Divert: Fine-grained Path Selection for Wireless LANs

Allen Miu, Godfrey Tan, Hari Balakrishnan, and John Apostolopoulos*
MIT Computer Science and Artificial Intelligence Laboratory
The Stata Center, 32 Vassar Street
Cambridge, MA 02139
{aklmiu, godfreyt, hari}@csail.mit.edu, japos@hpl.hp.com

ABSTRACT

The performance of Wireless Local Area Networks (WLANs) often suffers from link-layer frame losses caused by noise, interference, multipath, attenuation, and user mobility. We observe that frame losses often occur in bursts and that three of the five main causes of frame losses—multipath, attenuation, mobility—depends on the *transmission path* traversed between an access point (AP) and a client station.

In a typical WLAN deployment, different transmission paths to a client exist in places where overlapping coverage is provided by a set of neighboring APs. Using experimental measurements and analysis on a 802.11b testbed, we show that *fine-grained path selection* among a set of neighboring APs can significantly reduce path-dependent losses in WLANs. We design and implement a WLAN distribution system called **Divert**, which supports fine-grained path selection for downlink communications, on an 802.11b testbed. Divert reduces frame losses without consuming any extra bandwidth in the wireless medium. Our experimental results show that Divert can reduce frame loss rates in realistic scenarios by as much as 26% compared to a fixed-path scheme that uses the best available transmitter.

Categories and Subject Descriptors

Computer Systems Organization [**Computer-Communications Networks**]: Network Architecture and Design—*Wireless Communication*

General Terms

Measurement, Performance, Design, Experimentation

Keywords

802.11, Mobile Systems, Wireless LAN, Path Diversity

*Streaming Media Systems Group, HP Labs, Palo Alto, CA.

MobiSYS'04, June 6–9, 2004, Boston, Massachusetts, USA.

1. INTRODUCTION

Wireless communication channels have notoriously *time-varying* characteristics, where the quality of received signals changes dramatically even over time durations lasting just milliseconds. The complex behavior of wireless signal propagation, particularly indoors, is due to five main causes: *noise* at the receiver, typically caused by both thermal energy in the electronic components and external sources; *attenuation*, caused both by distance from the transmitter and by stationary or moving obstacles shadowing the signal's path to the receiver; *interference* from other transmitters that results in channel contention; *multipath* signal propagation that distorts reception; and *user mobility*, which causes the client to experience rapid channel variations. These properties lead to frame corruption at the link-layer, which in turn results in packet losses, and higher and more variable packet latencies, at higher layers.

An important requirement in the design of indoor wireless local area network (WLAN) infrastructures is loss resilience. To obtain good coverage inside a building, WLAN operators typically deploy multiple access points (APs), which are network elements that forward packets between WLAN clients and the rest of the network. In many cases, a client station can select which AP to use, and typically makes its choice based on factors like the signal quality or the packet loss rate. In current WLANs, a client station sends and receives data only via the AP it has associated with. The client station will only switch association to another AP via a handoff when it experiences severe performance degradation over a relatively long duration lasting many seconds or minutes.

In this paper, we decouple the process of associating a client with an AP from the process of delivering data frames to the client. The process of association usually entails AP scanning and a sequence of message exchanges used to authenticate and register routing information for a client, which do not happen frequently because they cause large interruptions in transmission flow. In contrast, the process of choosing an appropriate AP and path to deliver link-layer data frames to and from a client needs to adapt to short-term channel variations to obtain good performance.

Our measurements (detailed in Section 2) suggest that fine-grained path selection for each frame transmission to client stations can substantially reduce link-layer frame loss rates. There are two reasons why such fine-grained control is effective:

1. Frame losses occur in bursts, and many of these bursts are of long lengths on the order of tens of frames, implying that the conditional probability of losing a

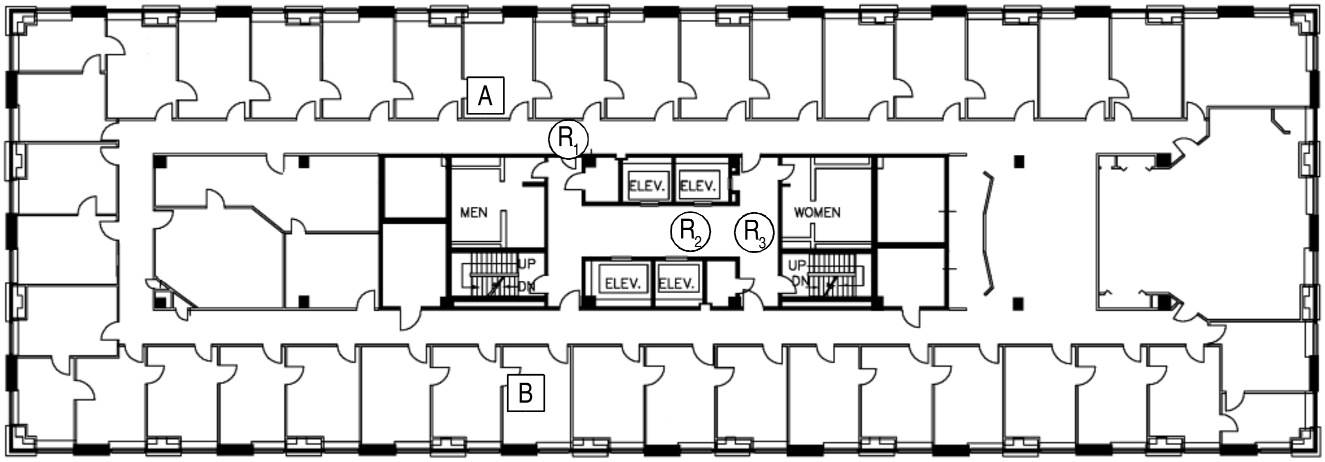


Figure 1: Floor-plan of the experiment setup. 802.11b transmitters at locations A and B each broadcast packets at 2.88 Mbps to a receiver at R_1, R_2 or R_3.

frame given that the previous one had been lost is often significantly larger than the average frame loss rate.

2. Three of the five causes of frame losses mentioned earlier—obstacle attenuation, multipath, and mobility—depend on the *path* traversed between an AP and a client. Thus, the choice of AP and the client's location can significantly affect performance. Furthermore, channel contention from other transmitters near a client's AP may prevent the AP sending a frame even when there is no contention near the client, yet another property that depends on the choice of AP.

These results motivate a WLAN data distribution system that permits fine-grained client-specific path selection among a set of neighboring APs. We describe the design and implementation of such a system. We call our system **Divert**, because it allows a controller to quickly and with little overhead "divert" the responsibility for delivering certain frames to a client from one AP to another. Divert attempts to choose a AP based on short-term frame delivery statistics, with the goal of adapting to short-term variations using path diversity. It runs in conjunction with a longer-term primary-AP selection mechanism, usually a card-specific proprietary mechanism, and can also be used with techniques for coping with high frame loss rates such as packet fragmentation [11], varying packet size [17], forward error correction (FEC), and adjusting data transmission rates [10], and mechanisms for improving performance in multi-rate WLANs [22, 25]. Section 3 describes the design and implementation of Divert.

We present a fine-grained path selection heuristic that can reduce the average frame loss rates without consuming any extra bandwidth in the wireless medium. Using this heuristic, our prototype system reduces the average frame loss rates by as much as 26% compared to a fixed-path scheme that uses the best available path when receiver is in motion. Divert also improves the transmission delay distribution by avoiding long burst losses. Because the two observed facts mentioned above are prominent for a moving client, we find that the benefits of Divert are especially significant for such clients. Section 4 gives performance results.

2. THE CASE FOR FINE-GRAINED PATH SELECTION

We present experimental evidence and examples to make the case for fine-grained path selection. First, we gather measurements to show the short-term loss characteristics of an 802.11b testbed deployed in our building. Our results confirm that frame losses occur in bursts and reveal that the losses have little spatial correlation among different transmission sites. Moreover, we find that when a frame loss occurs, the short-term probability of losing a subsequent frame transmitted from the same site is substantially greater than the short-term probability of losing a subsequent frame if it were sent from another site. We exploit this observation and design a system that seeks to avoid burst losses through fine-grained path selection.

We also analyze delay measurements from two concurrent packet streams transmitted from different sites, and show how localized interference causes intermittently high transmission delays. Our results demonstrate that transmission performance depends on path selection and further supports our case for fine-grained path selection.

2.1 Experimental Setup

Our setup, shown in Figure 1, consists of two 802.11b transmitters, A and B, and a receiving station placed at three different positions, R_1, R_2, and R_3. All nodes are configured to run in the 802.11b *ad hoc* mode. A central packet generator sends a constant bit rate stream to the two wireless transmission sites via a 100 Mbps wired link. For each packet it receives from the packet generator, each wireless site broadcasts the packet on its wireless interface. The packet generator is precisely calibrated to alternate packet transmissions successively between the two wireless transmitters to reduce potential collisions between them. The queues at each site are large enough to prevent any losses due to buffer overflows.

The packet generator sends a stream UDP/IP packets at 240 packets per second to each wireless transmitter. The combined throughput is 5.96 Mbps, which is similar to the previously observed saturation throughput [3]. We use a high packet rate to sample changes in the channel accurately. We use broadcast packets to measure the link-layer data frame loss rate; broadcast packets avoid the effects of link-layer retransmissions and exponential back-off delays, and the effects of link-layer acknowledgment frame losses in the reverse direction. Because frame loss rates tend to decrease with frame sizes [17], we use 1500 byte packets, a maximum transmission unit commonly used in ethernets, in all of our experiments.

