS has fewer dominating APs when the distance between neighboring APs becomes 50 m or larger and at an average performs 5.6% and 45.03% better than the hop count based algorithm and the signal strength based algorithm, respectively.

Figure 10 represents the number of control messages sent by APs. In dense case (i.e., D=10m), the signal strength algorithm has a very high number of control messages compared to others, since users change association very frequently. The total number of data packets forwarded on the broadcast tree is shown in Figure 11. As the distance between adjacent APs increases, the number of data transmissions also increases. In the heavy density case, the hop count algorithm performs 85.16% and 53.39% better than the signal strength and NCADS. However, the NCADS performs 60.3% and 11.18% better than the signal strength and hop count algorithms beyond an AP separation of 50 m.

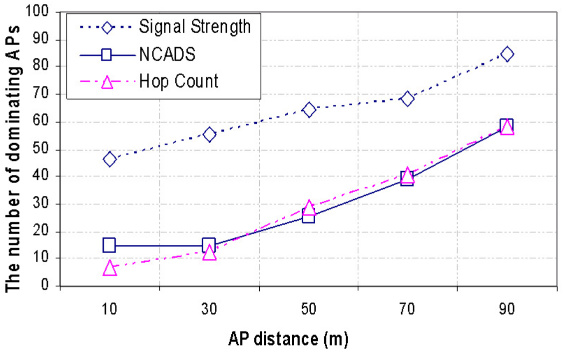

Figure 9: Number of dominating APs with respect to AP density.

6.3 Delivery Ratio

Figure 12 shows the number of successfully received broadcast data packets versus users with varying packet sending rates. The data packet size is 1024 bytes. As the packet sending rate increases, the delivery ratio goes down. The number of packets lost at high transmission rates (i.e. 1000 packets/sec) is much higher than the number of packets lost at low transmission rates. However, NCADS has 21% and 16% better performance than the signal strength and the hop count algorithms, respectively, at the rate of 100 packets/sec (800 Kbps). Figure 13 shows that the number of dominating APs is stable across different data rates as the control messages have higher priority in the interface queue than the data packets.

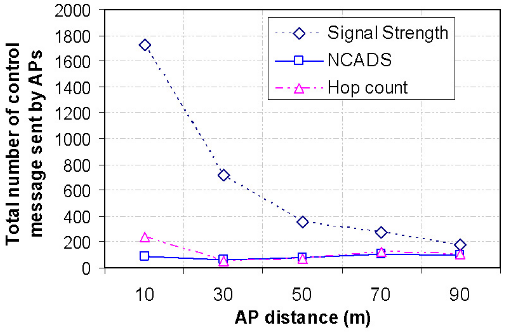

Figure 10: The number of control messages sent by AP with respect to AP density.

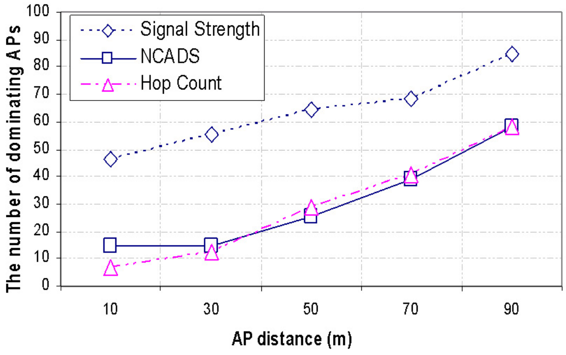

Figure 11: The number of data forwarded on the broadcast subtree with respect to AP density.

6.4 Channel Configuration

Figure 14 shows the number of dominating APs with varying number of users. In this scenario, the MAP sends 1024 byte packets every 5ms. Without traffic, we expect the three channel configurations to generate similar results. We observe that multi channel has smaller number of dominating APs and single channel has largest number of dominating APs for all user densities. The load on the channel used by the tree is more for the single channel and dual channel cases. Higher load results in increased loss and delay of control messages, resulting in higher number of APs. Figure 15 shows the total number of received data packets for each channel model for NCADS. In single channel and dual channel case, users can hear packets from several APs around it. Thus, if the neighboring APs of the users are participating in the dominating set, the user can receive a data packet multiple times. This leads to better performance for uni-channel and dual-channel configurations, as shown in Figure 15.

