The Case for a Multi-hop Wireless Local Area Network

Seungjoon Lee, Suman Banerjee, Bobby Bhattacharjee
Department of Computer Science, University of Maryland, College Park, MD 20742, USA
Emails: {slee,suman,bobby}@cs.umd.edu

CS-TR 4504 and UMIACS TR 2003-73

***Abstract*— We propose a multi-hop wireless LAN architecture and demonstrate its benefits to wireless clients. For this architecture, we define implementation paths that allow interoperation with existing wireless LANs and therefore lead to an incremental deployment of this system. We quantify the performance benefits of the proposed schemes through measurements in realistic wireless LAN environments. We also examine the performance of such multi-hop wireless LANs through detailed simulation studies. Our results show that such multi-hop extensions can significantly improve the wireless access experience (in terms of data throughput, latency, etc.) for clients who enable such mechanisms. More interestingly, when multi-hop extensions are enabled by some of the clients, it also positively impacts the performance at other clients that are completely unaware of such extensions.**

I. Introduction

IEEE 802.11 based wireless LANs (WLANs) are one of the primary enablers of untethered access to the Internet. In this paper we (1) define a *multi-hop* 802.11-based WLAN architecture, (2) demonstrate how such a system can provide significant performance benefits over existing single-hop counterparts, and (3) describe a deployment path that will enable it to seamlessly interoperate with existing WLAN infrastructures.

There are a number of benefits of enabling a multi-hop option for wireless access to the Internet. An obvious advantage of such an architecture is the increase in the wireless coverage area. In this paper we show that even from a data performance point of view there are significant benefits in deploying a wireless multi-hop architecture as an access mechanism to the Internet. For example, our measurements in deployed WLANs indicate that in many cases multi-hop extensions can improve the data throughputs by a factor of two or more.

One way to construct this multi-hop access infrastructure is to use a routing layer based solution. In fact, a number of on-demand routing protocols have been defined to provide network level connectivity between arbitrary pairs of wireless nodes in an ad-hoc wireless network, e.g. DSR [7], AODV [11], TORA [10], ZRP [3], etc. While these protocols can be used construct appropriate multi-hop paths from the wireless clients to the Access Points (APs) of a 802.11 WLAN, in this paper we argue that the benefits of multi-hop paths can be *better realized* by implementing them in the wireless medium access layer.

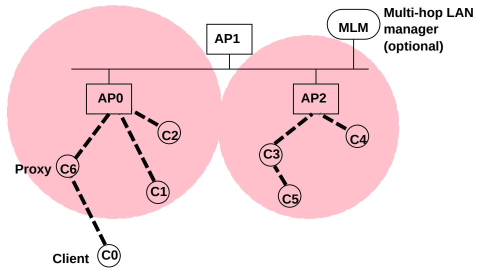

Fig. 1. The multi-hop 802.11 architecture. The circles represent the communication range for the specific APs.

A. Multi-hop Wireless LAN and its advantages

A typical WLAN consists of two different entities — Access Points (APs) and stations (STAs), which we refer to as clients in this paper. A client associates itself with an AP within its *direct* communication range. The set of all such clients for a specific AP is known as the Basic Service Set (BSS) for that AP. A single WLAN can consist of a number of such BSSs, one corresponding to each AP. The APs are connected via a backbone distribution system (DS), which also provides a conduit to the external network. All the BSSs together with the DS are known as the Extended Service Set (ESS). The entire WLAN is identified by a single ESSID.

In Figure 1 we illustrate our proposed multi-hop 802.11 architecture. In this architecture, each client can directly associate itself with an AP in the WLAN. However, the client can also have a multi-hop path, via other clients acting as intermediaries or proxies, to indirectly associate with the AP. In a typical scenario we expect the proxies to be “resource-rich” clients that take data forwarding responsibilities on behalf of resource-depleted clients.

There are a number of benefits of a multi-hop wireless LAN architecture. We discuss them in turn.

Enhanced performance: Some clients in a WLAN are resource-depleted. Consider the case when a specific client (say client C_5 in Figure 1) is low on battery power. The energy required for it to directly communicate with AP_2 is prohibitively expensive. However, the availability of a nearby client that can serve as a proxy (e.g. client C_3) significantly reduces the energy requirements for communication. Therefore the multi-hop path leads to increased lifetime for C_5.

Similarly, consider another scenario where the direct channel between C_5 and AP_2 is very noisy. Therefore, data transmitted on this channel will encounter significant errors and losses. The IEEE 802.11 protocol reacts to such losses by reducing the data rate. The bit error rate on a channel decreases with an increase in signal to noise ratio. Therefore, one way to maintain the higher data rate using the 802.11b protocol is to reduce the error rate on the channel by using a higher transmit power. This high power solution leads to increased interference in the WLAN. For example, transmissions from C_5 may now interfere with data transmissions between AP_0 and its clients, thus reducing the data throughput of the WLAN. In a multi-hop system, C_5 can use a "better-located" client (e.g. C_3) to communicate with the AP. We performed detailed measurements in existing WLANs to study the benefits of a multi-hop approach to clients. Our results indicate that in many such cases clients can leverage a multi-hop path to significantly improve their data throughput. Additionally, the performance improvement of these "resource-depleted" clients also positively impacts the performance of clients in the same WLAN that are not even aware of multi-hop extensions.

Extended wireless coverage: In the usual single-hop WLANs, a client must be located within the coverage area of some AP to receive wireless services. A multi-hop WLAN leverages participating proxies to extend the coverage area, e.g. client C_0 in Figure 1. Such a solution is particularly useful in *handling flash crowds.* If a transient user population moves into an area with no wireless coverage, multi-hop 802.11 can be used to provide immediate wireless services. Obviously the long-term solution to provide wireless connectivity in a popular user location is to add more APs in that area. However, the multi-hop solution is more appropriate to handle transience. This is because it requires no setup or administrative overheads and requires no additional hardware.

Enabling automated re-organization of AP distribution: The goal of a WLAN designer is to ensure that each location in the area is visible to at least one of the APs of the WLAN. Wireless LAN administrators currently use various techniques to monitor the expected performance of WLANs. One of the more popular methods is to perform signal strength measurements at various locations of the coverage area from the nearby APs. Such an approach is tedious and cannot be performed very frequently. As a result, WLAN administrators often do not have accurate radio maps that reflect the existing conditions in the wireless environment. (It is a common experience that new furniture brought into a room affects the channel noise characteristics significantly.)

The multi-hop WLAN presents a new opportunity to enhance the online performance monitoring as experienced by clients. For example, when proxies in a specific location get heavily used, (e.g. due to poor channel conditions in the direct path to the APs) the system can trigger alerts to the LAN administrators to appropriately add or re-distribute the APs in that location. In the proposed multi-hop 802.11 architecture, the proxies provide such information to the Multi-hop LAN Manager (MLM) and the latter is responsible for providing such notifications.

In some of the above examples, e.g. extended wireless coverage, the long term solution is to add more APs to the WLAN. In such cases the multi-hop architecture can be leveraged to (1) provide a short term solution, (2) handle transient situations, e.g. flash crowds, (3) provide performance benefits in cases where re-organization of the WLAN is too expensive, and (4) allow administrators to discover performance problems in the WLAN which can trigger the long-term re-deployment based solutions. In other cases, the multi-hop architecture provides the only logical solution to improve the performance of resource-depleted devices (e.g. a device with low residual battery power).

B. Pitfalls

While there are a number of benefits of the multi-hop architecture, it is important to evaluate some of the potential pitfalls that may arise in this environment.