We conducted two sets of experiments, *static* and *mobile*, done in separate trials, to examine the effects of stationary and mobile receivers. Although we conducted experiments at each of the three different receiver positions, due to space constraints, we only include the results when the receiver is at R_3. The results when the receiver is at R_1 and R_2 are similar. R_3 is approximately 15 meters from each transmitter. We conduct the static experiments during quiet hours to ensure that the channel is relatively static. For the mobile experiments, the receiving laptop was carried by a human subject moving with random motion over a small area (2 m $\times$ 2 m) centered at R_3 at a normal walking speed. R_3 is located in the middle of an elevator lobby where the walls are made of brick. There is no line of sight between the transmitters and the receiver. Such a location can be harsh for signal propagation but it is not unrealistic. Each experiment transmitted $144,000$ frames in 5 minutes. The results presented are the averages of three trials.

2.2 Burstiness and Spatial Correlation of Losses

We measure the loss characteristics of two concurrent packet streams transmitted from two 802.11b devices at different locations. We are interested in i) how bursty losses are, ii) how frame losses from different transmitters are related, and iii) how receiver motion affects loss characteristics.

Table 1 shows the frame loss rate (FLR) and the *burst loss rate* (BLR) for each packet stream averaged over three trials in both experiments. The BLR is the number of frames lost in a burst of two or more consecutive frames divided by the total number of frames sent in the stream.

For each stream, our mobile experiment has higher FLR and BLR than the static experiments. The BLR/FLR ratio is greater than 50% for mobile and less than 50% for static, which suggests that our mobile experiment has more lost frames that occur in bursts than our static experiment. Figure 2 shows the CDF of burst loss length for each transmitter in both experiments. Although static has proportionally fewer lost frames that occur in bursts, the CDF shows that static has a long tail. When the receiver is static, losses can occur in a few very long bursts (up to 146). In contrast, the maximum burst loss length for mobile did not exceed 53. While we cannot pinpoint the exact cause of this behavior, we believe that a receiver's movement can lower the maximum burst loss length; a mobile receiver can move out of a bad location where the channel quality is extremely poor while a static receiver can suffer long bursts of losses due to sustained, detrimental changes in the transmission path between the static sender and receiver.

Next, we examine how frame losses are correlated between different transmitters (spatial) and at different times (temporal). Let A_i and B_i represent the lost of frame i sent from transmitters A and B respectively. Then, $P(A_{i+k}|A_i)$ and $P(B_{i+k}|B_i)$, for $k > 0$, represents the "auto-conditional loss probability" that the $(i+k)^{th}$ frame is lost, given that the i^{th} frame is lost in the same packet stream. If losses occur in bursts, we expect $P(A_{i+k}|A_i) > P(A)$, where $P(A) = FLR_A$. In contrast, if losses are memoryless (non-bursty) or independent, we expect $P(A_{i+k}|A_i) = P(A)$.

Experiment	static		mobile	
Sender	A	B	A	B
$FLR(\%)$	4.79	10.2	17.1	15.3
$BLR(\%)$	1.5	4.6	10.8	9.2
BLR/FLR	31%	45%	63%	60%

Table 1: The average frame loss rate (FLR) and average burst loss rate (BLR) for the static and mobile experiments.

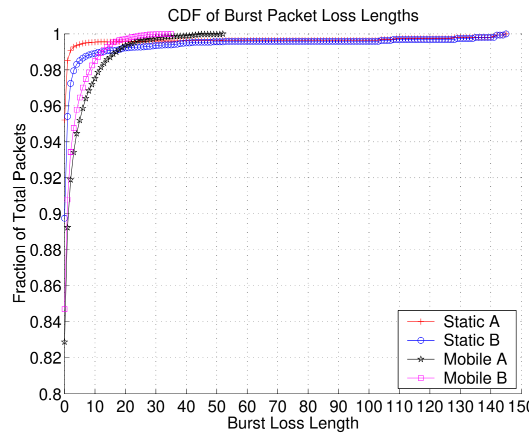
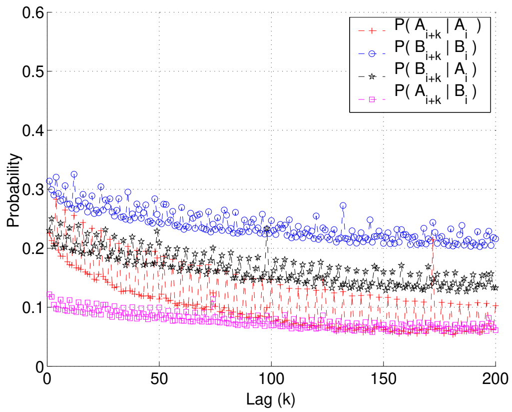


Figure 2: CDF of the length of burst losses for both static and mobile experiments.

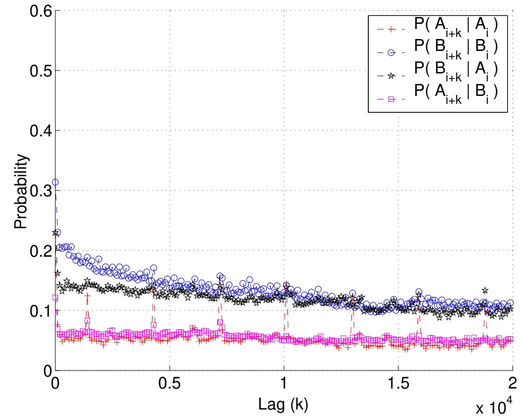
Similarly, we use $P(B_{i+k}|A_i)$ and $P(A_{i+k}|B_i)$ to represent the "cross-conditional loss probability". Thus, if losses are correlated between the streams, we expect $P(B_{i+k}|A_i) > P(B)$, where $P(B) = FLR_B$. If losses are independent between streams, we expect $P(B_{i+k}|A_i) = P(B)$.

Figures 3(a) and 3(c) shows the auto-conditional and cross-conditional loss probabilities for the static and mobile experiments at small values of k, $1 \le k \le 200$ (4.2 to 840 ms).

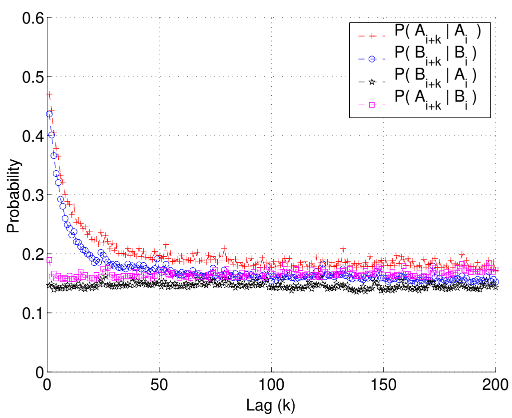
In our mobile experiment, losses are bursty. Figure 3(c) shows that the auto-conditional loss probabilities ($P(A_{i+k}|A_i)$ and $P(B_{i+k}|B_i)$) are much larger than the respective average FLR (Table 1) for A and B at small lags. Thus, given that a frame loss occurs, the probability of losing the next few frames is much higher than the average FLR, which suggests that burst losses are likely to happen. In contrast, the cross-conditional loss probabilities ($P(A_{i+k}|B_i)$ and $P(B_{i+k}|A_i)$) remain nearly the same as the respective average FLR, which suggests that frame losses have very little correlation between the different transmission sites. Observe that the average FLR of A is larger than that of B, yet $P(A_{i+k}|B_i) < P(B_{i+k}|B_i)$. This suggests that fine-grained path selection can be effective in avoiding imminent burst losses by switching to an alternate site whenever a loss occurs in the current site, even in cases where



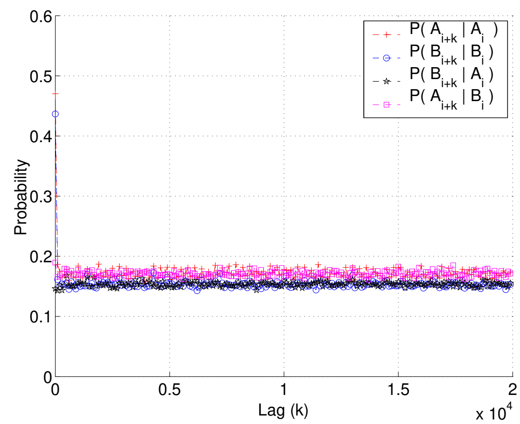
(a) Static (small k values)



(b) Static (large k values)



(c) Mobile (small k values)



(d) Mobile (large k values)

Figure 3: The auto-conditional and cross-conditional loss probabilities of frame losses at different frame lags k for the static (a) and (b) and mobile (c) and (d) experiments.

sites have different average FLR. *Thus, path diversity can effectively reduce time-correlated losses in dynamic wireless channel conditions.*

In our static experiment, losses are less bursty than our mobile experiment. Figure 3(a) shows that in almost all lags, $P(B_{i+k}|A_i) > P(A_{i+k}|A_i)$ but $P(A_{i+k}|B_i) < P(B_{i+k}|B_i)$. This is because the FLR of B is about twice that of A. While B can benefit by switching to A whenever a loss occurs, the converse is not true for A. Although fine-grained path selection is beneficial for our mobile environment, coarse-grained path selection based on long-term frame loss rates may be sufficient in our static environment (*i.e.,* when the channel is less dynamic). However, we will illustrate in the next section that in some cases, fine-grained path selection is beneficial for both static and mobile receivers.

