AirExpress: Enabling Seamless In-band Wireless Multi-hop Transmission

Bo Chen, Yue Qiao, Ouyang Zhang and Kannan Srinivasan
Department of Computer Science and Engineering
The Ohio State University, Columbus, OH 43210
{chebo, qiaoyu, kannan}@cse.ohio-state.edu, zhang.4746@osu.edu

ABSTRACT

This paper describes the design and implementation of AirExpress, a system that enables in-band wireless cut-through transmission. Unlike wired cut-through, wireless cut-through can reduce latency and improve throughput performance of the network at the same time. In AirExpress, all the forwarders along the cut-through path forward the signal they received immediately without decoding. The hierarchical structure of AirExpress enables its interference cancellation ability to handle all kinds of interference among the radios. Novel MAC and routing algorithms based on cut-through transmission are also proposed to support the realization of AirExpress in multi-hop mesh networks.

AirExpress is implemented on an NI-based SDR platform. Through experiments in the 2.4GHz ISM band, we show a throughput gain of up to 3.4 times with a 4-hop AirExpress system. Trace driven evaluation of AirExpress on the NS3 platform shows an average throughput gain of 2.85 for AirExpress over optimal TDMA.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication

Keywords

Full Duplex; Wireless Networks; Cut-through Transmission

1. INTRODUCTION

Cut-through routing is a strategy in which every router (or switch) in a path between a source and a destination forwards a packet before fully decoding it. This is a technique that is currently possible in Ethernet transmission [1]. Clearly, the end-to-end delay will significantly reduce when cut-through is employed in a wired network. In the history of wireless cut-through, people used to explore the latency gains using control mechanisms to enable cut-through routing using different frequencies [12]. Some similar works [9,14] got inspired by existing wireline network. They imported the pipeline philosophy from chip architecture and highlighted the importance of ad hoc networks when deployed with existing WLANs. Realizing in-band cut-through in wireless networks, however, is challenging in many ways as we will discuss later. However, wireless cut-through not only improves delay performance but also improves end-to-end throughput.

Figure 1(a) shows a path between source A and destination D in a multi-hop wireless network. When traditional routing is used and A is transmitting packets to B, C cannot transmit to D. Thus only one of the three hops can be active. When cut-through is enabled, however, nodes A, B, C and D, can all be simultaneously active. Thus the end-to-end throughput can be increased by up to three times. This shows that wireless cut-through routing can improve delay performance and throughput of the network.

One of the main requirements for in-band wireless cut-through is that a node (e.g., node B in Figure 1) should be able to forward while it is still receiving. Thus a node should be able to support full duplex operation. However, full duplex alone is insufficient to enable cut-through. In the same figure, when C forwards a packet, C's transmission affects B's reception. We refer to this interference as *forwarder interference*(FI). This additional interference also needs to be handled while realizing cut-through.

Forwarder interference cancellation is still not enough to realize cut-through. There will be intra-flow interference from nodes more than one-hop away. This *cross-hop interference*(CHI) also needs to be addressed in order to realize cut-through routing.

Another challenge is the cancellation overhead. Overall, we have self interference, forwarder interference and cross-hop interference to deal with. One nice feature of these three types of interference is that they are due to transmissions of symbols that we intend to receive. However, the interference channels are different for these different types of interference. For example, self interference is between a transmitting and a receiving antenna within a node, while forwarder interference is from the next node's transmitting antenna. The channels need to be measured before we can address these interference sources. Thus measuring these channels at every single node along the path itself could be a big overhead. Furthermore, any measured channel is only valid for a short duration (or coherence time), which is typically of the order of a few hundred milliseconds. Thus the measurement needs to be repeated frequently. Overall, the channel measurement overhead, if not handled carefully,

MobiCom'15, September 7–11, 2015, Paris, France.

DOI: http://dx.doi.org/10.1145/2789168.2790114.

could deteriorate the end-to-end throughput and delay gains in the cut-through implementation.

Furthermore, in order to activate a cut-through path, all the nodes along that path need to be available. A node may be unavailable because it is already participating in another cut-through flow or has an active node in its neighborhood. Thus we need a medium access control (MAC) and routing protocol suite that can dynamically handle cases where the entire path is not available. Otherwise, waiting for the whole path to become free itself could affect the end-to-end delay of a packet. Finally, since every node is simply forwarding a packet without decoding it, noise accumulates along hops in one transmission. If the cut-through path is very long, then the accumulated noise could deteriorate the throughput so that it is even below what is achievable by traditional routing.

This paper addresses these challenges and implements a fully functional wireless cut-through system AirExpress. Our system has the following features:

- Inside each radio AirExpress presents a unified cancellation module that handles all three sources of interference such that the channel training for all interference can happen simultaneously. Along the path, AirExpress forms a Hierarchical Cancellation Structure to protect cut-through transmission from interference saturation.
- It presents MAC and routing protocols needed to support cut-through.
- It presents extensive measurements and trace-driven simulation results to show the viability of wireless cut-through.

Overall, to our knowledge, AirExpress is the first prototype to demonstrate in-band wireless cut-through. A recent work [4] simply argued that wireless cut-through may be realizable. But it did not develop a fully working prototype and address cross-hop interference. Furthermore, it did not present MAC and routing schemes needed to realize cut-through routing either.

2. THE PROBLEM AND MOTIVATION

The cut-through transmission is promising as it increases network performance. With the existing full-duplex techniques, two-hop cut-through transmission is already realized. The relay radio can perform self interference cancellation which follows the design in [3]. This paper, however, focuses on a general wireless cut-through transmission problem. Extending beyond two-hop cut-through, the interference experienced at a node is not confined simply to **Self Interference**(SI). Thus, how to **deal with the additional interference** introduced into the network is the key challenge in the realization of cut-through transmission. This section elaborates the challenges and insights behind our solution – AirExpress.

2.1 Interference in Cut-through Transmission

Let us first look at the case of an ideal three-hop cut-through transmission, as shown in Figure1(a). In this example, A is in the transmitting mode while D is in the receiving mode. B and C in between are the forwarders which relay the signal. We assume that there is no signal outlet other than the three links A-B, B-C and C-D, i.e., there is no interference between A and C, B and D.

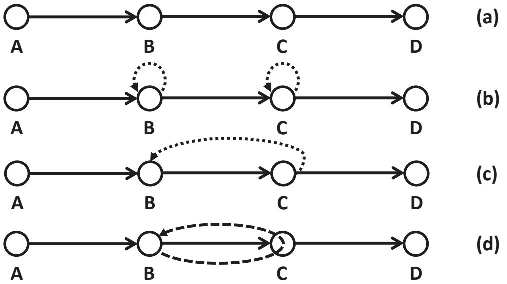


Figure 1: A 3-hop cut-through path. Node A wishes to send packets to D through B and C who are the forwarders. (b)(SI) and (c)(FI) together present the interference components in the ideal cut-through case. (d) illustrates the forwarding channel for the FI at B. Any node in the network will treat the signal from the afterwards nodes as a black box.

2.1.1 Problem

There are three interference streams in the network as illustrated in Figure 1(b) and (c). Cut-through transmission can fail because of any one of the three sources of interference. Among the three interference streams, two are the self interference within B and C separately. They can be removed with the assistance of the full-duplex capability enabled for these two radios. The problem reduces to cancelling the interference from C to B. This type of interference will be noted as **Forwarder Interference**(FI) as it comes from the next hop forwarder.

The nice property of the FI is that the forwarders (C) are sending the symbols that the previous nodes (B) along the path have already seen. In other words, the interfering symbols are known.