Increased channel contention: When a packet follows a multi-hop path to an AP, it uses the wireless channel two or more times. This would increase the contention of the channel and potentially allow reduced data throughput for the source as well as other clients in the vicinity. We study the effect of multi-hop paths data throughput using detailed measurements as well as simulations to quantify this effect. The results show that in many cases the data throughput increase due to better (multi-hop) path choices more than compensates for the loss due to channel contention. Our proposed mechanisms take channel contention into account when making such multi-path choices.

Resource consumption at proxies: Packets following multi-hop paths consume resources at the proxies, e.g. battery power, bandwidth, etc. Clearly, there is no incentive for wireless clients to operate in such an altruistic mode. Each client in the WLAN can choose independent policies on when it is willing to serve as a proxy. For example, some users may volunteer their laptop clients when they are powered from an electric outlet, and when the laptops are idle, i.e. not actively generating network traffic. Additionally, it is possible to define incentive based packet forwarding rules in such multi-hop environments as shown in [1].

Security threats: Allowing an intermediary to forward data packets on behalf of a client may potentially open the WLAN to new security threats. We argue that deployment of such multi-hop mechanisms does not add any security problems that current single-hop environments do not already have. We discuss this aspect in Section VI.

C. Incremental deployment

IEEE 802.11 based WLANs are currently widely deployed. Therefore a new multi-hop architecture that requires a change to existing entities (e.g. clients and APs) is not always feasible. Therefore, we explore the potential paths of deployment of multi-hop WLANs that require various degrees of change to existing entities. The proxies are new entities in the system and any client that is capable of being a proxy implements the multi-hop extensions. However, to maintain backward compatibility with existing systems we consider cases where the other entities, i.e. regular clients and APs, are not aware of multi-hop extensions to the WLAN. We consider the four different cases — (1) unaware-AP, unaware-client, (2) unaware-AP, aware-client, (3) aware-AP, unaware-client, and (4) aware-AP, aware-client — and define techniques for implementing a multi-hop 802.11 WLAN for each of these cases. While the basic principles of the protocols in these cases are similar, the mechanisms required to achieve the desired effect vary from case to case.

D. Roadmap

The rest of this paper is organized as follows. In Section II we provide detailed measurement studies on a deployed (single-hop) WLAN to demonstrate the potential benefits of a multi-hop implementation. In Section III we describe the protocols and mechanisms to construct a multi-hop 802.11 WLAN for the four cases mentioned above. In Section IV we present results from our simulation-based experiments that study the performance of the proposed schemes. In Section V we discuss some of the related work, and finally conclude in Section VI.

II. Measurement-based Evaluation

In Section I we identified some of the potential pitfalls of a multi-hop WLAN architecture. In particular, we identified the issue of increased channel contention as a potential disadvantage of multi-hop WLANs. In this section we primarily examine the channel contention effects and their impact on data throughput. Our results indicate that a carefully designed multi-hop WLAN protocol can lead to significant data performance benefits in all cases.

A. Experimental Setup

We performed our experiments on the 4th floor of A.V. Williams building (which hosts the Computer Science Department at the University of Maryland). The map of the floor is shown in Figure 2. In the experiments described in this section, we performed the experiments with respect to a representative AP located at the position marked in the figure. We measured the data throughput achieved by clients using both direct and multi-hop configurations. In both these configurations, the client performed a reliable data transfer (using TCP) of 51.12 MB of data to a sink, located in the same wired subnet as the AP. (This translates to 100,000 IP packets of size 536 bytes each, generated at the source.) In each experiment we measured the data transfer latency as observed at the application layer.

For the multi-hop measurements, we did not implement the full version of our proposed protocol (to be described in Section III). Instead we emulated the multi-hop link layer mechanisms using statically assigned IP addresses and routes, as shown in Figure 3. In this setup, the proxy device used two separate wireless cards — one to associate with the AP and operate in the managed mode, and the other to interact with the source client and operation in the ad-hoc mode. Due to physical constraints of the PCMCIA slots of laptops, we found it convenient to use two laptops, connected by 100 Mbps Ethernet, to operate as a single proxy as shown in the figure.

Note that such an arrangement is actually disadvantageous to the multi-hop experiment. Unlike multi-hop link layer mechanisms, the data packets encounter additional delay due to network layer processing. More importantly, this setup also leads to an additional latency due to data transfer between laptops A and B via Ethernet.

In these experiments we used IBM Thinkpad laptops running Linux with kernel version 2.4.19, equipped with Orinoco Silver PC cards.

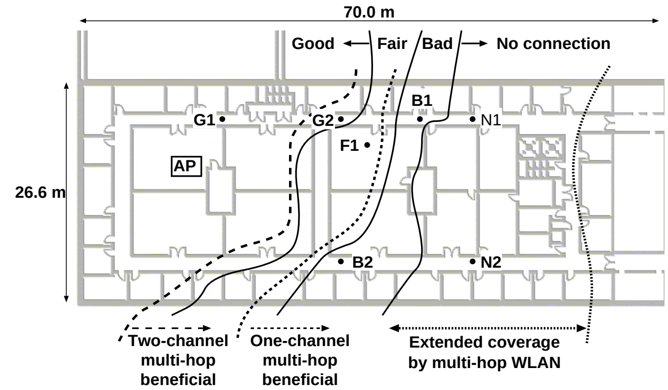

Fig. 2. Potential data throughput improvement by using multi-hop extensions to the currently deployed WLAN in the 4th floor of the A.V. Williams building. The "Good," "Fair," "Bad," and "No Connection" marks the performance the single-hop WLAN. The multi-hop benefits shown in this figure are obtained using two hop paths.

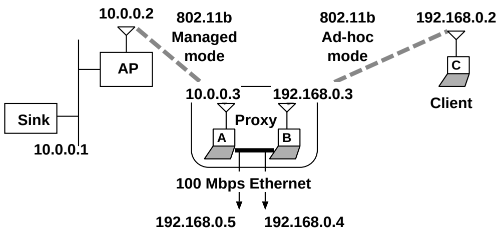

Fig. 3. The experimental setup to measure performance of a multi-hop WLAN.

To emulate the existing environment in the A.V. Williams Building, we informally surveyed laptop use habits of people in the different rooms on the 4th floor. We found that many laptop users, while at work, plug in their laptops to an electric power outlet [1]. For multi-hop paths, we only considered these locations to be candidates for proxies.

The IEEE 802.11 standard allows multiple channels to be used simultaneously. In the multi-hop experiments there are two wireless links, one from the source to the proxy, and the other from the proxy to the AP. We experimented with using the same channel as well as two independent channels for these two links and compare the performance of both these scenarios with the single-hop case. In an actual deployment whether multiple channels can be used depends on specific network conditions, administrative decisions, and other such factors.

B. Results

We performed this measurement study throughout the month of June 2003, in which we observed the data throughput of more than 30 sample positions. Not surprisingly, we found that the wireless data throughput fluctuated between different measurements. However, it was easy to identify a consistent ordering among the data throughput achieved at different locations.

In Figure 2 we present an approximate wireless coverage and direct-hop data throughput from different locations to a representative AP (marked in the figure). In the area marked "Good" users can get data throughput of more than 4 Mbps. (Although the maximum data rate in the 802.11b WLAN is 11 Mbps, it is not possible to achieve an 11 Mbps data rate due to overheads of RTS/CTS/ACK frames, channel contention effects, etc.) In the area marked "Fair" the throughput varies between 1 and 4 Mbps. In the area marked "Bad" the throughput is less than 1 Mbps, and finally the users lose connectivity with the AP in the area so marked [2].