Another interesting observation is that in our static experiment, $P(A_{i+k}|A_i)$ tend to be periodically higher for every k value that is a multiple of 4 (about 17 ms). Although we are not certain, we think that this behavior is due to collisions with beacon frames from nearby APs, each of which periodically broadcast beacons at every 100 ms.

Figures 3(b) and 3(d) show similar probabilities for our static and mobile experiments at large k values, $1 \leq k \leq 20,000$ (4.2 ms to 84 s). For clarity, we only plot data for each k value that is a multiple of 100. First, as k grows larger, the auto-conditional loss probabilities in both experiments converge to the corresponding frame loss rates. This expected behavior is due to the fact that frame losses become more and more independent as the lag increases. We note that $P(A_{i+k}|A_i)$ is periodically higher for each k value that is a multiple of 3000 (about 12.5 s), but we are not certain of the cause.

2.3 Impact of Localized Interference

Most WLAN medium access control (MAC) protocols use a carrier sensing (CS) mechanism to reduce the likelihood of collisions. Before sending a frame, the sender senses the channel for activity. If the sender senses energy in the chan-

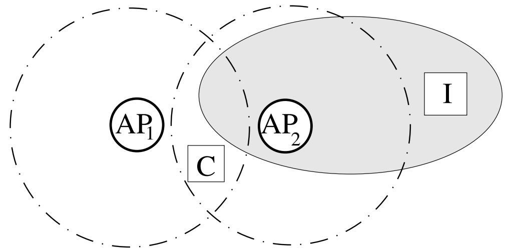


Figure 4: Carrier sense suppresses AP_2 from transmitting due to the interfering signal from I (*e.g.*, a WLAN client in another nearby network or an appliance that use the same frequency spectrum). However, AP_1 may be used to communicate with C because the interfering signal is not strong enough to affect either AP_1 or C.

Path	Loss Rate (%)	RSSI (dBm)	Jitter (ms)	DeferEngy Count
AP_1	1.82	-37.91	2.293	47389
AP_2	2.03	-44.46	0.351	22749

Table 2: Based on loss rate and average received signal strength, AP_1 is the preferred path. However, AP_1 has a much higher one-way delay jitter than AP_2.

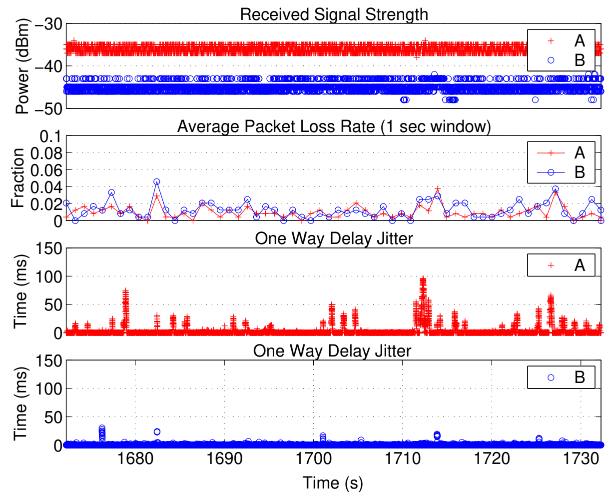


Figure 5: A 60-second snippet of a streaming experiment on two paths originating from transmitters AP_1 and AP_2. AP_1 (in red) offers higher signal strength and lower loss rate than AP_2. However, due to localized interference that triggers the carrier sensing mechanism in AP_1, AP_1 frequently suffers from high spikes of delay jitter, while the one-way delay from AP_2 remains low and relatively constant.

nel, it suppresses its transmission to avoid colliding with another potential ongoing transmission.

Carrier sense suppression depends on the relative positions of transmitting and receiving nodes, and can lead to the classical exposed terminal problem [4]. When one site's transmission is suppressed by CS, an alternate site may be used to transmit data frames. However, the alternate site's transmission cannot succeed if the interfering energy in the medium is too high at the receiver; the receiver must be in a location where the signal to interference ratio is sufficiently high (see Figure 4). A fine-grained path selection system can discover such transmission opportunities when they exist.

We gathered measurements that show that the scenario described above exists in a real network. During a busy hour, two transmitters, AP_1 and AP_2, alternatively send broadcast frames to a common receiver C. Table 2 shows that transmitter AP_1 offers both higher signal strength and slightly lower overall loss rate in a 30-minute packet trace of an experiment that involved transmitting 720,000 data frames. Thus, if AP_1 were an access point, the receiver would naturally associate with AP_1.

Figure 5 shows the received signal strength, the average loss rates of 1-second slices in the trace, and the one-way delay jitter (i.e., the delay variations above the minimum one-way delay value) as a function of time for a 60-second snippet of the packet trace. This 60-second snapshot is chosen to avoid cluttering the figure, and is representative of the characteristics manifested in the entire trace. The figure shows that the packet delay jitter from AP_1 is substantially higher than the delay jitter from AP_2. Lost packets were ignored from the delay jitter analysis. Because broadcast packets are not retransmitted and are not subject to the exponential back-off mechanism in 802.11 networks, the increased one-way packet transmission delays from AP_1 can only be attributed to the increased delay caused by the carrier sense mechanism (perhaps due to ongoing traffic from a nearby 802.11b network). We confirm this hypothesis by verifying the `DeferEngy` register in the transmitter's 802.11b interface, which counts the number of times that a packet has been deferred because energy was sensed in the carrier. Table 2 shows that the value of the `DeferEngy` register for transmitter AP_1 is much greater than the value for transmitter AP_2, indicating that transmitter AP_1 deferred transmission more than twice as many times as transmitter AP_2.

Table 2 shows that the loss rate in transmitter AP_2 is comparable to that of transmitter AP_1's, which suggests that site AP_2's transmission succeeds even though interfering energy is detected by AP_1. Thus, the losses from AP_2 are uncorrelated with the CS-triggered delays from AP_1. We have identified a real case of the example shown in Figure 4. *For these scenarios, fine-grained site selection can be used to reduce both loss and delay by switching data frame transmissions intelligently between the available transmission sites.*

3. DESIGN AND IMPLEMENTATION OF DIVERT

Divert deploys multiple APs within an area (Figure 6), interconnected over a wired network whose data rate is much higher than the wireless link rate, with each AP being able to detect whether a WLAN client is currently within its transmitting range or not (*e.g.*, using periodic probes). Both WLAN clients and APs use synchronous ACKs (as in 802.11 [1]) to immediately acknowledge every non-broadcast data frame received over the wireless link. This feedback is important, because it allows the data transmitter to determine path conditions at the granularity of individual frame transmissions.

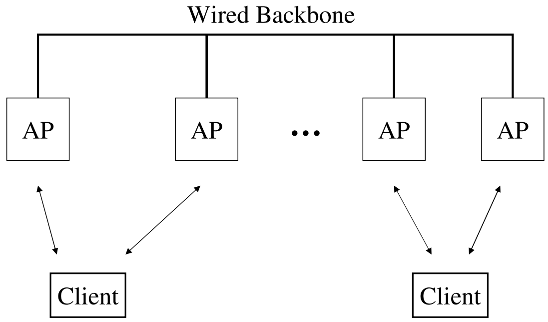


Figure 6: A cellular WLAN model where neighboring APs have overlapping coverage. To achieve the benefits of fine-grained path selection, Divert requires that the client switch between APs quickly and at low cost. Section 3.2 shows how 802.11-like systems can achieve this goal.

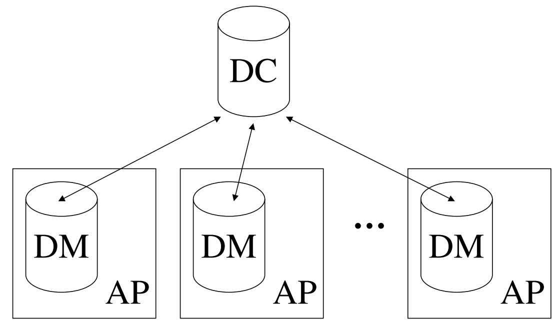


Figure 7: The Divert architecture to perform fine-grained path selection among access points, shown in the downlink direction. The Divert Controller (DC) determines which path (AP) to use. Each AP runs a Divert Monitor (DM) that monitors link conditions and reports these conditions to the DC.

Divert uses a path-selection heuristic to determine which AP, and hence which wireless path, to use for packets in the downlink and uplink directions. Section 3.1 discusses the details of the heuristic. Divert also requires the ability for packets to and from a WLAN client to change the data frame's forwarding AP, and hence its wireless path, on a fine-grained basis without disrupting communication or incurring overhead. Section 3.2 describes how fast path switching can be achieved in cellular WLAN networks.