One obvious way to realize cut-through is to let every node along the cut-through path forward the previous packet while it is receiving the current one. We refer to this method as **Decode-and-Forward** (D&F) cut-through as the forwarders decode the data before forwarding to the next radios. There are many techniques in the literature supporting D&F [2, 6, 8, 13]. Thus forwarder interference cancellation for D&F can follow the existing techniques. However, D&F has the following drawbacks. First, as packets need to be decoded before transmitted, the latency reduction because of cut-through disappears. Secondly, the throughput benefit would be limited if the cut-through path is used only for a few packets. Consider an extreme case where only one packet needs to be delivered. Then the packet needs to go through one hop after another. The throughput performance would be the same as traditional transmission. Last but not least, the efficiency of the interference cancellation relies on the dissimilarities of the samples in different packets since radios usually use correlation to detect and locate interference. When two similar packets are transmitted one after the other, the cancellation module is expected to treat the (new) data as interference also. Thus D&F's performance would be a function of traffic. Thus, we need to seek for some other ways to deal with the interference.

2.1.2 Intuition

The intuition in AirExpress is that each forwarder tries to measure the channel and model the interference before the transmission so that the *FI* component can be anticipated before the reception of the forwarder interference signal. With a simple subtraction, the useful data can be recovered from *FI*. Since in this way, data will be directly forwarded without being decoded, it can be treated as a method realizing **Amplify-and-Forward** (A&F) cut-through.

The channel that AirExpress needs to measure so as to cancel *FI* encompasses three entities: (i) the wireless channel from B to C, (ii) the amplification circuitry within C, and (iii) the wireless channel back from C to B. This is also shown in Figure 1(d). Sec.3.1 shows the structure of AirExpress forwarders. Specifically, we discuss the cumulative estimation of the three entities in the path, referred collectively as the forwarding channel. Then *FI* component can be predicted based on the signal output of the same radio.

2.2 More Interference in Practical Cut-through Transmission

The ideal channel condition discussed in the above subsection follows the binary interference model assumption: links either interfere with each other or not at all. However, in real scenarios, any two links can interfere with each other. We take a 4-hop cut-through as an example to show how complicated the observed interference is when all links are considered. Consider the topology in Figure 2(a), A is the transmitter, E is the final receiver, and B, C, D in between are the forwarders in order.

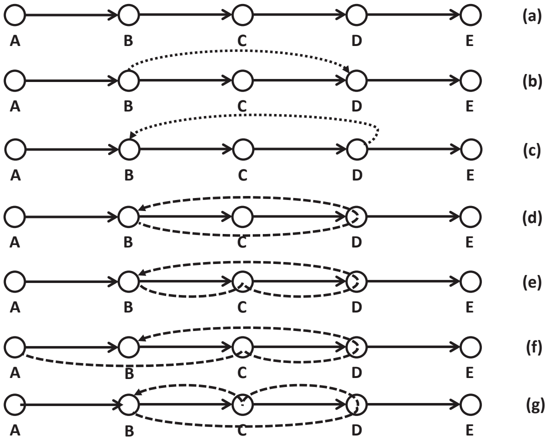


Figure 2: A 4-hop cut-through path. A wishes to send packets to E through B, C and D. Interference in (b) presents the *CHI* from B to D while the one in (c) shows the *CHI* from D to B. (d)(e)(f) are three different signal streams which contribute to the *CHI* from D to B. (g) shows a signal stream of *CHI* path from C to B.

2.2.1 Problem

In traditional routing, if the intra-flow interference is severe, one link can be disabled. However, for the cut-through system, such a setting is contradictory to the idea of enabling the links simultaneously. A working cut-through system should be able to support the successful transmission irrespective of the interference pattern. We refer to the interference introduced by the radios more than one hop away as **cross-hop interference**(*CHI*). Dealing with *CHI* is a significant challenge for wireless cut-through realization.

A straightforward approach could be modelling the channels for all the *CHI* and applying the cancellation following *FI* cancellation modules. However, two limitations render this approach impractical.

- Not all the constituent channels of *CHI* can be modelled. The reason *FI* cancellation works is based on the fact that the signal of *FI* is correlated with the output signal from the same radio so that *FI* can be predicted. However, this does not hold for all the interference components, like the one shown in Figure 2(b). We can see that even the existence of the *CHI* from B to D does not rely on the transmitted signal from D. Thus, the interference is not causal with respect to the transmitted signal from D. Therefore there is no way this *CHI* can be predicted.
- The composite *CHI* is very complicated. Let us consider the *CHI* from D to B as shown in Figure 2(c). There are multiple traces contributing to this *CHI*. Figures. 2(d)&(e) show two traces orginating from B. Since the trace could involve any of the relay nodes within the cut-through, it is expected that the number of traces grows exponentially with the number of forwarders in cut-through transmission. In addition to that, some components observed by B are not causal either. An example of such a trace is shown in Figure 2(f). Moreover, Figure 2(g) shows a case adjacent interference(C→B) also contains *CHI*.

2.2.2 Intuition

As at least the cancellation for some *CHI* is impossible, can we just randomly remove/cancel several *CHI* traces and leave the residual interference as noise? The answer is no. If we do that, it turns out in a short time all the radios would saturate. The reason for this consequence is that *CHI* is bidirectional. As shown in Figure2, there is *CHI* following the transmission direction(from left to right) while some *CHI* in the opposite direction(from right to left). If the interference streams form a loop within the cut-through path, there exists at least one channel condition under which the signal power will keep accumulating until all the radios are saturated. Thus, to validate the cut-through transmission, *CHI* cancellation needs to cut enough signal paths so that the interference streams within the cut-through transmission is loop-free.

To achieve this goal, interference cancellation in AirExpress is designed to follow a **Hierarchical Cancellation Structure**, in which a forwarder judges its descendant forwarders as a black box and cancels all the signals originating from its transmitter, retransmitted within the black box and received by its receiver. Specifically, in the 4-hop example shown in Figure 2, cancellation block in node C cancels the signal originating from C, forwarded by D and received back at C while node B cancels the signal originating from B, retransmitted by C and/or D and received back at B. Combined with the forwarder cancellation block design in AirExpress forwarding channels are modelled to capture this effect as shown in Figure 3. Two insights show that this design satisfies the cut-through *CHI* cancellation requirement: (i) All the *CHI* cancellations happening in our Hierarchical Structure are causal. This is simply due to the fact all the *CHI* components are originating from the radio's transmitter. (ii) This design structure guarantees loop-free property of interference streams. This is because, with any potential interference loop, there is one node at which all the other involved forwarders are the descendants and thus, the interference signal stream will stop at this node and not loop forward further.

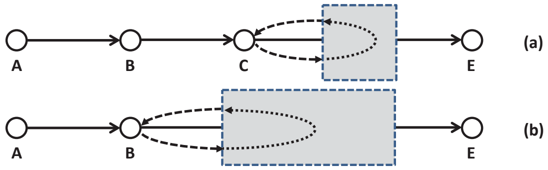


Figure 3: Hierarchical Cancellation Structure for a 4-hop cut-through example. (a) shows the forwarding channel for node C. (b) shows the forwarding channel for node B.

The remaining *CHI* components which can not be cancelled stay within the system. However, they are all copies of the original signal and thus, can be treated as multipath profile. For a n-hop cut-through system ($sr_1r_2\cdots r_{n-1}e$, where s is the start transmitter, e is the end receiver while $r_1r_2\cdots r_{n-1}$ are the n-1 relay radios), the longest path would involve $\frac{n(n-1)}{2}+1$ hops[1]. Thus it is quite critical to reduce the latency within each forwarder. Within each forwarder, 10ns latency for *SI* analog cancellation module and 50ns latency for ADC and DAC is unavoidable. For the digital cancellation (for *SI*, *FI* and *CHI*), we utilize the causal nature of the FIR filter to provide a 0 latency forwarding within the digital domain. This feature is also adopted in the FastForward [3] work. In a WiFi OFDM system, CP(cyclic prefix) duration is 800ns and multipath duration exceeds this time limit will introduce inter-symbol interference. Thus, in WiFi settings, theoretically 5-hop cut-through is the setting with highest supported hops. However, if we can change the OFDM configuration or apply AirExpress in other applications, like LTE, cut-through transmission with more hops is also achievable.

