SELF INTERFEROMETRY: SELF INTERFERENCE CANCELLATION AND ITS APPLICATIONS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Dinesh Bharadia
March 2016

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Sachin Katti) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Balaji Prabhakar)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Tsachy Weissman)

Approved for the Stanford University Committee on Graduate Studies

Abstract

Wireless radios are typically half-duplex radios, and hence, the current wireless networks are Time Division Duplex(TDD) or Frequency Division Duplex (FDD). Full-duplex for wireless communications is considered impossible i.e. radios cannot transmit and receive at the same frequency at the same time. If we could achieve full duplex radios we won't need TDD or FDD strategy, we could potentially double the spectral efficiency. The fundamental challenge in achieving full duplex radios, when a radio is transmitting while simultaneously trying to receive (hear another radio) on the same frequency, it cannot. It's own transmission acts as a very strong self-interference.

In this dissertation, we present the design, prototype and implementation of full duplex mimo radios. In particular, we built the first single antenna per chain full-duplex MIMO radios for 2.4 GHz WiFi-PHY i.e. to achieve an m-chain MIMO transceiver we need only m antenna. We design novel cancellation algorithms and circuits that reduce all self-interference to the noise floor and enable full-duplex MIMO PHY with almost no loss.

The cancellation algorithms designed for full duplex themselves are of independent interest and apply to many other interference problems in wireless. We exploit this to build a full duplex relay which is the first one to provide both range extension and increase the capacity, is oblivious to ongoing transmission the source and the destination don't even realise that relay exist. Further, we build on top of the cancellation BackFi; a system that provides high throughput connectivity using backscatter to IoT devices at a very lower power. BackFi backscatters all ubiquitous ongoing WiFi signals to provide connectivity. Thus, providing connectivity without using extra spectrum just leveraging full duplex link.

The cancellation, in essence, cancels all the reflections from the environment of self transmitted signal, inferring the reflection from cancellation provides us information about the environment. Towards the end, we abstract this information with a platform of Self-Interferometry, which provides with a unique way of looking at environment using wireless signals instead of light. Thus building a camera with wireless radios.

Previously Published Material

This dissertation is based in part on the following publications :

- [41] : Dinesh Bharadia, Emily McMilin, Sachin Katti. **Full Duplex Radios** in *ACM Sigcomm 2013*
- [39] : Dinesh Bharadia, Sachin Katti. **Full Duplex MIMO Radios** in *Usenix NSDI 2014*
- [38] : Dinesh Bharadia, Sachin Katti. **FastForward: fast and constructive full duplex relays** in *ACM Sigcomm 2014*
- [36] : Dinesh Bharadia, Kiran Raj Joshi, Sachin Katti **Full Duplex Backscatter** in *ACM HOTNETS 2013*
- [72] : Kiran Joshi, Dinesh Bharadia, Manikanta Kotaru, Sachin Katti, **WiDeo: Fine-grained Device-free Motion Tracing using RF Backscatter,** in *Usenix NSDI 2015*
- [37] : Dinesh Bharadia, Kiran Raj Joshi, Manikanta Kotaru, Sachin Katti, **BackFi: High Throughput WiFi Backscatter,** in *ACM SIGCOMM 2015*

Acknowledgments

My journey at Stanford has been incredible. I not only got a chance to drop out of Stanford for two years (like some cool Stanford students do) and work towards commercializing my research at a startup, but also come back to Stanford and complete my Ph.D., and hence this acknowledgement. I got an opportunity to bring my research into the real world as a product at Kumu Networks with substantial support from industry (45 Million $ funding [13]). The experience at Stanford has been a roller coaster ride filled with thrill and excitement. I feel not only acknowledging people during my stay at Stanford but also everyone who helped me get here. I also feel it will be best to write this acknowledgement as a story my story.

I hail from a small town of Ichalkaranji and belong to a Marwari (business) family. My father (Ramnivas Bharadia) is a relaxed and astute entrepreneur. He is easy-going and never pressured me in my studies, as some parents in India might. My parent's thought was that even if I didnt do well in my studies, they will help me financially in my life. My parents will always speak very proudly about my achievements, even if they were small ones. It might not always the best when your parents are relaxed and they are happy at whatever level you perform in the school. This is why I had to always push myself and my parents were always proud of my endeavors. When I was young, there was a small incident, my dad was teaching me to calculate compound interest but by iteratively adding interest into the principal amount. After a while, I gave him a simple formula to calculate it using one equation. My Dad was happy that I am good at math and so will be good at business as well. I was not really a good student in my primary school days. My father always said it is fine as long as I can get passing grades.

I remember when I was in 4th grade, my Dad was talking to his cousin (Ramprasad Bharadia) and praising my math skills. His cousin told him that her daughter is good at math too and is participating in a math competition and he encouraged me to compete as well. I believe this was first turning point in my life. I was introduced to a great teacher (Mr. Pankaj Urunkar), who encouraged me and greatly helped me in learning mathematics. Through him, I also met lot of motivated students, including my (cousin sister) Deepika Bharadia, (cousin brother) Ashish Kabra, (cousin sister) Rekha Kabra, Sunil Chapparwal, Kunal Todi, Deverath Golangade and others. Ashish is genius in mathematics and he is my mentor, friend and philosopher throughout my life. Many

times I feel his influence on my life is much more than many of my teachers. Deepika and Kunal were top students in the class and helped me in my studies. I still dont know what they saw in me in those early years and their support and encouragement helped me to do well in my school (not just in mathematics, but also in other subjects). A very important person in my life that I want to thank is my younger brother (Manish Bharadia). Many times I feel he believes in my abilities more than myself! All this great support and encouragement lead to me getting few scholarships including: Maharashtra Talent Search Scholar and National Talent search scholar. Since then, I have followed my heart to science, even though I come from a business family. My parents were happy and supportive of me. They used to say, "we need an engineer in our family, all entrepreneurs are not good."

The second turning point towards the academic career was getting admitted into IIT Kanpur, with the support and encouragement, of my teacher Vaman Gogate and support of my elder brother and my fathers role in making me chose to go to IITs instead of other engineering colleges. In India, you have to go through a competitive exam of IIT-JEE (Joint Entrace Exam) for entering into IITs. My elder brother introduced me to an amazing teacher Vaman Gogate, more popularly known as Gogate Sir. He provided guidance and mental energy to prepare for IIT-JEE exam. He created an inspiring atmosphere, where he motivated us to work hard and learn from each other (in a way that junior students can learn from the senior peers in the group). This provided me a unique opportunity to learn many things from my elder cousin brother (Ashish Kabra) and his friends Kokil Jain, Sushil Mantri, Abhinav Rawanka, Anup Walvekar. I felt nurtured in this group and I learnt how to teaching myself concepts in science, become a fast-learner and to apply the concepts in agile way to solve the problems. This skillset has been useful for me for my entire life. I am very thankful to all the students and Gogate Sir, who transformed my learning curve to another level. I was able to successfully pass the IIT-JEE exam and joined IIT Kanpur in 2006.

The third turning point towards the research was opportunities at IIT Kanpur. When I started at IIT Kanpur, my simple goal was to get a well-paid job after graduation. I was not too keen on higher studies at that point. At IIT Kanpur, I was exposed to a whole new world of opportunities in academics, research, sports and other fields. I had amazing teachers for every subject, and my habit of self-teaching formed ideas for my research. I not only learned how to methodically approach a technical research problem but also learned about current affairs, moral issues, gaming, politics in the numerous debates I had with some of the amazing colleagues at IIT. I also got opportunity to build circuits for fun at E-club and this made me learn how to build practical things and I participated in various campus level to international level programming contests. I was then selected by MITACS program for a summer research internship in Vancouver, Canada at University of British Columbia. This was also an opportunity for me to travel outside India for first time and have good fun summer. However I also really enjoyed the research project that I worked with Prof. Vijay Bhargava, Gaurav Bansal, and Praveen Kaligneedi. I published a research paper

from my internship work. Most importantly, I realized that I would love to conduct research and hence decided to puruse Ph.D. in Engineering. This decision was very unpopular with my extended family, who believed that it is actually hard to find a job with a Ph.D. degree. My parents were very supportive of my decision to pursue higher studies. I was at the top of my class at IIT Kanpur and also managed to publish a paper in my summer internship. Prof. Sachin Katti got interested in working me and I was admitted to Stanford after an interview with Sachin. I was also awarded with Thomas and Sarah Kailath Stanford Graduate Fellowship for my Ph.D. studies. Last five years at Stanford had been an incredible experience. I started working on building full duplex radios which was an open problem at that time. I joined a group of students who were working to make full-duplex radios a reality. Specifically, I had the opportunity to work with Mayank Jain, Jung Il Choi, Prof. Philip Levis, Kannan srinivsan, Tae Min Kim. I got the opportunity to generate ideas, analyze them mathematically, then design algorithms and prototype and demonstrate it in practice. We presented our demonstration to people at Industry and Academia. This was a perfect job for me, as I was getting a chance to realize my ideas in practice and build them while working with some of smartest people in the world. I was in heaven and at times even made me believe in God. I did all this research work as part of the Stanford Networked Systems Group (SNSG) which is lead by Prof. Sachin Katti. Sachin is an amazing advisor, the best way to describe him is a modern guru. He guided both my research pursuits (which ideas are great and which are not, which ideas could create impact and which won't, and accordingly chose the problems to work on) and the worldly affairs (life lesson, startup's, negotiation, motivating other students, how to work with other colleagues and motivate them). All this is very essential in constantly evolving world. He has infact motivated me to the extent, that I want to follow his foot-steps to become a researcher and an entrepreneur. In our first meetings, I was always hesitant to say some of the ideas that I knew right away wont work. I used to take my time and provide rigorous prove it analytically or by experiments before saying so. He encouraged me to challenge his ideas in our discussions and this lead to faster discussions and fruitful results. We both realized that I am able to get better guidance from him when we both challenge our process on how to approach the problem. This always resulted in the best outcomes. This is infact very different from a traditional gurukul discipline, and hence, I refer to him as a modern guru. We also had a lot of fun while working on these many complicated research problems. I remember numerous incidences where my navity on the existence of utopian research community made us laugh for days. Some of the most typical messages on chat from Sachin varied from innocuous "how is it going?", to after an hour from a discussion of an idea "Hey, Is it working?", to "Dinesh, do you have the results, can we pull this paper?". Sachin has always been very supportive of my ideas, at times they were just cool with no applications and over time we thought of applications where we can apply these ideas. He encouraged me to take my research out of the lab and build a commercial solution. He supported my decision to take leave of absence from Stanford and work at Kumu Networks. We also got a chance to work together at Kumu

Networks. Kumu was a fun work place with amazing environment and resources to design numerous full duplex designs and experiment with them. I again felt I was in right place! I greatly enjoyed working with Mayank, Jung Il, Sachin, Steven, Jeff, Ivan, Ricardo, Joseph, Rob, Steffen. Sachin was CEO at Kumu Networks, and he placed enormous trust in me to see if a design would work in a practical setting. He relied on my gut feeling at times. After two years of working at Kumu, I had made significant contributions and I felt that other can now take over to commercialize my research. I made a decision to quit Kumu Networks to pursue a faculty and an entrepreneur career (very similar to Sachins). I would continue to admire Sachin and strive to do better. I would also like to give my heartfelt thanks to my committee members Prof. David Tse, Prof. Arogyaswami Paulraj, Prof. Balaji Prabhakar, Prof. Tsachy Weissman. It was an incredible experience to my research on full-duplex radios to some of the renowned experts in the world. All these years, SNSG lab was a great place to hangout. In my first year, I spent all my time solving puzzles with Aditya and Manu, and many times disturbing them when they were trying to do actual research. Later years I was very focused on my research and at times spent endless nights in the lab and just came to my dorm for taking a shower! I learned a lot from every member of SNSG. Kiran Joshi is good at algorithms and has strongest work ethic. He was one of the amazing collaborator at my time at Stanford. Many times before the deadlines, I have worked at Kirans apartment and we used to be inseparable during these days. He has a two year old daughter (Rewa), and she considers me part of their family. Kiran is the collaborator who doesn't make excuses and gets things done. Aditya is super agile with the math and it amazes everyone the speed at which he can follow wireless theory papers. Manu can clearly articulate the system issue and can ask continuous questions (which can be little irritating), but these very questions lead to great system contributions! Steven is quick at rephrasing your ideas in a very simple way that everyone can follow and to this date some of his talks inspire me. Jeff has innate ability to analyze hardware design quickly and analyze feasibility. Mayank is the star at building FPGA logic and understanding arguments rapidly and explaining it to everyone. Jung Il writes code at the speed which I wish I can write one day! This group provided invaluable input on every paper that I wrote and on every talk I gave during my Ph.D work. Rakesh is also an amazing researcher and very articulate. Beyond this cohort, I had the chance to have the scintillating conversation with Kanthi, Sandeep, Kartik, Chinmoy, Yash, Ritesh for research and they provided me valuable comments. This group has an incredible capability to quickly grasp main concepts in a paper or a talk and understand the most important arguments and results. This provided various constructive criticism and was very helpful for me. I started at Stanford with my close batchmates from IIT Kanpur Kartik Venkat, Abhishek Arora and many other friends from all over the world. I had a lot of fun at Stanford, because of all these good friends. Each has helped in an unique way to make my journey fun. I have cherished discussing cultures to various course related homework. I remember many occasions having fun discussions with Kartik, Abhishek. Kartik was my housemate, and is a dear friend, philosopher and has helped me in a variety of aspects of life at

Stanford, and even today remains the same. He always has excellent perspective and gives excellent advice for working with faculty or advisor at Stanford. One of great things that Kartik told me (and which I remember till date), "for every one hour your advisor spends with you, one should spend at least 10 hours to prepare for that meeting". We had the group of friends who hangout a lot included Kartik, Ritesh, Aditya, Manu, Chinmoy, Yash, Preyas, Rahul Seth and Rahul Sharma. This group provided me a social support, and motivated and encouraged me to do better everyday. They have been helpful to point my weakness in academic writing, spoken English and helped me improve. They have provided constructive criticism in a positive way for my faith, and my involvement of my parents in many of my life decisions. We also numerous times (South, North and Rajasthani Indian food) in our first year and in later years we had a lot of meals together outside of campus. We use to frequent to this south Indian food place Madras Cafe. We have watched a lot of TV series, movies and played board games. I would remember watching movies in Packard 204 on Friday evening, sleeping all through the movies and then trying to summarize the story of the movie even when I was sleeping in it! Beyond Stanford community, I met a lot my batchmates and good friends in the bay area mostly working at the tech giants like Google and Facebook. I have enjoyed intellectual discussions with Gaurav Bansal on variety of topics. Gaurav is a good friend, a mentor to me, helping out every step of my stay at Stanford, from providing advice on research to life decision. Aditya Somani has been an awesome friend who has given a truthful opinion from an external perspective from Stanford bubble. He also has provided the Marwari perspective over the years. I had great discussions with Ashish Bhatia on topics like, "why to do a Ph.D.?". Dinner and hanging out with Ritesh, Nirmesh at Google was also always lot of fun. Without everyone, the stay wouldn't have been as much enjoyable and productive!

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Traffic demand over wireless networks has been consistently increasing over the past decade. The traffic demand is fueled by a variety of applications (IoT, sensors, etc.). Each of these application has crowded the world of wireless with devices. It's only going to get more crowded in the near future. In the last decade itself, the wireless devices have increased by a factor of 10. However, the resource (spectrum) to meet the traffic demand is not able to keep with it. Spectrum is the valuable resource needed for connecting these devices, but unfortunately, we only have limited spectrum. The efficiency of the spectrum is measured by the number of bits communicated per unit time per unit bandwidth. The average spectral efficiency has only doubled over the last decade, and the curve is flattening because we are running out of new link layer technologies that can provide gains. LTE (the latest link layer technology) is getting closer to Shannon channel capacity limits. Thus, the spectrum is not able to keep up with the growth of wireless devices.

Current techniques to utilize the spectrum are inefficient. Fig. 1.1 shows the frequency division duplex mechanism for using spectrum; one of the current popular method to use the spectrum. In this example, spectrum1 is used to receive (lower frequency) and the spectrum2 to transmit. One might wonder, why not use the spectrum1 for both receive and transmit, as shown at the bottom of Fig. 1.1. In essence, both receive and transmit on spectrum1 and free the spectrum2. This mechanism would immediately double the spectral efficiency. So, why aren't wireless devices full duplex (is the capability to be able to transmit simultaneously and receive at the same time on the same frequency)?

Full duplex radios for wireless communication has been generally considered impossible. A wireless textbook by Prof. Goldsmith from 2005 states,

> "It is generally not possible for radios to receive and transmit on the same frequency band because of the interference that results." Andrea Goldsmith, Wireless Communications [59]

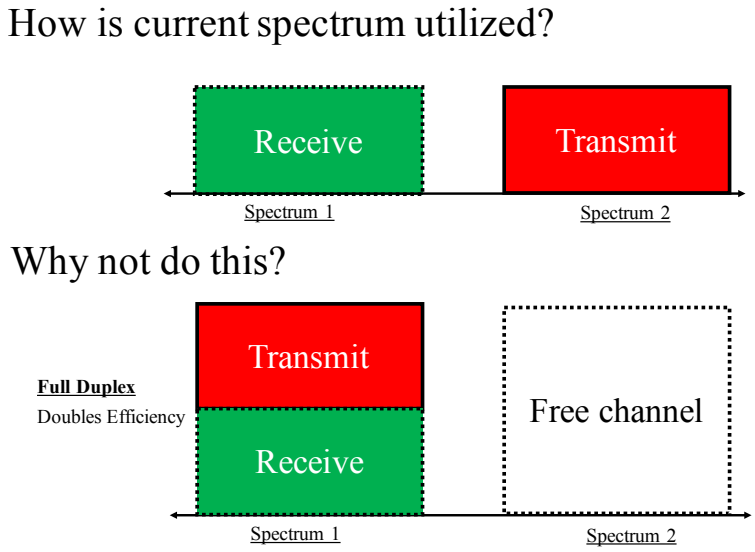

Figure 1.1: Shows the current frequency division duplex (FDD) mechanism of using the spectrum at the top. The bottom of the figure shows full duplex, which doubles the spectrum efficiency compared to FDD mechanism.

This is a long held assumption in wireless, which has impacted the design of the entire wireless system from physical layer to the higher layers. The key challenge as stated in the quote, is that radios very own transmitted signal act's as a very strong interference called as self-interference. This self-interference limits the radios to receive anything on the same frequency while they are transmitting on it. Another contemporary wireless textbook by the Prof. Tse and Prof. Vishwanath states that [117],

> "In addition to resource sharing between different users, there is also an issue of how the resource is allocated between the uplink and the downlink. There are two natural strategies for separating resources between the uplink and the downlink: time division duplex (TDD) separates the transmissions in time and frequency division duplex (FDD) achieves the separation in frequency. Most commercial cellular systems are based on FDD."

To paraphrase, in this context authors did not even consider the possibility of transmitting and receiving in the same frequency and at the same time. The fact is that over time, this practical limitation of radios has become fundamental or natural assumption in wireless system design.

In this thesis, *we invalidate this assumption. Specifically, we present the design, implementation and prototype of first fully functional full duplex mimo radios, that can achieve the theoretical doubling of capacity. This research has transcended into commercial product at Kumu Networks, where this technology is currently undergoing transformation into a commercial product at Kumu Networks.*

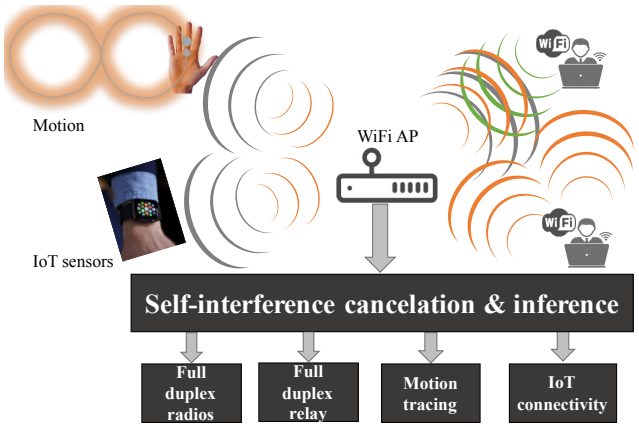

Figure 1.2: WiFi AP transmits the gray signal to the client on the top, while the signal is received by the client on top it can transmit back to the WiFi AP. The gray signal is also reflected back by Walls, furniture, humans or IoT sensors back to the WiFi AP. These reflections can be inferred to build applications shown. Finally, the WiFi AP can be used as a relay from client (source) on the bottom to the client on the top (destination).

Scalable full duplex mimo radios is just the tip of the iceberg, built upon the self-interference cancellation. However, self- interference cancellation has many applications beyond full duplex radios. Let me show you with the help of an example: Let's say we have a WiFi access point (AP) transmitting to a WiFi client as seen in Fig. 1.2. The transmitted signal not only travels to the WiFi client on top, but is also reflected off the walls and furniture, reflections from the humans, the human motion would also provide specific reflections, the transmitted signal also reflected from the IoT sensors. These all constitute self-interference; we can potentially exploit the self-interference to infer about human motion, or the IoT sensors can encode information in the reflections which later can be decoded using the self-interference. In summary, self-interference has a wealth of information. Beyond the inference of self-interference we can exploit capability to cancel self-interference, and then just use it to repeat the received signal from client at the bottom and extend the range to the client on the top in Fig. 1.2? Beyond self-interference cancellation, this thesis builds systems that can exploit the wealth of information in the self-interference. Specifically, we build a relay called FastFoward, which can extend both range and throughput which is atypical as relay typically extend range only. We also build a WiFi backscatter communication system called BackFi, which can allow IoT sensor to communicate with low power and achieve throughput as high as 1Mpbs at 5m range by just reflecting WiFi signals. Beyond this thesis, we have built system called Wideo, which allows motion tracing for human without them having to wear any device. In essence, it provides the platform to analyze the reflections or environment with the wireless radios as the camera. In the following section, we would describe each of this contribution in more detail.