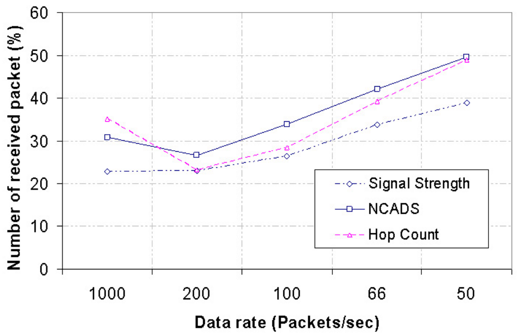

Figure 12: Number of successfully received data by users with respect to the data rate.

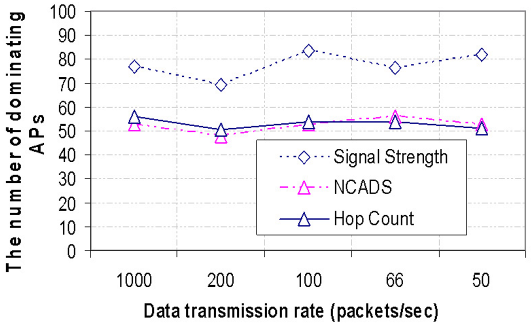

Figure 13: Number of dominating APs with respect to the data rate.

6.5 Impact of user speed

The movement of users causes changes in user-AP association. With higher speeds, users will be frequently changing associations, resulting in more control messages. Figure 16 shows the increased number of control messages with the increase in speed of the user from 1 m/sec to 15 m/sec. NCADS and hop count based algorithms have low control message overhead compared to the signal strength based algorithm. This is due to the smaller number of dominating APs for NCADS as shown in Figure 17. The figure shows that the number of dominating APs increases as the user speed increases. In terms of the number of APs, NCADS performs 30.74% and 8.56% better than the signal strength and the hop count algorithms, respectively, at a maximum user speed of 15 m/sec.

7 Related Work

In this section, we outline related work in the areas of mesh networking, controlled association in 802.11 networks, and sub-structure computation in ad-hoc networks.

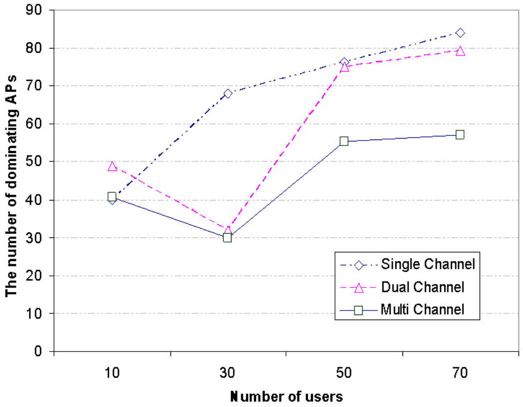

Figure 14: The number of dominating APs of NCADS with respect to channel scenario.

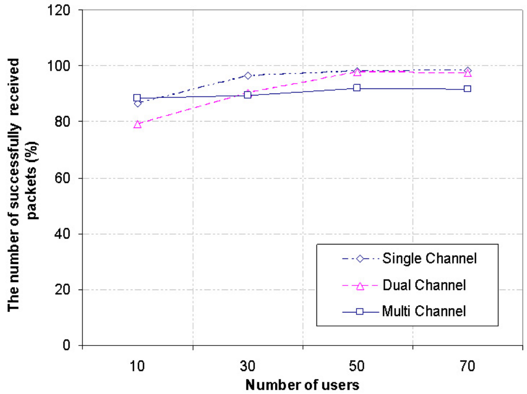

Figure 15: The number of received data of NCADS by user with respect to channel scenario.