3. REALIZING CUT-THROUGH TRANSMISSION

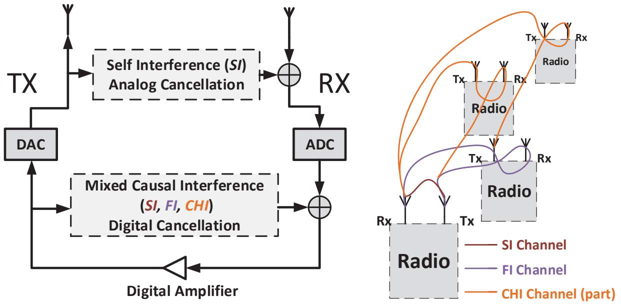
In general, a cut-through path involves a source node, a destination and a sequence of forwarders. The source node transmits the signal, while the destination node tries to decode the signal. Forwarders forward the signal in sequence. The source and destination nodes run as regular wireless radios capable of encoding and decoding messages. The interference component varies with the position of forwarders. Originally, to deal with varying interference components, different forwarders should be designed according to its position within the cut-through path. However, with the hierarchical cancellation structure already introduced in the previous section, the interference cancellation requirement is reduced to the causal interference from the descendent forwarders respective to the radio position within the cut-through path. In the rest of this section, we will first focus on the design of one forwarder radio in AirExpress and then, present AirExpress protocol realizing the cut-through transmission.

3.1 Forwarder Structure

Figure 4(a) illustrates the high-level block diagram of a forwarder node. As shown in the figure, in the forwarding mode, a node only uses the digital-to-analog-converter (DAC), analog-to-digital-converter (ADC) and radio frequency (RF) parts of its radio physical layer. The only processing for data stream a node does in the digital domain is interference cancellation, and then amplification of the received signal. It bypasses all of the other traditional physical layer data processing such as decoding and error correction.

The forwarder is built on a wireless full-duplex radio: As we show in the diagram, both the RF cancellation and digital cancellation of *SI* are also there. The *FI* and *CHI* cancellation modules are additions to the typical wireless full-duplex radio. Three signal characteristics enable us to implement a forwarder node with this design.

First, the interference is causal to the transmitted signal from the forwarder. This point has been illustrated in Sec. 2. Secondly, the power of *FI* and *CHI* is comparable to the regular received signal as they all come from the neighbouring radios: the two signals are within the resolution of analog to digital converter (ADC). This is different from self interference which is many orders larger than the received signal from a different radio. Thus digital cancellation is enough to meet the cancellation requirement in most of the cases. Thirdly, the received signal after interference cancellation is directly amplified and sent to the transmitter side for forwarding. This operation takes fixed time. Note that this forwarded data signal will be observed as *FI* or *CHI* at its ascendant. This means that the timing of the interference is usually small and predictable.



(a) (b)

Figure 4: (a) Overall block diagram of a forwarder in AirExpress. (b) Illustration of the interference received at each receiver.

As the channel is predictable, the forwarding channel has a lot of similarities as the self-interference channel. Thus the forwarder cancellation module in the system can be designed following the structure of digital cancellation in self-interference cancellation.

The digital amplification can be set to any value and it will not affect the cancellation performance. In AirExpress the amplification is adjusted according to the signal power from ancestor radio to maintain a constant output power at each forwarder node.

Is Frequency Offset an Issue for the Interference Cancellation?

It is widely known that any two wireless nodes will have a frequency offset: the notion of 2.4GHz could be differ-

---

[1] $sr_{n-1}r_{n-2}\cdots r_1r_{n-1}r_{n-2}\cdots r_2\cdots\cdots r_{n-1}r_{n-2}r_{n-1}e$ is the path

ent as the crystals used to generate these frequencies have imperfections [6]. A frequency offset between two nodes, if uncorrected, will rotate the phase of a signal continuously over time [11]. This continually changing phase can make channel estimation between two nodes extremely difficult: Even if the channel is relatively stable over time, the rotating phase will affect these estimates. To solve this problem, it is common to use a known sequence as a preamble for every packet. A receiver uses this preamble to estimate the frequency offset so that it can correct the consequent phase rotation.

Note that the digital cancellation for self interference (in full-duplex radios) does not have this problem. This is because the transmit and receive radios belong to the same node and so there is no frequency offset. The *FI* and *CHI* in AirExpress, however, are introduced by other nodes –the descendent forwarders. Therefore, in Figure 4(b), any two radios will have a frequency offset. Does this imply that a frequency offset correction mechanism is required at all the radios in the network? The answer is ***no***. Actually, the interference channel defined in AirExpress is a ***frequency offset free*** channel, even though there is frequency offset among multiple nodes.

To see why the interference channel is frequency offset free, let us first look at the signal between two forwarders. Assume forwarder B is the descendent of forwarder A. Center frequencies at A and B are f_A and f_B, respectively and $f_A \neq f_B$. Thus the signal received in forwarder B is affected by frequency offset $f_B - f_A$. Since nodes in AirExpress do not decode packets, the frequency offset is not corrected. Therefore, Node B directly forwards the signal with the frequency offset. As it goes through the channel B to A, an additional frequency offset $f_A - f_B$ is applied to the signal when it is received by Node A. Thus the two frequency offsets cancel out with each other and the FI/CHI signal received in node A is frequency offset free. The same argument can be applied to the stream going through three or more forwarders. Assume signal starts with A and comes back to A after going through $B_1, B_2, \cdots, B_k$, the frequency offset of the interference can be expected as $f(B_1) - f(A) + \sum_{i=1}^{k-1}(f(B_{i+1}) - f(B_i)) + (f(A) - f(B_k) = 0$. Thus all the interference is *frequency offset free*. It makes the *FI* and *CHI* cancellation robust to frequency offsets found in distributed radios.

3.2 AirExpress Protocol

So far, we have discussed the physical modules within each node to realize cut-through. However, each node still needs to know its role as a forwarder and the signal it requires to forward. To achieve this, AirExpress relies on the cooperation among all the nodes involved in the transmission. How to control the forwarders and process the cut-through transmission cooperatively is still a problem. In this subsection, the details for the process of AirExpress will be presented. We assume in this section the transmitter S has the data, knows the cut-through destination D and is aware of the order of all the k forwarders $R_1, R_2, \cdots, R_k$ in between. A routing algorithm to obtain this information will be presented in Sec. 4.3. Another preparation we need is that in the network, each node needs to generate a PN sequence according to its address[2] and assume the PN sequence for the node with ID X is PN_X. Also, we assume a pre-reserved PN sequence, PN_{AE}, which is reserved as AirExpress indicator and known to the whole network.

To support the transmission along an AirExpress path, three tasks need to be accomplished:

1. Member Notification. Transmitter S is the only node being aware of the cut-through plan before the transmission. It needs to notify the other radios for their roles within the AirExpress system. It is hard because radios cross multi-broadcasting domains and thus, multiple wireless messages are needed. A comprehensive and efficient notification process is quite important.
2. Channel training. The cancellation module within each forwarder needs to be tuned so that it can cover the cancellation of all the intended interference signal.
3. Forwarder Release. When the transmission is going on, all the forwarders blindly forwards the signal without decoding. When the transmission is done, forwarders need to be aware of the situation and stop forwarding data.