1.1 Thesis Contributions

In this first chapter of this thesis, I will not only present a working prototype of full duplex radio but also show that this technology has transcended in to commercial deployment. The key contribution is a new cancellation technique that completely eliminates all self- interference. We build on this cancellation technique and demonstrate a fully functional in- band full duplex radio that uses a single antenna. The first chapter presents the design and implementation of the first in-band full duplex WiFi radios that can simultaneously transmit and receive on the same channel using standard WiFi 802.11ac PHYs and achieves close to the theoretical doubling of throughput in all practical deployment scenarios. Our design uses a single antenna for simultaneous TX/RX (i.e., the same resources as a standard half duplex system). We also propose novel analog and digital cancellation techniques that cancel the self interference to the receiver noise floor, and therefore ensure that there is no degradation to the received signal. We prototype our design by building our own analog circuit boards and integrating them with a fully WiFi-PHY compatible software radio implementation. We show experimentally that our design works robustly in noisy indoor environments, and provides close to the expected theoretical doubling of throughput in practice. However, as for building full-duplex MIMO radios would need N^2 circuits to cancel self-talk and cross talk. The next chapter solves the full duplex MIMO radios in scalable fashion.

The next chapter is the design and implementation of the first in-band full duplex WiFi-PHY based MIMO radios that practically achieve the theoretical doubling of throughput. Our design solves two fundamental challenges associated with MIMO full duplex: complexity and performance. Our design achieves full duplex with a cancellation design whose complexity scales almost linearly with the number of antennas, this complexity is close to the optimal possible. Further we also design novel digital estimation and cancellation algorithms that eliminate almost all interference and achieves the same performance as a single antenna full duplex SISO system, which is again the best possible performance. We prototype our design by building our own analog circuit boards and integrating them with a WiFi-PHY compatible standard WARP software radio implementation. We show experimentally that our design works robustly in noisy indoor environments, and provides close to the expected theoretical doubling of throughput in practice.

This completes the story for full duplex radio for MIMO systems. However as pointed in the last subsection, the self interference cancellation has applications beyond full duplex. We build a full duplex relay on the prior mimo full duplex radio, called FastForward(FF). FF, a novel full-duplex relay that constructively forwards signals such that wireless network throughput and coverage is significantly enhanced. FF is a Layer 1 in-band full-duplex device, it receives and transmits signals directly and simultaneously on the same frequency. It cleanly integrates into existing networks (both WiFi and LTE) as a separate device and does not require changes to the clients. FF's key invention is a constructive filtering algorithm that transforms the signal at the relay such that when it reaches the destination, it constructively combines with the direct signals from the source and provides a

significant throughput gain. We prototype FF using off-the-shelf software radios running a stock WiFi PHY and show experimentally that it provides a 3× median throughput increase and nearly a 4× gain at the edge of the coverage area.

The next application we built on the self interference cancellation is backscatter communication system called BackFi. BackFi is a novel communication system that enables high throughput, long range communication between very low power backscatter IoT sensors and WiFi APs using ambient WiFi transmissions as the excitation signal. Specifically, we show that it is possible to design IoT sensors and WiFi APs such that the WiFi AP in the process of transmitting data to normal WiFi clients can decode backscatter signals which the IoT sensors generate by modulating information on to the ambient WiFi transmission. We show via prototypes and experiments that it is possible to achieve communication rates of up to 5 Mbps at a range of 1 m and 1 Mbps at a range of 5 meters. Such performance is an order to three orders of magnitude better than the best known prior WiFi backscatter system [74, 70]. BackFi design is energy efficient, as it relies on backscattering alone and needs insignificant power, hence the energy consumed per bit is small.

Chapter 2

Building SISO self-interference cancellation: Full Duplex radios

2.1 Introduction

A long held assumption in wireless is that radios have to operate in half duplex mode, i.e. either transmit or receive but not both simultaneously on the same channel. Recent work has attempted to invalidate this assumption. Researchers at Stanford [71, 48], Rice [56, 52] and several other groups in industry and academia [101, 42] have proposed various designs to build in-band full-duplex radios. Full duplex, if possible, has tremendous implications for network design, not least of which is the fact that cellular networks could cut their spectrum needs by half. For example, LTE uses equal width separate uplink and downlink channels to enable radios to achieve full duplex. With an in-band full-duplex system we could use a single channel to get the same performance. Consequently, the problem has attracted significant attention, both from industry and academia and has spurred significant follow-up work.

To achieve full duplex, a radio has to completely cancel the significant self-interference that results from its own transmission to the received signal. Since WiFi signals are transmitted at 20dBm (100mW) average power, and the noise floor is around -90dBm, the transmit self-interference has to be canceled by 20dBm$-(-90$dBm$) = 110$dB to reduce it to the same level as the noise floor and render it negligible. If self-interference is not completely canceled, any residual self-interference acts as noise to the received signal and reduces SNR and consequently throughput. For example, if the received signal's SNR without full duplex is 25dB but is reduced to 5dB due to 20dB residual self-interference, then the throughput with full duplex is that achieved using two 5dB SNR links. This is significantly worse than using the original half duplex link with 25dB SNR and it is better to turn off full duplex in this case. To sum up, the amount of self-interference cancellation dictates overall throughput and is a figure of merit for any full-duplex design.

Prior designs have made significant progress on the self-interference cancellation problem [71, 51, 48]. However the best performing prior designs can at best provide 85dB of cancellation, which still leaves about 25dB of residual self-interference and therefore reduces the SNR of each direction of the full duplex link by 25dB. A calculation similar to the previous paragraph's shows that to see throughput benefits with these full-duplex designs, the half-duplex SNR of the link has to be extremely high (45dB or higher). In terms of range, the two nodes would have to be closer than 5m to see such high SNRs. Outside this range, it is better to turn off full duplex and use the traditional half duplex mode. To be fair however, these designs were intended for low-power, narrow-band, fixed rate protocols such as Zigbee where 85dB of self-interference cancellation is sufficient for full duplex. WiFi is far more demanding both in terms of bandwidth as well as cancellation.

Prior designs also need to have at least two antennas [71, 51] in place of the one used by half duplex systems (one each for transmit and receive and possibly more [48]). However, with two or more antennas, the argument for full duplex becomes weaker since the same doubling of capacity could be obtained by using the two antennas as MIMO antennas to spatially multiplex two independent packets in half duplex mode instead of using them for full duplex.

In this chapter, we present the design and implementation of a full duplex WiFi radio that uses a *single antenna* [1] and delivers close to the theoretical doubling of throughput under all link SNR and distance ranges. Our key technical contributions are novel self-interference cancellation circuits and algorithms that provide the required $110dB$ *of self interference cancellation* for standard WiFi signals and thus eliminate all self interference to the noise floor. Our design is wideband: it works with the highest bandwidths (80MHz) and data rates used by the latest 802.11ac PHY in the 2.4GHz spectrum. We also experimentally demonstrate a complete full-duplex communication link which uses the full WiFi PHY (OFDM, constellations up to 256QAM and all the channel coding rates) and achieves close to the theoretically expected doubling of throughput. To the best of our knowledge, this is the first working implementation of a complete WiFi PHY single-antenna full-duplex link.

The reader might be wondering why full duplex is hard to realize. After all, as the sender knows the signal being transmitted, subtracting it should be relatively simple to implement. One of the key insight in this work is that in fact the *radio does not know what it is transmitting*. What it does know is the clean digital representation of the signal in baseband. However, once the signal is converted to analog and up-converted to the right carrier frequency and transmitted, the transmitted signal looks quite different from its baseband incarnation. The numerous analog components in the radio TX chain distort the signal in both linear and non-linear ways (analog circuits will create cubic and higher order components of the signal for example), add their own noise (e.g., power amplifiers add transmitter noise), are slightly inaccurate (e.g., your oscillator is

[1] Picasso [66] uses a single antenna, but it only allows the radio to simultaneously transmit and receive on *different* adjacent channels. Hence it fails to address the much harder problem of simultaneous TX/RX on the same channel. Our system does address this challenge, and offers novel and higher performance analog and digital cancellation techniques compared to Picasso.

tuned slightly off 2.45GHz), or delay it by different amounts at different frequencies and so on. In effect the transmitted signal is a complicated non-linear function of the ideal transmitted signal along with unknown noise. Unsurprisingly, naively subtracting a "known" baseband version of the transmit signal without accounting for all these analog distortions does not work. As we will show in Sec 4.5 prior designs fail to account for these distortions and hence are limited to at best 85dB of cancellation.

This chapter makes two key contributions over all prior work in this space. First, we design dynamic algorithms to estimate the distortions introduced by analog circuits and accurately model the actual self-interference being experienced by the received signal. Second, we design a novel programmable analog cancellation circuit using off-the-shelf components that allows us to implement the above algorithm in "analog" and dynamically cancel the self-interference. Such analog cancellation prevents receiver saturation from strong self-interference and allows us to use commodity radios. However, the analog cancellation stage does not completely cancel the self-interference. We complement it with a novel digital cancellation algorithm and implementation that cancels any remaining self-interference. Our digital cancellation algorithm differs from all prior work because it not only models the linear distortions, but also non-linear effects and other special effects such as oscillator noise. Thus, overall we use a hybrid analog-digital design that successfully models all linear and non-linear distortions as well as transmitter noise.

We implement our design via a combination of circuit designs and software implementations. Our analog cancellation is implemented on a PCB that we designed and populated using off-the-shelf components. We integrate our board with an off-the-shelf antenna and software radio transceiver [107, 106] based on test equipment from Rohde-Schwarz (RS) as well as on commodity WARP radios. We also implement our digital cancellation algorithms as well as a fully WiFi compliant PHY layer based on OFDM, supporting constellations up to the standard required 256QAM and all the channel coding rates. We deployed and evaluated our system in an indoor and noisy office environment in the 2.4GHz ISM band, operating the WiFi PHY over the 80MHz bandwidth on RS radios, and over the 20MHz bandwidth using WARP radios.

Our experiments demonstrate that our design delivers on the promise of full duplex. Under typical indoor deployment scenarios, our system delivers a median throughput gain of 87% in practice with WiFi radios which is close to the theoretically expected 2×. Looking into the cancellation itself, we show that our design consistently delivers the required 110dB of cancellation in a dense indoor office environment for both the RS 80MHz radios as well as the commodity 20MHz WARP radios. The system is robust to environmental changes, reflections, and can handle all the different constellations used in WiFi. We compare against the best known prior full duplex approaches [71, 56] and show experimentally that they can at best deliver 85dB of cancellation and therefore reduce the SNR of the received signal by at least 25dB.

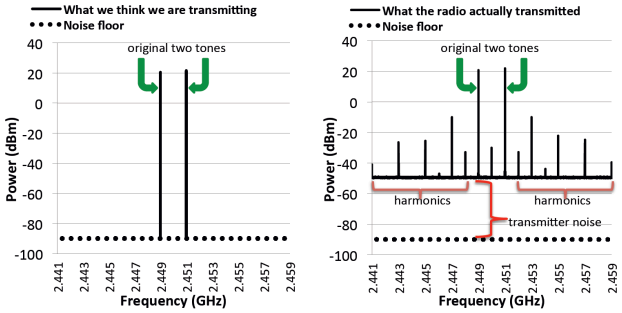

Figure 2.1: What we think we are transmitting in digital on the left side, and what the radio actually transmitted on the right side. The actual transmitted signal differs significantly from the two tones generated in digital baseband. Note transmitter noise and harmonics are generated in addition to the two main transmitter tones.

2.2 The Problem

Full duplex, in theory, should be simple to accomplish. After all, we know the signal we are transmitting and we are only designing circuits and algorithms to subtract it from the received signal. The intuition follows from the conventional abstraction that the analog radio (also known as the RF front-end) is a black-box that takes the digital baseband signal, converts it to analog, up-converts it to the carrier frequency, scales it to the right power and sends it. In other words, the assumption has been that the radio preserves the original baseband signal except for power scaling and frequency shifting. In practice this abstraction turns out to be incorrect. Radios in fact significantly distort the signal being transmitted, relative to the digital baseband representation.

To demonstrate the distortions, we use the following experiment throughout this section. We take a software radio transceiver [107, 106] and send the following signal: two tones at 2.449GHz and 2.451GHz. In other words, we are sending an extremely simple signal, two sine waves with frequencies 1MHz away from the carrier frequency of 2.45GHz. We do this by creating a digital baseband signal with samples of the sine waves at -1MHz and 1MHz which the radio then up-converts to 2.45GHz and amplifies to 20dBm average transmit power (the power used by WiFi radios). We then compare the signal output of the antenna to what we would ideally expect if the radio did not introduce any distortions. This experiment serves as some sort of lower bound on the quality of radios. If radios cannot transmit even this simplest of signals without distortion then more complex signals such as WiFi are likely to be significantly distorted. Fig. 2.1 plots the ideal and actual transmitted signals' spectra that resulted from our experimental set-up (we ensured that this was a clean environment with no other interference present in the environment at the time of the experiment).

Ideally, we expect to see only two tones at 2.451GHz and 2.449GHz as shown on the left side of Fig. 2.1. However in the transmitted signal, whose spectrum is plotted on the right side of Fig. 2.1, we can easily see that there are several other distortions present in addition to the two main tones

that were transmitted. The main components in self-interference can be classified into three major categories:

1. **Linear Components**: This corresponds to the two main tones themselves which are attenuated and could consist of reflections from the environment. These are linear components because the received distortion can be written as a linear combination of different delayed copies of the original two tones.
2. **Non-Linear Components**: These components are created because radio circuits can take in an input signal x and create outputs that contain *non-linear cubic and higher order terms such as* x^3, x^5. These higher order signal terms have significant frequency content at frequencies close to the transmitted frequencies, which directly correspond to all the other harmonics we see on the right side of Fig. 2.1. Harmonics, as the name suggests, are signal distortions which occur at equally spaced frequency intervals from the transmitted frequencies. As the right side of Fig. 2.1 shows, we see spikes at frequencies 2.447GHz and 2.453GHz, that are spaced 2MHz apart from the two transmitted tones 2.451GHz and 2.449GHz, on either side.
3. **Transmitter Noise**: The general increase we see in the base signal level which we can clearly see on the sides of the two main tones is noise from the radio transmitter. A radio will of course always have noise, which works out to a noise power level of -90dBm [106]). But as we can see, the power at the side-bands is significantly higher, on the level of -50dBm, or 40dB higher than the receiver noise floor. This extra noise is being generated from high power components in the radio transmitter such as power amplifiers. In the radio literature this is referred to as broadband noise [80]. Further radios have phase noise generated by local oscillators (LO), which is typically of level of -40dBm, or 50dB above (not seen in the Fig. 2.1 because its hidden under the main signal component).

2.2.1 Requirements for Full Duplex Designs

The above analysis suggests that any in-band full duplex system has to be able to cancel all the above distortions in addition to the main signal component itself, since all of these are within the frequency band we are transmitting and receiving on and act as strong self-interference to the received signal. In this section, we discuss how strong each of these components are for typical transceivers, and what are the requirements for full duplex. We will state all self-interference power levels relative to the receiver noise floor. The reason is that to implement full duplex, we need to cancel any self-interference enough so that its power is reduced to the same level as the receiver noise floor. There is no point in canceling beyond that since we won't see any benefits — the received signal's SNR will then be dictated anyway by the receiver noise floor which cannot be canceled or reduced, just as it is today in half duplex radios.

We use similar experiments for OFDM-wideband signals to quantify the power levels of the different distortions, shown in the left side of Fig. 2.2. In a typical WiFi radio using 80MHz

bandwidth, the receiver has a noise floor of −90dBm (1 picowatt). First, since the main signal component is being transmitted at 20dBm (100mW), self-interference from the *linear main component is* $20 - (-90) = 110dB$ above the receiver noise floor. Second, we observed experimentally that *the non-linear harmonics are at* $-10dBm$*, or* $80dB$ *above the receiver noise floor.* Finally, the *transmitter noise is at* $-40dBm$*, or* $50dB$ *above the receiver noise floor.* Note that these numbers are consistent with other RF measurement studies reported in the literature [35] for standard WiFi radios.

There are four takeaways from the above analysis:

- Any full duplex system needs to provide 110dB of **linear self-interference cancellation** to reduce self-interference to the receiver noise floor. This will ensure that the strongest component (the main signal) which is 110dB above the noise floor will be eliminated.
- A full duplex system has to reduce non-linear harmonic components that are 80dB above the noise floor, so any full duplex technique has to provide at least 80dB of **non-linear self-interference cancellation**.
- Transmitter noise is by definition noise and is random. In other words, we cannot infer it by any algorithm. Hence the only way to cancel transmitter noise is to get a copy of it where it is generated, i.e. in the analog domain and cancel it there. This implies any full duplex system has to have an analog cancellation component that provides at least 50dB of **analog noise cancellation** so that transmitter noise is reduced to below the receiver noise floor.
- A final constraint is that RX chains in radios get saturated if the input signal is beyond a particular level that is determined by their ADC resolution. Assuming a 12 bit ADC resolution typically found in commodity WiFi radios, we have a theoretical 72dB of dynamic range, which implies that the strongest signal level that can be input to the radio relative to the receiver noise floor is −90dBm+72 = −18dBm. However, in practice it is necessary to leave 2 bits worth of margin, i.e a 12 bit ADC should be used as if it is a 10 bit ADC to reduce quantization noise. So the maximum input signal level can be −90dBm+60 = −30dBm. Since in WiFi, the transmitted self-interference can be as high as 20dBm, a full duplex system needs to have an analog cancellation stage that provides 60dB of self-interference reduction (we keep a further 10dB margin for OFDM PAPR where instantaneously an OFDM signal's power level can rise 10dB above the average power).

To sum up, any full duplex design needs to provide 110dB of linear cancellation, 80dB of non-linear cancellation, and 60dB of analog cancellation.

2.2.2 Do Prior Full Duplex Techniques Satisfy these Requirements?

There are two state-of-the-art designs: ones which use an extra transmit chain to generate a cancellation signal in analog [52] and ones which tap the transmitted signal in analog for cancellation [71, 48]; both use a combination of analog and digital cancellation. Note that all these designs use

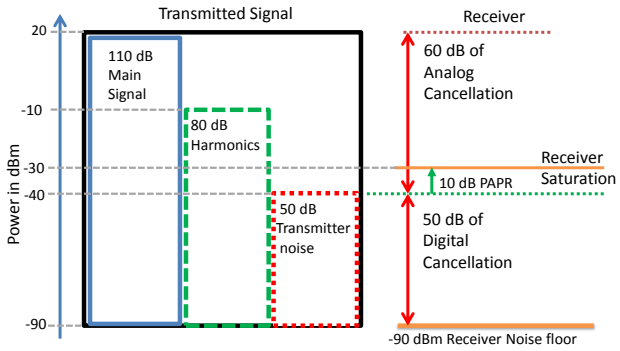

Figure 2.2: On the left hand side we see transmitted signal with sub-components. On the right hand side we see how this impacts the requirements of analog and digital cancellation.

at least two antennas for transmit and receive instead of the normal single antenna, and the antenna geometry ones use more than two.

Designs which use an extra transmitter chain report an overall total cancellation of 80dB (we have been able to reproduce their results experimentally). Of this, around 50dB is obtained in the analog domain by antenna separation and isolation between the TX and RX antennas of around 40cm (the designs also assume some form of metal shielding between the TX and RX antennas to achieve 50dB isolation). Note that this 50dB reduction applies to the entire signal, including linear and non-linear components as well as transmitter noise since it is pure analog signal attenuation. Next, these designs also use an extra transmit chain to inject an antidote signal [52, 60] that is supposed to cancel the self-interference in analog. However, the antidote signal only models linear self-interference components and does not model non-linear components. Further, it is incapable of modeling noise because by definition noise is random and cannot be modeled. Overall this extra cancellation stage provides another 30dB of linear self-interference cancellation in the best case. Thus, these designs provide 80dB of linear cancellation, 50dB of non-linear cancellation and 50dB of analog noise cancellation, falling short of the requirements by 30dB for the non-linear components. Hence if full duplex is enabled over links whose half duplex SNR is 30dB or lower, then no signal will be decoded. Further to see any throughput improvements with full duplex, the half duplex link SNR would have to be greater than 50dB.

The second design [71] gets a copy of the transmitted analog signal and uses a component called the balun (a transformer) in the analog domain to then create a perfectly inverted copy of the signal. The inverted signal is then connected to a circuit that adjusts the delay and attenuation of the inverted signal to match the self interference that is being received on the RX antenna from the TX antenna. We show experimentally in Sec. 4.5, that this achieves only 25dB of analog cancellation, consistent with the prior work's results. The cancellation is limited because this technique is very sensitive to and requires precise programmable delays with resolution as precise as 10picoseconds to exactly match the delay experienced by the self-interference from the TX to the RX antenna. Such

programmable delays are extremely hard to build in practice, at best we could find programmable delays with resolution of $100 - 1000$picoseconds and these were in fact the ones used by the prior design [71]. Hence the cancellation circuit is never able to perfectly recreate the inverted self-interference signal and therefore cancellation is limited to 25dB in analog. However this design also uses two separate antennas separated by 20cm for TX and RX and achieves another 30dB in analog cancellation via antenna isolation. Hence a total of 55dB of self-interference reduction is obtained in analog, this cancellation applies to all the signal components (linear, non-linear and noise). The digital cancellation stage of this design also only models the linear main signal component, it does not model the non-linear harmonics that we discussed above. Thus we found that we obtain another 30dB of linear cancellation from digital in this design.

Overall, the second design provides 85dB of linear self-interference cancellation, 55dB of non-linear cancellation and 55dB of analog noise cancellation. Thus this design falls short of the requirements by 25dB (especially for the non-linear component). Hence if full duplex is enabled over links whose half duplex SNR is 25dB or lower, then no signal will be decoded. Further to see any throughput improvements with full duplex, the half duplex link SNR would have to be greater than 45dB.

2.3 Our Design

In this section we describe the design of our self-interference cancellation technique. Our design is a single antenna system (i.e. the same antenna is used to simultaneously transmit and receive), wideband (can handle the widest WiFi bandwidth of 80MHz as well as all the LTE bandwidths) and truly full duplex (cancels all self-interference to the receiver noise floor). The design is a hybrid, i.e., it has both analog and digital cancellation stages. Note that our hybrid cancellation architecture is not novel, similar architectures have been proposed in prior work [71, 26, 25]. The novelty of our work lies in the design of the cancellation circuits and algorithms, as well as their performance. To the best of our knowledge this is the first technique that achieves 110dB of cancellation and eliminates self-interference to the noise floor.