In Figure 2, the two dotted lines on the left identify the regions where the emulated multi-hop wireless paths lead to improved performance over the existing infrastructure (e.g. >2 times better bandwidth in the "bad" region.) The two-channel multi-hop paths are useful even when users are located within the good wireless coverage region (e.g. location G2). It provides considerable performance improvement for users in "fair" and "bad" areas (e.g. F1, B1) as well as in "no connection" area. The single-channel scenario is expected to have worse performance than the two-channel case due to greater contention effects in the single channel. The results indicate that in spite of these effects, single-channel multi-hop wireless paths provide significantly improved performance in the areas marked "Bad" and "No connection" (e.g. B1, N1).

Finally we can observe that the multi-hop WLAN considerably extends coverage, as shown in the figure.

In Table I we tabulate some of the representative measurements at selected locations on the floor.

Position	Direct	Multi-hop	
		One-channel	Two-channel
G1	4.94	2.42	4.56
G2	4.12	2.58	4.50
F1	2.46	2.50	4.60
B1	0.84	2.26	4.30
B2	0.83	2.37	4.24
N1	-	1.83	3.77
N2	-	2.50	2.96

TABLE I
ACTUAL THROUGHPUT VALUES (MBPS) MEASURED AT REPRESENTATIVE POINTS

[1] There were other users who did not use the electric outlet by default, except to re-charge the laptop batteries.

[2] Note that these lines are an approximation, and we do not claim they are exact boundaries.

Using three hops: We also conducted some experiments with multi-hop paths with three hop paths. We observed that the bandwidth achieved in these experiments were similar or marginally worse than the two hop measurements (e.g. at location N1 it was 1.7 Mbps for a single channel experiment and 3.79 Mbps when three channels were used). Thus, the additional benefits of using three or more hops within the typical coverage areas of APs are marginal.

Overall, we believe that these experiments serve as evidence that multi-hop WLANs can be useful to clients in many cases.

III. Multi-hop WLAN Architecture and Deployment

We define three important constructs necessary to implement a multi-hop WLAN. We call them *composition, relaxation,* and *replacement* of proxies (Figure 4). In the examples in the figure we use three or more hops for the multi-hop paths. The protocol mechanisms generalize to an arbitrary number of hops. However, our measurements (Section II), indicate that in most typical scenarios, two hop paths are sufficient for performance benefits, and benefits of additional hops are marginal.

Let us consider any general metric, $\mathcal{M}$, e.g. bandwidth, loss rate, latency, energy consumption, etc. Composition defines the protocol mechanisms to add a proxy on the path from the client to the AP (Panel 1 $\rightarrow$ Panel 0). Such an addition is performed if and only if the path improves with respect to the given metric, $\mathcal{M}$, i.e. in the figure

$$\mathcal{M}_{X,Z} \bigoplus \mathcal{M}_{Z,Y} \text{ better than } \mathcal{M}_{X,Y}$$

(We use the $\bigoplus$ operator to denote composition). The definition of "better than" depends on the specific metric.

Replacement describes mechanisms where one proxy replaces another (Panel 1 $\rightarrow$ Panel 2) and leads to an improvement of the path quality with respect to $\mathcal{M}$. In the figure this implies that

$$\mathcal{M}_{X,Z} \bigoplus \mathcal{M}_{Z,AP} \text{ better than } \mathcal{M}_{X,Y} \bigoplus \mathcal{M}_{Y,AP}$$

Note that the proxy Z may be associated with a different AP within the same WLAN.

Finally, relaxation defines protocol mechanisms to remove a proxy on the path between the client and the AP (Panel 3 $\rightarrow$ Panel 4), to improve the path quality. In the figure this requires that

$$\mathcal{M}_{X,Z} \text{ better than } \mathcal{M}_{X,Y} \bigoplus \mathcal{M}_{Y,Z}$$

We describe the implementation of the constructs with respect to an example metric — bandwidth available on the path from the client to the AP.

Note that there are two key components that determine the bandwidth of a wireless path: (1) noise on the wireless channel, and (2) contention with other clients. As the noise on the channel increases, the 802.11b implementations on the wireless cards reduce the data rate, thus increasing the path latency and reducing the path bandwidth. Similarly as collisions occur on the wireless channel, the 802.11b clients perform contention resolution which leads to reduction in bandwidth and increase in latency.

In order to compute multi-hop paths with good bandwidth or latency performance, we need to estimate these metrics for individual wireless hops. In the appendix, we define a simple heuristic to compute these metrics through passive observations. There are two advantages of this proposed heuristic: (1) it requires no active measurement traffic and hence does not increase the contention of the data channel, and (2) an endpoint of a wireless link or any external entity with the capability to snoop packets can use this technique to estimate the the metrics for that link.

We have considered four different scenarios for deployment of a multi-hop WLAN. We now describe the multi-hop architecture that implements the composition, relaxation, and replacement constructs for improved performance in these scenarios.

A. Aware client

We independently consider the path from the client to the AP (forward path) and the path from the AP to the client (return path).

We use the following notation. For any link, $X \rightarrow Y$, let $b_{X,Y}$ denote the bandwidth on that link. For a client, C, we represent the end-to-end bandwidth on its single or multi-hop path to the AP, by b_C. Thus $b_C = \min\{b_{X,Y}\}$ over all $X \rightarrow Y$ hops on this path. b_C is our objective of maximization.

Forward Path: Let us assume that a client, C, currently uses some forward path (either direct or multi-hop) to an AP, where the client is the source of traffic. Consider a specific hop on this path, $X \rightarrow Y$ as shown in Figure 4. (If the client is using a single-hop path, then X is C, and Y is AP.) For each such $X \rightarrow Y$ hop, X computes the bandwidth available on that hop, $b_{X,Y}$, using the technique presented in the appendix. Each node, Y, on the path, periodically advertises its end-to-end bandwidth to the AP, b_Y with a low frequency (e.g. once every 20 seconds). Therefore, X can calculate b_X as $\min\{b_{X,Y}, b_Y\}$.

C also needs to periodically advertise the value of b_C along its path to the AP when it uses a multi-hop path to the AP. The periodic advertisement can be done either using local broadcast of additional packet types at a low fre-

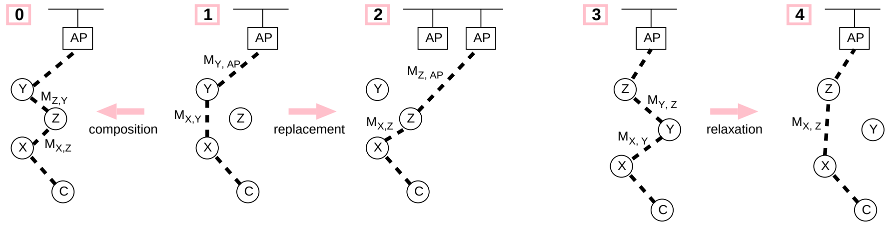

Fig. 4. The Composition, Replacement, and Relaxation constructs. C is a client. X, Y, and Z are proxies.

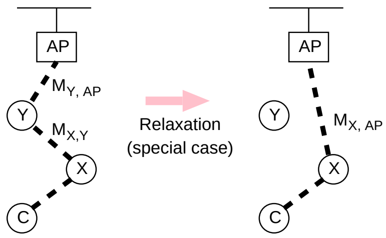

Fig. 5. Relaxation of the last proxy on a multi-hop path.

quency, or piggy-backing onto data packets along a multi-hop path. In the latter case, if the AP is unaware of multi-hop extensions, the last proxy on the multi-hop path needs to remove this field from the data packets before forwarding it to the AP. In this way, any proxy in the vicinity will be able to snoop this information.