3.2.1 Before Transmission: Notification & Training

The first two tasks need to be done before the transmission. To decrease the overhead of these tasks, AirExpress processes them jointly. For member notification, AirExpress tries to notify the forwarders one after another, following the order of cut-through path so that all the radios will be notified. Although the feasibility of the intended interference cancellation has been proved by our Hierarchical Cancellation Structure, channel training is still not trivial as the number of potential interference paths defined in the structure is still exponential with respect to the number of radios. To achieve this goal, AirExpress extends the 'hierarchical' feature from cancellation design to implement our channel estimation protocol. The forwarders start only when all the descendant forwarders are already operating in forwarding mode while the ancestors are silent. The running of the process can be separated into two phases:

Sequence of Notification: First of all, the source node sends the combined PN sequences of PN_{AE} and all the nodes in the path: $PN_{AE}PN_{R_1}PN_{R_2}\cdots PN_{R_k}PN_D$. When R_1 detects PN_{AE} followed by its PN sequence PN_{R_1}, it will transmit the sequence excluding its own PN sequence as $PN_{AE}PN_{R_2}PN_{R_3}\cdots PN_{R_k}PN_D$. The following forwarders will perform the same process: whenever anyone detects the occurrence of PN_{AE} plus its own PN sequence, it joins the path. In this way, the nodes in the route sequentially join the AirExpress system. At the same time, each forwarder also records the PN sequence of the next hop. For instance, R_1 records PN_{R_2} and R_k records PN_D. In this way, all the forwarders are aware of PN sequence of the next hop radio, which will be used in the next phase. Thus we introduce PN_{AE} to ensure we retain the notification in order of the participating forwarder nodes in spite of cross-hop interference. Under the scenario without PN_{AE}, forwarders would just join the transmission based on the reception of its own PN sequence. It is highly possible that more than one radio detect their PN sequences for the same transmission. In this way, the order information of the forwarders is lost.

---

[2] PN sequence, Pseudo Random Sequence has already been widely used in recent years [2,10]. It is a predefined sequence of information. A receiver is able to detect this sequence by using correlation. The advantage of PN sequence is that it can be detected even under interference and its duration is short which leads to a small control overhead.

Backward Training: At the beginning of this phase, all the forwarders are already aware of their relative positions in the cut-through path through the knowledge of next-hop PN sequence(based on the recording process illustrated in the previous phase). For channel training, D broadcasts its PN sequence first. When R_k captures this information (D is R_k's next hop), it starts transmitting its training sequence and measures the parameters for *SI* RF cancellation and the digital cancellation parameters for *SI*, *FI* and *CHI* interference streams. When the training is done, this forwarder will set the digital gain according to the received signal power level (from the PN sequence in the previous phase) so that its output power is maximized. Afterwards, the forwarder sends out its PN_{R_k} as an indicator to trigger its previous radio (transmitter or forwarder) and immediately switches to forwarding mode and then begins forwarding. R_{k-1} follows the same procedure to do the training and notify the previous radio. In the end, when S detects PN_{R_1}, it knows the training process is done and all the forwarders are forwarding their received signal.

Throughout these two back and forth phases, whenever a forwarder tries to measure the interference channel, all its descendant forwarders are in forwarding mode while its ancestor forwarders are waiting for the trigger signal. Thus, it satisfies our requirement for hierarchical channel training.

3.2.2 After Transmission: Forwarder Release

After the data transmission is complete, the source node S has to release all the nodes involved in forwarding. To do this, S just sends PN_{R_1}. When R_1 detects its PN sequence, it stops forwarding and sends out PN_{R_2}. The same process repeats sequentially at all the forwarders until D receives PN_D as an indication of the end of the current AirExpress session.

AirExpress SNR Implications

It is straightforward that the channel of an AirExpress path is limited by the worst channel in its path. The reason is that there is no error correction module inside the forwarders. So, the physical layer throughput cannot exceed this lower bound. In addition to that, when signal is forwarded, the noise is also forwarded. So the cut-through path also accumulates noise. On the other side, when all the forwarders transmit the signal, they can be treated as multipath components. As long as CP (cyclic prefix) in an OFDM system is able to cover the duration of multipath component, the power from different multipath will also accumulate in the final receiver. In this way, it alleviates noise accumulation in AirExpress.

4. MEDIUM ACCESS CONTROL AND ROUTING

Now that the PHY capability for wireless cut-through transmission is realized, we can enable the cut-through transmission in a network. However, the underlying carrier sensing module in traditional MAC cannot perform the collision avoidance or even detection for cut-through transmission. Thus to enable multiple simultaneous transmissions in the network, a cut-through friendly MAC algorithm is needed. In addition to that, the traditional routing algorithms for mesh network were not designed with consideration of cut-through capability. So, we need to revisit the routing layer as well. In this section, MAC and routing layer algorithms supporting AirExpress capability will be presented.

4.1 Medium Access Control (MAC)

Traditional carrier sensing in wireless radio can only detect the transmission within the broadcast domain of that radio. AirExpress extends beyond a single broadcast domain. Thus cut-through requires carrier sensing beyond a single broadcast domain. A naive solution is to have each participating AirExpress node sense the channel and piggyback this information down to its ancestor nodes. However, this introduces intolerable latency. We achieve this by skipping this step altogether. The transmitter just acts as if the channels are free. The forwarders themselves will cut the length of the cut-through transmission in-between if they can detect transmission around themselves. This means that an n-hop AirExpress could end up to be a k-hop if the (k+1)th forwarder can not forward the signal. The AirExpress MAC protocol is shown in Algorithm. 1.

The major steps of the algorithm following the protocol are described in Sec. 3.2. The carrier sensing modifications are integrated. Nodes will transmit or react to forwarding requests only when channel is free (Line 4). Here, we add a Δt to the carrier sensing to prevent the sequence notification phase, as discussed in the previous section, from generating a false busy channel trigger. When this scenario happens at the next hop forwarder/destination, the current radio can immediately detect collision if it fails to detect the expected PN_{AE}(Line 14,29). When it happens at the transmitter, transmission is aborted(Line 16). For forwarder, the current radio will turn to be the destination for this AirExpress path(Line 31). The last modification to the AirExpress protocol is that the destination will transmit an additional PN_{AE} to trigger the process in the previous forwarder (Line 41).

4.2 Virtual Hop

It seems that combining the source, destination and all the forwarder nodes in between into a single cut-through path is the right way to deploy cut-through in a mesh network. However, through the analysis in previous sections, several realistic situations have already been shown to limit its performance:

(1) Worst link bottleneck: Cut-through routing does not take full advantage of the diversity of the link quality in a path. Since no intermediate node can modify the received symbols, the source needs to pick the transmission rate suitable for the bottleneck link along the path. This is the data rate at which all the links in that path will operate. It means that every link will carry the packets at a data rate supported by the bottleneck link even if the other links are far superior. In traditional routing, however, a packet can be transmitted at higher data rates on other links and sent at a low rate on the bottleneck. This frees up the medium sooner around (superior) links than around the bottleneck allowing other flows to start early.