2.3.1 Analog Cancellation

We introduce a novel analog cancellation circuit and tuning algorithm that robustly provides at least 60dB of self-interference cancellation. Fig. 2.3 shows the high level design of the circuit and where it is placed in the radio architecture. A single antenna is connected to a circulator (at port 2), which is a 3 port device that provides limited isolation between port 1 and port 3 while letting signals pass through consecutive ports as seen in Fig. 2.3. The TX signal is fed through port 1, which routes it to the antenna connected to port 2, while the received signal from the antenna is passed from port 2 through to port 3. Circulator cannot completely isolate port 1 and port 3, so inevitably the TX signal leaks from port 1 to port 3 and causes interference to the received signal. From our

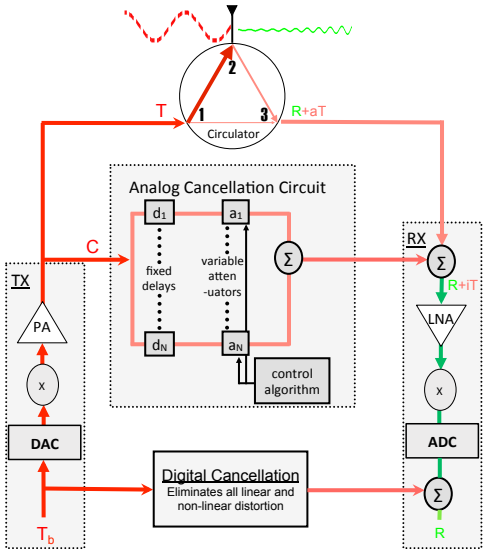

Figure 2.3: Full duplex radio block diagram. T_b is intended baseband signal we think we are transmitting, but in fact the transmit signal is T (red). The intended receive signal is R (green), however we see strong components of the red signal the RX side. Some of these red signals are undesirably leaked through the circulator. The analog cancellation circuit is trying to recreate a signal that matches the leaked interference signal for cancellation. The digital cancellation stage eliminates any residual self interference.

experiments we find that the circulator only provides 15dB of isolation, i.e., the self-interference that is leaking to the RX circuit is reduced only by 15dB. To get to the noise floor, we still have to provide 95dB of cancellation, and at least 45 dB of that has to come in analog to ensure transmitter noise is sufficiently canceled and we do not saturate the receiver. We accomplish this using our novel analog cancellation circuit that we describe next. Note that when we report analog cancellation performance numbers, we include the 15dB of reduction we get from the circulator for simplicity of description.

Fig. 2.3 shows the design of our analog cancellation circuit. We tap the TX chain to obtain a small copy of the transmitted signal just before it goes to the circulator. This copy therefore includes the transmitter noise introduced by the TX chain. The copy of the signal is then passed through a circuit which consists of parallel fixed lines of varying delays (essentially wires of different lengths) and tunable attenuators. The lines are then collected back and added up, and this combined signal is then subtracted from the signal on the receive path. In effect, the circuit is providing us copies of the transmitted signal delayed by different fixed amounts and programmatically attenuated by different variable amounts. The key challenge is to pick the fixed delays, as well as to dynamically program the tunable attenuators appropriately so that the we maximize self-interference cancellation. Note that unlike prior work our design uses components that are all available off-the-shelf and is therefore easy to manufacture, we do not need sophisticated high resolution programmable delays that are

hard to build like in prior work [71].

The design of our cancellation circuit is based on a novel insight: *we can view cancellation as a sampling and interpolation problem.* The actual self-interference signal has a particular delay and amplitude that depends on the delay d and attenuation a through the circulator. Our insight (the reason for which will become clear shortly) is that we should pick the fixed delays in our cancellation circuit such that they straddle the delay of the self-interference signal through the circulator. So if we have N fixed delay lines, $N/2$ of those lines should be placed at equidistant intervals all of which have delays that are less than the delay of the self-interference d, and we should do the same for the other half of the delays but greater than d. In practice it is hard to know the precise value of d since it is a function of how the circuit is put together, but we can always find the range over which it varies and place our fixed delays outside of that range on either side.

At this stage we have leading and lagging copies of the transmitted self-interference signal, how might we use them to approximate the actual self-interference itself at some intermediate instant? If we take a step back, this is essentially an interpolation problem, similar to Nyquist digital sampling. In Nyquist digital sampling, we have discrete samples of the signal at a time period equal to the inverse of the sampling frequency. The Nyquist theorem [92] tells us that sampling (at the Nyquist rate) does not lose information, in other words we can always reconstruct the signal at any instant as a weighted linear combination of samples taken before and after the instant at which we want to recreate. The weights of the linear combination can be determined by using a standard algorithm called *sinc interpolation.* The basic idea is that you overlay sinc pulses at each sampling time instant and calculate the value of the sinc pulse at the time instant t where you wish to recreate the signal. This value gives the weight you should apply to this sample when you take the linear combination for reconstruction. We repeat this algorithm for every sample to determine the corresponding weight to apply to it. The value of the signal at time t is then given by the linear combination of all the samples with weights calculated by the sinc trick discussed above.

Our analog cancellation circuit is in effect implementing the same trick, at every instant we have copies of the signal at different equally spaced delays just like in digital sampling. The programmable attenuators essentially function as the weights we need to apply in the linear combination for reconstruction. Similar to digital sampling, we need to estimate the self interference at an instant d that lies somewhere in between these fixed delays $d_1, \ldots, d_N$ as shown in Fig. 2.4. To do so, the weights for each sample, i.e., the value of the attenuator that we need to set on each line i is equal to the value of the sinc pulse centered at the fixed delay d_i at instant d. If we adjust the attenuators for each delay line to those values, then we will be able to perfectly reconstruct the self interference and cancel it from the receive path. Fig. 2.4 shows this algorithm visually in action.

In practice however, there is an important difference with digital sampling. In digital, we can take linear combinations of a very large number of samples since memory is essentially free. To do that in analog we would need a correspondingly large number of delay lines. In practice, this is

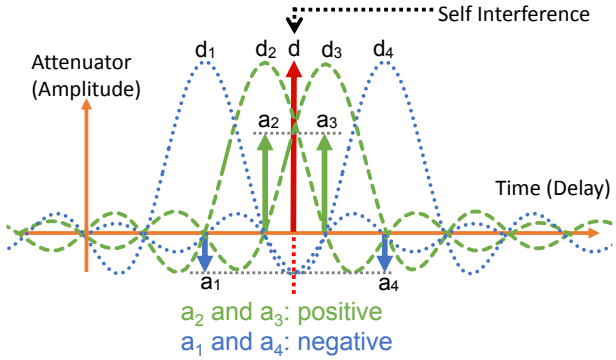

Figure 2.4: This figure shows how we can recreate the self interference signal which is located at instant d, positioned between the fixed delay lines d_i. The value of the attenuator a_i for delay d_i is given by the value taken by the sinc centered at d_i at instant d.

not possible due to a variety of reasons, ranging from space limitations to power consumption to electromagnetic radiations. Our key insight is that in interpolation, the samples that matter most are the ones that are closest to the instant t at which the signal is being reconstructed. Intuitively, the value of a signal at a much further/before time than t should not affect the value of the signal at t. This is reflected in the fact that the weights in the linear combination for these further out samples are nearly zero. This allows our analog circuit to therefore use a small number of delay lines and still approximate the self interference fairly well. We show in Sec. 4.5 that sixteen delay lines are sufficient to approximate the self interference signal leaking through the circulator. Further, we will show in Sec. 4.5 that our analog cancellation delivers at least 60dB cancellation comfortably exceeding the requirements we developed in Sec. 2.2.1.

2.3.2 Digital Cancellation

The goal of digital cancellation is to clean out any remaining residual self-interference. Assuming that analog cancellation provides 60dB, digital cancellation has to cancel the linear main signal component by another 50dB and non-linear components by another 20dB. We address each of these components separately.

Canceling Linear Components

The first part of digital cancellation eliminates the residual linear components of the self-interference. This consists of the main transmitted signal that is leaking over through the circulator after analog cancellation, as well as any delayed reflections of this signal from the environment. The reflections are also delayed and attenuated by different unknown amounts.

The basic idea is that this part of the self-interference can be modeled as a linear and *non-causal* function of the transmitted signal, as we know it in digital (recalling that we know the baseband IQ samples of the transmitted packet). The non-causal bit is important. Since we know the samples

of the entire packet that was transmitted, we can use samples from the future to estimate the self-interference at the current instant. In other words, the received sample $y[n]$ at any instant can be modeled as a linear combination of up to k samples of the known transmitted signal $x[n]$ before and after the instant n. The parameter k is empirically chosen and is a function of the amount of memory in the channel. So we can write the equation as:

$$y[n] = x[n-k]h[k] + x[n-k+1]h[k-1] + \ldots + x[n+k-1]h[-k+1] + w[n]$$

where $h[k], \ldots, h[-k+1]$ represents the attenuations applied by the channel to the transmitted function, and $w[n]$ is the receiver noise floor.

How can we estimate the coefficients $h[n]$? We leverage the fact that most wireless transmissions have known packet preambles (e.g. WiFi uses a preamble of two known OFDM symbols at the start of the packet). Let the samples representing the preamble be $x_{pr}[n]$. Let the receive samples corresponding to the preamble be $y[0], \ldots, y[n]$. Then the above channel equations can be written specifically for the preamble as:

$$y = Ah + w$$

where A is Toeplitz matrix of $x_{pr}[n]$.

$$A = \begin{pmatrix} x_{pr}(-k) & ... & x_{pr}(0) & ... & x_{pr}(k-1) \\ ... & ... & ... & ... & ... \\ x_{pr}(n-k) & ... & x_{pr}(n) & ... & x_{pr}(n+k-1) \end{pmatrix}.$$

Our goal is to find a maximum likelihood estimate of the vector h, i.e.,

$$minimize \;\; ||y - Ah||_2^2$$

Note that the matrix A is known in advance since we know the values of the preamble samples. Hence it can be *pre-computed*. Additionally, we know from prior work [43] that the coefficients for the above problem can be computed by multiplying by the ith received sample of the preamble, as the samples arrive serially as follows:

$$h = \sum (y_i a_i^\dagger)$$

where $a_i^\dagger$, is the ith column of pseudo inverse of A matrix. Thus our estimation algorithm computes the linear distortions that the transmitted main signal has gone through for every packet, and is capable of dynamically adapting to the environment.

Canceling Non-Linear Components

The second task for digital cancellation is to eliminate the residual non-linear components whose power is around 20dB after being reduced by 60dB due to analog cancellation. However, it is quite hard to guess the exact non-linear function that a radio might be applying to the baseband transmitted signal. Instead, we use a general model to approximate the non-linear function using Taylor series expansion (as this is a standard way to model non-linear functions)[50]. So the signal that is being transmitted can be written as:

$$y(t) = \sum_m a_m x_p(t)^m$$

where $x_p(t)$ is the ideal passband analog signal for the digital representation of $x(n)$ that we know.

The above general model contains a lot of terms, but the only ones that matter for full duplex are terms which have non-zero frequency content in the band of interest. A little bit of analysis for passband signals (taking the Fourier transform) of the equation above reveals that the only terms with non-zero energy in the frequency band of interest are the odd order terms (i.e., the terms containing $x_p(t)$, $x_p(t)^3$, $x_p(t)^5$ and so on), so we can safely ignore the even order terms. The first term that is the linear component, i.e., the terms for $x_p(t)$ is of course the one corresponding to the main signal and is estimated and canceled using the algorithm discussed in the previous section. In this section, we focus only on the higher-order odd power terms. We can therefore reduce the above model and write it in the digital baseband domain as:

$$y(n) = \sum_{m \in \ odd \ terms, n=-k,\ldots,k} x(n)(|x(n)|)^{m-1} * h_m(n)$$

where $h_m[n]$ is the weight for the term which raises the signal to order m and is the variable that needs to be estimated for cancellation, and k is the number of samples in the past and future which significantly influence the value of the signal at instant n.

To estimate these coefficients, we can use the same WiFi preamble. The WiFi preamble is two OFDM symbols long of length 8μs, and assuming a sampling rate of 160MHz, it consists of a total of 1280 digital samples at the Nyquist sampling rate. However, if we look at the above equation, the number of variables $h_m(n)$ that we need to compute is a function of $2k$ (i.e., how far in the past and future is the current self-interference signal influenced by) and the highest value of m that exhibit strength greater than the receiver noise floor. A naive model assuming that just the $1, 3, 5, 7, 9, 11$th order terms matter, and that upto 128 samples from both the future and the past influence the self-interference signal at any instant [2] would require us to estimate $128 * 2 * 6 = 1536$ variables using 1280 equations. Clearly, this is under-determined system, would increase the noise floor significantly.

[2]The number of samples required is a function of the amount of multipath, the higher the mutlipath, the higher the number of samples in the past and future that matter but 128 is the number suggested by the WiFi standard and is equal to the length of the WiFi OFDM Cyclic Prefix

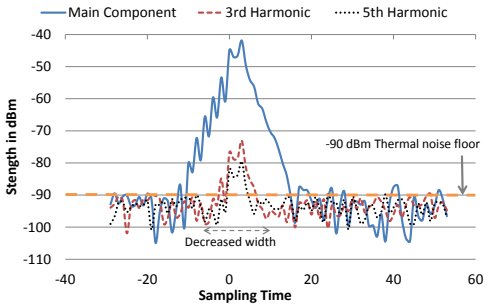

Figure 2.5: Signal strength of various harmonics that make up the transmitted signal. Note that higher order harmonics are much weaker relative to main component and therefore any reflections of these harmonics have to be quite closely spaced in time for them to be stronger than the receiver noise floor.

In practice we found empirically that many of these variables do not matter, that is their value is zero typically. The reason is that higher order terms have correspondingly lower power since they are created by the mixing of multiple lower order terms and each mixing reduces power. So the 7^{th} order term has lower power than the 5^{th} order term which has lower power than the 3^{rd} order term. Fig. 2.5 shows a plot of the strength of of the main signal and higher order non-linear terms relative to the receiver noise floor. As we can see higher order terms have weaker strength relative to the main signal, and consequently their multipath components also decay quickly below the receiver noise floor. In other words, far fewer than 128 samples from the past and future impact the value of the self interference harmonic component at this instant. We find empirically that for indoor WiFi systems, across all the non-linear higher orders, a total of only 224 such variables are all that we need to estimate which we can easily accomplish using the WiFi preamble (over-determined system of linear equation). Hence our digital cancellation algorithm calculates all these coefficients using the WiFi preamble and applies them to recreate the harmonics and cancel them. The method for estimating the coefficients is the same as the one used in the linear digital cancellation step described by Eq. 2.3.2, but the matrix A is formed using the higher order odd powers of the preamble samples.

Complexity

The complexity of digital cancellation is the same as solving 1280 (say W, width of preamble in general) linear equations with 224 unknowns. Further the matrix that forms the linear equations is known in advance (this is the known preamble trick as discussed above). Hence the pseudo-inverse of this matrix can be pre-computed and stored. Thus the complexity of digital cancellation reduces to $O(W)$ multiplications. The design is therefore relatively simple to implement and can be efficiently realized in hardware.

2.3.3 Dynamic Adaptation of Analog Cancellation

To provide a robust full duplex link, we need to ensure that sufficient cancellation is maintained to reduce self interference to the noise floor, even as things such as environment, transmit power, temperature and other such parameters change. These changes would clearly reduce the cancellation achieved by any static configuration, since they change the distortions that are imposed by the self interference. Digital cancellation can cope since it essentially estimates these distortions on a per-packet basis, however analog cancellation might be degraded and hence performance might be worsened. In this section, we describe how we can quickly tune the analog circuit to provide the required amount of cancellation (60dB at least).

The goal of tuning is to pick the attenuation values $a_1, \ldots, a_N$ such that self-interference is minimized. More formally,

$$\min_{a_1,\ldots,a_N} \quad (y(t) - \sum_{i=1}^{N} a_i c(t - d_i))^2$$

where $c(t)$ is the reference signal that is tapped from the transmit path, $y(t)$ is the self interference, $d_1, \ldots, d_N$ are delays associated with the taps as shown in Fig. 2.3.

A simple and obvious technique to solve the above problem in practice is a iterative gradient descent algorithm, which other prior works in full duplex have also used to tune their own analog cancellation [71]. However, we found that this algorithm is extremely slow (requires nearly 40ms) because of the larger number of variables (16) that need to be estimated in our design unlike prior work. That's an unacceptable overhead, since we found empirically that we need to re-tune analog cancellation once every 100ms on average in our setup. So taking 40ms to tune implies a 40% overhead.

Our key contribution here is an approach that solves the tuning problem in the frequency domain. The idea is that the self interference $y(t)$ can be modeled in the frequency domain as a function of the tapped signal $c(t)$ as

$$\mathbf{Y}(f) = \mathbf{H}(f)\mathbf{C}(f)$$

where $\mathbf{H}(f)$ is the frequency domain representation of the distortion introduced by the circulator, antenna and the environment and $C(f)$ is the frequency domain representation of the tapped signal. Recall that the tapped signal is essentially a scaled replica of the transmitted signal input to the circulator, hence the above equation can be written in terms of the tapped signal. This frequency response $\mathbf{H}(f)$ is easier to measure, it is essentially an FFT of the self interference channel which can be measured using the WiFi preamble. In fact, standard OFDM is doing exactly this, it is estimating the frequency domain channel using the preamble and pilot symbols.

The goal of the optimization problem then is to pick the attenuator values such that the overall frequency domain response of the cancellation circuit approximates $\mathbf{H}(f)$ as closely as possible. So the above optimization problem can be restated as

$$\min_{a_1,\ldots,a_N} \quad (\mathbf{H}(f) - \sum_{i=1}^{N} \mathbf{H_i^{a_i}}(f))^2 \tag{2.1}$$

where, $\mathbf{H_i^{a_i}}(f)$ is the frequency response for delay line i for attenuation setting of a_i.

How might we solve this problem? The problem is two fold. First, we have to find the frequency response of each delay line of the cancellation circuit for every attenuation value, i.e., $\mathbf{H_i^{a_i}}(f)$. Second, once we have the frequency response of the self-interference channel $\mathbf{H}(f)$, we need to search on the space of possible attenuation values for every delay line(attenuator), to come up with best possible solution to the optimization problem. Each delay line can take 128 different attenuation values, and there are 16 delay lines, so in total we have $128^{16} = 2^{112}$ values, a computationally expensive search.
Modeling the frequency response of delay lines $\mathbf{H_i^{a_i}}(f)$: Measuring the frequency response of individual delay line is impossible — The entire circuit is well connected, thus isolating individual delay line is impossible. Our key observation, is if we can measure the frequency response of a delay line at one attenuation value, then the datasheet of the attenuator provide measurements called S parameters (specifically frequency response measurements between different ports of a device) that can be used to extrapolate the frequency response of the delay line for all attenuation values. The S parameter data provides the relative change in frequency response with changing attenuator value. To calculate the frequency response at this initial point, we use the following trick. We set the attenuators for all the lines to their highest attenuation setting, except the one being measured. The idea is to essentially emulate a board where none of the delay lines, except the one being measured, let any signal through. The highest attenuation value approximates that setting but doesn't fully accomplish that, hence we apply a second least squares fit to find a more accurate response (collecting more data for different attenuation's for this delay line, keeping the rest all others at highest attenuation setting). Then, the frequency response of this delay line for all 128 attenuator values can be calculated. We repeat this process for all the delay lines in the circuit. Note that all of this has to be done once and can be stored, since this frequency response of the delay line and attenuation is independent of the environment or other such changing parameters.
Optimization Algorithm : Now to actually find the attenuation settings in real time to optimize the cancellation, we use the following algorithm.

1. Measure the frequency response of the self interference $\mathbf{H}(f)$ using the WiFi preamble. This is relatively simple since we have two OFDM symbols and as part of the baseband decoding we can perform an FFT to measure the frequency response.
2. Solve the frequency domain integer linear optimization problem posed in Eq. 2.1 by relaxing it to a linear program and then use random rounding to find a solution for attenuator settings, which achieves required cancellation of 60dB. The intuition behind the algorithm is that it reduces the search space of attenuator values to a polynomial set compared to the exponential search space. This is due to the fact that we are looking for a point which provides required

cancellation, instead of the optimal point (achieving optimal point is a NP hard problem). Note all the aforementioned calculations are offline and are implemented using the frequency response model. Essentially the model is used for looking up the frequency response of the circuit, for any combination of attenuator values. This offline algorithm implementation is therefore extremely fast — a non-optimized C++ implementation takes less than 1μ sec to converge.

In practice, we find that offline solution calculated above might yield a point that provides an analog cancellation of $45 - 50$dB due to manufacturing variation of attenuator (the S parameter data provided is accurate to 2%, thus every attenuator has its own response different from the provided standard data). To further improve the cancellation, we use an additional gradient descent step. Typically, gradient descent takes several hundreds of iterations, however here since we are starting the descent from a much more accurate starting point, the gradient descent converges to the required point in 10-12 iterations. So in the worst case, we show experimentally that analog cancellation tuning can take around 900-1000μs. Assuming we have to do such tuning once every $100ms$ (which is what we needed in our testbed), that represents less than 1% overhead for tuning.

2.4 Implementation

Fig. 2.6 shows the prototype of a single full duplex radio. To implement it we designed our own analog circuit boards for cancellation and integrated them with existing software radios. We also implemented the digital cancellation algorithms in the software radio. Below we discuss the different pieces.

Analog Cancellation Board: The analog cancellation board is a 10×10 cm PCB board designed and built using Rogers 4350 material. The fixed delay lines are implemented using micro-strip trace lines of different fixed lengths. The attenuators are programmable step Peregrine PE43703 [19] attenuators which can be programmed in steps of 0.25dB from 0 to 31.5dB for a total of 128 different values.

Radio Transceiver and Baseband: Our goal was to design and implement a full duplex system that was capable of supporting the latest WiFi protocol 802.11ac with least 80MHz of bandwidth in the 2.4GHz range and 20dBm average TX power. Unfortunately none of the widely used software radios, such as USRPs or WARPs, support such high performance; at best they are capable of supporting 20MHz bandwidths. For that reason, we prototyped our design using radio test equipment from Rohde and Schwarz. For our transmitter, we used a SMBV 100A vector signal generator [107] to send our desired WiFi signals. Since the SMBV is not capable of generating 20dBm power, we use an external power amplifier [21]. For the receiver, we use the RS spectrum analyzer [106].