Providing connectivity to large communities using wireless backhaul networks, also known as mesh networks, has lately received a lot of attention [1, 8]. Several companies including Meshnetworks, Firetide, Strix, and BelAir Networks have various products based on the concept of mesh networking.

In a recent work [9], authors have explored the problem of fairness across flows between the APs in an architecture similar to XNET. The APs are referred to as Transit Access Points (TAPS). The authors propose a fairness model and an approach at layer 2 for providing fairness. The study considers the roles of the MAC protocol, end-to-end congestion control, antenna technology and traffic types. The critical relationship between fairness and aggregate throughput is captured by the reference model. This work is orthogonal to our work as it only pertains to unicast traffic.

In 802.11 networks user nodes often use the signal strength as the key metric in selecting the AP. Recent work [10] has explored the idea of association control to balance the network load and provide max-

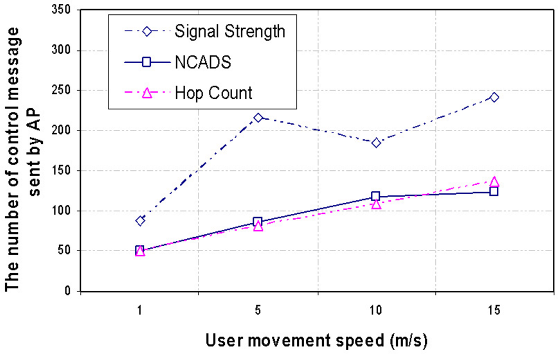

Figure 16: The number of control messages with respect to the user speed.

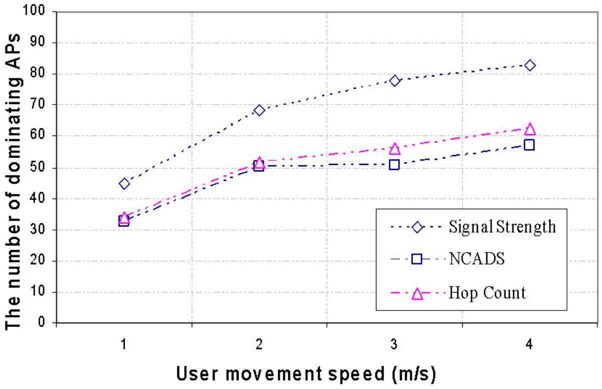

Figure 17: The number of dominating APs with respect to the user movement speed.

min fairness among users. The authors prove that balancing the network load is equivalent to achieving the max-min fairness. Authors propose a rigorous formulation of the association control problem and propose 2-approximation algorithm for unweighted greedy users and a 3-approximation algorithm for weighted and bounded demand users. Although our objective function was different from [10], in the presence of unicast flows load-balancing and fairness will make the RCDS problem more interesting. Study of combined unicast and broadcast traffic with goals of fairness and efficient construction of sub-structures is important and is left for future study.

The idea of constructing backbones or sub-strictures in ad-hoc networks to limit the number of transmissions has been explored by several protocols. The concept of MCDS has been used in designing various routing protocols for ad-hoc networks [11, 12]. The importance of constructing and maintaining an MCDS in an ad-hoc network has spurred research on finding better approximation algorithms [6].

8 Discussion and Future Work

In this paper, we have focused on unreliable data dissemination with the goal of distributing data efficiently. To keep our discussion and study focused, we have ignored various other facets of the problem. We outline some such extensions that we are currently pursuing.

- *Multiple MAPs:* In the more general case, there may be multiple APs with backbone Internet connection. Although we noted in Section 3 that theoretically the problem is equivalent to the problem with a single MAP, there are some details that need to be worked out for the distributed protocols with multiple MAPs.

- *Coverage from multiple APs:* In the uni-channel and bi-channel scenarios, it is possible to configure the users to receive packets from nearby APs to which they are not associated. In such scenarios the reliability of data reception can be improved while constructing the sub-structure to guarantee coverage from multiple APs for each users. However, to obtain a performance gain by using coverage from APs, the extra overhead created must also be accounted for. To benefit from multiple coverage in the multi-channel scenario, users need to be equipped with multiple wireless cards.