Consider another proxy, Z, that is within direct communication range of X. Z receives the bandwidth advertisement, b_C, on this path. C has a better path to an AP through Z rather than its existing path, if

$$\{\min(b_{X,Z}, b_Z) - b_C\} > b_{thresh} \qquad (1)$$

where b_{thresh} is the bandwidth advantage threshold. Note that Z is also a regular client in the system, and therefore, computes and maintains the available bandwidth, b_Z, to its AP. Z estimates the value of $b_{X,Z}$ using the passive estimation technique described in the appendix.

If using Inequality 1, Z detects that the path $C \rightarrow \ldots X \rightarrow Z \rightarrow \ldots \rightarrow AP$ has higher bandwidth, it sends a *ForwardProxyBid* message to X. This message includes the values of $b_{X,Z}$ and b_Z. If X receives multiple such *ForwardProxyBid* messages, it chooses a proxy that leads to the best bandwidth improvement. X sends a *ForwardProxyAccept* message to the chosen proxy and starts forwarding data packets to Z.

If the path from Z to the AP has Y as its first hop, then this operation would be a *Composition* (shown in Panel 1 $\rightarrow$ Panel 0, Figure 4). If the first hop from Z is some node other than Y, this would be a *Replacement* operation (Panel 1 $\rightarrow$ Panel 2, Figure 4). Finally, if in the original multi-hop path from the client C to the AP, Z was the next hop to Y, then the operation describe above is equivalent to a *Relaxation* (Panel 3 $\rightarrow$ Panel 4, Figure 4).

The proxy state is soft. Therefore, in absence of data packets, X is required to periodically refresh the state at Y by sending gratuitous *ForwardProxyAccept* messages. Y can revoke proxy services to its previous hop, X, by using a *ForwardProxyRevoke* message. This can be invoked due to many reasons. For example, the laptop serving as the proxy is unplugged from the electric outlet and, hence, is no longer willing to serve as a proxy. Alternatively it can also be that the proxy is dissociated from its AP. As a final fallback mechanism, X can also detect the failure of Y, when it fails to acknowledge a threshold number of consecutive RTS packets forwarded to it (in the RTS/CTS access method).

Special case for unaware-AP: All the above operations work independent of whether the AP is aware or unaware of multi-hop extensions to the MAC protocol, except one special case. This special case arises for the unaware-AP case, when the original multi-hop client path was $C \rightarrow \ldots \rightarrow X \rightarrow Y \rightarrow AP$, and a relaxation operation is required to eliminate the last proxy, Y, from the path (Figure 5).

Note that in the aware-AP case, we implement the same *ForwardProxyBid* mechanism in the AP that leads to this relaxation operation. We call such a relaxation step *Relaxation assisted-by Access Point (RAP)*. However, if the AP is unaware, such an operation is not feasible. An unaware-AP will not attempt to evaluate the bandwidth of the $X \rightarrow AP$ link, nor send a *ForwardProxyBid* message to eliminate Y from the path. Therefore to enable the elimination the last proxy from a multi-hop path, if and when necessary, we need to define additional mechanisms for the unaware-AP case.

In this case, X actively probe the quality of the direct path between itself and the AP. In this active probe technique, X periodically sends a *NULL* frame to the AP. The *NULL* frame is a special frame which is automatically dropped by the AP and therefore does not add any extra load on the Distribution System. However, like any data packet, the AP will perform the four-way handshake to receive this packet (i.e. RTS-CTS-NULL-ACK). Using this low frequency stream of *NULL* frames, X estimates two parameters — (1) the packet error rate, p, on this link, and (2) the latency of the four-way handshake for a successful data transfer across the link, τ. Estimation of these two parameters is sufficient for X to infer $b_{X,AP}$ using the technique described in the appendix. If $b_{X,AP} - b_X > b_{thresh}$ then X directly eliminates Y from the multi-hop path, by sending a *ForwardProxyRevoke* message.

MAC Address Translation: Consider a forward multi-hop path from the client C to the AP, $C \rightarrow P \rightarrow AP$, where P is the proxy. When P forwards data frames to the AP, on behalf of the client, it uses its own MAC address as the source address for those data frames [3]. (Alternatively P can use a specially chosen independent MAC address when forwarding packet for each specific client.) The proxy therefore performs *MAC-level Address Translation(MAT)* for data frames transmitted by C. This is true for a multi-hop return path as well, as described next.

Return Path: On the return path, i.e. from the AP to the client, there are multiple choices. If the client is within direct communication range of the AP, then the AP can directly forward data to the client. Alternatively the data frames can follow the same multi-hop path in reverse that is used in the forward direction. Finally, it is possible that the AP to client path is another multi-hop path, independent of the forward path. We focus on the first two choices in this paper and briefly describe the third option later.

In general, the path from the AP to the client should use the direct single hop path if one is available. This is because the AP is typically a resource-rich device and can transmit with adequate power to tide over moderate noise levels in the channel. However, it is possible that the client is outside the communication range of the AP. In such a scenario the same multi-hop path, as used in the forward direction, can be used in reverse for the return path. In our proposed protocol, we allow the aware client to choose between these two alternatives.

The client, C, computes the bandwidth on the direct single-hop path (b_{direct}) from the AP to itself using Equation 5. Consider the case when the current forward path is multi-hop. In such a scenario the client maintains an estimate for the bandwidth on the reverse direction on this multi-hop path (b_{multi}). It can infer the bandwidth of the link to the immediate proxy using the same inference equation above. C also needs to know the bandwidth of the remaining path, which is included in a periodic proxy advertisement by the immediate next hop described above. As a result, a periodic proxy advertisement should contain bandwidth information of both forward path to the AP and return path from the AP.

If $(b_{multi} - b_{direct})$ exceeds the bandwidth advantage threshold, b_{thresh}, then C would switch to the multi-hop path for the reverse traffic. In such a case, C sends a *ReverseProxyRequest* message to its first-hop proxy, X, in the forward multi-hop path (Panel 4, Figure 4). X may respond with a *ReverseProxyAccept* or a *ReverseProxyReject* message, as appropriate. The return path proxy request propagates from X along the forward multi-hop path to set up the reverse proxy state in the subsequent hops. The proxy state in the return path is also soft, and is periodically refreshed by the client using the *ReverseProxyRequest* message. The *ReverseProxyRequest* message carries the IP address of the C's MAC interface. This is necessary to appropriately handle the ARP *who-has* message.

The reverse proxy path is activated by the last proxy, Z, on the path, (i.e. closest to the AP) using ARP mechanisms. However, C continues to stay associated with its AP and continually estimates the bandwidth on the direct hop from the AP to itself. This estimation can be performed by snooping the channel. On detecting an improvement of this direct hop path, it reverts back to this path. C sends a *ReverseProxyRevoke* message to its first-hop proxy to effect this change. Alternatively C stops refreshing the proxy state on the return path and this state at the proxies time out.