A practical concern is how to kick-start re-tuning of analog cancellation. Specifically if analog cancellation drops below a threshold, then the receiver might get saturated and the feedback needed

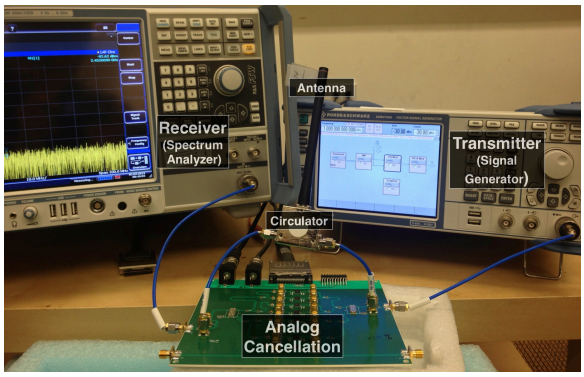

Figure 2.6: Experimental set-up of our full duplex transceiver

to tune is distorted. To tackle this we implemented an AGC via a digital tunable attenuator in front of the LNA. The idea is that if the baseband detects that the receiver is getting saturated, then it programs the attenuator to a large value which brigs the whole signal down to within the dynamic range. After cancellation is tuned, this attenuation is turned off. The FSW is capable of receiving 100MHz signals at 2.45GHz, down-converting and digitizing it to baseband, and then giving us access to the raw IQ samples, which we can then freely process using our own baseband algorithms. The noise floor of this receiver is -90dBm at 100MHz bandwidth. It has a 16 bit ADC capable of sampling a 100MHz signal, however to ensure that we are only using resources found in commodity WiFi cards we configure the ADC to only use 12 bits of resolution.

The IQ samples are transported via ethernet to a host PC, on which we implement our cancellation and baseband software. We implemented a full WiFi-OFDM PHY that can be configured to operate over all the standard WiFi bandwidths (20MHz, 40MHz, and 80MHz). We support all the WiFi constellations from BPSK to 64-QAM for 40MHz, and 256 QAM for 80MHz. We also support all the channel codes with coding rates (1/2, 2/3, 3/4 and 5/6 convolutional coding). Finally we also implement our digital cancellation algorithm in software on the same host PC.

However to show that our design is general and does not benefit from using expensive test equipment, we also develop an implementation using standard WARP radios. Due to their radio limitations, these results will be for 20MHz signals which is the widest that the WARP supports.

2.5 Evaluation

In this section we show experimentally that our design delivers a complete full duplex WiFi PHY link. We prove the claim in two stages. First, we show that our design provides the 110dB of self interference cancellation required to reduce interference to the noise floor. We also show experimentally that the received signal is received with almost no distortion in full duplex mode (the SNR of the received signal is reduced by less than 1dB on average), and that these results are consistent

across a wide variety of bandwidths, constellations, transmit powers and so on. Second, we take the next step and design a working full duplex communication WiFi link. We show experimentally that it delivers close to the theoretical doubling of throughput expected from full duplex.

We start with an experimental evaluation of the cancellation system. We define two metrics we use throughout this section:

- *Increase in noise floor*: This is the residual interference present after the cancellation of self interference which manifests itself as an increase in the noise floor for the received signal. This number is calculated relative to the receiver noise floor of the radio of −90dBm. For example, if after cancellation we see a signal energy of −88dBm, it would imply that we increased the noise floor by 2dB.
- *SNR loss*: This is the decrease in SNR experienced by the received signal when the radio is in full duplex mode due to any residual self interference left after cancellation. To compute this we first measure the SNR of the received signal when the radio is in half duplex mode and there is no self interference, and then with full duplex mode. The difference between these two measured SNRs is the SNR loss.

We compare our design against two state-of-the-art full duplex systems presented in prior work.

- *Balun Cancellation*: This design [71] uses a balun transformer to invert a copy of the transmitted signal, adjust its delay and attenuation using programmable attenuators and delay lines and cancel it. The design also uses two antennas separated by 20cm one each for TX and RX which automatically provides 30dB of self interference reduction. We implement this design and optimize it to produce the best performance.
- *Rice Design*: This design uses an extra transmit chain in addition to the main transmit chain. The extra chain generates a cancellation signal that is combined with the signal on the receive chain to cancel self interference. This design also uses two antennas and to make a fair comparison we use a 20cm separation as the balun based design. However we also provide results with 40cm separation since that was the value used in the prior work. We implement this design by using an extra signal generator as an extra transmit chain for cancellation.

Note that our design uses a single antenna and therefore does not have the benefit of the 30dB of self interference reduction that prior schemes enjoy from using two physically separate antennas.

2.5.1 Can we cancel all of the self interference?

The first claim we made in this chapter is that our design is capable of canceling all of the self interference for the latest operational WiFi protocols. To investigate this assumption, we experimentally test if we can fully cancel a 80MHz WiFi 802.11ac signal upto a max transmit power of 20dBm (all of which are the standard parameters used by WiFi APs), as well as the smaller bandwidths of 40MHz

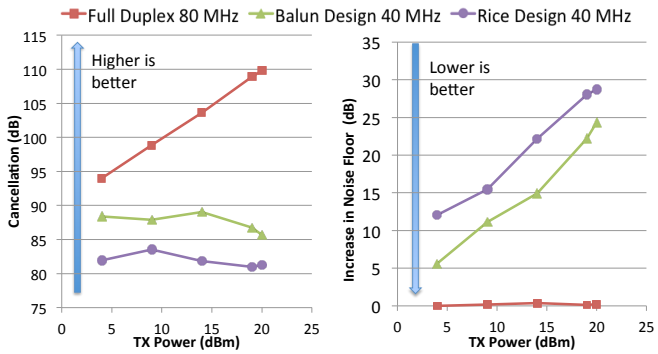

Figure 2.7: Cancellation and increase in noise floor vs TX power for different cancellation techniques with transmission of WiFi 802.11 signal. Our full duplex system can cancel to the noise floor standard WiFi signals of 20dBm at highest WiFi bandwidth of 80MHz, while prior techniques still leave 25dB of self interference residue, even for the narrower bandwidth of 40MHz.

and 20MHz. We conduct the experiment by placing our full duplex radio in different locations in our building. Further we increase the transmit power from 4dBm to 20dBm (typical transmit power range). For each TX power and location (in total 100) we conduct 20 runs and compute the average cancellation across those runs and locations. The goal is to show that we can cancel to the noise floor for a variety of transmit powers up to and including the max average TX power of 20dBm. Fig. 3.12 plots the average cancellation as a function of TX power. It also plots the corresponding observed increase in noise floor on the other axis.

Fig. 3.12 shows that our design essentially cancels the entire self interference almost to the noise floor. In the standard case of 20dBm transmit power, the noise floor is increased by at most 1dB over the receiver noise floor. The amount of cancellation increases with increasing TX power, reaching the required 110dB for the 20dBm TX power. The takeaway is that as the TX power increases, self interference increases at the same rate and we need a correspondingly larger amount of cancellation, which our design provides.

PAPR: Note that these are average cancellation numbers, in practice our WiFi transmissions exhibit transient PAPR as high as 10dB, so the peak transmit power we see is around 30dBm. We do not report the specific numbers for these due to lack of space, but our cancellation system scales up and also cancels these temporary peaks in the self interference signal to the noise floor.

The prior balun and Rice designs however fare far worse. Further, since these designs perform very poorly at 80MHz, we only report their results for the smaller 40MHz WiFi bandwidth and 20dBm TX power. As we can see, these designs can at best provide 85dB and 80dB of cancellation respectively. In other words they increase the noise floor by 25dB and 30dB respectively. The reasons for this are the ones we discussed in Sec. 2.2.2, the inability to adequately cancel transmitter noise in analog and the inability to model non-linear distortions produced by radios. To check if these designs could be made to work with larger antenna separation, we repeated the experiment

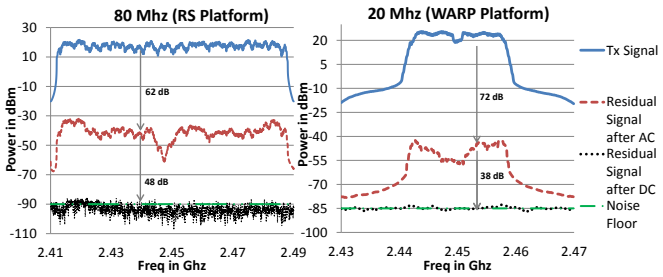

Figure 2.8: Spectrum Response for our cancellation with the Rohde-Schwarz (RS) radios and the WARP radios. The figure shows the amount of cancellation achieved by different stages of our design. It also shows that our design provides the same 110dB of cancellation even with WARP radios.

with an antenna separation of 40cm instead of 20cm. We found that even with an impractical rough half meter separation in antennas, the noise floor increase is at least 20dB.

Does our design work with commodity radios?

We repeat the above experiment, but instead of the Rohde-Schwarz test equipment, we use off-the-shelf WARP radios in the setup. The goal is to show that our design can work with cheap commodity radios and does not depend on the precision of test equipment. Since the widest bandwidth that the WARP can support is 20MHz, we only report results for that bandwidth. Fig. 3.11 shows the spectrum plot of canceled signals after different stages of cancellation. For comparison, we also plot the spectrum plot of cancellation using the Rohde-Schwarz equipment.

As we can see, our cancellation completely eliminates self-interference even with commodity WARP radios. The WARP has a worse noise floor of -85dBm compared to the -90dBm of the RS equipment. Hence if we used 20dBm transmit power, then a slightly smaller 105dB of self-interference cancellation is required to eliminate it to the noise floor. However for consistency, for the WARP experiments we increase the transmit power to 25dBm to show that our design can still achieve 110dB of cancellation and eliminate self-interference to the noise floor.

SNR loss of the Received Signal in Full Duplex Mode

The previous section provided evidence for the amount of cancellation and increase in noise floor. However the experiments had only one radio transmitting. A natural question is how well does the system work when we are in true full duplex mode, i.e. the radio is transmitting and simultaneously receiving a signal. In this section, we evaluate the SNR loss for the received signal when operating in full duplex mode.

The experiment is conducted as follows. We setup two nodes capable of full duplex operation in our building. The two nodes first send 20 WiFi packets (with the following PHY parameters: 80MHz bandwidth, 20dBm TX power, 64QAM constellation) to each other one after the other, i.e.

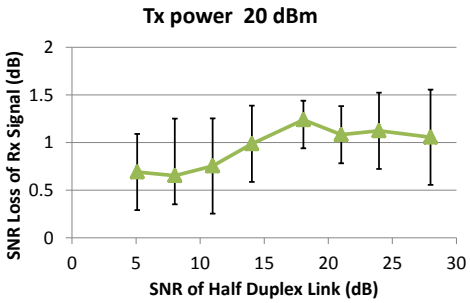

Figure 2.9: SNR loss vs half duplex SNR at fixed TX power = 20 dBm, constellation = 64 QAM, bandwidth = 80MHz with transmission of WiFi 802.11 signal. Our full duplex system ensures that the received signal suffers negligible SNR loss regardless of the SNR it was received at.

they take turns and operate in half duplex mode. They then send 20 WiFi packets to each other simultaneously, i.e. they operate in full duplex mode. For each run we measure the average SNR of the received packets across the 20 packets in half duplex mode, and then with full duplex mode. We then compute the SNR loss which is defined as the absolute difference between the average half duplex SNR and full duplex SNR measured above. We repeat the experiment at several different locations of the two nodes in our testbed. We plot the SNR loss as a function of the half duplex SNR in Fig, 2.9.

As Fig. 2.9 shows the SNR loss is uncorrelated with the half duplex SNR value and is almost identical to the increase in noise floor value we saw in the previous experiment. The takeaway is that self interference cancellation is not impacted by the received signal's strength, whether it is weak or strong. Further, the SNR loss is typically around or less than 1dB which implies that even in full duplex mode the received signal should retain almost the same throughput as in clean half duplex mode.

2.5.2 Digging Deeper

Impact of Constellation and Bandwidth

We conduct two experiments. First we use the same setup as the SNR loss experiments and fix the bandwidth to 80MHz, but vary the constellation for the transmitted signal for the full duplex node from QPSK to the densest constellation in WiFi 256-QAM. Once again we calculate the SNR loss of the received signal across different measurements and locations from the half duplex node. In the second experiment we fix the constellation to 64-QAM but vary the bandwidth from 20 to 40 to 80MHz and once again calculate the SNR loss of the received signal. We repeat this experiment for different locations of the two nodes. Fig. 2.10 plot the CDFs of the SNR losses for different choices of constellations and bandwidth.

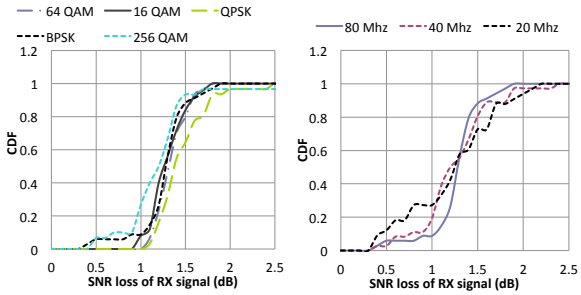

Figure 2.10: Shows CDF of SNR loss with changing bandwidths and constellations. Left: we see the SNR loss for different constellations with TX power = 20 dBm and bandwidth = 80MHz. Right: we see the SNR loss for different bandwidths (20 MHz, 40 MHz and 80 MHz) for TX power = 20 dBm and constellation = 64 QAM. Observe we can support all WiFi modulation schemes and bandwidths with low SNR loss.

As the figures show, our design performs consistently well for all constellation choices and bandwidths. Our cancellation technique makes no assumptions about what constellation and other parameters the PHY is using: for us all of them are a self interference signal and hence the design is unaffected by constellation choice. Our design also works equally well for all the bandwidths used by 802.11ac in the 2.4GHz band. The reason is that our analog cancellation, as we will show in the next section, has sufficient flexibility to provide an almost flat wideband cancellation, while prior designs are extremely narrow-band and cancellation tapers off quickly with wider and wider bandwidths.

Deconstructing Analog Cancellation

In this section we dig deeper into the analog cancellation component of our design. The key parameter in our analog cancellation circuit board is the number of fixed delay lines as discussed in Sec. 2.3.1. We conduct an experiment to examine the impact of the number of such lines. However since these are circuit boards, we do not have the flexibility to vary the number of lines in increments of one. The granularity of our board design allows us to only test two configurations, one with 8 lines and one with 16 lines. We conduct the same self interference cancellation experiment as described in Sec. 4.5. We measure the signal after analog cancellation (without digital cancellation) and plot the frequency response of the canceled signal for the two cases in Fig. 2.11. The plot should be read as the power of the self-interference signal after analog cancellation as a function of the frequency.

As Fig. 2.11 shows, with 8 lines we can achieve 45dB of cancellation over 80MHz, while we can achieve 63dB of cancellation with 80MHz. The reason for the difference is the higher capability of 16 lines in canceling signal reflections in addition to the main self interference component that is leaking through the circulator. When the full duplex node is transmitting, the response from the circulator and antenna in the RX chain has two primary leakage components from the TX signal: one due to the direct leakage from the TX port of the circulator to the ("isolated") RX port of the circulator, and one due to reflections from impedance mismatch between the circulator and the

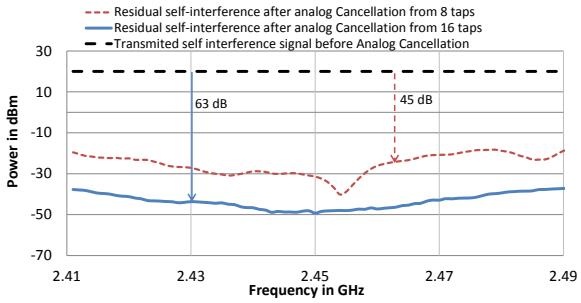

Figure 2.11: Frequency domain representation of self interference before analog cancellation and self interference after analog cancellation using 8 taps and 16 taps. Note that with 16 taps we can provide at least 63 dB of analog cancellation over the entire 80 MHz of bandwidth.

antenna. Because these two components travel different paths in the circulator from TX port to RX port, they undergo different delays as deduced from time domain measurements. These delays are fixed and are a function of the particular circulator and antenna we choose to use. In our implementation we find the delay of the direct leakage component is 400 picoseconds, while the reflected component is centered around 1.4 nanoseconds. With 16 lines we have the capability to center the first 8 lines to have delays around 400 picoseconds, and the other 8 lines around 1.4 nanoseconds. We can then use the interpolation trick discussed in Sec. 2.3.1 to cancel both the direct and reflected self interference components precisely. As expected with 8 lines, our flexibility is reduced in terms of placing our delay lines around the actual delays experienced by the self interference and consequently cancellation is reduced.

Deconstructing Digital Cancellation

After 62dB of analog cancellation, digital cancellation needs to clean up 48dB and 16dB of linear and non-linear self-interference components respectively. In this section, we deconstruct the amount of linear and non-linear cancellation achieved by our design. To conduct this experiment, we tune our analog cancellation circuit to provide 62dB of cancellation. We then progressively add more components to our digital cancellation design. We first implement only our "linear " digital cancellation which cancels only the linear main self interference components and multipath reflections from the environment. We then add the capability to model non-linear components which we christen "non-linear cancellation" . We calculate the cancellation achieved by these two variants of digital cancellation techniques. For comparison with prior work, we also implement only the digital cancellation technique described in the balun based design [71]. We plot the increase in noise floor for all the techniques as a function of Transmit power in Fig. 2.12.

As we can see, our full digital cancellation technique cancels everything to the receiver noise floor. Further, notice that just our linear digital cancellation stage leaves 16 dB of self interference residue above the receiver noise floor. Being able to model the non-linear harmonics allows us to

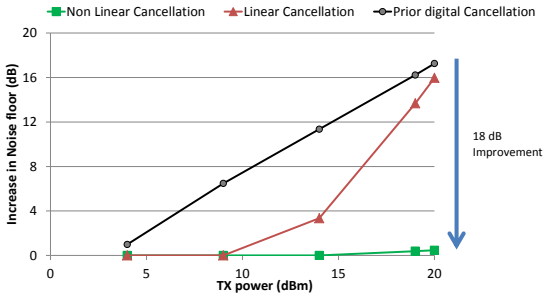

Figure 2.12: Performance of digital cancellation showing impact of different components of the algorithm vs TX power with fixed constellation = 64 QAM, bandwidth = 80MHz. Our algorithm cancels the main component, reflections and harmonics, thus ensuring that self interference is completely eliminated, and the increase in noise floor less the 1dB. Prior techniques can not cancel harmonics, and therefore increase the noise by 18dB.

reduce self interference by a further 16 dB and cleans out the non-linear distortions almost to the receiver noise floor. In comparison, the prior work's digital cancellation technique falls far short, leaving nearly 18dB of self interference residue over the noise floor since it cannot model non-linear distortions. Note that we have given prior work the benefit of an analog cancellation of 62dB from our circuit, as we saw before in Sec. 3.5.1 if we used their implementation of analog cancellation the numbers are worse.

Dynamic Adaptation

As environmental conditions change, the level of cancellation drops since the values of the attenuators used will be off w.r.t to the new conditions. In this section, we evaluate how long it takes to re-tune analog cancellation, as well as how often it needs to be re-tuned in our indoor environment. Note that digital cancellation is tuned on a per-packet basic, hence it is not a concern. Analog cancellation has to be tuned via a special tuning period during which no data is transmitted, hence quantifying that overhead is important.

We conduct this experiment in our busy indoor environment with other WiFi radios and students moving around. Note that an indoor environment is the worst case scenario for full duplex, because of the presence of a large number of reflectors near the transmitter. Outdoor LTE scenarios are less likely to have such strong near-field reflectors, hence we believe our design extends relatively easily to outdoor LTE scenarios. We place the full duplex node and conduct analog cancellation tuning as described in Sec. 2.3.3. Specifically, we use the WiFi preamble to determine the initial settings of the attenuators to be used to match the frequency response of the circulator and antenna. Next we run a gradient descent algorithm to further improve the cancellation from that initial point. Each iteration of the gradient descent consumes 92μs since we have 16 different directions to compute the gradient one (corresponding to the 16 different attenuators). We compute the time it takes for the analog cancellation to converge. We repeat this experiment several times for different node placements and

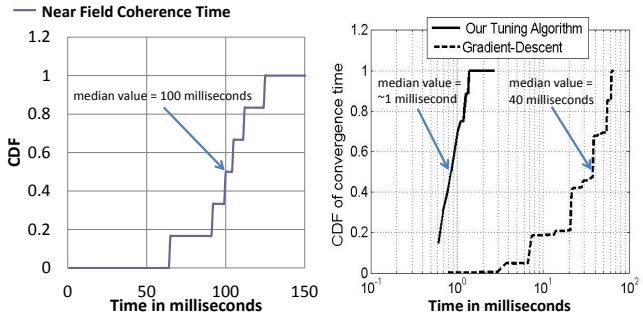

Figure 2.13: Left figure shows CDF of near field coherence time. This implies that we have to retune analog cancellation on an average of every 100 milliseconds. Right figure shows how long it takes for our tuning algorithm to converge to the required cancellation, after the initiation of tuning. We observe exponential improvement compared to the gradient descent algorithm which takes an order of magnitude longer.

environmental conditions and plot the average convergence time. We also conduct an experiment where we do not use the initial frequency based tuning and only use gradient descent from a random starting point for the attenuator values. We show the cancellation achieved as function of tuning time on right side of Fig. 2.13.

As we can see in right side of Fig. 2.13, our analog tuning converges in around 920μs, compared to the 40 or more milliseconds it takes for a pure gradient descent based approach. The reason is that the frequency based initial point estimation provides a point very close to optimal, and from that point a few gradient descent iterations allow us to find the optimal point. Our cancellation algorithm therefore tunes an order of magnitude faster than a simple gradient descent based approach.

But an important question is how often do we have to tune? Analog cancellation has to re-tuned when there is a change in the near-field reflections, since it cancels only the strong components (components 50 dB above noise floor, farther out reflections are weaker than this 50dB threshold). Hence the question is how often do the near-field reflections change? As expected, this depends on the environment, for the indoor office deployments we used in our experiments we found that we needed to retune once every 100ms on average (outdoor scenarios would be easier since changes in near field occur less frequently, and we leave mobile hand-held scenarios to future work). We show this experimentally in Fig. 2.13, the left plot shows the amount of cancellation observed as a function of time after we have found the optimal operating point from a large collection of different experimental runs in our testbed. We define the "near field coherence time" of analog cancellation as the time upto which the receiver remains unsaturated from when it was tuned, which we also use as the trigger to rerun the tuning algorithm. As we can see the near field coherence time for the cancellation is roughly 100 milliseconds. In other words, we have to retune the analog cancellation once every 100 milliseconds, which leads to an overhead of less than 1%.