(2) Noise accumulation: With cut-through, more number of links are enabled simultaneously. Therefore, throughput is improved compared to traditional routing. However, since every node in cut-through needs to forward without decoding, noise accumulates in every hop. This leads to deteriorating SNR as the packet traverses towards the des-

Algorithm 1: AirExpress Protocol

```
1  Input:
2  1. Global PN sequence $PN_{AE}$. Unique PN sequence $PN_{this}$ for current Radio.
3  2. For a specific transmission request at transmitter S: Its own PN sequence $PN_S$. PN sequences of forwarders/destination for the next AirExpress path $PN_{R_1}$, $PN_{R_2} \cdots PN_{R_k}$, $RN_D$.
4  Initial State:
5  while Carrier Sense($\Delta t$) do
       // Operate when channel is free
6      if Detect $PN_{AE} PN_{R_{this}}$ then
7          goto Forward State.
8      else if New Transmission Request then
9          goto Transmit State.
10     endif
11 endwhile
12 Transmit State:
13 Transmit($PN_{AE} PN_{R_1} PN_{R_2} \cdots PN_{R_k} RN_D$)
14 WaitingFor($PN_{AE}$)
15 if Timeout then
16     goto Initial State.// Abort transmission
17 else
18     WaitingFor($PN_{R_1}$)// Training Done
19     Transmit(Data)// Data Transmission
20     Transmit($PN_{R_1}$)// Channel Release
21     goto Initial State.
22 endif
23 Forward State:
24 if No signal after $PN_{this}$ then
25     goto Receive State.// Destination Node
26 endif
27 Calculate digital gain based on PN sequence Power.
28 Transmit($PN_{AE} PN_{this+1} PN_{this+2} \cdots PN_{R_k} PN_D$)
29 WaitingFor($PN_{AE}$)
30 if Timeout then
31     goto Receive State.
       // Timeout, forwarder turn to be receiver
32 else
33     WaitingFor($PN_{R_{this+1}}$)// Descendent Training Done
34     Train for cancellation channel and start forwarding.
35     Transmit($PN_{this}$)// Trigger Ancestor Training
36     WaitingFor($PN_{this}$)// End of Data
37     Transmit($PN_{this+1}$)// Channel Release
38     goto Initial State.
39 endif
40 Receive State:
41 Transmit($PN_{AE} PN_{this}$)// Trigger Channel Measure
42 Receive(Data) & WaitingFor($PN_{this}$ )// End of Data
43 goto Initial State.
```

tination. If the deterioration is significant, then it can lead to a throughput even lower than traditional routing. Additionally, an erroneous packet is dropped at an intermediate switch in traditional routing. While in cut-through, errors go undetected until a packet is decoded by its destination. The propagation of errors wastes network resources.

(3) Latency limitation: It has been shown in Sec.2.2.2, the duration of multipath grows quadratically with the number of hops. Although an OFDM system can tolerate multipath effect if the FFT and CP length can be configured. For any fixed setting, there is always an upper bound.

It seems that, in a real scenario, cut-through routing over a long path is not efficient. To resolve this limitation, we introduce the concept of virtual hop. Traditionally, one hop refers to a transmission from one node to its neighbouring node. Comparatively speaking, a virtual hop represents a transmission from one node to a node several hops away. Within a virtual hop, AirExpress is carried out. Thus between a source and a destination, there could be multiple virtual hops. We refer to such a cut-through routing structure as *virtual hop* structure.

By limiting the number of hops in AirExpress, virtual hop mitigates the bottleneck effect and error propagation problem and provides an upper bound for the multipath duration of the system. At the same time throughput improvement is still significant.

4.3 Routing Algorithm

Current routing table in the mesh network tells only the next hop receiver. However, AirExpress protocol requires the transmitter to know the path before the transmission (Line 3, Algo. 1). Thus it is important to come up with a routing algorithm to support AirExpress protocol. In addition to that, the cut-through capability has changed the throughput for a certain route. For example, a simple routing scheme considering the minimum number of hops as the metric to determine the routing is inapplicable for AirExpress transmissions, since there is no fundamental difference between 1, 2 or 3 hops - it is a single virtual hop. In this subsection, we will present a routing protocol based on AirExpress PHY, which provides the routing algorithm to minimize the latency between the source and the destination when employing AirExpress. As the delay between the source and the destination reflects the time during which data stays within a network, it is a good index of the efficiency of the network resource utilization. Thus it should increase the overall network throughput performance.

Consider a mesh network with allowed virtual hop transmission of up to k hops. Our routing algorithm in the following subsection will show how it maintains and updates the routing table so that it would support AirExpress PHY and MAC. For any radio in the network and a given destination node, this routing table provides the list of forwarders and destination of the next virtual hop. Our computation is based on one approximation that the cut-through channel is equivalent to the worst channel within the path.

Routing Table Structure: Let us consider the routing table in the radio R. Usually for a certain destination, routing table will provide one path (or trace). This is also true for our routing algorithm. However, in our routing table, multiple traces are recorded for the same destination. If the data is originated from R itself, then one path is sufficient. However, R could also be a forwarder in an AirExpress path. When R is a forwarder, the trace to the destination depends on the SNR at R from its ancestor. The following example illustrates this point.

Assume there are two routes X and Y from a radio to the same destination. X consists of only one virtual hop and takes $200\mu s$ to directly deliver a 1000 bytes packet to the destination while Y takes two virtual hops with $75\mu s$ each. To deliver data direct from this radio, Y is a better choice. However, when the radio is a forwarder and the channel to its ancestor is bad, delivering the packet alone in that link takes $200\mu s$. Combining this link with the first virtual hop in X and Y, X still takes $200\mu s$ while Y takes 275 μs. Thus X is the better route in this case.

Therefore any path which could potentially outperform all others based on the possible ancestors' channel conditions will be recorded. The paths are recorded in the format as shown in Figure 5. For any path X, $X.i$, $X.d1$, $X.d2$ and $X.list$ are used to denote the items in the format shown in the figure. For any two paths X and Y, if X strictly outperforms Y (X performs better than Y under any ancestors' channel conditions), Y will be removed. We denote this as $X \geqslant Y$. Under our assumption that cut-through channel is equivalent to the worst channel within the path, it is not hard to get the sufficient and necessary condition as follows:

$$\begin{cases} Y.d2 \geqslant (X.d1 + X.d2) & : X.i > Y.i \\ (Y.d1 + Y.d2) \geqslant (X.d1 + X.d2) & \\ \&\&\ Y.d2 \geqslant X.d2 & : X.i = Y.i \\ (Y.d1 \geqslant X.d1\ \&\&\ Y.d2 \geqslant X.d2) & \\ ||\ Y.d2 \geqslant (X.d1 + X.d2) & : X.i < Y.i \end{cases}$$

Directly from the definition, the transition property *strict outperform* also holds as $X \geqslant Y\ \&\&\ X \geqslant Z \Rightarrow X \geqslant Z$.

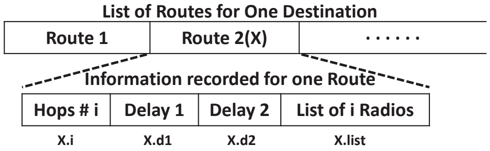


Figure 5: Recording for one destination within the Routing table. It contains a list of traces. For each trace, it records the number of hops within the next virtual hop (i), time to transmit unit size packet in the virtual hop (Delay 1), time to transmit unit size packet to the destination after next virtual hop (Delay 2) and the trace for next virtual hop(List of Radios).

Trace Output: When radio R has the data to transmit, it will search for the trace X in the routing table list that minimizes $X.d1 + X.d2$.

Distributed Routing Table Update: First of all, radios can update their routing table based on new channel measurements. And they periodically broadcast their routing tables to their neighbours. Radios can also update their routing tables based on the neighbours' routing table. Assume radio R receives the routing table from its neighbour R_{nei}, for any route X in R_{nei}, R could generate two potential traces Y_1 and Y_2 based on X and the time d to transmit a unit size packet to R_{nei}. Y_1 treats R_{nei} as the destination for next virtual hop and thus ($Y_1.i$, $Y_1.d1$, $Y_1.d2$, $Y_1.list$) = (1,d,$X.d1 + X.d2$,RR_{nei}). Y_2 treats R_{nei} as the first forwarder in the next virtual hop and thus ($Y_2.i$, $Y_2.d1$, $Y_2.d2$, $Y_2.list$) = ($X.i + 1$, $max(d, X.d1)$, $X.d2$, $R_{nei}X.list$). Of course, if $X.i + 1 > k$ the second item will be directly ignored since it violates the virtual hop setting. These new traces will be added to the routing table if no existing traces *strictly outperform* them and all the traces they *strictly outperform* will be removed.