In addition to the two methods described above, it is natural to consider a return path independent of a forward path. Even though this alternative might be able to provide the most efficient solution with respect to the metric of interest (e.g. bandwidth, latency, etc.) computing the return path would require additional advertisement in the WLAN. It is because client C needs to know the bandwidth of the return path, which requires periodic advertisements of reverse path performance by all proxies. For example, in Panel 4, Figure 4, suppose that $AP \rightarrow Y \rightarrow C$ is best as the return path, while the forward path is $C \rightarrow X \rightarrow Z \rightarrow AP$. Although C can infer the link bandwidth from Y, it cannot know the link bandwidth from AP to Y. As a result, Y needs to advertise the bandwidth of the link from AP to Y [4]. Note that in case of a forward path, only proxies on the path periodically advertise their end-to-end bandwidth of the path to the AP. For a return path, on the other hand, all proxies are potentially required to advertise, which makes this alternative expensive.

MAC Address Resolution: First consider the direct single hop return path. In this case, when the AP sends an ARP request for C's IP address, C sends the ARP response with its own MAC address. Hence the AP transmits all data packets addressed to C using C's MAC address as the destination.

Next consider the case when the return traffic uses the multi-hop path. In this case, when the AP sends an ARP

[3] If P spoofs the MAC address of C, it can lead to ambiguities and incorrect operation at the MAC layer.

[4] We can have Y infer link bandwidths and bid for a return path. Nevertheless, it needs to be informed of the link bandwidth from Y to C, which also requires message exchange between Y and C

request for C's IP address, the last proxy (Z in Panel 4, Figure 4) on the multi-hop path sends proxied ARP responses with its own MAC address. Subsequently all traffic destined for C will be forwarded by the AP to Z's MAC address. Whenever C switches between the two paths, an explicit ARP response is sent to appropriately update the cache entry at the AP.

All interaction between the clients and proxies in the aware-client case, takes place using the *Ad-hoc* mode of the Distributed Coordination Function (DCF) of 802.11b operation.

B. Unaware client

We briefly summarize the implementation path for a multi-hop WLAN for the unaware client scenarios. In these scenarios, since the clients are unaware of multi-hop extensions, they will not associate with any entity other than APs with the designated ESSID. Therefore the key problem in this scenario is to compose a proxy on the path from the client to the AP.

In this scenario, a multi-hop path can only be constructed if the proxies operate as APs in the WLANs. All these active proxies (acting as APs) need to interact with the actual APs in the WLAN to form a Wireless Distribution System (WDS). Some implementations of WDS are already commercially available today, e.g. Orinoco AP-2000 from Agere Systems [5] and WX-1520 from SparkLAN [6].

If all possible proxies act as APs, then the number of APs in the system can become very large. Therefore unlike existing implementations of WDS, the proxies in our proposed system emulates AP functionality on-demand, i.e. only when it is needed by resource-depleted clients.

Consider a client C that is directly associated with an actual AP (which we call wired AP in this description). A proxy, X, enables its AP functionality when it detects that the $C \rightarrow X \rightarrow AP$ has a higher bandwidth than the direct path. Like before, X maintains the estimate of bandwidth from itself to its wired AP, i.e. b_X and computes the direct bandwidth from C to itself (i.e. $b_{C,X}$). However, unlike the aware client case, C does not periodically advertise b_C. Hence, X estimates this value by snooping the wireless traffic sent by C to the AP.

Let us first consider the *Composition* operation in case of an unaware AP. Low bandwidth to client is typically due to two reasons: (1) poor channel conditions, i.e. high noise in the wireless medium on the path from C to AP, or (2) high network traffic which leads to significant channel contention. 802.11b clients respond to both these scenarios by trying to identify a "better" AP and associating it. If C attempts such a re-association, it will be able to locate the proxy X operating as an AP. Of course, in this unaware client scenario, there is no guarantee that the client will try to locate a better AP. Hence, it is not possible to guarantee bandwidth-optimal paths for the clients.

[5] See http://www.agere.com
[6] See http://www.sparklan.com

Client	Access Point: Unaware	Access Point: Aware
Unaware	WDS	WDS + RAP/ CAP
Aware	MAT	MAT + RAP

TABLE II

MECHANISMS REQUIRED TO DEPLOY MULTI-HOP WLANS FOR THE FOUR DIFFERENT SCENARIOS.

If the APs are aware of multi-hop extensions, it can actively participate in a *Composition* operation as follows. X, on detecting a better path for C, can optionally send a *ClientDissociateRequest* to the AP. The AP on receiving this message will explicitly dissociate C. This will force C to locate an alternate AP, and in the process will find proxy X. We call this process *Composition assisted-by Access Point(CAP)*. Note that it is possible that there are (unnecessarily) multiple proxies for one client in the unaware AP case, while an aware AP can designate only one proxy for a client and thus avoid such inefficiency.

In the unaware client scenarios, the *Relaxation* step is also hard to guarantee. For the aware AP case, we rely on the AP to initiate the relaxation step (RAP). When the AP detects that the direct path has better bandwidth than the composed multi-hop path, it sends a *ClientDissociateRequest* to the proxy, X, which has been emulating AP functionality. The proxy X subsequently dissociates the client, and the latter eventually re-associates directly with the wired AP. In the unaware AP case, relaxation is possible only if the channel conditions on the path between the client and the proxy becomes bad, and the client automatically attempts to locate a better AP for itself. Therefore, to force the client to locate better alternate and possibly direct paths, the proxy should periodically dissociate the client, forcing the latter to locate a better AP. This is the only possible mechanism that can enable path relaxation when both a client and an AP are unaware.

When a proxy is eliminated from a multi-hop path through the relaxation process, and it is not serving as a proxy for any other client, it stops operating as a wireless AP and reverts back to the regular client mode.

We summarize the mechanisms to implement all the four scenarios in Table II.

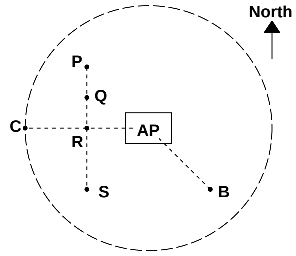

Fig. 6. Location of clients and AP in the some of the experiments. The radius of the circle is 250 m.

IV. Simulation Studies

To evaluate the performance of our proposed protocol in the aware-clients case, we performed detailed simulations using the *ns-2* network simulator [7]. Apart from static scenarios, we have also performed detailed experiments that involve mobile clients. In this section we focus on the impact of our proposed techniques for the bandwidth and latency metrics for the aware client scenarios. Due to space constraints, we only present results from a representative set of our experiments.

A. Simulated Environment

In our experiments, we used *ftp* traffic to model reliable TCP-based data transfer between sources and destinations. These data sources were typically mobile clients that sent traffic through APs to a wired sink node. Since our study focussed on the data performance of the WLAN, we assumed that the link between the AP and the wired sink is not a bandwidth bottleneck. Typical simulation durations were between 300 to 600 seconds. In this paper, we primarily present results for multi-hop extensions where all communication used a single channel. We present a brief summary of results for the two-channel experiments.

We model the environment as a noisy channel. We assume that the underlying physical layer uses the Binary Phase Shift Keying (BPSK) modulation scheme in which the bit error rate experienced on the channel is given by $p_b = 0.5 \times erfc(\sqrt{\frac{P_r}{N \times f}})$ where P_r is received power, N is the noise spectral density, f is transmission bit rate, and *erfc* is the complementary error function. We also assume that signal strength is reduced proportionally to the square of distance. Therefore the quality of the channel depends on the noise in the environment and the distance between the endpoints. In our simulation experiments, the clients were distributed in an area of upto 250 m away from the AP. In these experiments we assumed that all clients and the AP are within the transmission range of each other.