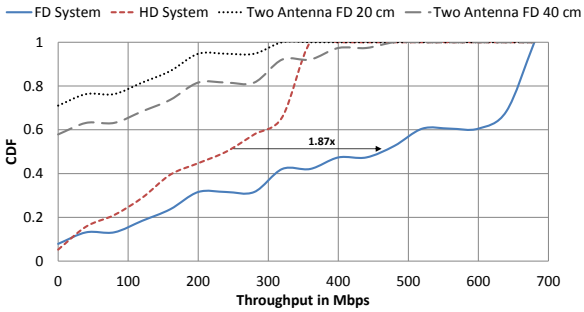

Figure 2.14: CDF of throughput for full duplex link using TX power = 20 dBm, bandwidth = 80MHz. We see a median gain of 87% using full duplex as compared half duplex. Further, prior full duplex with two antenna's separated by 40cm show gains, only in 8% of cases.

2.5.3 Does Full Duplex Double Throughput?

This section demonstrates experimentally that our design delivers close to the theoretically expected doubling of throughput for a full duplex WiFi link. Note that this is a PHY layer experiment, a full MAC design for full duplex WiFi is beyond the scope of this chapter.

We conduct these experiments as follows. We place the two full duplex nodes at different locations and send trains of 1000 packets in full duplex mode, and then similar trains for each direction of the half duplex mode. Each train uses a particular bitrate (from WiFi) and we cycle through all the bitrates for each location. We pick the bitrate with the best overall throughput for full duplex, two antenna full duplex and half duplex respectively. We repeat this experiment for different locations. We found the SNRs of the links varied uniformly between $0-45$dB across locations as we would find in a typical indoor deployment. We plot the CDF of the throughput for half duplex and full duplex link in Fig. 3.15. Note that all of these throughput numbers account for the overhead introduced by the periodic analog cancellation tuning. As we can see, our full duplex system achieves a median throughput gain of $1.87\times$ over the standard half duplex mode. As we known from the experimental analysis in Sec. 2.5.1 that there is a small SNR loss due to a small amount of self interference residue. This SNR loss is the reason that instead of the theoretical $2\times$, we see a slightly reduced gain of $1.87\times$.

How do prior designs perform? We found that in 60% of the scenarios, the throughput with prior full duplex techniques was zero. This is because these designs leave at least 25dB of self-interference residue that acts as noise and if the link SNR is below 30dB no signal is decoded (WiFi requires a minimum of $4-5$dB to decode even the lowest rate packet). As the half-duplex link SNR increases, performance improves but is still not sufficient to beat the system throughput achieved by half duplex. The reason is that even if the link half-duplex SNR is 35dB, it implies that we only have two 10dB links for full duplex. The throughput achieved with a single 35dB half duplex link is still higher than two 10dB links. Consequently the only region where we could find improvements for

full duplex over half duplex with prior techniques was when the link SNR was greater than 40dB.

2.6 Discussion & Conclusion

We believe this chapter marks an important step in proving that full duplex is not only possible, but feasible and practical. Further, it can be deployed with no overhead in terms of antennas used and yet achieve the theoretical doubling of throughput.

MIMO: The current design targets SISO scenarios. For MIMO we could use the same design, but a key challenge is that the cross-talk between different antennas also has to be canceled in analog. Hence, an analog cancellation circuit has to be designed that models not just the distortions through a circulator and a single antenna, but also the distortions that happen when signals travel across antennas. Designing an efficient space-compact circuit for this problem is part of next chapter.

Finally, we would like to comment that full duplex radio design is a problem that spans three different research areas: RF circuit & system design, digital signal processing and networking. The problem cannot be solved in any one domain alone, the solution in our opinion requires understanding trade-offs across all these domains and architecting it appropriately. Historically however, these communities have been separate, RF system designers expect baseband IQ samples as the interface and view their job as sending and receiving signals in RF from these baseband IQ samples. DSP designers view their job as converting between bits and IQ samples efficiently in the presence of noise. Finally, networking researchers transact in bits and packets and design medium access while abstracting out the underlying details. Realizing and taking advantage of full duplex requires research that spans across these domains, and this work represents a step in that direction.

Chapter 3

Scalable MIMO self-interference cancellation: Full Duplex MIMO radios

3.1 Introduction

Full duplex radios have garnered significant attention recently in academia and industry [57, 34, 71, 69, 51, 60, 58, 48, 101, 66, 77, 73]. Several efforts are now underway to include full duplex technology in future cellular 5G standards [12], as well as explore applications of the technology in current wireless infrastructure. However these efforts are hampered by the fact that there aren't viable and efficient full duplex designs that can work in conjunction with MIMO. Specifically, no current practical designs are known which can enable one to build a M antenna full duplex MIMO radio that can transmit and receive from all antennas at the same time and double the throughput. The best known prior MIMO full duplex system, MIDU [34] requires $4M$ antennas for building a full duplex M antenna MIMO radio, and even then fails to provide the needed self-interference cancellation for WiFi systems (20 MHz bandwidth) to achieve the expected doubling of throughput.

Recent work has however demonstrated that a single antenna (SISO) full duplex system is practically possible [41]. Specifically, it demonstrates the design and implementation of a cancellation system for a SISO system that completely cancels self-interference to the noise floor and consequently achieves the theoretical doubling of throughput. A natural question therefore is why not just replicate the same design M times to build a MIMO M full duplex radio? After all, a MIMO radio can be conceptually and physically viewed as a collection of M single antenna SISO full duplex radios.

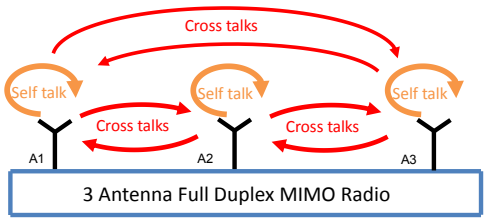

Figure 3.1: Shows a 3 Antenna MIMO Full Duplex node, with different interference's referred as talk. Every chain sees 2 other cross-talks other than the self-talk.

The challenge is cross-talk interference as seen in Fig. 3.1. When a full duplex MIMO radio transmits, the transmission from any one of the M antennas (interchangeably referred to as transceiver chains) propagates to the other antenna (chains) and causes a large amount of interference. For the sake of clarity, in this chapter we will refer to the self-interference at a receive chain caused by a transmission from the TX-chain with which the receive chain shares an antenna as "self-talk", and the interference from a neighboring TX chain's transmission as "cross-talk". Since MIMO antennas are closely spaced due to size constraints, this cross-talk is extremely strong, almost 75-80dB stronger than the desired signal that is being received on that chain. Consequently, even if we have cancellation circuits and algorithms that cancel every chain's self-talk, there is an extremely strong cross-talk interference that can saturate the receive chain.

A naive solution is to introduce a separate copy of the cancellation circuit and DSP algorithm for each pair of chains that experiences cross-talk. If there are M antennas, then it would imply a total of M^2 circuits and DSP algorithms. In other words complexity grows quadratically with the number of antennas, which is untenable as MIMO systems go towards 4 to 8 antennas. Supporting 16 cancellation circuits and DSP implementations (for 4 antenna MIMO) on even a WiFi AP based form-factor is untenable (our analysis suggested that with the current SISO design we would need 400sq.cm of analog circuit area and a high-end Virtex FPGA that consumes 80W of power to accommodate the DSP computations). Complexity impacts more than space and power consumption, cancellation systems (both analog and digital) need to be tuned continuously to adapt to environmental changes. The time for tuning scales linearly with the complexity, hence it would take M^2 time longer to tune such a design's MIMO self-interference cancellation system. The best known prior algorithm for tuning [41] requires around a millisecond to tune, so we would need 16 ms to tune for a 4 antenna MIMO system which would be untenable even in a slowly changing environment like indoor WiFi (coherence times are on the order of tens of milliseconds), let alone mobile environments such as LTE.

A second problem is performance itself. The key metric is the residual interference left after cancellation at each receive chain, the residual directly translates to decrease in SNR for the desired received signal. As we will show in Sec. 3.3, even if one could accommodate a quadratic number of circuits and DSP cancellation implementations, the performance degrades linearly with the number

of MIMO chains. In other words, the residual interference after cancellation at each receive chain increases linearly with M. This is due to the accumulation of the residual interference from all the cross-talk and self talk cancellation systems. Once again, as MIMO systems scale to support many antennas, this essentially limits the performance gains of full duplex.

This chapter presents the design and implementation of a MIMO WiFi full duplex radio. Our M antenna full duplex MIMO radio uses each antenna for simultaneous transmit and receive, i.e., it uses the same number of antennas as a standard half duplex M-antenna MIMO radio unlike prior designs. The design uses slightly more than $M\times$ cancellation circuits and DSP algorithms (w.r.t to SISO full duplex design) to cancel all the self and cross talks. In other words, complexity scales linearly with the number of chains, which is the best performance one could expect. Further, the performance does not degrade linearly with the number of MIMO chains, i.e., the residual interference is the same as the SISO design and does not increase linearly with the number of chains. We prototype our design and integrate it with the off-the-shelf WARP software radios [28] running a stock WiFi baseband and demonstrate experimentally that it achieves close to the theoretical doubling of throughput.

Our design solves the key challenge of efficiently and effectively achieving the MIMO full duplex using two major ideas as follows.

- First, a key insight is that MIMO chains are co-located, i.e., "they share a similar environment". Intuitively, the signals transmitted by two neighboring antennas (separated by a few cm) go through a similar set of reflectors and attenuations in the environment [68]. Cancellation systems are essentially trying to model these distortions, so when we want to model cross-talk, we can reuse the work that has been done for modeling the chain's own self-talk interference. This results in a novel "cascaded" filter structure for cancellation that results in an overall design that has near-linear complexity scaling with the number of MIMO antennas.
- Second, the reason performance degrades linearly with the SISO replication based design is that each of the M independent cancellation algorithms for self-talk and cross-talk at a receive chain produce their own estimation error which add up to the linear degradation. Our key insight here is to leverage the fact that we have M transmitters available that can concurrently send training symbols. Specifically, we design a training preamble for WiFi that allows each receive chain to estimate each of the self-talk and cross-talk channels with an error that is M times lower than the SISO design by combining information from all M training symbols. Consequently, in our design when the estimation errors add up for the self-talk and cross-talk cancellations, *the overall error or residue is the same as a SISO system would have achieved, which is the best one can hope for. Further the algorithms are modular and structured in a way that, if in the future the SISO full duplex design manages to improve its performance even further, the MIMO design in this chapter immediately benefits.*

We prototype our design using our own custom designed analog cancellation circuits, and integrate them with novel implementation of our digital cancellation algorithms using off-the-shelf

Power and Interference relative to noise floor of -85 dBm

	Power level in dBm	Cancellation needed in dB
Total TX signal	20	105
Linear component	20	105
Non-linear component	-10	75
Transmitter Noise	-20	65

Figure 3.2: The different components of the transmitted signal (self-talk) for a typical WiFi radio. The second column tabulates the amount of self-talk cancellation needs to eliminate the corresponding self-talk component to the noise floor.

WARP radios [28]. Our experiments demonstrate that in a 3×3 configuration, our system achieves a performance that leaves a negligible 1dB of self-interference after cancellation. We also show that our system achieves a 95% throughput gain over half duplex radios using a standard WiFi compliant OFDM PHY of 20MHz for 802.11n for all different modulations (BPSK, QPSK, 16QAM and 64 QAM) and coding rates of (1/2,2/3,3/4,5/6), supporting three streams for 3×3 MIMO.

3.2 The Problem

In this section, we describe the nature of interference in a MIMO full duplex radio and then discuss the architectural challenges in designing a cancellation system.

Self-talk or cross talk (or for that matter any transmitted signal) is made up of three major components [29, 35, 22]:

- Linear Signal: This is the signal that the baseband modem wanted to transmit and is then distorted by channel reflections. It's linear because it can be represented as a linear combination of delayed and summed copies of the same signal that arise from environmental multi-path reflections.
- Non-linear Signal: This is the signal that is generated due to non-linear transformations that the linear signal goes through when it is passed through analog radio components such as mixers, power amplifiers in the transmit chain [87].
- Transmit Noise: This is the noise that is generated by active components in the TX chain such as power amplifiers and local oscillators (we club things such as broadband noise and phase noise into this term for the sake of brevity).

The relative strengths of these components depends on the quality of the radio. Fig. 3.2 tabulates the strengths of the different components we empirically measured for a commodity 20dBm WiFi SISO radio, and the amount of cancellation needed to eliminate them in a full duplex system. Note that this is a cheap radio widely used in many commercial WiFi devices [22, 28], so we believe this is representative of the WiFi radios in general.

The above analysis is of course true even for a single antenna radio without MIMO, and recent work [41] describes cancellation techniques that eliminate all self-talk. However, what is unique

Power and Interference relative to noise floor of -85 dBm

MIMO FD, Receiver 1	Power in dBm			Cancellation needed (dB)		
	Self-talk	Cross talk 1	Cross talk 2	Self talk	Cross talk 1	Cross talk 2
Overall signal at antenna 1	15	-9	-15	100	76	70
Linear component	15	-9	-15	100	76	70
Non-linear component	-15	-39	-45	70	46	40
Transmitter noise	-25	-49	-55	60	36	30

Cancellation Requirement

MIMO FD, Receiver 1	Self-talk	Cross Talk 1	Cross Talk 2
Analog cancellation	65 dB	41 dB	35 dB
Digital cancellation	35 dB	35 dB	35 dB

Figure 3.3: Interference components and cancellation requirements for 3 antenna MIMO full duplex. The first table describes the levels of different interference components (linear, non-linear and transmit noise) that make up self-talk and cross-talks at one receiver in a 3 antenna MIMO radio. Cross-talk 1 is from the neighboring antenna and cross-talk 2 is from the farther neighboring antenna. The second table lists the overall cancellation needed, here the values are bumped up by 5dB relative to the first table to ensure that even when the residues left from the self-talk and the two cross-talk cancellations are added up, the overall noise floor does not go up (else it would go up by 5dB if the cancellation requirement for each component did not have a 5dB margin).

with MIMO is cross-talk. In other words, the interference that results at a receive chain due to a transmission from a neighboring co-located MIMO antenna/chain. In a 3 antenna full duplex MIMO radio, each receiver chain would see two cross-talk signals from the other two antennas as seen in Fig. 3.1.

Cross-talk is slightly weaker than the self-talk generated by the chain's own transmission, but is still quite strong and has all the above three enumerated components. Like the earlier SISO design [41] (as shown in Fig. 3.4), the transmit noise component of the cross-talk signal has to be canceled in the analog domain, whereas the non-linear and linear components could be canceled in both analog and digital domains. Fig. 3.3 tabulates the strengths of the various components that make up a cross-talk and self-talk signal in a typical 3-antenna MIMO WiFi radio with 20dBm[1] transmit power (note that the power is divided equally among all three transmitters, so the power out of each antenna is 15dBm).

3.2.1 Why can't we reuse the SISO full duplex design by replicating it?

At first glance, the MIMO interference cancellation problem looks quite similar to a SISO full duplex problem, only replicated a few times. After all the cross-talk signal that needs to be canceled looks like an attenuated version of a chain's own self-talk signal that the SISO design manages to cancel completely. So why couldn't we replicate the SISO design $M^2 - M$ times for each of the cross-talk signals in a M antenna MIMO radio and be done with it (as shown in Fig. 3.5)?

[1]The FCC specifies that the peak power can be 30 dBm [10]. However OFDM signals have a high PAPR, i.e. the peak power of the output signal is significantly higher than the average power. For WiFi we find that the PAPR is 10dB, so the average power we can use is actually 20dBm.

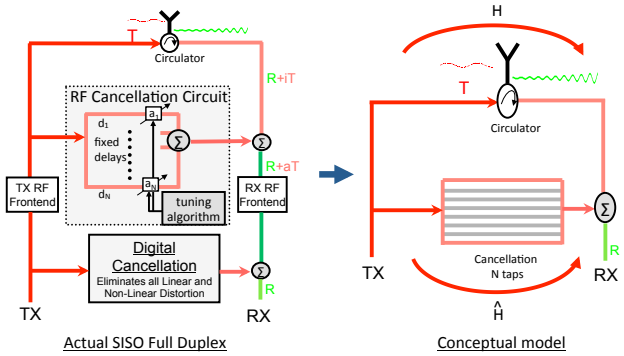

Figure 3.4: Prior best performing SISO full duplex design. The figure on the right shows an equivalent conceptual filter based view of self-talk cancellation. The filter is parameterized by its complexity, the number of taps. The filter subsumes both analog and digital cancellation.

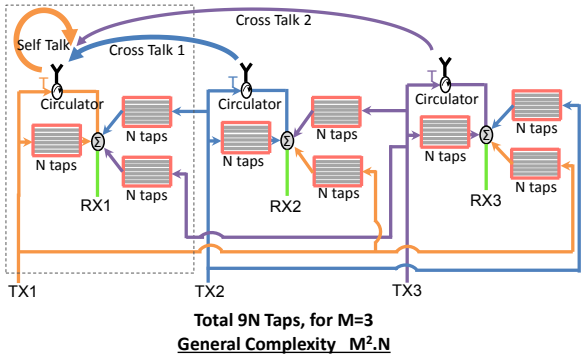

Figure 3.5: **SISO Replication Based Design**: Shows a 3 antenna full duplex MIMO radio, using nine SISO cancellation circuits (SISO replication design). This design uses in total 9N taps for M=3 assuming each circuit requires N filter taps. In the general case this design would require M^2N for a M antenna full duplex MIMO system.

To understand the reason this might not work, it will help to have a conceptual understanding of what a SISO self-talk cancellation system accomplishes. At its core, the self-talk cancellation technique can be thought of as shown in Fig. 3.4. The input is the baseband signal that is being transmitted, to which transmit noise is added and the combined signal is passed through a linear and non-linear unknown transfer function that captures the distortions introduced by the analog components and the wireless channel and is denoted by H. The cancellation circuits and algorithms are trying to calculate an estimate – $\hat{H}$ – of this unknown transfer function H as accurately as possible (to the tune of 105dB resolution), and then pass a copy of the input baseband transmitted signal and noise through this estimated transfer function $\hat{H}$ to recreate the self-talk and cancel it (shown in Fig. 3.4). The estimated transfer functions are created using tunable **analog and digital FIR filters**, for example the prior SISO design's analog cancellation circuit requires 12 delay-attenuation taps that each represent a single analog FIR filter tap (refer Fig. 3.4), and what

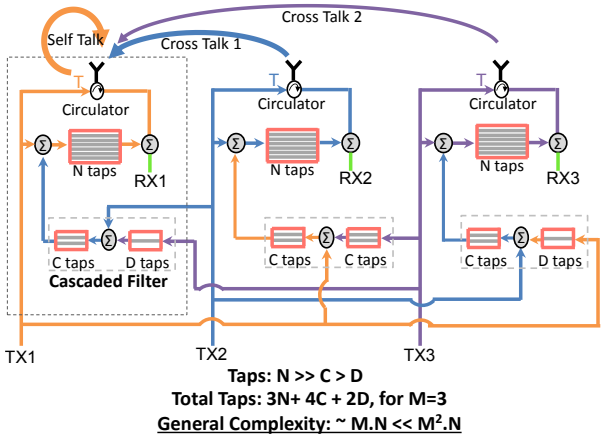

Figure 3.6: **Cascaded Cancellation Design**: Shows a 3 antenna full duplex MIMO radio design with cascaded filter structure for cancellation. The structure is shown for receiver chain 1 only, but the same structure is repeated for the other chains. For, self-talk cancellation we have N filter taps on every chain. Further we have C and D taps feeding in a cascading fashion at the input of the N tap self-talk cancellation circuit. Notice cross talk 1 is stronger so we need more taps ($C > D$) as compared to cross talk 2. However both C and D are significantly smaller than N.

is being controlled is the weight on each tap (practically this translates to controlling the attenuator on that delay-attenuation analog line). A similar FIR filter structure is used for digital cancellation and the challenge is calculating the weights to use on each of the taps. So the key challenge the SISO self-talk cancellation system is solving is calculating a set of FIR filter weights that can accurately model this unknown and time-varying transfer function.

Consequently, there are two metrics that characterize these estimation circuits and algorithms.

- **Complexity**: can be quantified by the number of filter taps that are used in the implementations that represent the estimated $\hat{H}$. The more taps we need, the more analog circuitry is needed as well as DSP resources in FPGA to implement them. Keeping the number of taps low is important so as to reduce the space and power consumed by analog circuits [33] and DSP logic for FIR implementations (the baseline is the SISO design that requires 12 analog taps and 132 digital FIR taps). To get a sense of the impact, 12 analog taps consume roughly 24sq.cm of board area. A second consequence of complexity is the amount of time it takes us to re-tune the cancellation when the environment changes (including things such as temperature). The larger the number of taps, the longer it will take to tune since there are more variables to estimated. When cancellation is being tuned, the radio cannot be operated in full duplex mode. Hence tuning time is pure overhead, and needs to be minimized.
- **Estimation error**: A second key metric is estimation error which manifests as residual interference left after cancellation and directly reduces the SNR of the desired received signal. A perfectly accurate cancellation system would leave no residue. The baseline for this metric

is the best performing prior SISO self-talk cancellation design that leaves 1dB of residue over the noise floor. In other words, the receiver noise floor is increased by 1dB and therefore the SNR of the received signal is also decreased by 1dB. To put this number in context, this is extremely accurate since at most normal receive link SNRs, a 1dB decrease will have negligible impact. The reason for this residue is estimation and quantization error in the algorithms that calculate the weights for the filter taps used in analog and digital cancellation. Estimation error is inevitable and cannot be avoided, but its important to keep it as small as possible.

How well would the SISO replication based design for MIMO perform on these two metrics? The optimal scenario is that the complexity of a M antenna full duplex MIMO radio would be $M\times$ the complexity of the SISO design, and it would have the same estimation error as the SISO design. We cannot do better than a linear increase in complexity and no increase in estimation error. However, the SISO replication based design has a complexity of $M^2\times$ the complexity of the SISO design. This is because it requires us to replicate the SISO design for each cross-talk factor, and therefore we need a total of M^2 versions of the SISO design. In terms of taps this implies $12 \times M^2$ taps in analog circuits alone, along with the corresponding increase in digital cancellation FIR taps.

Second, this design's estimation error turns out to be worse compared to SISO design. At each receiver chain, we show in Sec. 3.3.2 that the residual interference scales linearly with the number of MIMO chains M. Intuitively the reason is that each replica of the SISO design is running an independent estimation algorithm for determining the values of the filter taps to use for cancellation. Since at each receiver chain we have M versions of the SISO design running, we will have a $M\times$ increase in estimation error and consequently the interference residue.