Initialization: When the routing table is generated the first time, all the radios just broadcast their routing table for enough rounds until the routing tables become stable for most of the radios. Since it only needs to be done once, the overhead can be amortized during the running of the system.

5. IMPLEMENTATION

5.1 Hardware Platform

We have built a prototype of AirExpress on the NI-based SDR platform. Each radio has one RF chain and NI PXIe-1082, an RTOS-based controller. The transceiver RF chain consists of NI-5791 (RF frontend and data converter module) and NI PXIe-7965R (Xilinx Virtex-5 FPGA) for baseband processing. The ADC and DAC of each transceiver chain sample at 130 MSps. The ADC/DAC resolution is 14-bit. Our full duplex implementation follows the design in [7]. Transmit and receive antennas are 15cm apart. Together with the Balun cancellation circuit and preconditioning block to cancel the non-linear signal, our RF module supports 68dB self-interference cancellation over 20MHz band. The digital module for self interference alone can provide another 42dB cancellation. With the combination of both the RF and the digital modules, our full-duplex system supports 110dB cancellation. The details of our implementation and performance is presented in our previous work [5].

5.2 Implementation Details

All the radios can transmit and receive OFDM signal with QPSK, QAM16 and QAM64 constellations and convolutional codes with coding rate 1/2, 2/3 and 3/4. Each radio is preassigned a unique PN sequence and a global PN sequence. Two correlation blocks are implemented on the FPGA for the detection of these two PN sequences. Radios support transmission of 20MHz band and operate on the 2.4GHz. Output power of the radio varies from -20dBm to 20dBm.

There is one harware limitation on our AirExpress implementation: NI-5791 introduces $0.5\mu s$ latency in both transmitting and receiving trace. Thus the lower bound of the latency introduced in one forwarder in our implementation is $1\mu s$. As we know the radio latency varies from platform to platform. For instance, WARP only introduces a delay up to 50ns as shown in [3]. So the latency performance is not the fundamental limitation for the deployment of AirExpress. However, to realize AirExpress in the current platform, the communication system needs to tolerate a $1\mu s$ latency. Therefore we keep the same ratio (1/4) for the CP length of the OFDM system, while the FFT size of OFDM system is set to 512. Thus theoretically, the CP is able to tolerate multipath latency up to $6.4\mu s$. To model the channel for *SI*, *FI*, *CHI*, our digital FIR filter is implemented with a duration of 5 μs so that most of the signal can be covered. In this way the physical layer throughput supported by the system matches the setting in WiFi.

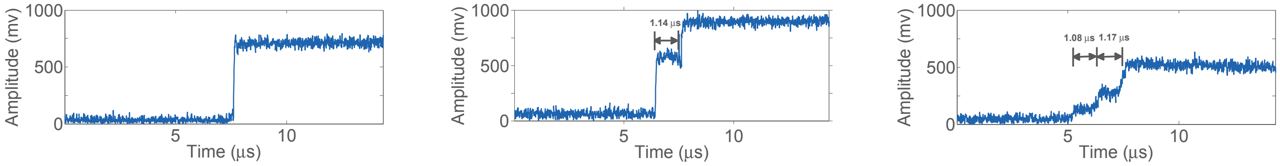
Our implementation of AirExpress is deployed on 5 such radios, with which we can show the working of a 4-hop AirExpress system.

6. EVALUATION

6.1 Micro-benchmark

6.1.1 Forwarder Interference Cancellation

We have shown that the *SI* cancellation in digital domain is 40+dB (see results in [5]). In this experiment, we evaluate the *FI* cancellation performance. To study the capability of *FI* cancellation in AirExpress , we use two AirExpress



(a) 1 hop (b) 2 hops (c) 3 hops

Figure 6: Power of the received sine wave in 1-hop(a), 2-hop(b) and 3-hop(c) AirExpress system. The multi-path component can be seen as discrete steps.

radios. In the first step, the first radio trains for self interference and then switches to the forwarding mode. In the second step, we turn on the second radio and let it train for the interference from the first radio and perform the interference cancellation. We vary the distance between these two radios so that the power of *FI* changes. Both radios are transmitting with power 0dBm. The experiment is conducted in a relatively quiet environment and we measure the cancellation performance over a 100-ms duration after the *FI* channel measurement.

The *FI* cancellation performance is shown in Figure 7. We can see that our digital cancellation module can cancel around 36dB interference, which is much better than the existing correlation based cancellation [2, 8]. In addition we can see that the *FI* from a radio beyond 4 meters is canceled to the noise floor in our system.

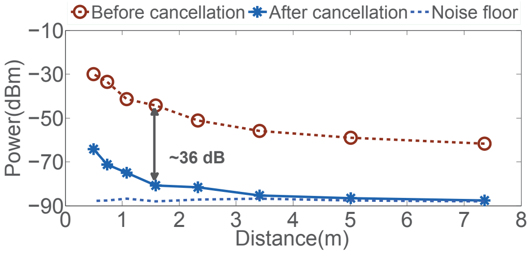


Figure 7: Cancellation performance of the interference with distance.

6.1.2 Cancellation for Interference Going through Multiple Forwarders

Although 36dB cancellation is much better than the other cancellation technique, compared to the *SI* cancellation capability (42dB), there is a performance loss. We believe this is due to the clock drift in different radios. So it is natural to wonder whether the clock drift will affect the cancellation performance further when multiple forwarders are involved. To answer this question, we measure the cancellation performance of the forwarding interference which travels through the most number of forwarders in a multi-forwarder setting. To isolate interference caused by intermediate forwarders, we use multiple center frequencies. For example, to evaluate the interference through two forwarders we use three radios A, B and C. Radio A transmits at f_1 and receives at f_3 while radio B transmits at f_2 and receives at f_1, and radio C transmits at f_3 and receives at f_2. A transmits a signal, and B and C are forwarders. We implement *FI* cancellation on A. Therefore, at A, we can measure *FI* cancellation of the interference forwarded through B and C.

Motivated by the previous experiment, we place all the radio close to each other to get the worst case *FI* cancellation performance. The performance is shown in Figure 9. The cancellation stays stable irrespective to the number of radios. Surprisingly, from 1 hop to 4 hops, on average, performance only drops by 3 dB. Given the fact that interference power decreases as the interference stream goes through more radios, the cancellation requirement also decreases with more radios involved. Therefore it is expected the interference stream going through multiple forwarders will eventually be entirely removed.

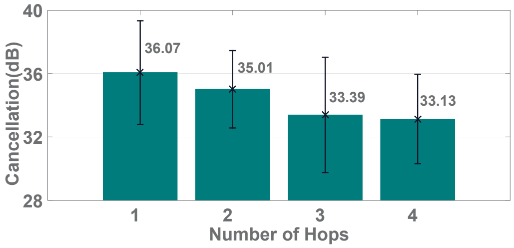


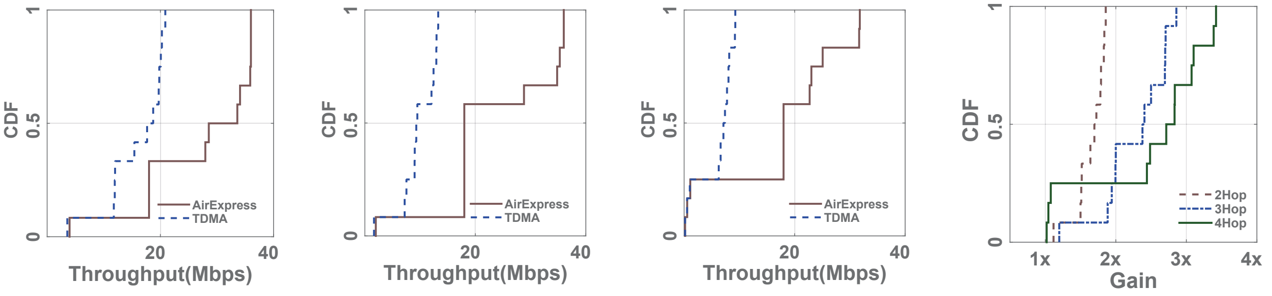
Figure 9: Cancellation performance of the interference going through multiple radios.