[7]Available at: http://www.isi.edu/nsnam/ns

Most wireless cards incorporate a mechanism called Automatic Rate Fallback (ARF) [8] to handle noisy channel conditions. In this mechanism, each node initially uses a data transmission rate of 11 Mbps. On detecting repeated data transmission failures, it reduces its transmission data rate to 5.5 Mbps, 2 Mbps, and 1 Mbps successively. Later, if the node receives ACKs for several successive data packets, it increases its transmission bandwidth until the bandwidth reaches 11 Mbps. Note that the IEEE 802.11 standard does not specify any ARF algorithm, and implementations of this mechanism varies between different card vendors. We incorporated this ARF mechanism into the *ns-2* simulator, and our implementation was based on the description presented in [8].

We explain the experiment scenarios using Figure 6.

In the first experiment, an *ftp* sender is placed at C (Figure 6). We consider two mobility cases for a proxy-capable client: (1) it is initially co-located with the AP, and moves towards C (westbound), starting at time 25 seconds, at the speed of 1 m/s. It reaches C at 275 seconds. (2) it is initially at P, and moves towards S (southbound) with the same speed. Both these scenarios capture how the location of a proxy affects bandwidth performance at the client.

Figure 7 illustrates the achieved bandwidth averaged over 20 second intervals for these two cases, and compares it with the no multi-hop scenario. In absence of multi-hop extensions, the client achieves a data throughput of about 0.5 Mbps. The data throughput achieved in the multi-hop scenario depends on the location of the proxy. For example, when the westbound client is close to the AP, it is not useful as a proxy to the sender. Therefore, the sender continues to use the direct path to the AP. At time 75 seconds, the westbound client has moved sufficiently away from the AP, and the sender starts using it as a proxy. Note that the bit error rate is higher for a channel with larger distance. Hence the best data performance is observed when the proxy is located at R (mid-way between the client and the AP) at time 150 seconds. As expected, we observe that the proxy-enabled client moving along the Y-axis is better located for bandwidth performance at C.

Next, we show that the proposed protocol adapts its multi-hop path to a better proxy, when one becomes available. In this experiment, the sender is at C as before. There are two proxy-enabled clients, at Q and at R, respectively. Furthermore, the client at Q is enabled to act as a proxy after 50 seconds from the start of the simulation. The other client (at R) is enabled to act as a proxy after 150 seconds. (We can imagine that these two proxy-enabled clients are plugged into the power source and become willing to serve as proxies from those respective

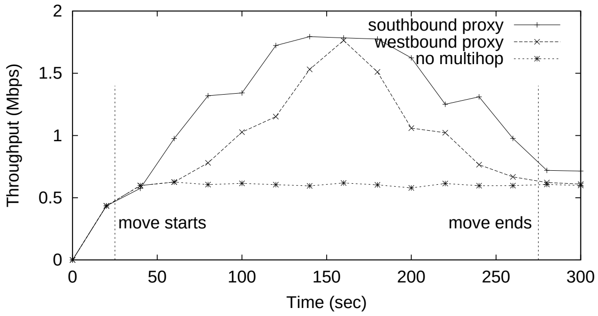

Fig. 7. *Bandwidth benefits of multi-hop extensions for a single sender.* The sender is at C in Figure 6. The westbound proxy-enabled client moves from AP to C The southbound proxy-enabled client moves from P to S.

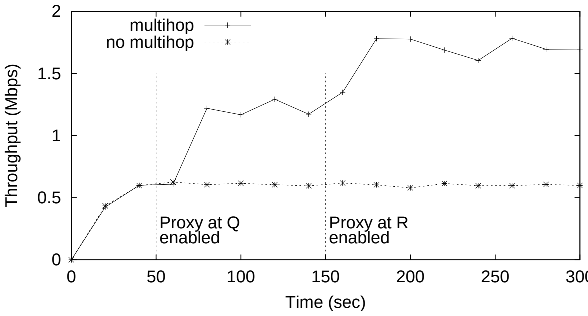

Fig. 8. *Adaptation of multi-hop path using the* Replacement *operation.* The sender is at C in Figure 6, and two proxy-enabled clients are at Q and R, respectively. The two upward transitions in bandwidth, corresponds to the adoption of each proxy in the multi-hop path.

time instants.)

In Figure 8 we present the results from this experiment. The sender starts to use the client at Q as a proxy starting at around 70 seconds. This corresponds to an increase in the bandwidth in the plot (from 0.5 Mbps to 1.3 Mbps). Subsequently, when R is available, it is evaluated to be a better proxy. R sends an appropriate *ForwardProxyBid* which is accepted by the sender in a *Replacement* operation. This happens at time 165 seconds and the bandwidth increases to about 1.8 Mbps.

B. Impact on other sources

We now examine the impact of such multi-hop paths on other sources. Intuitively it appears that a source using a multi-hop path incurs a higher channel contention in the common wireless medium and adversely affects the performance of other sources. However, in these set of experiments we demonstrate that when sources with poor bandwidth to the AP use a multi-hop path instead of the direct path, it positively impacts the performance of other data sources sharing the same wireless medium.

We first consider a scenario with two senders, located at B ("near" sender) and C ("far" sender) respectively (in Figure 6). At time 200 seconds, a proxy-enabled client is activated at location R. At time 400 seconds, the far client starts to move eastbound from C (to R) at the speed of 2 m/s. We examine the bandwidth and latency experienced by the two clients in Figures 9 and 10 respectively.

In the first 200 seconds, both the clients gets about 0.5 Mbps data throughput on the channel (Figure 9). Note that the far client experienced higher noise than the near client, and therefore due to ARF mechanisms, typically uses a lower data rate (1 Mbps) than the near client (which often can use 11 Mbps). Consequently when the far client gets access to the channel, it occupies the channel for a longer time duration than the near client to transmit the data packet of the same size. This is because it transmits the a data frame at a lower data rate. Although the near client transmits at a higher data rate, the far client gets a larger time share of the channel, effectively canceling out the benefits of the higher data rate of the near client. Similar observations of 802.11 WLAN behavior were made in [4].

C. Impact on source using multi-hop extensions

Now we observe how multi-hop extensions used by the far client positively impacts the near client. Note that the near client itself does not use multi-hop extensions. At time 200 seconds the proxy-enabled client is activated at R and the far client starts using this proxy to enhance its own bandwidth. We can observe in Figure 9 that simultaneously, the bandwidth of the near client also improves. This can be explained as follows. With the availability of the proxy, the far client is able to use higher data rates, and consequently reduces the time occupancy of the channel. Consequently the near client is able to occupy the channel for a higher proportion. This leads to its improved data throughput. In Figure 9 we can see that the availability of the proxy-enabled client increases the aggregate data throughput (line marked 'sum') from 1.2 Mbps to about 2.05 Mbps. The use of multi-hop paths by the far client also positively impacts the end-to-end latency experienced by both the clients (Figure 10). When the far client starts using the proxy, the latency of the two clients drop from 80 and 60 ms respectively to about 33 ms for each of them.

Finally, as the far client starts to move towards the AP (at time 400 seconds), the noise on its the direct path to the AP reduces. When it reaches location R, the direct path is obviously more efficient than the multi-hop path. It switches back to a direct single-hop path to the AP, and we observe another increase in aggregate bandwidth for the two clients (Figure 9).