3.3 Design

We present a new cross talk cancellation technique for full duplex MIMO which is scalable and efficient. The key technique behind our MIMO cancellation design is a cascaded filter structure. Specifically, we exploit the fact that in MIMO, cross-talk and self-talk share a similar environment (or similar set of multi-path reflection and attenuation profiles in the channel). Further, cross-talk across chains is naturally reduced compared to the chain's own self-talk because of physical antenna separation. We exploit these insights to design a low complexity and highly accurate cross-talk cancellation system. For canceling the chain's own self-talk we use the design from prior work [41].

3.3.1 Reducing Complexity: The Cascade

Our design builds on a key insight: co-located MIMO antennas share a similar environment. In other words, the transfer function (i.e., the channel response across the frequency) that transforms the cross-talk signal from a neighboring transmit chain at the receive chain has a close relationship with the transfer function that the chain's own self-talk undergoes. Intuitively, this is because the

environment around a radio looks essentially the same to neighboring antennas since they share the same reflectors in the environment, and the distances to these reflectors are almost the same from the closely-spaced antennas. The difference however is that any cross-talk signal experiences an additional delay before it arrives at a receive chain as compared to the chain's own self-talk signal [68, Sec. 2]. Technically this means that the phases of self-talk and cross- talks at a given receive chain might become different due to the delay, but can still be determined by a fixed relationship depending on antenna location and the environment. [2] What's important for MIMO full duplex design however is that cross talk and self talk transfer functions can be expressed as a function of each other, with a modifying factor to account for the antenna separation.

The above insight can be mathematically modeled as a cascade of transfer functions. Let $H_i(f)$ and $H_{ct}(f)$ be the transfer functions of the chain's own self-talk and cross-talk respectively, which are due to environment only, these cannot be directly measured. The overall relationship between these functions can be modeled as follows:

$$H_{ct}(f) = H_c(f).H_i(f) \tag{3.1}$$

where $H_c(f)$ is the cascade transfer function. The key observation is that $H_c(f)$ which cascaded with $H_i(f)$ results in the cross-talk transfer function, is an extremely simple transfer function. Typically $H_c(f)$ is a simple delay that corresponds to the fact that the two antennas are separated and the cross-talk signal experiences slightly higher delay compared to the self-talk.

How might we exploit this insight? The idea is to mimic the cancellation design in a cascade similar to the equation above as seen in Fig.3.6. Specifically, we could design simple low-complexity analog cancellation circuits and digital cancellation filters that model the cascade function $H_c(f)$. These circuits and filters would then feed into the cancellation circuits and digital cancellation filters for the chain's own self-talk cancellation and thus reuse all that circuitry to model the cross-talk channel. Remember that the circuits and digital filters for the chain's own self-talk are modeling $H_i(f)$, hence the cascaded structure is essentially recreating the above Eqn. 3.1. So the only additional complexity compared to the optimal MIMO design would be from the circuits and filters that model the cascade transfer function $H_c(f)$.

The natural question is how to design the cascade circuits itself? The intuition behind the design is to consider what the cascade circuits are exactly canceling compared to the self-talk cancellation circuits. The interference in the self-talk comes from two major factors. The first are the reflections from the antenna (impedance mismatch) and other components such as circulators. The second are

[2]Note that having a deterministic relationship between the self-talk and cross-talk channel responses does not contradict the assumption in MIMO channels that they form spatially independent streams as long as the antennas are separately by half a wavelength. The phase difference typically results in spatially independent streams [75]. Second, note that what we are exploiting is the fact that both the self-talk and cross-talk channels are correlated in their changes across frequency, i.e. the way the self-talk channel and cross-talk channels change across frequency are related and is a function of the environment. This fact has been studied in prior work, for example, a typical point to point LOS indoor MIMO channel can have a specific relationship across frequency across the different MIMO paths and still form spatially independent streams [68, 121, 75].

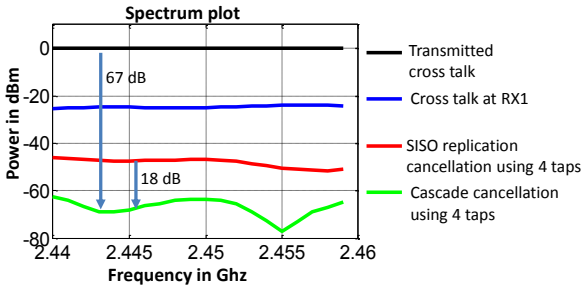

Figure 3.7: Cancellation performance in the frequency domain for the cascaded design and the replication based design with the same complexity for a 3 antenna MIMO full duplex radio operating a WiFi PHY in a 20MHz band at 0dBm TX power(WARP radios [28]).

the reflections from the environment. The reflections from the antenna are only part of the self-talk and are not part of the cross-talk, whereas the reflections from the environment are part of both self and cross-talk. Hence, the cascade cancellation circuit's job is to only cancel the environmental reflections.

The second insight is that the environmental reflections of the cross-talk are related to the environmental reflections the chain's own self-talk cancellation circuit is trying to cancel. To discover this relationship, we conduct the following experiment in a wide variety of locations in indoor scenarios. We first transmit a signal from a single antenna and measure the environmental channel response of the reflections at the same antenna [8]. We then measure the environmental reflection response at the neighboring MIMO antennas. Measuring the responses is possible because we know what we are transmitting, and we can use classic channel estimation techniques to measure the channel impulse response[3]. We then calculate the cascade transfer functions as described in Eqn. 3.1. We collect these calculated transfer functions and then check what is the complexity of the cascade cancellation circuit that can approximate these responses. This is an optimization problem, where the parameter is the number of taps that we are allowed to use in the cascade circuit, and the calculated responses are what we are trying to fit for. The goal is to minimize the number of taps in the cascade circuit, while fitting the cascade responses to a level of 40dB of cancellation (assuming we get 30dB of interference reduction from antenna separation in the cross-talk). The details of the technique are described in [8].

The number of analog taps required to realize the required performance for MIMO using the cascaded design calculated via the optimization above is tabulated in Fig. 3.8. For a typical 3 antenna MIMO WiFi radio with 12cm separation between antennas (typical of APs), the antenna separation itself provides about 24dB of isolation, so we need another 41dB of cross-talk cancellation in analog (see Table. 3.3 for requirements). As we can see we need only four analog taps with the cascaded structure compared to the 12 taps required by the naive design for canceling cross-talk

[3]This experiment is done via WARP software radios as discussed in evaluation.

Resource Comparison between SISO replication and Our design

	SISO replication design	Our design
Analog Cancellation taps (3X3)	108 (12*9)	56 (reduced by 1.92x)
Digital Cancellation taps (3X3)	1188 (132*9)	485 (reduced by 2.45x)
Tuning time (3X3)	9 ms (1ms*9)	.024 ms (reduced by 375x)
Analog Cancellation taps (mXm)	$O(M^2N)$	O(MN)
Digital Cancellation taps (mXm)	$O(M^2R)$	O(MR)
Tuning time (mXm)	$O(M^2)$	O(M)

Figure 3.8: Table showing the reduction in complexity and tuning time with the cascaded design compared to the replication based design for both a 3 antenna full duplex MIMO radio as well as the general case of a M antenna full duplex MIMO radio.

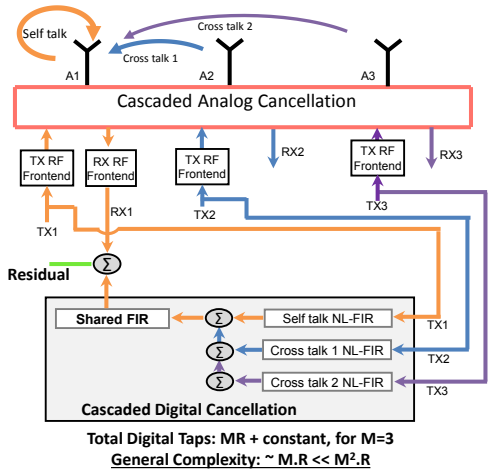

Figure 3.9: Shows the **cascaded digital cancellation** architecture for receiver chain RX1. Similar cascaded digital cancellation is applied to every receiver i.e., RX2 and RX3, not shown in this figure. The cascaded analog cancellation is implemented as shown in Fig. 3.6. The shared FIR brings significant saving of taps for overall MIMO cancellation. The NL-FIR's are the non-linear finite impulse response filter, recreating the digital copy of the unique component for the self-talk and cross-talks to be canceled at a receive chain.

at an adjacent antenna and only two taps, when canceling to the farther out antenna as shown in Fig. 3.6. The cascaded design therefore requires 1.92× lower number of taps compared to the SISO replication design for a 3 antenna full duplex MIMO radio as seen in Fig. 3.8. The reduction factor approaches the optimal 3× number as the number of antennas increases.

To verify the improvement for digital cascading (seen in Fig. 3.9), we conduct a similar experiment with the same setup (but with 20 dBm of total TX power). However, we provide the SISO replication design the required number of taps to meet the requirement on analog cancellation so we can specifically evaluate the benefits for digital cancellation with cascading. As seen in Fig. 3.8, we need a total of 485 taps to cancel self-talk and cross-talk to the noise floor for a 3 antenna MIMO radio. Further, for the SISO replication based design using the same number of taps (485), the residual interference is still an additional 7dB. To achieve the same performance as our cascaded

design with the SISO replication based design, we would need 1188 or 2.45× more taps as tabulated in Fig. 3.8. Once again the reduction factor approaches the optimal number M and the number of antennas (M) grows. Finally in terms of cancellation performance, a 7dB increase in noise floor or reduction in desired signal's SNR is quite high by itself, and when we take into account the reduction in cancellation for analog of 18dB, we are looking at a 25dB reduction in overall cancellation for the SISO replication based design with the same complexity as our cascaded structure.

There are two main benefits to reducing complexity:

Reduction in size, cost and tuning time: Each additional filter tap increases the size of cancellation boards in analog and FPGA resource consumption in digital cancellation. For analog cancellation, our circuits consumed 110sq.cm of board area compared to nearly 216sq.cm for the SISO replication based design for a 3-antenna MIMO full duplex system. For example, we found experimentally that reducing the number of digital filter taps from 1185 to 485 for a 3 antenna MIMO radio means that a lower class Xilinx Kintex series FPGA has sufficient DSP resources to implement the cancellation, whereas the SISO replication based design would require the higher end Virtex FPGA [30]. This translates to enormous power savings, a Virtex FPGA consumes nearly 80W of power whereas a Kintex consumes only 40W on twice as less [31]. Power reduction translates to less heat and consequently simpler AP designs. Also to ultimately realize the design in compact boards, reducing the number of taps as much as possible is a must. A final consequence is the tuning time to compute the weights for each of these taps also reduces linearly with lesser number of taps (tuning time is pure overhead since during tuning the radio cannot be used for communication).

Reduction in Tx power waste: The amount of power that needs to be coupled off from the transmit paths to powering cancellation circuits depends linearly on the number of taps in the cancellation circuits. This is because each tap is of course only useful if some copy of the transmitted signal is passed through it, and in addition each tap has loss associated with it that adds up. Thus reducing number of taps helps reduce TX power waste.

3.3.2 Reducing Residue: Joint Training

The goal of digital cancellation is to clean out any remaining residual self-interference. Once again, a natural question is why not reuse the digital cancellation algorithms designed for SISO? In other words, for each receive chain in a M antenna full duplex MIMO radio, run M separate digital cancellation algorithms that estimate the chain's own self-talk and the other $M-1$ cross-talk interference components. These algorithms work by estimating the distortion experienced by each of the interference (both for linear and non-linear components). They then apply the estimated distortion functions to the known baseband copy of the transmitted signal and subtract it from the received signal.

The above approach doesn't work because every additional and independent digital cancellation algorithm we use in the receive chain linearly increases the residual interference after cancellation. In other words, performance worsens linearly with the number of MIMO chains. To see why, we

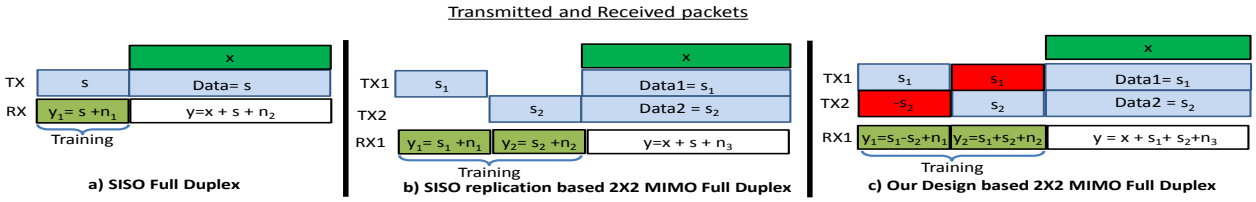

Figure 3.10: This figure shows the transmitted and received packets for a SISO full duplex, 2 antenna MIMO full duplex with the traditional training technique, and our design with the novel training technique. Notice the training symbol structure in the last figure, this allows us to reduce the estimation error by half for the self-talk and cross-talk components for a 2 antenna MIMO radio.

start with describing why even a simplified SISO digital self-interference cancellation algorithm will have some residual interference that cannot be canceled.

Digital cancellation works in two stages, first there is a training phase and then cancellation phase. The training phase uses training symbols (e.g. the WiFi preamble), and the assumption is that there is no desired received signal from the other full duplex node. The training symbols are used to estimate the self-interference. Let's say the training self-interference symbol is s as seen in Fig. 3.10.a. The self- interference symbol is being received after transmission from the same radio (for simplicity assume there is no distortion from the channel), and the receiver adds its own noise n_1 (variance σ^2) to the received signal (this noise comes from effects such as quantization in the ADC). Hence the received signal y_1 can be written as,

$$y_1 = s + n_1$$

The best estimate of the self-interference s in this case is simply y_1 However this estimate $\hat{s}$ has some estimation error, which in this case is simply the power of the receiver noise as show below:

$$\hat{s} = y_1, \quad E((s - \hat{s})^2) = E(n_1^2) = \sigma^2$$

How can we use this estimate to cancel subsequent self-interference? For simplifying the description, let's assume the packet that is being transmitted and is acting as self-interference is simply the same training symbol repeated throughout the packet (real world packets are of course not trivial like this, but this assumption does not change the basic insight below). To cancel this self-interference throughout the packet, the algorithm will simply subtract the above estimate from the overall received signal. Lets say x is the actual desired received signal, the overall signal received is y, and the signal after cancellation, are given by:

$$\begin{aligned} y &= x + s + n_3 \\ \underbrace{y - \hat{s}}_{\text{cancellation}} &= x + \underbrace{s - \hat{s}}_{estimation\ error\ =\sigma^2} + \underbrace{n_3}_{\text{RX noise}} \end{aligned}$$

As we can see, the estimation error shows up as residual interference with variance of σ^2. As the best known prior design has shown this is on the order of 1dB over the half-duplex noise floor.

SISO Replication based MIMO design: It's now easy to see why a design for MIMO that simply uses M replicas of the digital cancellation algorithm at each receive chain for the self-talk and the $M-1$ cross-talk interference signals increases the estimation error roughly by a factor of M. The training symbol structure for a 2×2 MIMO transmission is shown in the Fig. 3.10.b. above, essentially there are two training symbols s_1 and s_2 sent over two slots from the two different transmit chains. The algorithms at a particular receive chain use these symbols like in the SISO case to estimate the self-talk and the cross-talk, and each of them will have their own estimation error. When these estimates are used for cancellation, the estimation errors add up, and the overall estimation error (or residual self-interference) at each receive chain is theoretically two times the SISO case. The math below shows the above intuition formally. First, the estimates for the self-talk and cross-talk symbols are given by:

$$\hat{s_1} = y_1, \quad E((s_1 - \hat{s_1})^2) = \sigma^2$$
$$\hat{s_2} = y_2, \quad E((s_2 - \hat{s_2})^2) = \sigma^2$$

When canceling to attempt to recover the desired received signal x, we can calculate the estimation error as follows:

$$\begin{aligned} y &= x + s_1 + s_2 + n_3 \\ \underbrace{y - \hat{s_1} - \hat{s_2}}_{\text{cancellation}} &= x + \underbrace{s_1 - \hat{s_1}}_{\sigma^2} + \underbrace{s_2 - \hat{s_2}}_{\sigma^2} + \underbrace{n_3}_{\text{RX noise}} \end{aligned}$$

As we can see, the estimation error shows up as residual interference with variance of $2\sigma^2$, both self-talk and cross-talk estimation introduce σ^2 error. We can recursively show that for a general M antenna full duplex MIMO radio, the estimation error and consequently residual interference on each receive chain goes to $M\sigma^2$.

Our Design: Our key contribution is a novel training symbol structure and estimation algorithm that reduces the estimation error for each interference component at each receiver chain (self-talk or cross-talk) to σ^2/M for a full duplex $M \times M$ MIMO radio. The key insight is to re-design the training symbols to reduce the estimation error. Specifically instead of sending training symbols from each of the transmit chains separately in consecutive time slots, we send a combination of all of them from each transmitter in parallel. The idea is to actually leverage the fact that there are two transmitters that could be leveraged to transmit training information jointly and thereby improve accuracy, there is no need to treat each of them separately. Doing so requires an intelligent joint training symbol design so that each symbol can be estimated as a linear combination of the received transmissions. Fig. 3.10.c. shows the main idea. We use a similar set of equations as before to show formally why this works. As seen in Fig. 3.10.c., the training symbols are transmitted by chain 1

and chain 2 simultaneously. In time slot 1, transmitter 1 and 2 transmit s_1 and $-s_2$, respectively. And in time slot 2, transmitter 1 and 2 transmit s_1 and s_2 respectively. Receiver 1, receives the combined symbols in time-slot 1 and time-slot 2, y_1 and y_2. Thus:

$$y_1 = s_1 + s_2 + n_1, \quad y_2 = s_1 - s_2 + n_2$$

Lets assume the rest of the transmissions from the two chains are just repetitions of the same symbols s_1 and s_2 respectively (again this is for description simplicity and suffices to explain the insight). We need to get estimates for the data symbols s_1 and s_2 using the two received training symbols y_1 and y_2. The best estimates are given by:

$$\hat{s_1} = \frac{y_1 + y_2}{2}, \quad E(s_1 - \hat{s_1})^2 = E((\frac{n_1 + n_2}{2})^2) = \frac{\sigma^2}{2}$$
$$\hat{s_2} = \frac{y_1 - y_2}{2}, \quad E(s_2 - \hat{s_2})^2 = E((\frac{n_1 - n_2}{2})^2) = \frac{\sigma^2}{2}$$

As we can see, the error in each of these estimates (self-talk and cross-talk) is $\sigma^2/2$. Now when these estimates are used for cancellation, the following equation results:

$$\begin{aligned} y &= x + s_1 + s_2 + n_3 \\ \underbrace{y - \hat{s_1} - \hat{s_2}}_{\text{cancellation}} &= x + \underbrace{s_1 - \hat{s_1}}_{\frac{\sigma^2}{2}} + \underbrace{s_2 - \hat{s_2}}_{\frac{\sigma^2}{2}} + \underbrace{n_3}_{\text{RX noise}} \end{aligned}$$

As we can see the residual interference is only σ^2, rather than the $2\sigma^2$ that would have resulted from the SISO replication based design. Further, we can show by recursion that this residual is the same as the SISO design, i.e. there is no linear increase with the number of MIMO chains as the number of antennas increases. Implementation of this technique for wide-band OFDM systems is detailed in [8] based on [81].

Training in presence of another signal: While describing our algorithm above, we implicitly assumed that there is no other signal during the training phase, although in practice that might not be the case. This assumption however is not necessary. That is, even if there is a signal x as in the case of data, the algorithm would still work; the only change would be that the effective noise would now be $x+n_j$ instead of n_j at a given RX chain j and we use regularized least-squares estimation [41]. The downside is that the additional signal increases the interference during the training, thereby also increasing the number of samples or time required for convergence. Specifically, if interference to noise ratio after projecting the received signal on to the Tx signal space in least-squares is z, then it would take z times more samples to converge to the optimal point.

3.4 Robust MIMO Interference Cancellation

Interference cancellation needs to be robust to enable consistent full duplex operation in the face of frequent channel changes. To accomplish this, both analog and digital cancellation need to continuously tune their filter taps to maintain cancellation. The main bottleneck is tuning analog cancellation, since digital cancellation can be tuned on a per-packet basis in software as prior work has shown [102, 71, 41]. Tuning analog circuits requires measuring the residue in digital and then sending control signals to analog components, which is relatively slow. Minimizing the amount of time required to tune here is therefore critical, since during the time spent tuning packets likely cannot be received. We focus on this problem in this chapter and re-use the algorithms from prior work for tuning digital cancellation.

The prior SISO full duplex design demonstrated a technique to tune a single analog cancellation in around a millisecond. However, as before if we were to naively replicate the same algorithm for all the self-interference components, we would need M^2ms for a M antenna full duplex MIMO radio (e.g. 9ms for a 3 antenna full duplex). Such a high overhead is untenable for moderately mobile environments where the channel changes on average every 60ms (e.g. WiFi hotspots).

In this chapter we propose a novel technique that reduces tuning time by three orders of magnitude, i.e. an algorithm that tunes the circuit in 8μs. Note that this algorithm also applies to the SISO case, and therefore improves on the best known prior SISO design too. Our insight is to model the cancellation circuit as a filter whose response we are tuning to match as closely as possible the frequency response of the self-interference channel. Like prior work, we estimate the frequency response of the cancellation circuit for different combinations of filter tap values. The pre-calculated response is represented in a matrix A, whose each column is the frequency response of the analog cancellation circuit for a particular value of the filter tap at K different frequencies in the band of interest (e.g. K=128 for a 20MHz bandwidth in our current prototype for WiFi). Now assuming $H(f)$ is the frequency response of the self-talk channel in the frequency domain (i.e. the channel introduced by the antenna, circulator and any strong environmental reflections), the analog cancellation tuning problem reduces to:

$$\min_{x} ||H - Ax||^2$$

Where, H is the column consisting of $H(f)$ at different frequencies, and x, represents a binary indicator vector for selecting the corresponding filter tap values as in [41].