In a traditional WLAN architecture, the different APs deployed in a single WLAN requires little explicit coordination between one another. On the other hand, Divert requires explicit coordination because it makes path choices on a frame-by-frame basis (including sending frame retransmissions along a path different from the original transmission), with control over the path resting on the *transmitter-side* rather than on the receiver. To enable this coordination, Divert extends the WLAN architecture by adding two components, the *Divert Controller (DC)* and the *Divert Monitor (DM)*, as shown in Figure 7. The DC and DM run on the transmitter-side of the system—the wired backbone network and AP in the downlink direction, and the WLAN client in the uplink direction. The system also works if Divert's DC and DM components are deployed on only one of the two directions. In this case, fine-grained path selection is enabled in one corresponding direction only.

In the downlink direction, the DC is responsible for forwarding each packet via one of the APs that is within transmission range of the client. The DC runs a fine-grained path-selection heuristic, which makes a forwarding decision for each packet based on feedback sent by the DMs, each of which runs at an AP. A DM monitors the wireless link at its AP and sends two types of messages to the DC, *registration event* messages and *path-condition update* messages.

Registration event The DM sends a periodic registration event to the DC whenever the DM detects that a particular client is within its transmission range. The registration event allows the DC to maintain a set of usable transmission paths for fine-grained path selection. The event is maintained as soft-state at the DC so that the registration can timeout when a client moves out of an AP's transmission range.

Path-condition update Each DM monitors the channel conditions in the direction of the data flow; in the downlink direction, the DM at the AP maintains this information per client. The DM periodically sends updates of this information to the DC. The DM observes a failed transmission if the sender does not receive a synchronous ACK after a frame transmission. This failure can occur when either the data frame or the returning ACK is lost. The DM may also observe the receiver's received signal strength of the transmitted data frame if it is reported in the synchronous ACK. To reduce the overhead of reporting feedback to the DC, the DM does not send per-frame level information to the DC. Instead, it sends an update at regular intervals or whenever a threshold condition (see Section 3.1) has been satisfied.

To support retransmissions, the DC wraps each data packet with a header that contains a field indicating the retransmission limit. If the sender (AP or client) fails to successfully transmit a data frame to the receiver (client or AP) and receive an ACK, the corresponding DM decrements the retransmission limit field and returns it to the DC for retransmission if the retransmission limit has not been exceeded. Because the DC runs the path-selection heuristic for the packet, including those being retransmitted, the retransmission may be done along a different path.

In the downlink direction, the DM runs on each AP, maintaining per-client statistics on path conditions. The DC coordinates a set of APs and decides which one to use for any given frame. In the uplink direction, the DC and DM both run on the WLAN client; in this direction, the DC decides which AP to use as its next hop for data frames. The client requires a method to determine which APs are within its transmission range at any time and can obtain this information using periodic probing. Note that Divert can be used in both directions or only one; our current Linux implementation and experiments are for the downlink direction alone.

We made a deliberate design decision to put the path-selection decision control at the transmitter-side of the system. Alternatively, a receiver can monitor channel conditions and select different wireless paths. But receiver-side control has several drawbacks. First, the receiver can only detect lost frames from gaps in the sequence numbers of the transmitted frames, which means that it cannot detect a loss before it receives a successful transmission. When losses occur in bursts, a receiver may not be able to switch paths in time to avoid them. Second, when a receiver decides to switch paths, it must send a wireless control message to notify the DC when to switch paths. Such a message is unreliable and is prone to loss when channel conditions are poor. None of these problems will occur when the transmitter makes path-selection decisions.

Moreover, a downlink DC that resides in the distribution system has a global view of the wireless activities at all the different APs. Thus, the downlink DC can, for example, measure traffic load among APs, track a client's movement or detect which APs are suffering from localized interference from their carrier-sense mechanism, and adapt path-selection decisions accordingly.

3.1 Divert Path Selection Heuristic

The goal of Divert's fine-grained path selection heuristic is to reduce losses in the wireless medium without consuming extra wireless bandwidth. The heuristic, at any given time, selects only one AP with a good transmission path to transmit a downlink/uplink data frame to/from a client. Our goal is different from techniques proposed in [24, 21, 19], which seek to aggregate bandwidth by using multiple orthogonal paths in parallel. Our goal is also different from schemes that employ forward error correction (FEC) across multiple paths. For example, a simple FEC scheme might replicate and transmit every data frame via all the APs that are within range of the client. Such schemes use redundancy to reduce loss rates in the wireless medium, while Divert achieves the same goal through intelligent, fine-grained path selection without consuming extra wireless bandwidth.

In theory, a path-selection algorithm should select the best path for each data frame transmission. To do so, a system must acquire accurate knowledge of the wireless channel condition of each available path within a few milliseconds. In practice, accurate sampling of the channel conditions is difficult and might incur large overhead.

We observe that selecting the best path for every data frame is unnecessary to achieve good results. As observed in Section 2.2, frame losses usually occur in bursts, especially when the receiver is mobile, and different transmission paths often exhibit weakly correlated channel conditions. Therefore, a path selection heuristic can be effective if it can determine whether the currently used transmission path has fallen into a bad state (*i.e.,* predict whether the next few frame transmissions will fail with high probability), and divert the subsequent transmissions to an alternate path. As long as the alternate path's average loss rate is not substantially higher than that of the current path, diverting the frame transmissions will likely avoid burst losses in the original path.

In Divert, the DM running at each AP or the client keeps track of the per-path history of the losses of the last data frames sent to each station within a time window of H. The DM then monitors the loss rate within this window. If the observed number of lost data frames is greater than a certain threshold T, the DM notifies the DC to forward subsequent frames via a different AP. After a path switch occurs, the DM at the newly-selected AP waits for at least H data frame transmission attempts before signaling another switch to the DC. Thus, H defines the switching time granularity (hysteresis), while T governs the sensitivity to the losses on the current path.

This heuristic is simple, and it uses feedback information only from the currently used transmitter. Active channel probing is unnecessary because the sender can detect a lost unicast frame by the absence of its synchronous ACK.

However, this heuristic is sub-optimal and will not work well under all channel conditions if the values H and T are fixed. A small value of H is desirable for bursty and dynamic channel conditions, so that the heuristic can adapt quickly. On the other hand, a larger value for H allows the heuristic to obtain a better estimate of the channel's average loss rate; a larger value is suitable under static channel conditions where the signal quality does not vary quickly. As suggested in Section 2.2, often, a better selection strategy for static channel conditions is to "lock on" to the access point that has a lower average loss rate. Similarly, when the loss rates of the alternate paths are significantly higher than or are highly correlated with the current path, a large T is desirable to prevent switching to a potentially poorer path when only a small number of losses are detected in the current path. In other cases, a small T diverts packets early, which helps to avoid imminent burst losses in the current path.

Our experiments in Section 4 indicates that a choice of $H = 1$ frame[1] and $T = 1$ works reasonably well for dynamic channel conditions when the receiver is mobile, and a choice of about $H = 10$ and $T = 5$ works well when the channel is less dynamic. Fortunately, our experiments suggest that the observed loss rates are not too sensitive to the exact values of H and T. Nonetheless, we envision that the current heuristic can be improved by making H and T adaptive, *e.g.,* using simple machine learning techniques for learning parameters [18].

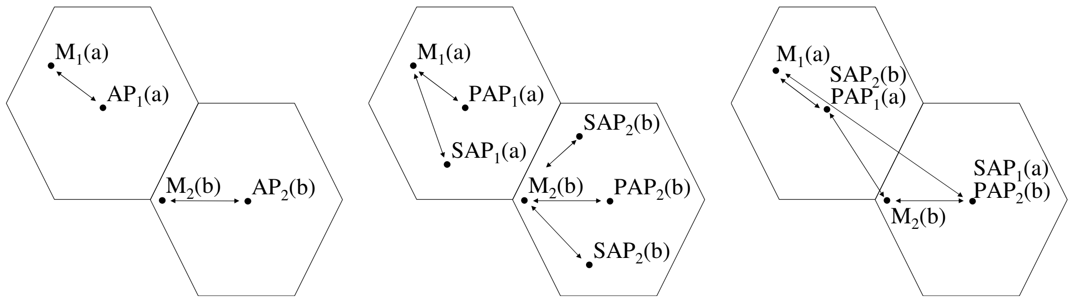
Finally, when the heuristic has access to more than one available path, it needs to pick a path. A simple mechanism is to randomly pick an alternate path from among a set of APs within communication range of the client. As we explain in Section 3.4, the method of discovering the APs within communication range of the client is implementation-specific. Currently, we have not investigated the problem of selecting between more than two paths and plan to address it in future work.

We emphasize that the fine-grained path selection heuristic presented here is different from the handoff algorithms that are used to initiate a handoff process in many common cellular WLANs (described in the next section). Due to the high overhead in a typical handoff procedure, handoff algorithms often use a strong hysteresis to prevent a receiver from flapping handoffs among APs [5] when it finds multiple APs within range. In contrast, the Divert heuristic can switch paths among APs on a frame-by-frame basis; thus, transmission paths are selected only as a function of channel conditions estimated by the per-client data frame loss history at each AP.

3.2 Reducing Path Switching Cost

The Divert design assumes that the WLAN incurs negligible cost when the transmission path is switched between different APs. This assumption is reasonable for WLAN ar-

[1]We used a simpler implementation where the loss history H is specified by a frame window (*i.e.,* the number of most-recently sent data frames), and not by a time window.