- *Extensions for Supporting QoS constraints of Real Time Flow:* Real time flows typically have various QoS requirements such as end-to-end delay and jitter. Our study in this paper has not considered such QoS requirements.

- *Extensions for Reliable Data Dissemination:* For applications such as scientific data visualization, or software upgrade, the data dissemination scheme needs to be enhanced with recovery mechanisms for lost packets. If the fraction of lost packets is significant, the recovery traffic will impact the protocol's performance. Smart mechanisms to enhance the proposed approaches to handle reliability is part of current research.

- *Ad-hoc Communication between users:* The construction of the sub-structure did not explore user to user communication. If the users are allowed to receive traffic from other users and if APs are allowed to receive traffic from users, the problem becomes equivalent to multicasting in an ad-hoc network. This may work well in a uni-channel scenario. However, for the bi-channel and the multi-channel scenario with single adapter users, solutions such as [13] can be used to enable simultaneous user operation in infrastructure and ad-hoc modes.

9 Conclusion

The need for easily deployable and quickly reconfigurable Wireless LAN architecture has led to research on network of APs with wireless inter-connection. Such extensions of wired backbone networks are termed XNETs in this paper. We have focused on the construction and maintenance of an efficient sub-structure for providing broadcast services over XNETs. The sub-structure consists of the association of users to APs and the tree connecting the APs to the Internet connected APs. We proposed a distributed approach for constructing the sub-structure based on a new metric called the normalized cost that is advertised by the APs in their beacons. Users periodically scan the channels and associate with the AP with the least normalized cost. We have proved lower and upper bounds on the hardness of the problem using approximation preserving reductions with other well known problems. Using simulations, we observe that the number of APs in the sub-structure can be reduced by up to a factor of 6 using our approach as compared to a signal strength based association approach. This results in a heavy reduction in control and data packet overhead in the sub-structure, leading to higher packet delivery ratio. We also observe that a cost metric based on the number of hops of the AP from the current tree, is easier to compute and maintain, and performs similar to our normalized cost metric in most scenarios. Based on the simulation results, we claim that NCADS is highly suited for supporting broadcast services in XNETs.

References

[1] "Meshnetworks Inc.," http://www.meshnetworks.com.

[2] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LANs," in *Proc. ACM SIGCOMM*, Oct. 1994, vol. 24, pp. 212–215.

[3] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," in *Mobicom*, 1999.

[4] ns2: Network Simultor, ," http://www.isi.edu/nsnam/ns/.

[5] D. West, *Introduction to Graph Theory*, Prentice Hall, 1996.

[6] S. Guha and S. Khuller, "Approximation Algorithms for Connected Dominating Sets," *Algorithmica*, vol. 20, no. 4, pp. 374–387, 1998.

[7] V. Vazirani, *Approximation Algorithms*, Springer Verlag, 2001.

[8] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, " Link-level Measurements from an 802.11b Mesh Network," in *Sigcomm*, Aug. 2004.

[9] V. Gambiroza, B. Sadekhi, and E. W. Knightly, " End-to-End Performance and Fairness in Multihop Wireless Backhaul Networks," in *Mobicom*, Sept. 2004.

[10] Y. Bejerano, S.-J. Han, and L. Li, " Fairness and Load Balancing in Wireless LANs Using Association Control," in *Mobicom*, Sept. 2004.

[11] B. Das, R. Sivakumar, and V. Bharghavan, " Routing in Ad Hoc Networks Using a Spine," in *Proc. of the 6th International Conference on Computer Communications and Networks* , Sept. 1997, p. 34.

[12] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: a Core-Extraction Distributed Ad-hoc Routing algorithm," *Journal on Selected Areas in Communications (Special Issue on Ad-Hoc Networks)*, vol. 17, no. 8, pp. 1454–1465, 1999.

[13] R. Chandra, P. Bahl, and P. Bahl, "MultiNet: Connecting to Multiple IEEE 802.11 Networks Using a Single Wireless Cards ," in *IEEE INFOCOM*, Hong Kong, Mar. 2004.