The efficacy of the tuning that results from the above problem depends on the accuracy in the measurement of $H(f)$. We can measure $H(f)$ using the preamble of the received interference signal $y(t)$ (e.g. the first two OFDM symbols of a transmitted WiFi packet which are known preamble symbols). The challenge is measuring the frequency response of the interference channel accurately. The accuracy is limited by the linearity of the transmit-receive chain, which is 30dB, By this we mean that any initial measurement can only have an accuracy of 30dB. The main reason is that the

transceiver produces non-linearities which act as noise to the channel estimation algorithm. In other words the received interference signal $y(t)$ has non-linearities that are only 30dB below the main linear signal component. Our key contribution in this chapter is a technique to accurately measure this channel quickly in the presence of non-linearities and tune analog cancellation.

Source of error and its magnitude: The transmitter produces non-linearities 30 dB lower than the transmitted signal. To show mathematically, say $x(t)$ is the baseband signal that is being transmitted after up-conversion and amplification, we can write

$$x_{tx}(t) = x(t) + a_3x(t)^3 + a_5x(t)^5 + a_7x(t)^7 + \ldots + w(t)$$

This transmitted signal $x_{tx}(t)$ is somewhat known to us because we know $x(t)$, however its non-linear components and the transmit noise $w(t)$ are unknown . This signal further undergoes the circulator and antenna channel $H(f)$ (which we wish to estimate), so when its received at the receiver the frequency domain representation of the received signal is given by:

$$Y(f) = H(f) * \mathcal{F}(x(t) + a_3x(t)^3 + \ldots) + \ transmit\ noise$$

Here, a_3 is around $10^{(-30/20)}$, i.e., its 30 dB lower. Further transmit noise distortion is 40 dB lower than the signal level of $x(t)$. The challenge is that our channel estimation algorithm is only going to use its knowledge of $x(t)$ to estimate the channel $H(f)$, and the other terms in the received interference signal limit the accuracy of the estimation to 30dB (the estimation noise is 30dB lower).

Accurate, Iterative method: The key idea is to run the estimation algorithm in an iterative fashion. Remember that the WiFi preamble has two OFDM symbols, each of length $4\mu s$. After the first OFDM symbol, we solve the above equation to produce an inaccurate estimate of the interference channel H_a and tune the cancellation circuit to achieve (at best) 30dB of cancellation (we cannot cancel more than our estimation accuracy). Now when we obtain the second preamble symbol, we know that the non-linearities and the transmit noise components that were producing the error are reduced by 30dB. We can exploit this fact by the following trick:

We transmit one OFDM symbol to estimate the inaccurate H_a, which can be written as a function of accurate H as, $H_a = H + e_1$. Note e_1 is 30 dB lower than H. We use the same algorithm as [41] to optimize the following,

$$\min_x ||H_a - Ax||^2$$

which produces the solution as $\hat{x}$, which gives us the values to use in the filter taps. We program the cancellation circuit using these values and achieve a 30 dB cancellation. Next, when we transmit second OFDM symbol and measure the channel response we get:

$$H_b = (H - A\hat{x}) + e_2$$

Notice that e_2 is 30 dB lower then $H - A\hat{x}$ and $H - A\hat{x}$ is 30 dB lower than H. So in essence e_2 is 60 dB lower than H. Define,

$$\tilde{H} = H_b + A\hat{x}$$
$$\tilde{H} = H + e_2$$

Thus, we can this new estimate $\tilde{H}$ with an error that is 60 dB lower. We use this estimate to re-tune the optimization algorithm and find a solution $\tilde{x}$ that tells us what values to use for the analog filter taps. This new solution provides 60 dB cancellation. Further, we only needed two OFDM symbols of $4\mu s$ each to get to this cancellation.

Extension to Cascaded Filter Structure: The above description is for a single cancellation circuit, but our MIMO design has a cascaded structure of multiple circuits. This leads to a combinatorial explosion in the parameter space that makes the problem NP hard to solve if we use the above approach. In this subsection we present a trick to approximate the overall combinatorial problem via two reduced complexity problems which can be solved using the same technique as the SISO one presented above.

We describe the algorithm in the context of tuning the cancellation circuits at receiver 1 for self and cross-talk in a 2 antenna MIMO radio. Lets say H_{11} is the self-talk channel response and H_{12} is the cross-talk channel response. The general tuning problem can be stated as:

$$\underset{x_1,x_2}{\text{minimize}} \quad t \tag{3.2}$$
$$\text{subject to} \quad norm(H_{11} - A_1x_1) \leqslant t \tag{3.3}$$
$$norm(H_{12} - (A_1x_1) \odot (A_2x_2)) \leqslant t \tag{3.4}$$

Where, $\odot$ represents the element wise multiplication of the column, and t represents the analog cancellation achieved, and A_1 is the response of the self-talk cancellation board with N taps in Fig.3.6 and A_2 is the response of the cascade cancellation board with C taps. The second constraint Eq. 3.4 renders the problem irreducible to a convex solvable form, and in fact the columnwise multiplication of the indicator variable vectors explodes the problem space and makes it a NP hard combinatorial problem.

We use a novel trick to approximate and help solve this problem practically. Since the first constraint in Eq. 3.3 is trying to find $A_1x_1 = H_{11}$, we can approximate A_1x_1 in the next constraint, Eq.3.4 with H_{11} which is known (since we measured H_{11}). This is of course an approximation, but it suffices to solve for x_2 using this substitution since we are after all trying to emulate the same cascaded channel response structure using our circuits as described in Sec. 3.3. Thus instead of a

cascade of unknown variables, the new problem to solve is

$$\underset{x_1,x_2}{\text{minimize}} \quad t \tag{3.5}$$

$$\text{subject to} \quad norm(H_{11} - A_1 x_1) \leqslant t \tag{3.6}$$

$$norm(H_{12} - H_{11} \odot (A_2 x_2)) \leqslant t \tag{3.7}$$

This new problem is no longer a combinatorial problem. This can be reduced to an integer program, which can be solved using randomized rounding in fraction of micro seconds practically [41]. Thus in effect the substitution trick reduces the non-tractable combinatorial problem into a tractable problem, whose solution can be found using the techniques described above. The tuning time for each MIMO chain is still two OFDM symbols, and the overall tuning time for the MIMO radio therefore scales linearly with M, the number of chains.

3.5 Evaluation

In this section, we experimentally demonstrate that our MIMO full duplex design almost completely cancels all self-talk and cross-talk interference to the noise floor with a low-complexity design. We also show that this translates to a doubling of throughput for the link performance.

We implement our design using four WARP v2 boards for building a 3×3 MIMO full duplex link. We design our own boards for analog cancellation and integrate them with the WARP boards. At each receive chain, we have analog circuits with 12 taps for the self-talk cancellation, 4 taps for the first cross talk and 2 taps for the farthest transceiver. In total we have 56 taps in the analog cancellation circuits for a 3 antenna full duplex MIMO radio, and total of 485 filter taps in digital cancellation. Since the WARP cannot generate 20dBm transmit power, we use an external off-the-shelf power amplifier [21].

We compare against the SISO replication based design primarily. This is the straightforward replication of the recently published SISO full duplex design as discussed at the start of Sec. 3.3. We compare against two variants of this design. One is a design that fully replicates the analog and digital cancellation implementations for all self-talk and cross-talk cancellations. As discussed before the complexity of this design is a factor of two higher for analog and $2.5\times$ higher for digital compared to our design. We call this design **SISO Replication**. However to make an apples to apples comparison with our design we also implement a SISO replication design with the same complexity as our design. The difference compared to our design is that, it neither use the cascaded structure nor the novel estimation algorithm, but simply replicates the SISO design with lower number of taps. We experiment with the tap distribution between self-talk and cross-talk to obtain the best overall cancellation. We call this compared approach **SISO Low Complexity Replication**.

The best recent work that we could compare for MIMO full duplex is **MIDUs [34]**. However

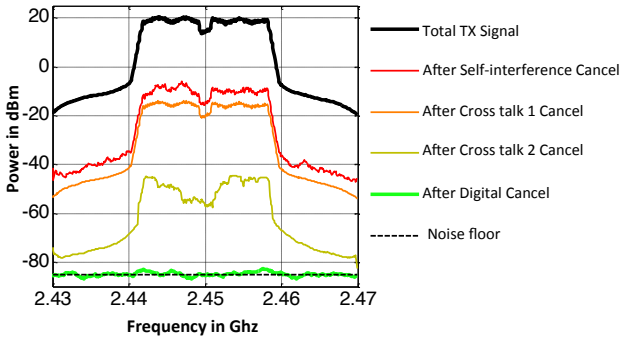

Figure 3.11: Spectrum plot after cancellation of various self-talk and cross-talk components for RX1 of a 3×3 full duplex system using our design.

this design only works for small bandwidths (i.e. 500KHz). Further, it relies on obtaining 50dB of cancellation using antenna cancellation (which itself requires more antennas per MIMO chain and is problematic), and then complements it with another 30dB of digital cancellation. However when we go to normal bandwidths of 20MHz found in WiFi signals, then the antenna cancellation reduces to 40dB at best, and hence we are limited to a total of 70dB of cancellation. This is significantly worse than SISO replication, and hence we omit comparisons against MIDU. SISO replication is in fact the best comparable technique that we can compare our design to.

Unless stated otherwise, all experiments are conducted by placing the two full duplex nodes at various locations in our department building. At each location, we repeat the experiment ten times and calculate the average performance.

3.5.1 Can we cancel all the interference for 3 antenna full duplex MIMO ?

The first claim made in this chapter is capability of canceling all of the interference for the 3×3 MIMO. To prove this, we experimentally test if we can fully cancel a WiFi 802.11n 20MHz signal upto a max transmit power of 20dBm for a 3×3 MIMO. To demonstrate we first pick one instance of this experiment, and show the spectrum plot of the received self-interference after various stages of cancellation in Fig. 3.11. Remember, that in analog we first cancel the chain's own self-talk leaking through the circulator, and then the cross-talk from the other two antennas. Finally, we apply our digital cancellation step to clean up the residual. We see that overall in analog we achieve 68-70dB of self-interference cancellation after all three stages. This satisfies the requirements outlined in Sec. 3.2.

We now place the node at several different locations in the testbed. At each location we vary the overall TX power from 16dBm to 20dBm and plot the average cancellation for each power across all locations. At each location and for each power, we conduct 40 runs. The goal is to show that we can consistently cancel to the noise floor for a variety of transmit powers up to and including

Figure 3.12: Increase in noise floor vs TX power on the left side and Cancellation vs TX power on the right side. For different MIMO cancellation designs, we present the performance of a full duplex 3 antenna full duplex MIMO system.

the max average TX power of 20dBm. In each instance of the above experiment, we also measure the increase in noise floor due to any residual self-interference that is not canceled. Note that the increase in noise floor represents the SNR loss the received signal will experience when the node is used in full duplex mode. Fig. 3.12 plots the average cancellation and the increase in noise floor as a function of TX power.

Fig. 3.12 shows that our 3-antenna MIMO full duplex design cancels the entire self interference almost to the noise floor. In case of max average transmit power of 20dBm [41], the noise floor is increased by 1.6dB over each receive chain's noise floor. The SISO replication design increases the noise floor by 4dB per receive chain, while the SISO low complexity replication approach increases the noise floor by 25dB. Finally, the performance of our design and the SISO replication design scales with increasing TX power, while the other replication based design is limited due to its inability to cancel the increasing transmit noise and non-linearities due to the reduced number of taps available to it.

3.5.2 Scaling with the number of MIMO antennas

A question with MIMO is how does full duplex performance scale with increasing number of transmit chains. The ideal case would be to maintain the same level of cancellation at each RX chain as the number of transmit antennas increase, starting from one antenna. In other words, even with increasing number of transmit antennas and cross-talk components that need to be canceled, we retain the same performance as if there was a single transmit antenna and a single self-interference signal to deal with. Fig. 3.13 plots the increase in the noise floor at one receive chain as we go from one transmit chain to three transmit chains for a MIMO radio for both our design as well as the SISO replication technique. The overall TX power is fixed to be 20dBm (additional 10 dB of PAPR for WiFi [41], i.e., total 30 dBm) to adhere to ISM band EIRP requirements. Hence if we use a single transmit chain, then all the 20dBm is used for a single antenna. If we use two chains, then each antenna produces a 17dBm signal and so on.

As we can see from the figure, our design maintains a near-constant performance even as we go

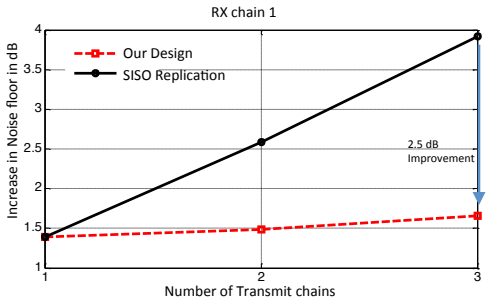

Figure 3.13: Increase in noise floor at a RX chain as the number of MIMO chains and consequently the number of cross-talk components increase from 1 to 3. With our design we observe a 2.5 dB improvement for 3×3 MIMO per RX chain compared to the SISO replication design.

from one to three transmit chains. In other words, the performance is roughly the same regardless of the number of cross-talk components (We do wish to note that we could not go beyond three transmit chains due to hardware limitations, verifying the above claim for higher number of transmit chains is future work). On the other hand, the SISO replication design shows the noise floor increasing linearly with increasing number of transmit chains, a fact we provided theoretical intuition for in Sec. 3.3.2. Thus this design will look worse as we scale to higher MIMO configurations. We omit the SISO low complexity replication approach because its results are significantly worse.

3.5.3 Dynamic Adaptation

An important metric for analog cancellation is how quickly can it be tuned, and how often do we need to tune? The best know prior technique [41] required around 1 millisecond to tune a single SISO analog cancellation circuit. So for a 3×3 MIMO, applying the same algorithm will take at least 9ms for the SISO replication based design. In this section we show the efficacy of our new tuning algorithm which cuts the tuning time to $8\mu s$ per receive chain. Fig. 3.14 shows the tuning time as a function of the amount of analog cancellation. To achieve the 70dB analog cancellation, our algorithm takes $8\mu s$ per chain, for a total of $24\mu s$ for the full radio. The prior work as we can see take a millisecond per chain. The interesting takeaway is that both schemes achieve 40dB of analog cancellation fairly quickly (with one preamble symbol, i.e. $4\mu s$), but our scheme covers the final 30dB in one more step of $4\mu s$, while the prior scheme takes an exponential number of symbols to achieve that. The reason for this improvement is precisely our ability to get a precise measurement of the self-interference channel using the trick described in Sec. 3.4.

A second question is how often one needs to tune? This depends on the environment and the amount of analog cancellation that needs to be maintained. In this chapter, we tune for challenging indoor environments which have strong multi-path (this is the main source of analog cancellation degradation). We define a near-field coherence time which depends on the amount of analog cancellation and is essentially the time for which that analog cancellation can be maintained on average

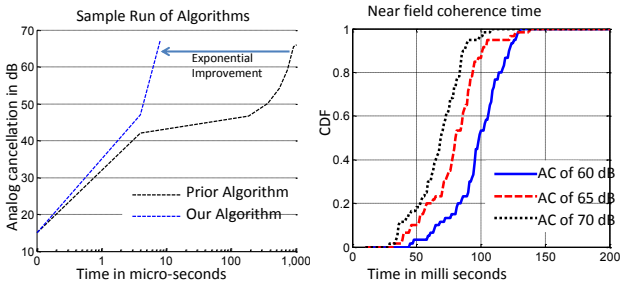

Figure 3.14: Tuning time for analog cancellation. The first figure shows the three orders of magnitude improvement in tuning time with our algorithm compared to the best known prior approach. The second figure shows how often this tuning algorithm needs to be run for an indoor environment.

before the circuits need to be retuned. Fig. 3.14 plots the near-field coherence time for three different analog cancellation targets. As we can see, to maintain an analog cancellation of 70dB, we need to retune roughly every 60ms. Our tuning overhead is $24\mu s$, which is negligible.

3.5.4 Does Full Duplex Double Throughput?

A final question is whether all this cancellation performance translates to a the desired doubling of overall throughput. We show experimentally the throughput gains of our 3×3 MIMO full duplex design compared to the SISO replication based design. Two full duplex 3-antenna MIMO nodes are placed at different locations and we send 1000 packets in full duplex mode between them, and then send 1000 packets for each direction of the half duplex mode. We repeat this experiment for each bitrate that is available in WiFi. We pick the bitrate which maximizes the overall throughput for all of the compared full duplex designs and half duplex respectively. We repeat this experiment for 50 different locations. We found the received power of the links varied uniformly between -45 *to* -80dBm, across locations as found in typical indoor deployments. To put these numbers in perspective, this implies that the SNR of the links in half duplex mode ranges from 5dB to 40dB. We plot the throughput for half duplex and full duplex designs in Fig. 3.15. Note that all of these throughput numbers account for the overhead introduced by the periodic analog cancellation tuning. As we can see, our full duplex system achieves a median throughput gain of $1.95\times$ over the half duplex mode, but the SISO replication based design with full complexity only achieves a $1.36\times$ gain. The reason is the higher increase in noise floor from the SISO replication based design. For example, if the link SNR in half duplex mode is 10dB, a 4dB increase in noise floor will result in worse overall throughput for full duplex compared to running the link in half duplex mode. Our ability to keep the noise floor constant results in a performance close to the theoretical optimum.

The SISO replication based design with lower complexity is quite poor, in fact in 70% of the scenarios, the throughput was zero. This is because it increases the noise floor by at least 25dB which acts as noise and if the SNR is below 30dB no signal is decoded (WiFi requires a minimum of $4-5$dB SNR to decode the lowest rate packet). As the half-duplex link SNR increases, the

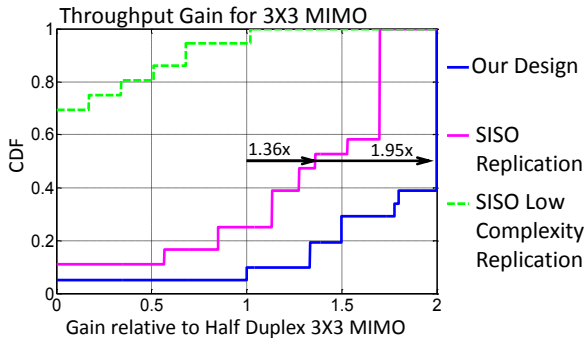

Figure 3.15: CDF of throughput gain relative to half duplex 3×3 WiFi MIMO. Our 3×3 MIMO system provides a median gain of 95% relative to half duplex, whereas the SISO replication design only provides a $1.36\times$ relative gain.

performance improves but is still not sufficient to beat the system throughput achieved by half duplex. The reason is that even if the link half-duplex SNR is 35dB, it implies that we only have two 10dB links for full duplex. The throughput achieved with a single 35dB half duplex link is still higher than two 10dB links. Consequently the only region where we could find improvements for full duplex over half duplex with this design was when the link SNR was greater than 38dB.

3.6 Conclusion

This chapter brings towards completion a line of work on PHY layer of full duplex radios, and shows that practical full duplex is achievable for the most common wireless protocols and for MIMO while using commodity radios. The cancellation techniques developed in this chapter are fundamental and apply to a wide variety of problems [66, 36, 58] where self-interference cancellation is needed. While this work wraps up work on board level realizations of full duplex, much work remains in realizing these designs in a chip. Tackling these problems is future work.

Chapter 4

Application: FastForward Full duplex relay

4.1 Introduction

We have all often experienced perplexingly poor wireless performance. For example, it's not uncommon to find that one's connection is flaky and offers very low throughput even when one is the only user of the WiFi AP in a home. Similarly, for LTE networks, even at nights when the network is lightly loaded, performance can be poor indoors or in urban concrete jungles, with raw link speeds varying between a few hundred Kbps to a couple of Mbps. This is despite continuous evolution of wireless standards over the last few years to provide very high link bitrates. For example, the 802.11ac WiFi standard promises bitrates of up to 1.3Gbps, while LTE downlink speeds are expected to be up to 300Mbps [17, 14]. These gains are coming from two factors: use of higher modulation (up to 256QAM for both LTE and WiFi) and higher MIMO spatial multiplexing (up to 4 parallel streams for both LTE and WiFi). Both these features should work well when there is little to no contention/interference and a single or a few users are connected to the WiFi AP or the LTE basestation. Yet often users don't realize these benefits in practice, experiencing raw speeds that are one to two orders of magnitude less than the advertised speeds.

There are two fundamental reasons for the poor performance

described above: propagation loss and MIMO rank degradation. Propagation loss is a natural and expected cause of the drop in link rates. Fig. 4.1 shows a typical 2000 sq. ft. home with a WiFi AP at one corner of the house in the living room. We model propagation and other effects using commercial grade wireless ray propagation modeling software [16] that is used for planning wireless deployments. As we can see, except for the immediate area around the AP, most of the coverage area in the middle of the home experiences SNRs between 10-15dB (as seen in Fig. 4.1), and at the edge the performance is worse, with SNRs between 0-6dB. This cuts down the highest modulation that can be used to QAM/16-QAM from 256-QAM, a 4× reduction in bitrate. An analogous argument can be made for LTE networks where the coverage area is larger, and signals often have to propagate through large buildings in urban areas which further cause signal loss due to shadowing effects.

The second fundamental reason is MIMO rank degradation as seen in Fig. 4.2. To send multiple data streams via MIMO spatial multiplexing, the channel between the AP and the client needs to have several independent strong propagation paths available (in other words, the MIMO channel matrix needs to be full rank and have strong eigenvalues [32]). But in most indoor and urban scenarios, often we find that only a single strong path exists between the AP and the client, and the rest are weak or non-existent. This happens because of the geometry of homes, offices and hotels which typically have a single or few corridors with rooms off the corridors. The corridor acts like an RF pinhole [32, 76] since it is typically the only strong path available between the AP and the client, and focuses all the signals to go through a single path which makes all of the paths correlated at the destination. The consequence is that the MIMO channel rank is reduced, and the AP cannot send multiple independent streams, reducing the bitrate significantly. LTE signals behave analogously, in that the only path indoors for the signal is through windows or doors (walls tend to block signals almost completely), and the doors/windows acts as RF pinholes. Combined with the propagation loss described above, this results in nearly a 6-10x reduction in bitrate in the middle and edge of the coverage from the AP.

Our goal in this chapter is to design a general, practical and easily deployable system that provides high-throughput uniform wireless coverage. By general, we mean the fundamental technique should be applicable to any OFDM based standard. By practical and easily deployable, we mean that the system should require minimal to no changes to the existing infrastructure of APs , clients and/or standards.