6.1.3 Multi-path Profile

To illustrate the multi-path profile in AirExpress system, in this experiment, we first transmit a narrowband sine wave of duration 0.2ms after the training. With 1-hop to 3-hop AirExpress setting, we can see the difference in the power of the received signal in Figure 12(b). The small steps before the received sine wave shown in the figure are caused by the cross hop interference. We can see that each step lasts for around $1\mu s$, which is mainly due to the latency introduced by each forwarder.

6.1.4 Effect of External Interference

As shown in our MAC protocol 4.1, the transmission in the AirExpress has already taken into account the carrier sensing result. So that multiple transmissions can happen simultaneously in the network. However, collision avoidance is not always perfect. At the same time, WiFi channels are also experiencing interference from other technologies, like Zigbee signal or cordless microphones. What happens when external interference happens after the carrier sensing period of AirExpress is the case we want to discover. To illustrate the scenario, we articulate the collision in the following way. We will transmit 10 packets in a roll periodically and carrier sensing is done every cycle. Thus the 10 packets in a roll can be seen as a continues running AirExpress transmission.



(a) 2 hops (b) 3 hops (c) 4 hops (d) AirExpress over TDMA

Figure 8: Figure (a),(b) and(c) depict the throughput of AirExpress and TDMA respectively with 2-hop, 3-hop and 4-hop routes. Figure (d) depicts the throughput gain of AirExpress over TDMA with 2-hop, 3-hop and 4-hop routes.

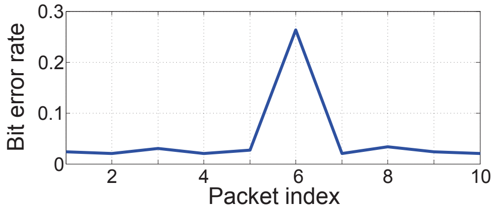


Figure 10: Bit error rate for ten continuous packets. An external packet is introduced as interference aligned with the 6th packet.

We externally introduce an interference packet synchronized with the 6th packet in each cycle. And such cycle is repeated for 100 transmissions. The average bit error rate for all the 10 packets is shown in Figure 10

It is expected packets 1 to 5 are not affected by the interference as shown in the Figure 10. The good phenomena is although Packet 6 can not be recovered due to collision, the interference does not affect the ongoing transmission when the collision signal stops. This is quite important for the AirExpress design. If we can pack multiple packets together after one channel measurement, the mac overhead can be easily amortized.

6.2 System Evaluation

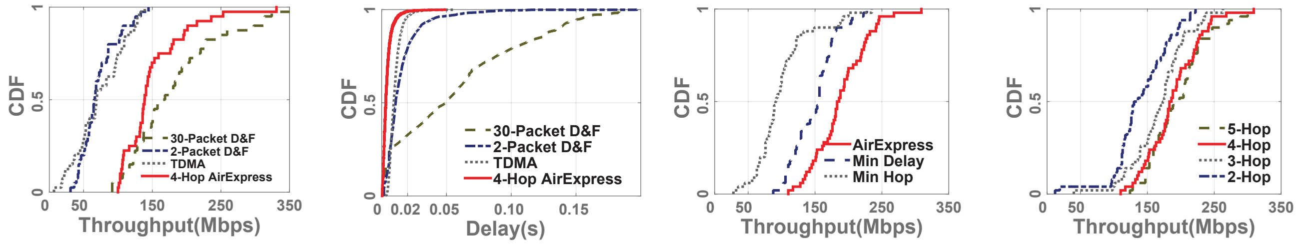
We evaluate the throughput performance of AirExpress via experiments in an indoor setting. 20 locations are chosen as shown in Figure 11 for nodes' placement. For several source-destination pairs picked from the 20 locations, we choose the multi-hop traces which maximize the throughput in TDMA setting. For the same source-destination pair, we apply AirExpress on the same trace to perform cut-through transmission. Rate adaptation is not implemented in the system. To demonstrate the system performance, we calculate the throughput by comparing the performance of transmission under each date rate. To be more specific, for each transmission (AirExpress or single hop transmission in TDMA setting), we deliver 500 packets with different physical layer rates (1/2QPSK, 2/3QPSK, 1/2QAM16, 2/3QAM16, 1/2QAM64, 3/4QAM64). The one that maximizes throughput is chosen as the date rate for that transmission. All the carrier sensing overhead is ignored for the demonstration of throughput comparison. The result is shown in Figure 8.

Figure 11: Floor plan for the system evaluation. AirExpress radios are placed in these 20 possible locations.

The throughput result for TDMA is calculated based on the assumption of a perfect scheduler that could assign different time to different hops to maximize its throughput performance. Even without considering the contention overhead of the TDMA scheduling, we can see the improvement of AirExpress is significant. For the 2-hop case in Figure 8(a), AirExpress increases the throughput by 164% and it provides a median gain of 1.71. In 3-hop scenario as shown in Figure 8(b), AirExpress improves the network throughput by 227% and provides a median gain of 2.39. In the 4-hop case, the throughput improvement is 246% as shown in Figure 8(c) and the median gain is 2.75. We can see in all figures, several cases show limited performance gain of AirExpress. This is usually in the scenario where one link in the trace is relatively bad. As the worst channel in the path dominates the throughput performance in both AirExpress and TDMA cases, it is expected their performance is similar.

6.3 AirExpress in Large Mesh Network

To evaluate AirExpress in a large mesh network, especially to evaluate the routing performance, we use NS3 to do the simulation. In the experiment, we randomly place



(a) Throughput (b) Delay Performance (c) Different Routing (d) Virtual Hop Size

Figure 12: Performance comparison in NS3 simulation. (a)(b)show the general throughput and delay performance among TDMA and 4-hop AirExpress and *D&F* schemes. For *D&F* we illustrate the performance when 2/30 packets are combined in one set transmission. (c) shows the performance of 4-hop AirExpress working with different routing scheme. (d) shows the performance of AirExpress with different virtual hop settings.

100 nodes in a 1000m×1000m field. All the radios are operating on output power 20dBm. Among these 100 nodes, we randomly pick source-destination pairs to generate traffic. Measurements of the cancellation performance from experiment are fed into the network. As shown in Figure 7 and Figure 9, the cancellation is not perfect. There is residual interference after interference cancellation and the residual interference is related to the channel condition. We take this into account and feeds the measurement from our experiment into the simulator.

To evaluate the performance of AirExpress, in addition to the AirExpress protocol, we implement two more schemes – an omniscient TDMA scheme and a perfect *D&F* scheme. In the TDMA setting, a central controller is assumed to be aware of the global information. It can coordinate the transmission among all the potential links with zero mac overhead. The carrier sensing and contention overhead are all removed. In the *D&F* setting, we assume a *D&F* scheme capable of canceling out all the *FI* and *CHI* components perfectly.

For the routing choices, besides AirExpress routing described in section 4.3, we implement another two routing schemes: the minimum hop scheme and minimum delay scheme. Minimum hop scheme always finds the path with the minimal number of hops, in which each hop is guaranteed to be a link with SNR higher than a basic threshold. Minimum delay scheme grants the routes, through which if a packet is transmitted one hop after another, the transmission time is minimized. The supported virtual hop size ranges from 2 to 5.