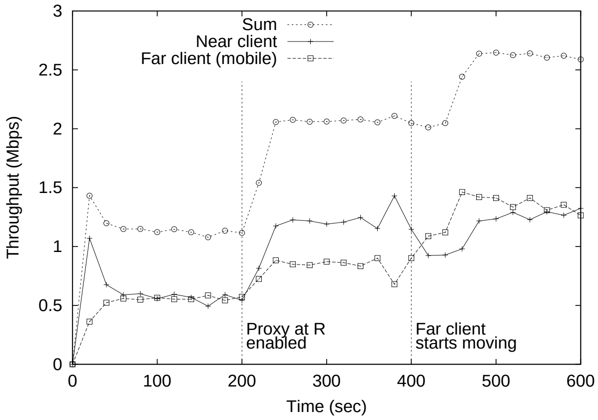

Fig. 9. *Impact of multi-hop extensions on bandwidth at other senders.* Two senders are located at B (near client) and C (far client) in Figure 6. A proxy-enabled client (located at R) is activated at time 200 seconds. At time 400 seconds, the far client starts moving towards R at the speed of 2 m/s.

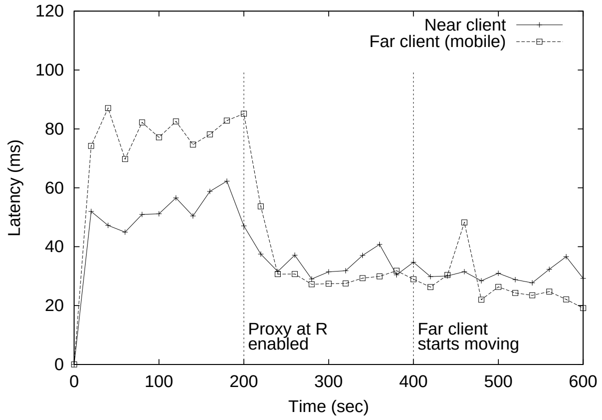

Fig. 10. *Impact of multi-hop extensions on latency at other senders.* This is the latency plot corresponding to Figure 9.

	No multi-hop	Multi-hop 1-channel		Multi-hop 2-channels	
Client	Mbps	Mbps	Ratio	Mbps	Ratio
Direct	0.28 (0.02)	0.45 (0.07)	60.7%	0.48 (0.04)	71.4%
Proxied	0.32 (0.01)	0.37 (0.03)	15.6%	0.49 (0.04)	53.1%
All	0.30 (0.01)	0.41 (0.05)	36.7%	0.48 (0.04)	60.0%

TABLE III

PERFORMANCE IMPROVEMENT OF MULTI-HOP EXTENSIONS IN FOR DIRECT, PROXIED, AND ALL CLIENTS FOR BOTH 1-CHANNEL AND 2-CHANNEL CASES. NUMBERS IN PARENTHESIS INDICATE STANDARD DEVIATIONS.

Finally we performed experiments with a larger number of wireless clients associated with an AP, and the impact of multi-hop extensions in such a scenario. In this paper we report the result of one such set of experiments. In these experiments there were 20 wireless clients randomly distributed around an AP. Five of these clients were *ftp* sources. We classify these sources into two groups — those that leveraged a multi-hop path ("proxied"), and those for which the direct hop path provided good bandwidth ("direct"). In Table III we present a summary of the bandwidth received by all these clients. All the values are averaged over 50 runs of the simulations.

Multi-hop extensions lead to better bandwidth performance for both direct as well as proxied clients. For the single channel case the improvements are 61% and 16% for direct and proxied clients respectively. For the two-channel case, they are 71% and 53% respectively. Note that the clients close to the AP use direct paths. Their data performance were significantly impacted by the distant clients in the single-hop WLAN. The distant clients used proxied paths in the multi-hop WLAN environment and allowed the near clients to significantly improve their path bandwidths.

Control Overheads: The extra control overheads due to the multi-hop extensions was marginal. This is because most of the inferencing was done using passive measurement techniques. In all our experiments, the extra control traffic was < 1 packet per second.

V. RELATED WORK

Multi-hop wireless networks have received significant attention over the last two decades. The main goal of work in this area has been to define auto-configuration mechanisms to organize a set of wireless device into an ad-hoc network. Defining efficient routing techniques for such environments is one of the challenges that have been well addressed in prior literature [7], [11], [10], [3], [6]. As briefly discussed in Section I, these ad-hoc routing solutions can be leveraged to construct a multi-hop wireless access infrastructure. We, however, believe that the benefits of a multi-hop wireless access infrastructure can be better realized when implemented at the wireless medium access layer due to the following reasons. (1) As we demonstrate in this paper, multi-hop wireless paths can lead to better data performance by closely interacting with with MAC and physical layer properties (e.g. contention on the wireless medium, error characteristics of the channel, etc.) to gain significant performance benefits. These interactions can be best implemented at the MAC layer. (2) In most popular wireless environments (e.g. office buildings, homes, and WiFi hotspots), wireless clients typically need mechanisms to access the wired infrastructure. Consequently, the goal of the access infrastructure is to construct appropriate (single-hop or multi-hop) paths to the nearest AP of a WLAN. A full routing protocol that allows flexible routing between arbitrary pairs of nodes is not necessary for such purposes. Note that some of the proposed route construction mechanisms (e.g. network-wide flooding to locate destination nodes) are based on arbitrary separation between the source and the destination. In contrast, th clients in a WLAN are in a much more limited region, where typically the clients are in direct communication range of the APs. In fact, as our experimental results show, most data performance benefits are gained by using short (one or two-hop) paths to between the clients and the APs.

Lin and Hsu [9] had defined multi-hop cellular as a new architecture for wireless communication. They examine the general principles of using multi-hop paths to base stations in cellular networks. Based on useful but simplifying assumptions (e.g. static configurations, centralized routing table construction at all nodes based on an all-pair shortest path algorithm, etc.) they demonstrate that such a multi-hop architecture is beneficial in improving data throughputs of cellular architectures. In contrast, our work significantly builds on these general observations made in [9]. We propose multi-hop extensions at the *MAC-layer,* define detailed protocol mechanisms for interoperability with existing IEEE 802.11b standards, and present detailed performance studies through actual measurements as well as simulations involving both static and mobile scenarios.

Hseih and Sivakumar [5] presents performance comparisons of conventional cellular networks with ad-hoc wireless networks, and briefly introduces a hybrid network model that switches between a purely cellular structure and ad-hoc routing mechanisms. The base station of the cell is responsible for making the switching decision. In their proposed scheme, at any instant, all wireless nodes operate in the same mode, i.e. only cellular or only ad-hoc wireless based. The base station uses a centralized algorithm to compute all routes in the ad-hoc wireless based mode and disseminates this information to the wireless nodes. The route computation requires accurate location information of each wireless node (e.g. from GPS). Therefore, such a mechanism may be practical in outdoor wireless cellular environments, but is not currently feasible in indoor WLANs.

Ben Salem et. al. [1] have examined the construction of a multi-hop wireless packet forwarding technique in the context of cellular networks. The goal of their work was to define incentive-based mechanisms such that cellular users provide multi-hop forwarding services for each other. Therefore the techniques developed in [1] define a solution to an useful and complementary problem (in the context of cellular networks) to what we address in this paper. Our work can leverage such an approach to provide incentives to mobile clients to serve as proxies in a multi-hop WLAN.

Dousse et.al. [2] has recently proposed a hybrid network to improve the connectivity of an ad-hoc network. In their definition, a hybrid network is an ad-hoc network which is interconnected by a sparse set of wired backbone nodes. Liu et. al. [5] subsequently analyzed the capacity of such hybrid networks and identified the scaling behavior of capacity with increasing number of wireless and wired nodes.