(a) Cellular WLAN (b) Divert WLAN (c) Divert WLAN (Low-cost)

Figure 8: Clients M_1 and M_2 belong to cells 1 and 2, which operate in channels a and b respectively. For (b) and (c), all APs in the same cell operate in the same channel. Arrows mark possible WLAN communication path(s) for each client. In a traditional cellular WLAN (a), clients only communicate with one AP. In the extended Divert WLAN (b), the primary access points (PAP) associate with clients while the secondary access points (SAP) provide alternate communication paths within the same cell. A low-cost deployment alternative for Divert is shown in (c), where the SAP is co-located with the PAP of another cell to reduce the number of AP deployment locations. In (c), M_1 has a long communication path to SAP$_1$ but in many cases, M_1 will communicate with a closer SAP located in another cell (not drawn).

chitectures that support *soft handoff*. In a soft handoff, during the transfer of communication from one AP to another, a client maintains an undisrupted communication flow with both APs until the transfer completes. For example, code-division multiple access (CDMA) wireless networks support soft handoff, during which neighboring APs transmit signals simultaneously. Clients use RAKE receivers to resolve and decode the combined signals and maintain connectivity.

A WLAN that uses the same frequency channel for all its APs may also support soft-handoffs. However, typical WLANs such as 802.11 uses a *cellular* architecture (Figure 8(a)) in which operators configure neighboring APs to use *orthogonal* channels to achieve spatial frequency reuse that increases the capacity of the network. In order to communicate with an AP in the network, a client needs to switch its communication channel to the one being used by the AP. Thus, in cellular WLANs, Divert needs to explicitly notify the client to switch channels whenever it selects a new path for downlink communication. The overhead associated with switching paths can be significant (lasting from a few to hundreds of milliseconds [15]) especially when it occurs on a frame-by-frame basis. Moreover, forcing a client to communicate with an AP when the client is outside of that AP's cell boundary may increase co-channel interference and reduce the capacity of the network, as we will explain later.

One method of reducing the path-switching overhead is to install multiple radios on each client and statically associate each radio with a different access point that is within range of the client. This solution has two drawbacks: 1) the solution is not scalable; to take full advantage of the path diversity offered by N available APs, a client needs to install N radio devices, where N can be as large as the number of orthogonal channels offered by the WLAN (twelve for 802.11a) and 2) multiple radios consume more power and may not be suitable for battery-powered clients.

Our approach for reducing the path switching cost is to use one radio on the client and deploy additional *secondary* access points in the WLAN. In the extended Divert architecture shown in Figure 8(b), each *primary* access point (PAP) defines a distinct WLAN cell and may be assigned a frequency channel that is orthogonal to a neighboring primary AP. The primary AP handles *authentication* and *association* procedures to allow clients to join its cell. Then, one or more secondary APs are placed within a cell, *i.e.,* within the coverage area of their primary AP, but at spatially diverse locations to achieve path diversity gains. The secondary APs are used to provide alternate transmission paths to the clients within their cell. All APs within a cell operate in the same frequency channel to minimize the path switching cost among them. A DM runs at each AP within a cell to monitor the wireless link condition and report feedback to the DC, as described earlier. The DC performs fine-grained path selection as previously described, except that the set of possible alternate paths for a particular client is limited to the APs within the client's cell.

Because common WLAN systems (*e.g.,* 802.11) offer link-layer services (*e.g.,* security) that processes packets in an AP, it is convenient to place the DC inside the primary AP. This allows the DC to forward data frames to an AP *after* they have been processed. Essentially, the existing wireless services should run without modification. More importantly, the processing takes place in a central location for every cell. Central processing obviates the need to distribute states across the secondary APs to run the existing services; hence, the design choice of placing the DC at the primary AP greatly reduces the complexity of integrating Divert into existing WLAN systems.

Although a secondary AP is similar to a primary AP, it is different in the following ways. A primary AP defines a distinct cell and a pair of neighboring primary APs define a cell boundary. Although a secondary AP may transmit frames within a cell, a secondary AP does not increase the

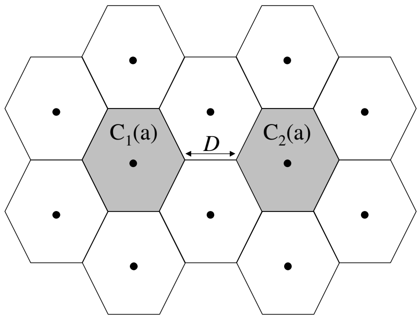


Figure 9: A hexagonal cell model. The cells C_1 and C_2 operate in channel a and may cause co-channel interference between each other. If all cell sizes are identical, the worse-case co-channel interference between C_1 and C_2 is a function of their minimum separation distance D.

size of the cell defined by its primary AP (when the operators deploy secondary APs within their primary AP's cell boundary). The primary AP is the only AP within a cell that handles *authentication* and *association* procedures to allow clients to join its cell. Thus, a secondary APs has no effect on a client's handoff policy, which dictates when a client initiates a handoff as it crosses cell boundaries. We made these design choices to eliminate the potential interference problems described in Section 3.3.

A significant advantage of the extended Divert architecture is that it allows the WLAN to increase its capacity using the well-tested cellular architecture, while facilitating low-cost fine-grained path selection. Because access points are commodity devices, we expect that secondary APs will not significantly increase deployment cost. If the cost of installing and wiring secondary APs at different physical sites become significant, WLAN operators may co-locate the secondary APs of one cell with the primary APs in the neighboring cells (see Figure 8(c)). In this case, Divert operates in the same way as before except that alternate paths of a cell will extend into the neighboring cells. Consequently, the channel quality of the alternate paths may decrease and co-channel interference between cells that operate at the same channel may increase. However, in practice, the channel quality of alternate paths and co-channel interference between cells are related to the individual cell's location and traffic load. While Divert does not restrict where secondary APs are deployed, an WLAN operator needs to make the appropriate trade-offs between cost and performance when deploying a Divert system. The next section describes how Divert, with strategic placement of secondary APs, limits the potential increase of co-channel interference.

3.3 Co-channel Interference

Co-channel interference arises when two or more wireless devices that operate at the same frequency are placed within each others' radio interference range. Thus, APs that operate in the same frequency channel should be placed carefully so that they do not interfere with one another and reduce the overall capacity of the network. Since the extended Divert architecture requires additional (secondary) APs that operate in the same frequency channel, it is important to understand how their deployment might affect the overall capacity of the network.

We use a simple hexagonal cell model as depicted in Figure 9 to examine the impact of adding secondary APs into a WLAN. To simplify our analysis, we assume that an AP is located in the center of each cell and all cells have identical size. The results are general and remain valid even if the cell sizes and shapes are different. Without loss of generality, we assume cells C_1 and C_2 operate in the same frequency. In a traditional WLAN without secondary APs, clients can move to the edge of a cell's boundary. Thus, the *worst-case* co-channel interference between the two cells is a function of the cells' minimum separation distance D, *i.e.,* the minimum distance between a client from C_1 and a client from C_2.

Suppose the primary AP is located at the center of every cell, adding secondary APs *does not increase* the worst-case co-channel interference, if the following conditions hold:

- The secondary APs do not affect a client's handoff policy. As a client crosses a cell boundary, it disassociates with the primary AP of the cell that it is leaving and associates with the primary AP of the cell that it is entering.
- Secondary APs are always placed within the boundary of the same cell as their primary AP.

The first condition maintains that a client cannot join a cell, *e.g.,* C_1, unless it is within C_1's cell boundary. Thus, regardless of the secondary APs' existence, all of C_1's active clients are still contained within C_1's boundary, and similarly for clients in C_2. The second condition ensures that the secondary APs of C_1 are placed within C_1's boundary, and the secondary APs of C_2 in C_2's boundary. With these two conditions, there is no way to place a wireless station from C_1 at less than D length away from the closest wireless station in C_2. Thus, the worst-case co-channel interference remains unchanged.

3.4 802.11 Implementation

We use Linux PCs equipped with a Intersil Prism-II based 802.11b PCI card to implement the primary and secondary APs, and a dedicated 100 Mbps Ethernet to serve as the wired backbone between the primary and secondary APs of a single Divert WLAN cell. We modify the HostAP (ver. 0.0.1) driver [12] to incorporate the DC and DM. We configure the wireless interfaces to run in 802.11 AP mode, and our prototype implementation works with regular, unmodified 802.11b managed mode clients.

We configure the wired backbone and the wireless network as different subnets. The AP host uses Linux *iptables* to forward packets with an IP address destined to a WLAN client from the wired network to the wireless network. Because we implement the DC and DM within the HostAP driver, we need a way to deliver Divert control messages to a DC or DM running at a remote AP. We achieve this behavior by configuring each AP host with an IP address in the wireless subnet. An AP sends a Divert control message to the wireless IP address of the target AP host via the wired backbone. Unfortunately, IP packets that reach the destination host will be consumed by the host. Thus, the target Divert component running within the wireless interface's driver will not receive the control packet. To solve this problem, we add to every AP host one static ARP entry that contains the wireless IP address of the corresponding AP host. As long as there is an ARP entry with the AP's wireless IP address, the target host will forward all IP packets to the wireless interface, independent of the MAC address value in the ARP entry.