6.3.1 AirExpress vs TDMA and D&F

We compare the throughput and delay performance among TDMA, *D&F* and AirExpress. We generate 100 sets of topology and within each topology we generate 10 traffic requests. Routing for TDMA setting is based on the minimum delay routing. AirExpress is operating in the 4-hop virtual hop configuration with corresponding AirExpress routing scheme. *D&F* uses the same routing scheme as AirExpress. There are two configurations of *D&F*: 2-packet *D&F* and 30-packet *D&F*. The number of packets here denotes the number of packets in one batch of transmission.

The CDF of the experiment performance is plotted in Figure 12, in which Figure 12(a) shows the throughput performance while Figure 12(b) illustrates the delay performance. Compared with TDMA scheme, the improvement of AirExpress is clear. AirExpress outperforms TDMA by 2.85 in the throughput performance. At the same time, the latency is reduced by three times comparing to TDMA.

The performance of *D&F* is quite bad through our emulation. The 2-packet *D&F* throughput performance is comparable to TDMA, while its latency is higher than TDMA scheme. 30-packet *D&F* shows its potential in the throughput performance. It outperforms AirExpress by 20 percentage in average. However the latency trade-off is also quite severe. To deliver one packet from source to destination, it takes 15 times more delay comparing to AirExpress and 5 times more delay than the TDMA. Partial reason for the bad performance of *D&F* is that the routing scheme does not match the *D&F* protocol. However, the policy of *D&F* itself hides the fundamental reason. When the number of packets in a batch is small, there is not much improvement of the throughput. At the same time, the time a *D&F* path reserved is proportional to the hop numbers. Thus there are less simultaneous transmissions in the network. The aggregation of these two facts ends up in a worse performance even comparing to TDMA. When the number of packets in the batch increases, the latency performance is extremely high, because packets can only be resolved when all the packets in one batch arrive at the destination.

6.3.2 Routing Algorithm in AirExpress

To illustrate the importance of a matching routing algorithm in presence of the cut-through capability, in this experiment, we compare the throughput performance of 4-hop cut-through in different routing schemes. To make minimum hop or minimum delay routing work with AirExpress, we grant nodes the ability to get the route and the ability to stop at any of the middle forwarders just like the function in AirExpressrouting protocol. Figure 12(c) shows the comparison among their performance. AirExpress routing outperforms minimum hop by 1.88 times and increases the throughput by 1.26 compared to minimum delay.

6.3.3 Virtual Hop Size in AirExpress

When we vary the maximum number of hops in a virtual hop in AirExpress, the network throughput varies as shown in Figure 12(d). We can see that more than four hops within a virtual hop will not further increase the network performance.

This observation somehow matches the expectation. It is true that with more hops involved, more transmissions can be processed together. However the equivalent channel of the AirExpress path becomes worse. The improvement of the cut-through transmission disappears at some point. We believe the optimal virtual hop size is a somehow a topology and flow demand related statistic.

7. RELATED WORK

Cut-through routing is mainly a wired networking concept. In wired cut-through routing, packets are routed after the header of the packet is decoded so that latency is reduced. Along with the delay reduction, cut-through routing also reduces the necessity of storing packets which requires additional resources.

In the history of wireless cut-through, people used to explore the latency gain and the control mechanism to enable cut-through routing using different frequencies [12]. Some similar works [9, 14] got inspired by existing wireline network. They respectively imported the pipeline philosophy from chip architecture and highlighted the importance of ad hoc networks when deployed with existing WLANS.

After the introduction of wireless in-band full-duplex techniques in recent year, literature starts to explore the opportunity of cut-through transmission within channels of the same center frequency. [3] looks at the the opportunity to filter the signal at the forwarder so that in a 2-hop cut-through trace, the signal could constructively add up at the receiver so that the coverage range of WiFi could be extended. The main focus of [3] – constructive signal adding up could potentially improve the performance of the AirExpress transmission in the two-hop setting. However it requires all the transmitters to be aware of the channel information among the radios. [4] shows the possibility of in-band cut-through transmission for more than three hops. But, it treats all the *CI* components as interference, and thus the performance will be quite bad in the real world deployment

AirExpress, however, is the first system realizing 2+ hop in-band cut-through transmission in a practical channel setting. It considers and deals with cross-hop interference. Also efficient MAC and routing algorithms in a mesh network supporting AirExpress are presented.

8. CONCLUSION AND FUTURE

In this paper, we realize the first fully functional in-band wireless cut-through system AirExpress. AirExpress is able to deal with self interference, forwarder interference and cross-hop interference efficiently altogether at the same time. A MAC protocol of AirExpress supporting its running and carrier sensing in the multiple broadcast domains is provided. A routing algorithm adapted to AirExpress PHY and MAC properties is also presented.

We believe with the cut-through capability enabled, the fundamental description of transmission pattern in wireless ad hoc network is challenged. Although AirExpress's protocol going back to the circuit-switched networking looks like a step back in the networking technology evolution, actually it is a big step forward. Our previous belief in the packet-switched networking in the wireless communication area reduces the spacial reuse opportunities. Enabling more transmissions in the same space is just the fundamental resource AirExpress seeks to take advantage of. Although it is true that current AirExpress system has several remaining issues for the real world realization, like how to do the dynamic data rate selection, its potential in the performance improvement from the network perspective will surely drive us to the future realization.

Acknowledgments

We would like to thank our anonymous shepherd and reviewers for their comments and suggestions to improve our paper. The work reported in this paper was supported in part by the NSF under Grant CNS 1254032.

9. REFERENCES

[1] Cut-Through and Store-and-Forward Ethernet Switching for Low-Latency Environments. `http://www.cisco.com/c/en/us/products/collateral/switches/nexus-5020-switch/white_paper_c11-465436.html`.

[2] Bansal, T., Chen, B., Sinha, P., and Srinivasan, K. Symphony: Cooperative Packet Recovery over the Wired Backbone in Enterprise WLANs. In *ACM MOBICOM* (2013).

[3] Bharadia, D., and Katti, S. Fastforward: Fast and constructive full duplex relays. In *ACM SIGCOMM* (2014).

[4] Chen, B., Tummala, G. K., Qiao, Y., and Srinivasan, K. In-band wireless cut-through: Is it possible? In *ACM HotWireless* (2014).

[5] Chen, B., Yenamandra, V., and Srinivasan, K. Flexradio: Fully flexible radios and networks. In *USENIX NSDI* (2015).

[6] Gollakota, S., and Katabi, D. Zigzag decoding: Combating hidden terminals in wireless networks. In *ACM SIGCOMM* (2008).

[7] Jain, M., Choi, J. I., Kim, T., Bharadia, D., Seth, S., Srinivasan, K., Levis, P., Katti, S., and Sinha, P. Practical, real-time, full duplex wireless. In *ACM MOBICOM* (2011).

[8] Katti, S., Gollakota, S., and Katabi, D. Embracing wireless interference: Analog network coding. In *ACM SIGCOMM* (2007).

[9] Lee, S., Banerjee, S., and Bhattacharjee, B. The case for a multi-hop wireless local area network. In *IEEE INFOCOM* (2004).

[10] Magistretti, E., Gurewitz, O., and Knightly, E. 802.11 ec: Collision Avoidance Without Control Messages. In *Proc. of ACM MOBICOM* (2012).

[11] Rahul, H., Kumar, S., and Katabi, D. JMB: Scaling Wireless Capacity with User Demands. In *ACM SIGCOMM* (2012).

[12] Ramanathan, R. Challenges: A radically new architecture for next generation mobile ad hoc networks. In *ACM MOBICOM* (2005).

[13] Tan, K., Liu, H., Fang, J., Wang, W., Zhang, J., Chen, M., and Voelker, G. M. Sam: Enabling practical spatial multiple access in wireless lan. In *ACM MOBICOM* (2009).

[14] Velayutham, A., Sundaresan, K., and Sivakumar, R. Non-pipelined relay improves throughput performance of wireless ad-hoc networks. In *IEEE INFOCOM* (2005).