VI. Conclusions

In this paper we have defined a multi-hop WLAN architecture and quantified its benefits. We also define deployment paths for these multi-hop extensions that can interoperate with existing deployed WLANs. Through detailed measurements and simulation studies we show that the proposed mechanisms benefit all WLAN users: those that use the proposed multi-hop extensions, as well as those who do not adopt these extensions.

While multi-hop WLANs have significant benefits, enabling multi-hop paths from clients to APs involving untrusted proxies can lead to potential security threats, e.g. a malicious proxy can (1) mount a denial of service attack by dropping all frames forwarded to it by the clients, or (2) tamper sensitive data sent through it. However, we believe that multi-hop extensions do not add any *new* threat that is not already present in WLAN environments. For example, in current WLANs it is relatively easy to mount a denial of service attack by using simple channel jamming techniques. Similarly, all sensitive data should be encrypted using end-to-end mechanisms even in existing WLANs, since the entire network between the endpoints should be considered to be untrusted for such applications.

As a logical next step to this work, we are currently implementing our proposed mechanisms in a prototype system. We are also examining how an incentive-based multi-hop mechanism (similar to [1]) can be incorporated within our multi-hop WLAN framework.

References

[1] N. Ben Salem and M. J. L. Buttyan, J.P. Hubaux. A charging and rewarding scheme for packet forwarding. In *ACM MobiHoc*, June 2003.

[2] O. Dousse, P. Thiran, and M. Hasler. Connectivity in ad-hoc and hybrid networks. In *IEEE Infocom*, June 2002.

[3] Z. Haas, M. Pearlman, and P. Samar. The zone routing protocol (ZRP) for ad hoc networks, July 2002. IETF draft, Work in progress.

[4] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda. Performance anomaly of 802.11b. In *IEEE Infocom*, Apr. 2003.

[5] H.-Y. Hsieh and R. Sivakumar. Performance comparison of cellular and multi-hop wireless networks: a quantitative study. In *ACM Sigmetrics*, June 2001.

[6] G. J., C. A., N. M., and B. C. Paro: Supporting transmission power control for routing in wireless ad hoc networks. *ACM/Baltzer Journal on Mobile Networks*, 2002.

[7] D. Johnson and M. D. *Dynamic Source Routing in Ad Hoc Wireless Networks*. Kluwer Academic Publishers, 2001.

[8] A. Kamerman and L. Monteban. WaveLAN-II: A high performance wireless LAN for the unlicensed band. *Bell Labs Technical Journal*, 1997.

[9] Y.-D. Lin and Y.-C. Hsu. Multihop cellular: A new architecture for wireless communications. In *IEEE Infocom*, Mar. 2000.

[10] V. Park and M. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *IEEE Infocom*, Apr. 1997.

[11] C. Perkins and E. Belding-Royer. Ad hoc on-demand distance vector (AODV) routing. In *IEEE Workshop on Mobile Computing Systems and Applications*, Feb. 1999.

APPENDIX

In the protocol description (Section III) we use a simple heuristic to estimate the latency and bandwidth of a 802.11b wireless link. This is a passive inferencing technique. Consider a link $X \rightarrow Y$. Either X, Y, or any external device, snooping the wireless channel, can use this technique to infer the latency and bandwidth of the $X \rightarrow Y$ link. No additional measurement traffic is needed.

Let p be the packet error rate on this channel. Let τ be the latency incurred during a successful data exchange. For the RTS/CTS access method of 802.11b, this is the time difference between the RTS and the ACK packets in a successful transmission attempt [8].

The 802.11b standard uses a backoff counter which corresponds to the number of slots a client should wait before transmitting after it detects the wireless channel to be idle. This counter is decremented by one for each idle slot [9]. The initial value of the backoff counter is chosen uniformly at random between $[0, CW]$, where CW is a "congestion window" parameter. CW is initialized to $CWMin$ and on each transmission failure, the CW parameter is doubled until it reaches a maximum value.

Let β denote the initial value of the backoff counter, on average, at the transmitting node, in units of time. On average, this would be $CWMin * SlotSize/2$. Then the first data transmission attempt will take on average $\beta + \tau$ time units. The probability of successful transmission on this attempt is $1 - p$. However, with a probability p this attempt fails, in which case the contention window will be doubled for the the second transmission attempt. Hence the second transmission attempt will take $2\beta+\tau$ time units on average. The total latency is $3\beta + 2\tau$. The probability that the transmission is successful due to failure in the first attempt and success in the second, is $p(1-p)$. In general, the ith attempt will take $2^{i-1}b + \tau$ time units. The total latency is $\sum_{j=1}^{i}(2^{j-1}\beta + \tau) = (2^i - 1)\beta + i\tau$, when $p < 0.5$. The probability that the packet is successfully delivered in the ith attempt and had failed in all $i - 1$ previous attempts is $p^{i-1}(1-p)$.

We assume that the wireless link continues to re-transmit the data frame until it is successfully received at the receiver. We also assume that there is no upper bound on the congestion window. Note that in typical wireless scenarios, data frames normally get through in a few attempts. Hence, in practice these simplifying assumptions do not significantly impact the accuracy of the technique. With these assumptions, we can estimate the data latency of a wireless link by knowing p and τ:

$$l = \sum_{i\geq 1}\{(2^i - 1)\beta + i\tau\}p^{i-1}(1-p) = \frac{\tau}{1-p} + \frac{\beta}{1-2p} \tag{2}$$

An external passive observer (that is not an endpoint on the link) can estimate the error rate on the link using empirical observations of gaps in the MAC sequence number space. For example, if this observer, which is operating in the promiscuous mode, receives a sequence of MAC frames with sequence numbers 1,3,4,5,7, it can infer that it has correctly received 5 out of 7 data packets and two were lost. Note that it is possible that the observer receives a MAC frame with the same sequence number multiple times. This is due to losses at the receiving endpoint of the link, which led the sender to re-transmit. Each independent copy of the same sequence number is treated as successful transmission.

The data transfer latency for a successful transmission attempt, τ, can be calculated by observing the instantaneous data rate, B_{inst} (which is either 1, 2, 5.5, or 11 Mbps), used to transmit the data packet. For the RTS/CTS access method, it is given by

$$\begin{aligned}\tau \;=\; & \frac{S_{RTS} + S_{CTS}}{1.0 \times 10^6} + \frac{S_{DATA} + S_{ACK}}{B_{inst}} \\ & + 3 \times D_{SIFS} + D_{DIFS}\end{aligned} \tag{3}$$

where, S_x is the size of packet x in bits, D_{SIFS} and D_{DIFS} are the lengths of SIFS and DIFS respectively. Note that the RTS and CTS packets are usually sent at 1 or 2 Mbps. Note that a passive observer can learn the

[8] We also include a DIFS duration in τ because that is the minimum time duration for the channel to be available after any successful transmission.

[9] The length of a slot, $SlotSize$, is a fixed time duration defined in the IEEE 802.11 standard.

value B_{inst} from the PHY layer header of the data frame which carries this information[10].

Alternatively, τ can be estimated by monitoring the time gap between the transmission instant of the RTS frame, T_{RTS}, and that of the ACK frame, T_{ACK}, in the successful data transfer. Then

$$\tau = D_{SIFS} + T_{ACK} - T_{RTS} \tag{4}$$

Then the bandwidth on that link can be computed as:

$$b = \frac{S_{DATA}}{l} \tag{5}$$

[10]The *signal* field in Physical Layer Convergence Procedure (PLCP) sublayer header has this information.