We implement the DC inside the data path of the HostAP driver so that the primary AP can forward a client's packets to the secondary APs via the wired backbone. The DC contains a table of wired ethernet MAC addresses of all the secondary APs in the primary AP's cell. Before the DC forwards a packet to a secondary AP via the wired interface, it changes the packet's destination MAC address to the Ethernet address of the selected secondary AP listed in the table. We disable packet retransmissions in the native wireless interface layer to allow the DC to assume control of retransmissions. We have not yet implemented DC-controlled retransmissions, so the wireless interface simply drops all packets that fail their first transmission attempt. We plan to incorporate this retransmission functionality soon.

A DC also needs to determine which APs are within transmission range for a particular client. For the primary AP, client detection occurs automatically when the client sends an association request. Secondary APs currently do not detect whether a client is within range. However, we can implement client detection by configuring a dedicated wireless interface in every secondary AP to sniff for the client's upstream transmissions (either a data or an ACK frame). Similar techniques have recently been proposed to build connectivity graphs to improve 802.11 handoff performance [23].

We implement the DM inside the HostAP driver as an interrupt handler, which receives a callback triggered by a packet transmission, indicating if its delivery has succeeded or failed. The DM runs the Divert path-selection heuristic. We use a simpler implementation from the one described earlier, where the loss history H is specified by a frame window (*i.e.,* the number of most-recently sent data frames), and not by a time window. Because all of our experiments send packets at a constant packet rate, the frame-based loss history is a good approximation of the time-based loss history. When the DM detects that an AP's channel condition has fallen into a bad state, it sends a path-condition update to the DC. The DC then selects a different AP by cycling through the table of secondary APs. The DC sends a control message to clear the packet history of the DM running at the selected secondary AP and to start forwarding subsequent packets to it.

The primary AP runs like an ordinary 802.11b access point. It broadcasts periodic beacon messages to advertise its existence to clients and accepts their association requests. The secondary AP receives and forwards packets over the wireless interface, but does not participate in broadcasting beacons or associating with clients. We change the MAC address of the secondary AP's wireless interface to the MAC address of the primary AP's wireless interface. Hence, the secondary AP is configured to spoof the primary AP's identity. The MAC address spoofing allows a client in 802.11b managed mode to receive packets from different APs transparently and without interrupting the data flow.

One important detail concerns the use of link-layer sequence numbers in 802.11. Ideally, the primary and secondary APs should use synchronized sequence numbers so that the packet's origination is completely indistinguishable. However, the Prism-II chipset used in our implementation does not export an API that allows us to synchronize the sequence numbers or to modify them in the 802.11 header. In practice, the sequence numbers are used only for duplicate packet detection and reassembling fragmented data frames. As long as we restrict the fragmented frames to the same wireless interface that initiated the link-layer fragmentation, the system will handle fragmented frames properly.

Unfortunately, the system can no longer detect duplicate data frames transmitted by different APs, which can happen due to a ACK packet loss.[2] Fortunately, link-layer packet duplication is usually not a problem in practice because the best-effort service model allows for occasional packet duplication; the link layer is not required to filter all duplicate packets for the higher layers of the protocol stack. End applications and transport layer protocols such as TCP can usually detect duplicated packets and discard them if necessary.

We have not yet implemented the client-side modifications to support fine-grained path selection in the uplink direction. Uplink transmissions from the unmodified clients are received and acknowledged by the primary AP in exactly the same way as a regular AP in the 802.11 network.

3.5 Security Issues

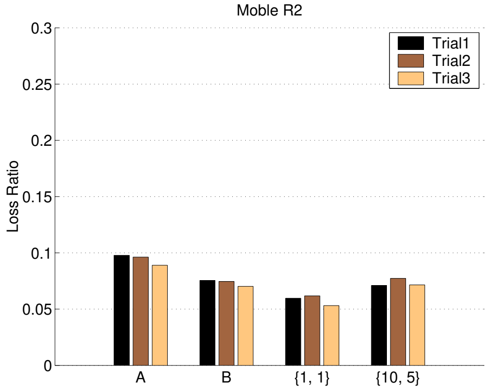
Divert does not affect link-layer security services such as the Wired Equivalent Privacy (WEP) and the 802.1x security extensions [2]. An unmodified client associates and authenticates with a primary AP in the same manner as it would in the original 802.11 WLAN. For downlink communication, the DC can let the WEP/802.1x security service perform the necessary processing to a data frame before forwarding it to the selected AP for immediate transmission. Since the security layer is typically implemented in the device driver of the wireless interface, it is important to run the DC component inside the primary AP.

For clients that do fine-grained path selection in the uplink direction, the DC at the client may modify the next-hop address in the 802.11 link-layer header of an uplink frame. Since the 802.11 header is not protected by either WEP or 802.1x, the security service for uplink packets should be unaffected, as long as the secondary APs forward all received packets to the primary AP for proper processing.

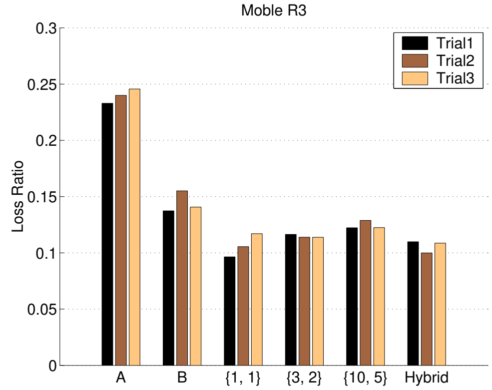
4. EXPERIMENTAL RESULTS

We evaluate our implementation of Divert to demonstrate the benefits of fine-grained transmission path selection. The experimental setup is the similar to the one in Section 2.1. The major difference is the use of unicast frames as opposed to broadcast frames, and the transmitters at A and B are APs running our Divert implementation. To measure link-layer frame loss rate (FLR), we disabled packet retransmissions. We stream 1500 byte unicast UDP packets to the receiver at each of the three locations in Figure 1 at a rate of 240 packets per second using i) only transmitter A (referred to as scheme A), ii) only transmitter B (referred to as scheme B) and iii) Divert with several settings of H and T values. As explained earlier, Divert will use AP A or B to transmit each frame. Except for the *Hybrid* configura-

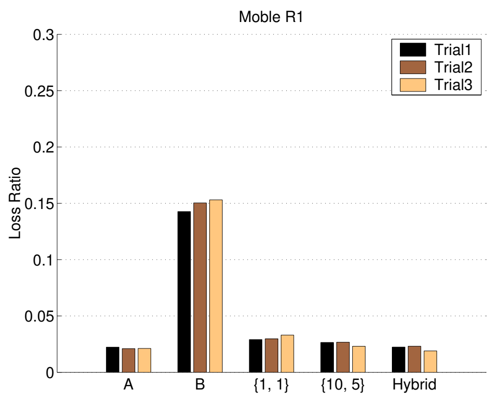
[2] In the future, the availability of an 802.11 chipset that allows higher-layer control of the frame sequence numbers can solve this problem in a way that permits duplicate detection.



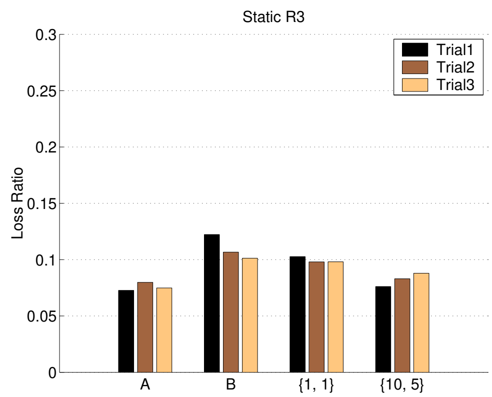
(a) Mobile at R_2



(b) Mobile at R_3



(c) Mobile at R_1



(d) Static at R_3

Figure 10: Average frame loss rates of different transmission schemes at three different receiver positions. The label denotes that values $\{H, T\}$ used in a Divert transmission scheme. Hybrid uses $\{1, 1\}$ for transmitter A and $\{3, 2\}$ for B.

tion, the same set of H and T values is used as the switching criteria from A to B and from B to A. Under the $Hybrid$ configuration, Divert uses $H = 1$ and $T = 1$ as the switching criteria from the transmitter with a higher average FLR and $H = 3$ and $T = 2$, from the transmitter with a lower average FLR.

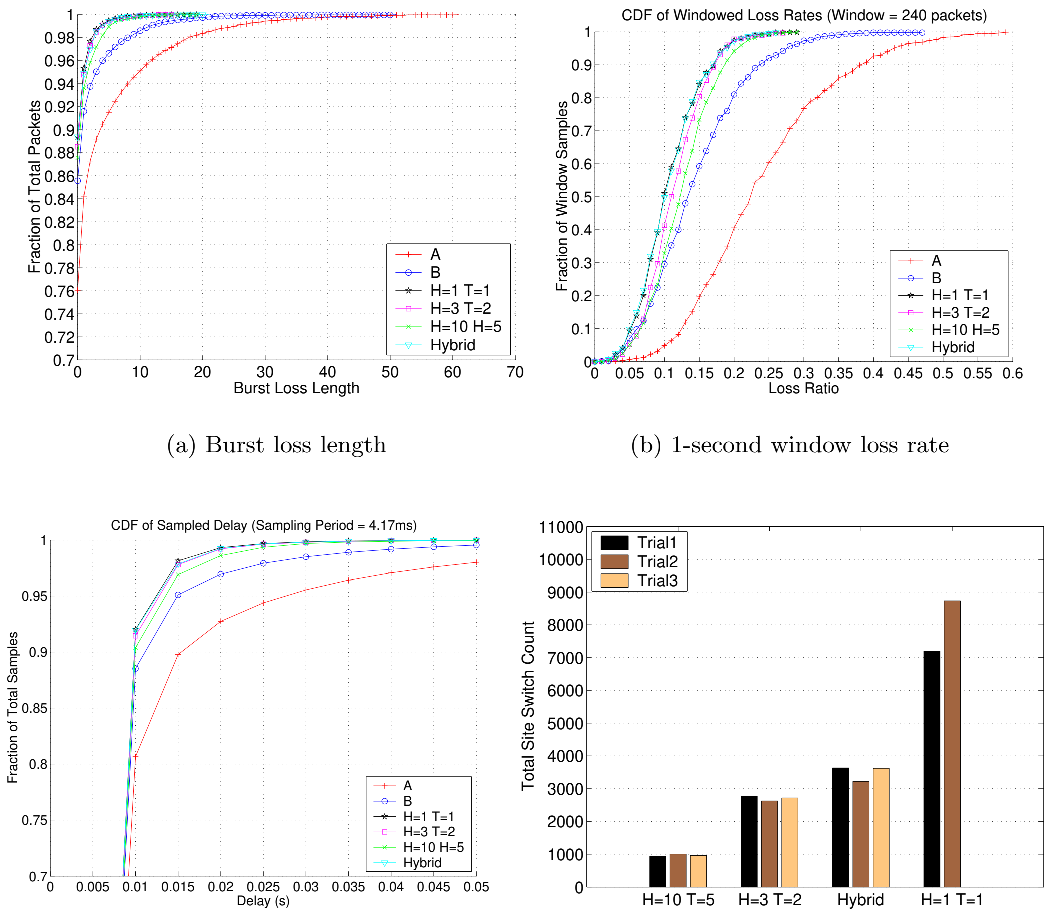
We disabled roaming at the receiver to prevent it from initiating a handoff during the experiment. Again, we conducted our experiments in late evening to avoid biases from the building's daily activity. We repeated each experiment over three trials. To avoid biases from the human subject performing the mobile experiments, the order of the experiments' trials was randomized and was unknown to the human subject. Each trial transmitted 72,000 packets in 300 seconds.

These experiments are by no means exhaustive. Nonetheless, they illustrate how a simple fine-grained path selection heuristic such as the one used by Divert can offer significant performance improvements in terms of reduced delay and loss rate in realistic scenarios.

4.1 Frame Loss Rate

In our mobile environment, Divert performs significantly better than both schemes A and B when the receiver is at R_2 and R_3, for all the values we used for H and T. Figures 10(a) and 10(b) show that at R_2, Divert $H = 1$ and $T = 1$ reduces the average FLR by about 38% from scheme A and 21% from scheme B, and as the receiver moves further to R_3, the loss reductions increase to 56% from A and 26% from B. As predicted in Section 2 (see Figure 3), Divert effectively reduces losses by avoiding burst losses in the wireless channel. Divert performs better with $H = 1$ and $T = 1$ than with $H = 10$ and $T = 5$ because it is more responsive with smaller H and T values.

At R_1, the receiver is much closer to transmitter A than B. As expected, the average FLR of scheme A (2.1%) is much lower than B (15%). Due to the large difference in the average loss rates, it is unlikely that the auto-conditional loss probability of transmitter A ($P(A_{i+k}|A_i)$) will exceed the cross-conditional loss probability of B ($P(B_{i+k}|A_i)$) for any lag k. Figure 10(c) shows that non-$Hybrid$ configurations



(a) Burst loss length

(b) 1-second window loss rate

(c) Sampled channel delay

(d) Number of site switches

Figure 11: CDFs of various measures for the mobile experiments at R_3

of Divert perform slightly worse than schemes A and B. To compensate for the large differences in the average loss rates between the two transmitters, Divert may use different H and T path-switching thresholds for each path. The *Hybrid* case at R_1 uses a more conservative threshold (*i.e.,* $H = 3$ and $T = 2$) for the transmitter with lower FLR (A), and maintains an aggressive threshold (*i.e.,* $H = 1$ and $T = 1$) for the transmitter with higher FLR (B). Figure 10(c) shows that *Hybrid* Divert performs as well as the better transmitter (A) when it uses different path-switching thresholds for different paths. Thus, Divert can adapt remarkably well to extremely asymmetric, dynamic channel conditions (*e.g.,* R_1) by making small adjustments to H and T. In our mobile environment, Divert performs no worse than the best available path when the available paths are extremely different (*e.g.,* R_1). But when the paths are less asymmetric (*e.g.,* R_2 and R_3), Divert drastically reduces the average FLR compared to the fixed-path schemes.

When the receiver is stationary at R_3, Figure 10(d) shows that Divert has a lower average FLR than scheme B but a higher average FLR than scheme A. This is because losses are seldomly bursty in our static environment. In our case, choosing the transmitter that has a lower average FLR (which is AP A) is better than fine-grained selection. However, we believe that there are cases when fine-grained path selection is beneficial even when the receiver is static, *e.g.,* when the channel condition is dynamic or when there is localized interference at the transmitter.

The measured FLR suggests that the performance gains of Divert is much higher in dynamic conditions than in static conditions. To understand Divert's potential gains in other performance aspects, we focus on mobile experiments at R_3 for the rest of this section. In general, the trends described earlier hold for the following evaluation: Divert performs no worse than the best available path at R_1. When the receiver is stationary at R_3, Divert's performance is about the same as the average of the two available paths.

4.2 Burst Loss Length and Window Loss Rate

Figure 11(a) shows the CDF of the burst loss length. Divert is able to significantly cut the tail of the distribution. In particular, the largest burst loss length of Divert with $H = 1$ and $T = 1$ is less than 20 whereas that of scheme A is 52 and that of scheme B is 61.

Some applications such as multicast video streaming require low loss rates over short intervals. Figure 11(b) shows the CDF of frame loss rates over 1-second windows. Divert's distribution of the 1-second window FLRs is much lower (and narrower) than that of both schemes A and B. The worst-case 1-second window loss rate is also much lower: the highest loss rate in a 1-second window for schemes A, B and Divert are 59%, 47%, and 29% respectively.

4.3 Channel Delay

As we have explained throughout the paper, losses are bursty in our mobile environments. Transmitters often experience periods of degraded channel conditions, lasting for several tens of milliseconds. Any frame transmission attempt during such periods leads to failure. We define the *per-packet channel delay* as the difference between the time when a packet is first transmitted in the wireless medium and the time when it is successfully received. Channel delay is a very important metric for voice and video applications that require low one-way packet delay and delay jitter.

To accurately compute the per-packet channel delay, we need the transmit and receive times of each packet. It is technically difficult to synchronize the clocks accurately and consistently among the transmitters and the mobile receiver in each experiment. Thus, we use a sampling approximation to estimate per-packet channel delay. We transmit packets at periodic intervals of I and assume that packet i is sent precisely at $t_i = i * I$. Let r_j be the time when the j^{th} data frame is successfully received, *i.e.,* when the jth data frame logged in the receiver's data trace. Then, the i^{th} channel delay sample d_i is computed as: $d_i = r_j^{min} - t_i = r_j^{min} - i * I$, where r_j^{min} is the minimum r_j in the data trace that satisfies $r_j - t_i \geq I$. In our analysis, we combine all three trials of each experiment to generate $216,000$ delay samples. We used a packet transmission period of $I = 4.17$ ms.

Figure 11(c) shows the CDF of the channel delay samples for the mobile experiment under various schemes when the receiver is at R_3. As shown, all of the Divert schemes have a lower channel delay distribution than the fixed-path schemes (A and B), *e.g.,* in Divert, 98% of the packets have a channel delay less than 15 ms but in the fixed-path schemes A and B, fewer packets (90% and 95%) are transmitted successfully within the same delay. In terms of delay reduction, the $99th$-percentile delay is reduced from 70 ms and 40 ms for fixed-path schemes A and B to 20 ms for Divert.

4.4 Number of Path Switches

We measured the number of path switches that took place in each of our experiments. A high number of switches is indicative of a large number of Divert control messages being sent over the wired backbone. Because a typical wired backbone usually has a much higher capacity than a 802.11b WLAN, the additional traffic caused by Divert's control messages in most cases will produce little congestion in the wired network. However, for low-bandwidth wired backbone networks, a smaller number of AP switches may be more desirable especially with higher-capacity WLAN technologies such as 802.11a.

Figure 11(d) shows the number of AP switches for various Divert configurations. As expected, the number of switches decreases as H increases. The number of switches for the third trial of Divert with $H = 1$ and $T = 1$ is missing because the file containing that data was corrupted.

5. RELATED WORK

Our results in Section 2.2 are consistent with prior WLAN loss measurements in indoor environments in various settings [27, 26, 7, 9]. Our work adds to the considerable evidence in the literature that wireless channel losses often occur in bursts, especially when the receiver is in motion.

Applying spatial diversity techniques in WLANs to avoid bursty losses is a natural evolution in the advances in wireless communication. Cellular phone networks have long used various downlink transmission techniques, often referred to as transmit diversity techniques, which exploit spatial diversity of strategically placed transmitters or antennas to mitigate the effects of multipath and shadowing [6, 20]. In most cases, the techniques are tightly integrated with the physical layer and require stringent synchronization (on the order of 100μs among transmitting elements [14]) made possible by specialized and expensive base station hardware (*e.g.,* an antenna array). Adapting such techniques to existing WLANs designed for low-cost applications will certainly require substantial changes to the physical layer and greatly increase the hardware cost.

In Site Selection Transmit Diversity (SSTD) [8], the client continuously measures the pilot signals emitted by the surrounding base stations and signals the network to perform a soft-handoff to the base station that transmits a pilot with the highest received signal strength. The soft-handoff can happen on a frame-by-frame basis. The idea is similar to Divert except that it relies on the receiver to make switching decisions. The architecture works well in cellular phone networks because medium access is synchronized by the base station. In WLANs such as 802.11b, medium access is randomized and distributed. Thus, feedback information may not be received by the distribution system in a timely fashion for an effective fine-grained switching to occur.

DIRAC [28] is a system framework that facilitates the implementation of an intelligent backbone routing system for wireless networks. DIRAC's design was motivated by a number of channel-adaptive and mobility-aware protocols, such as IP mobility, adaptive FEC, and quality of service, that benefit from link-layer feedback. Divert is an example of such protocols, and may be implemented within DIRAC's system framework.

A distributed radio bridge architecture that exploits spatial diversity in WLANs is proposed in [13]. Divert differs from distributed radio bridges in several important ways. The architecture of distributed radio bridges assumes that all radios in the system communicate in the same frequency. No explicit handoff is required (thus, simplifying mobility) and multiple radio bridges may participate in forwarding packets between the wired and wireless medium (thus, achieving spatial diversity). In comparison, Divert is a hybrid of the traditional cellular architecture and the radio bridge architecture. Frequency reuse is achieved by assigning different frequencies to each primary AP's and diversity is achieved through using secondary APs. Moreover, in the radio bridge architecture, one or more radio bridges are randomly chosen for each downlink frame transmission whereas Divert selects a single transmission site based on loss history.

A previous measurement study evaluated how fine-grained path selection can help reduce bursty losses and thus reduce one-way packet latency in indoor mobile environments [16]. The results show that interactive video applications that have low-latency requirements can significantly benefit from such techniques. Building the results of this work, we i) conduct a much more comprehensive set of experiments and analysis, ii) design a fine-grained path switching system and implement a prototype WLAN that works with standard, unmodified 802.11b clients, and iii) evaluate the system's effectiveness in realistic settings.

6. CONCLUSION

In this paper, we showed that a fine-grained path selection technique for wireless networks can yield substantial performance benefits under the following conditions: i) strong temporal loss correlation within a path in which the short-term (10-100ms) frame loss rate is significantly higher than the steady state frame loss rate, and ii) weak spatial loss correlation across paths. Using a number of real-world experiments on an indoor 802.11b WLAN, we showed that such conditions can occur when the receiver is in motion. Our results show that the simple and practical fine-grained path selection technique proposed in this paper can help reduce loss rates—without consuming extra wireless bandwidth—by as much as 26% compared to a fixed-path scheme that uses the best available transmission path under realistic settings.

Our choice of fixed algorithm parameters (loss history and loss threshold) for fine-grained selection may not be appropriate in some environments. We plan to explore adaptive path selection algorithms so that the scheme is suitable under a variety of dynamic conditions.

7. ACKNOWLEDGMENTS

We thank John Ankcorn, Mitchell Trott, David G. Andersen, John Guttag, Srikanth Kandula, Eugene Shih, Can Emre Koksal, Vladimir Bychkovskiy and the anonymous reviewers for their insightful comments and constructive critiques. We also thank Sanjit Biswas, John Bicket, and Douglas S. J. De Couto for sharing their expertise on hacking 802.11 devices. Finally, we thank Michel Goraczko for always providing attentive technical support.

8. REFERENCES

[1] IEEE 802.11b/d3.0 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, Aug. 1999.

[2] IEEE Standards for Local and Metropolitan Area Networks: Standard for Port Based Network Access Control, Oct. 2001.

[3] Arranz, M. G., Aguero, R., Munoz, L., and Mahonen, P. Behavior of UDP-based applications over IEEE 802.11 wireless networks. In *Proc. of IEEE PIMRC'01* (San Diego, CA, Sept. 2001).

[4] Bharghavan, V., Demers, A., Shenker, S., and Zhang, L. Macaw: a media access protocol for wireless lan's. In *Proc. of ACM SIGCOMM'94* (London, United Kingdom, Aug. 1994), pp. 212–225.

[5] Camp, T., Lusth, J., Matocha, J., and Perkins, C. Reduced cell switching in a mobile computing environment. In *Proc. of ACM MOBICOM'00* (Boston, MA, Aug. 2000).

[6] Derryberry, R. T., S. Gray, D. M. I., Mandyam, G., and Raghothaman, B. Transmit diversity in 3G CDMA systems. *IEEE Communication Magazine* (2002), 68–74.

[7] Eckhardt, D., and Steenkiste, P. Measurement and analysis of the error characteristics of an in-building wireless network. In *Proc. of ACM SIGCOMM'96* (Palo Alto, CA, Aug 1996), pp. 243–254.

[8] Furukawa, H., Hamabe, K., and Ushirokawa, A. SSDT–site selection diversity transmission power control for cdma forward link. *IEEE JSAC 18* (2000), 1546–1554.

[9] Gross, J., and Willig, A. Measurements of a Wireless Link in different RF-isolated Environments. In *Proc. of European Wireless 2002* (Florence, Italy, February 2002).

[10] Holland, G., Vaidya, N. H., and Bahl, P. A rate-adaptive MAC protocol for multi-hop wireless networks. In *Proc. of ACM MOBICOM'01* (Rome, Italy, July 2001), pp. 236–251.

[11] Jean Tourrilhes. Fragment adaptive reduction : Coping with various interferers in radio unlicensed bands. In *Proc. of IEEE ICC'01* (St.-Petersburg, Russia, June 2001).

[12] Jouni Malinen. Host AP driver for Intersil Prism2/2.5/3. `http://hostap.epitest.fi/`, 2003. Version 0.0.1.

[13] Leung, V. C. M., and Au, A. W. Y. A wireless local area network employing distributed radio bridges. *Wireless Networks 2* (1996), 97–107.

[14] Marchent, B., and McTiffin, M. J. Handover and macro diversity for 3rd generation mobile systems within atm fixed networks. In *Proc. of IEEE GLOBECOM'96* (London, UK, Nov. 1996), pp. 1151–1155.

[15] Mishra, A., Shin, M., and Arbaugh, W. An empirical analysis of the IEEE 802.11 MAC layer handoff process. Tech. Rep. 75, University of Maryland, 2002.

[16] Miu, A., Apostolopoulos, J., Tan, W. T., and Trott, M. Low-latency wireless video over 802.11 networks using path diversity. In *Proc. of IEEE ICME'03* (Baltimore, MD, July 2003).

[17] Modiano, E. An adaptive algorithm for optimizing the packet size used in wireless ARQ protocols. *Wireless Networks* (1999), 279–286.

[18] Monteleoni, C., and Jaakkola, T. Online learning of non-stationary sequences. In *Proc. of Advances in Neural Information Processing Systems* (2003).

[19] Phatak, D., and Goff, T. A novel mechanism for data streaming across mutiple ip lnks for improving throughput and reliability in mobile environments. In *Proc. of IEEE INFOCOM'02* (New York, NY, June 2002).

[20] Rappaport, T. *Wireless Communications*. Prentice Hall, Upper Saddle River, N.J., 1996.

[21] Rodriguez, P., Chakravorty, R., Pratt, I., and Banerjee, S. MARS: A commuter router infrastructure for the mobile internet. In *Proc. of ACM MOBISYS'04* (Boston, MA, June 2004).

[22] Sadeghi, B., Kanodia, V., Sabharwal, A., and Knightly, E. Opportunistic media access for multirate ad hoc networks. In *Proc. of ACM MOBICOM'02* (Atlanta, GA, Sept. 2002).

[23] Shin, M., Mishra, A., and Arbaugh, W. A. Improving the latency of 802.11 hand-offs using neighbor graphs. In *Proc. of ACM MOBISYS'04* (Boston, MA, June 2004), ACM Press.

[24] Snoeren, A. Adaptive inverse multiplexing for wide-area wireless networks. In *Proc. of IEEE GLOBECOM'99* (Dec. 1999), pp. 1665–1672.

[25] Tan, G., and Guttag, J. Time-based fairness improves performance in multi-rate WLANs. In *Proc. of USENIX'04* (Boston, MA, June 2004).

[26] Tang, C., and McKinley, P. K. Modeling multicast packet losses in wireless LANs. Tech. rep., Computer Science and Engineering Department, Michigan State University, May 2003.

[27] Willig, A., Kubisch, M., Hoene, C., and Wolisz, A. Measurements of a wireless link in an industrial environment using an IEEE 802.11-compliant physical layer. In *IEEE Transactions on Industrial Electronics* (2003).

[28] Zerfos, P., Zhong, G., Cheng, J., Luo, H., Lu, S., and Li, J. J.-R. Dirac: a software-based wireless router system. In *Proc. of ACM MOBICOM'03* (San Diego, CA, Sept. 2003), pp. 230–244.