We design and implement a novel system called **F**ast**F**orward (FF) that achieves the above goals. FF's core operation is simple to describe. It is a single device that operates independently listening to the signal from the source, digitizing it to IQ samples, processing it by passing the IQ stream through a filter (in both digital and RF domains), and up-converting and amplifying the processed IQ stream to RF signals that are then transmitted to the destination on the same frequency. The filtering and amplification are done in such a way that the SNR of the signal at the destination is significantly increased and the number of independent MIMO paths at the destination is also increased, enabling

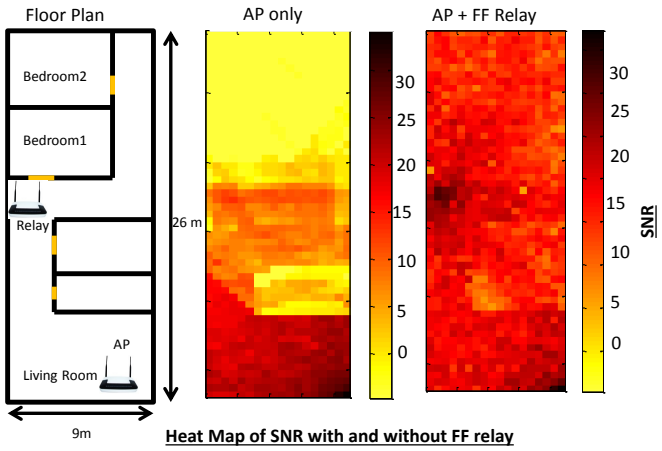

Figure 4.1: Heatmap of SNR with AP alone and with AP and FF relay. A majority of the home has poor SNR due to propagation loss in the AP only scenario.

a significantly higher bitrate. Thus it acts as a controlled strong multi-path creator of the signals that is completely transparent to the AP and the client, they do not even realize that an FF device exists. A glimpse of its performance is shown in the heatmaps (Figs. 4.1,4.2).

The key insight behind FF is a novel technique that we invent called **construct-and-forward full-duplex relaying**. The basic idea is best described in terms of a single SISO transmission from the AP to the client. With a simple full-duplex amplify-and-forward relay that has been discussed in the literature [83], the client would receive two signals: one directly from the AP and the other amplified version from the relay. A naive implementation of the relay will result in both these signals acting as destructive interference to each other, and the relay potentially amplifying noise. FF's innovation is to control the properties of the relayed multi-path signal to in fact turn such potential interference into a constructive SNR gain. The design relies on the fact that if an OFDM receiver receives multiple reflected copies of a signal, then as long as the reflections are within the OFDM cyclic prefix (CP) interval (around 400ns for WiFi and 4.69μs for LTE), they do not cause inter-symbol interference (ISI) to each other. If we can ensure that the processing delay through the FF relay is minimized so that the relayed signal does not fall outside the OFDM CP interval at the receiver, we can achieve no inter symbol interference. FF's low latency cancellation technique achieve this purpose. Thus, FF's relay acts as an amplified multi-path component at the receiver.

While limiting the processing delay ensures that inter-symbol interference is avoided, it still does not provide a constructive SNR gain. The second aspect of construct-and-forward relaying is to intelligently process the received signal at the relay before transmission such that the relayed signal adds up constructively with the other signals that the destination is directly receiving from the source to significantly enhance the effective SNR. The basic idea is that the relay first collects the channel state information about three links: source-relay, relay-destination and source-destination.

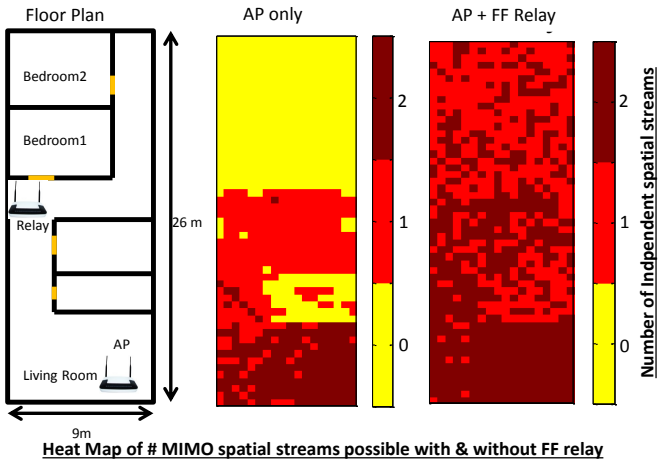

Figure 4.2: Heatmap of number of MIMO spatial streams possible with AP alone and with AP and FF relay. A majority of the home has poor MIMO channel rank due to pinhole effects and poor link propagation through walls.

Now, when it receives the transmission from the source, it passes the signal through a filter such that cumulative effect for the received signal at the destination (which has now gone through the channel from the AP to the FF relay, the filter at the relay and the channel from the relay to the client) is such that it adds coherently (in almost complete alignment) at the destination with the direct signal received by the destination from the source. Fig. 4.5 shows the effect visually, the relay rotates the incoming signal such that it aligns up with the vector representing the channel between the source and the destination. The constructive addition significantly increases the SNR at the destination (client), enabling a higher bitrate to be used by the source (AP). A similar effect happens when the FF relay is combating the pinhole effect, it computes a filter that increases the number of spatial streams and the SNR at the destination (client), enabling the source (AP) to use a higher level of spatial multiplexing and therefore higher bitrates. Note that the relay can be used to improve the link from the client to the AP as well.

The challenge in realizing such construct-and-forward relaying while ensuring that processing delays is much smaller than the OFDM CP is the full duplex nature of the relay. The FF relay is transmitting and receiving signals at the same time on the same frequency. Further, the transmitted signal is essentially a slightly delayed and amplified version of the received signal. To receive the signal from the AP, the FF relay has to cancel the transmitted signal. The amount of cancellation puts a limit on the amount of amplification that we can apply at the relay, since if we amplify more than the cancellation, residual signal is left over and is recycled for transmission, creating an unstable positive feedback loop. Maximizing the amount of cancellation is therefore crucial to maximizing amplification. However, unlike prior work on full duplex, the cancellation has to be performed within a time budget as small as possible (e.g. within 100ns for WiFi since the CP is

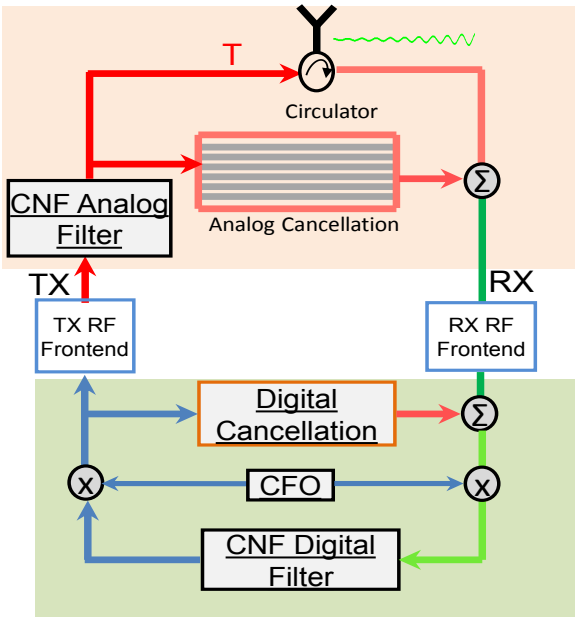

Figure 4.3: Overall Block Diagram of a FF relay. There are two key pieces: construct-and-forward (CNF) analog and digital filters, and self-interference cancellation.

only 400ns long) to ensure that the relayed signal can take advantage of FF's constructive relaying capability. A second key contribution of this chapter is a novel cancellation technique for relays that achieves nearly 110dB of cancellation while operating within a processing time budget of 100ns.

We design and implement FF on the WARP radio platform [28] and by designing our own self-interference cancellation RF boards. We evaluate FF in an indoor testbed and show that FF provides a 3× median increase in throughput and nearly 4× at the edge of the coverage area. The gains come from different aspects for different clients. For clients with decent SNR already, the gains come from MIMO rank expansion. For clients at the edge of the coverage area where the SNR is already quite poor, the gains come from the SNR gain constructive relaying provides. We also compare against the half duplex packet-level relay (e.g. the Apple Airport Express) and show than FF provides at least 2× better throughput and coverage.

4.2 Related Work

A natural question is whether there are other approaches that can be used to solve the problem of coverage and capacity that FF aims to? There has been of course a large body of work in recent years that have proposed several PHY and MAC layer enhancements to increase network capacity and robustness, FF however is operating on signals directly and is therefore orthogonal to those approaches.

However there is one approach that could help and is immediately deployable: a half-duplex mesh

router like the Apple Airport Express. These devices help extend WiFi coverage by connecting to the AP as a client, and then turning around and transmitting to the actual client in the next slot (hence the name half duplex router). Theoretical literature on relaying refers to such techniques as **decode-and-forward** relaying. However, as we show in Sec. 4.5 these devices do not provide capacity gains except in the edge of the coverage area. This is because they essentially require close to twice the number of time slots for transmitting the same amount of traffic. Further for clients with decent SNRs to the AP, the half-duplex mesh router is a bad option, it is better to have a single-hop medium-SNR link rather than using two hops over high-SNR links.

There are several products in the market that are called repeaters. These devices are simple **amplify-and-forward** relays. They receive a signal, and then immediately amplify it and transmit it. Such devices are available for both WiFi and LTE networks. However these devices cannot amplify too much, they are severely limited by the amount of isolation between the signals they are receiving and relaying as we show in Sec. 4.3.5. Second, since they are blindly amplifying signals, they amplify noise and often hurt performance as we show in Sec. 4.5.5. FF also belongs to the class of **amplify-and-forward** relays, however this chapter makes three novel contributions:

- FF is selective and smart about relaying, it exploits the knowledge of channel state information to intelligently filter and amplify signals such that they appear as a constructive multipath component at the destination, rather than increase noise and/or add up destructively like a standard repeater would.
- FF designs a novel low-latency self-interference cancellation technique which ensures that relayed signals fall within the CP for OFDM signals and do not cause inter-symbol interference. The technique is applicable to standard repeaters too and they can benefit from being able to use a higher amplification factor due to the increased amount of cancellation.
- This chapter also provides a full design, implementation and evaluation of full-duplex relays, to the best of our knowledge we are not aware of prior work that provides an experimental characterization of how well other kinds of relays work in practice.

Finally, there is a large body of theoretical work on relays in the information theory literature [93, 53, 115]. Starting from early work by Shannon, there have been several proposals on relaying [49, 85, 47, 103]. Apart from the amplify-and-forward and decode-and-forward relaying techniques; a third well known class of techniques is **compress-and-forward**: this is an intermediate version between the above two relays. Here the relay may not decode the entire packet, but only the symbols and re-encodes them in a more efficient way [123, 64, 122, 79]. The destination has to combine the relayed information with the direct transmission from the source to recover the original packet. This method is typically quite complex to implement since it requires changes at the client with techniques such as soft interference cancellation and combining, as well as sophisticated processing at the relay.

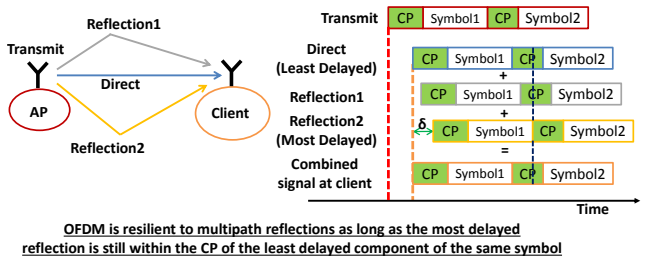

Figure 4.4: OFDM is resilient to multipath reflections as long as the extra delay experienced by the slowest reflection compared to the quickest arriving signal at the destination is less than the cyclic prefix (CP).

4.3 Design

FF is a layer 1 full-duplex relay, i.e. it receives signals from the source, processes them both in the analog and digital domains, and then converts them back to RF signals and transmits them on the same channel they were received on. Fig. 4.3 shows the high-level block diagram of an FF single-antenna relay. Note that an FF relay can have multiple antennas and can relay MIMO signals, however we use the single-antenna SISO FF relay for describing the key ideas in a concise manner. However the techniques and algorithms naturally translate to a MIMO relay implementation.

As we can see there are three main components in the design: cancellation, constructive filtering (CNF) and amplification. The insight underpinning these components is exploiting the structure of OFDM such that relaying can produce a constructive SNR gain at the receiver. We start by describing first the basics of OFDM.

4.3.1 OFDM Background

OFDM was introduced to combat the negative effects of multipath and inter-symbol interference. The basic idea is widely known and described in textbooks [59], but we include it here because it helps explain some of FF's algorithmic design choices later.

The basic idea of OFDM is to divide the available bandwidth B into N smaller subcarriers (e.g. 802.11ac with 80MHz bandwidth is divided into 512 subcarriers whereas LTE divides into subcarriers of width 15KHz). Each subcarrier can be conceptually treated as an independent orthogonal channel carrying independent symbols. Hence the symbol time is N/B, i.e. the symbol is N times longer, as compared to a typical communication system transmitting symbol at $1/B$, for bandwidth B. Further to each symbol, a guard period known as the cyclic prefix (CP) (typically 25% of the symbol time) is added. Hence as long as the extra delay of a multipath reflection of an OFDM symbol w.r.t. the first arriving version at the destination is less than the CP length, no inter-symbol interference is caused as seen in Fig.4.4. The length of the cyclic prefix is 400ns in WiFi and 4.69μs in LTE. Hence in WiFi there is tolerance for a distance spread of 400 feet whereas for LTE its almost 5000 feet, which is expected since WiFi is designed for covering homes whereas LTE is designed for covering

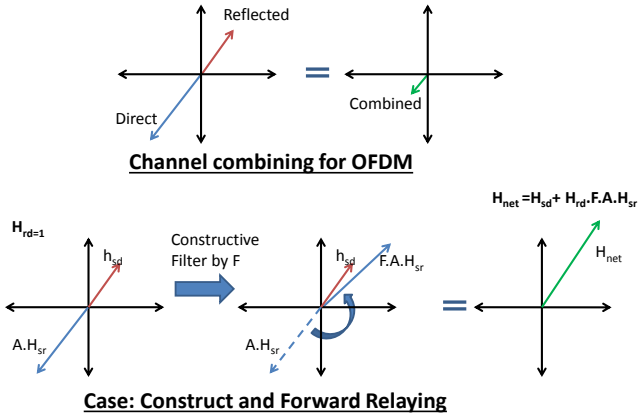

Figure 4.5: FF's construct-and-forward relaying rotates the relayed signal such that it aligns with the direct signal from the source to the destination and provides a constructive SNR gain. The top figure shows what happens with normal OFDM where instead of the relay there is a normal reflection of the same delay. The channel gains add up destructively and reduce SNR at the destination.

larger outdoor areas.

Given the above fact, how does the effective channel look at the receiver? In other words how do the multipath reflections add up if they are not causing ISI with each other? To visualize this, consider Fig. 4.5. We are plotting the channel gains for a single OFDM subcarrier (i.e. the attenuation and phase shift applied by the direct path channel to any signal on that subcarrier). Now suppose there is another multipath reflection with a slightly longer path and higher attenuation. The channel gain for this second path shows up as a second vector that is rotated w.r.t the first channel gain. Assuming the extra delay is within the CP, the overall channel perceived by the receiver is the sum of these two channel gains. The effective SNR at the client therefore depends on the relative orientation and gains of the direct and reflected channel paths, if they are aligned with each other in the same direction SNR increases, if they are in opposite directions SNR decreases.

4.3.2 Construct-and-Forward Relaying

FF's construct-and-forward relaying builds on top of OFDM. Our basic insight is to make FF look like another strong multipath reflector, albeit with the ability to amplify and modify the signals. Since FF operates at the signal level, at the receiver the signal from the relay looks like yet another multipath component, albeit a strong one. As long as the extra delay of this component is within the CP, the receiver will not perceive any inter-symbol interference. The constraint then is that the overall delay of the signal going through the FF relay has to be as low as possible, and definitely well within the CP interval. Since we still have to account for normal propagation delay from the source to the relay and then from the relay to the destination, ideally we want to completely minimize the processing delay in the FF relay.

As we see in Fig. 4.6, by minimizing the relative delay below cyclic prefix between direct and

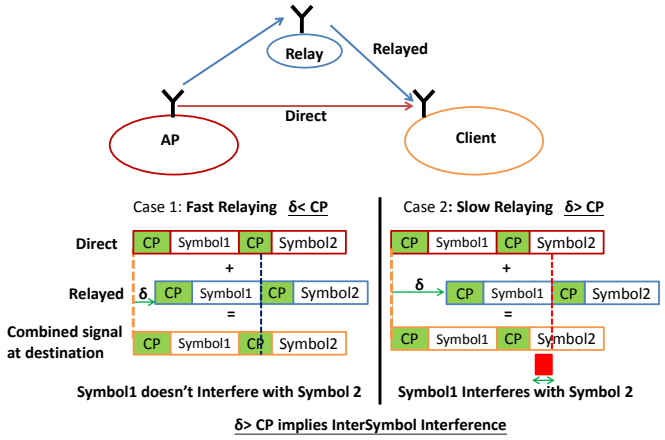

Figure 4.6: Low latency processing at the FF relay is critical. If the delay of processing in the FF relay is greater than the OFDM CP, then the relayed signal will cause inter-symbol interference at the destination [59].

reflected (or relayed) path we can avoid inter-symbol interference. However, depending on the relative phase of the channel gains from the relay to the destination (relayed) and from the source to the destination (direct), we might hurt overall SNR at the receiver as shown in of Fig. 4.5.a. So might FF be hurting the SNR by relaying ?

FF's key invention is a novel technique that leverages its relaying capability in a way to actually significantly enhance the SNR at the client. Remember that the relay has the opportunity to modify the signal before it amplifies and sends it to the destination. FF's novel idea is to apply a filter before amplifying and relaying the signal such that it adds up *constructively* at the destination to maximize the SNR gain. Mathematically, let us say the channel from the source to the destination is h_{sd}, and from the source to the relay is h_{sr} and from the relay to the destination is h_{rd}, for a particular subcarrier. Further the noise at the destination is n_d, and at the relay is n_r. The relay would amplify the signal by a factor A and then pass the signal through a constructive filter whose response is F at that subcarrier. The SNR at the destination for that subcarrier, is given by

$$SNR_d = \left| \frac{h_{sd} + h_{rd} F A h_{sr}}{N_o} \right|^2 \tag{4.1}$$

where $N_o = n_d + h_{rd} F A n_r$. The second term in N_o is small, since the amplification (A) is controlled as described in Sec. 4.3.5, which makes sure that noise is not amplified at the destination. For now we will assume the controlled amplification is represented by, $A < A_{max}$ and we will ignore N_o in optimization of Eq. 4.1. Visually this is demonstrated in Fig. 4.5.

Note that the constructive filter can introduce additional processing delay, however as before the overall delay still has to be well within the CP so that we can take advantage of OFDM. Further constructive relaying assumes that the relay knows all three channels. The channels from the source

to the relay and from the relay to the destination are easy to measure by the relay itself. However the channel from the source to the destination cannot be measured by the relay and has to be explicitly fed to it. We discuss in Sec. 4.4.2 how this can be done in both WiFi and LTE using existing mechanisms in the standards.

The above discussion has focused on the SISO case. However the same arguments hold for the MIMO case. In effect the relay adds a separate independent strong MIMO path which increases the rank of the MIMO matrix. For constructive relaying, instead of optimizing the above equation, the relay would perform the following optimization

$$\begin{aligned} \max_{F,A} \quad & \det(H_{sd} + H_{rd}FAH_{sr}) \\ \text{subject to} \quad & A < A_{max} \end{aligned} \tag{4.2}$$

where H_{sd} is an $N \times M$ channel matrix where the source and destination have M and N antennas respectively, H_{sr} is a $K \times M$ channel matrix to the relay (the relay has K antennas) and H_{rd} is an $N \times K$ matrix, A is again the scalar amplification factor (power) and F is the constructive filter which is a $K \times K$ rotation matrix in this case. Intuitively, the path through the relay acts as a strong independent MIMO path and adds rank to the overall matrix. Since a K antenna relay has only K dimensions, it can increase the MIMO rank at the destination at most by K. The filter again in this case acts as a mechanism to maximize the SNR. The optimization problem described in Eqn. 4.2 is non-convex and is solved using non-linear optimization technique. Note that it can be solved for $F.A$ as a single variable, and only needs to be solved whenever any of the three channels are updated, and not for every packet. The solution to this problem is referred to as $H_c(f_i)$ in the later sections, overall filter response is referred as H_c.

The takeaway from the above algorithm is that FF needs to implement two key blocks: **amplification** and **constructive filtering**. Note that both these blocks need to be as low latency as possible, ideally within a 100ns budget given that the WiFi CP is 400ns. If we can design it with that delay then the techniques will work for LTE too since it has a longer CP. We turn to the design and implementation of these blocks next.

4.3.3 FF: Low-Latency Amplification

As we saw in the previous section, FF enables constructive relaying by applying an amplification A and a filter F to the received signal at the relay. As expected the relay cannot receive a signal if it is also transmitting an amplified signal at the same time on the same frequency. Hence to build a relay we need to isolate the received signal from the transmitted signal, i.e. remove the transmitted signal from the received signal. Further, the amount of isolation directly dictates how much amplification the relay can apply on the received signal, which in turn dictates how much the relay expands the range and capacity of the network.

To see why, consider what happens if we amplify beyond the achievable isolation as seen in

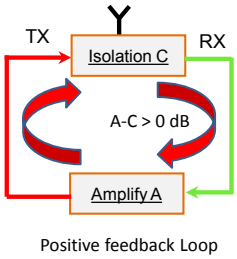

Figure 4.7: Amplification A is limited by the amount of isolation C. Amplifying more than the isolation implies there is still some residual left over after isolation by C dB, which is then again amplified and relayed in the next time instant and so on. This creates an unstable positive feedback loop.

Fig. 4.7. In effect this means that some of the signal that is being transmitted is still left over in the received signal after isolation since amplification is greater than isolation. But remember that the transmitted signal is simply a delayed version of the received signal. So in the next instant the transmitted signal would contain a copy of the transmitted signal that was left over in the previous instant. This iteratively accumulates and creates a positive feedback loop where ultimately the relayed signal simply consists of leftover copies of the same signal from previous time instants. The positive feedback loop is unstable and ultimately leads to poor performance. On the other hand if the amplification is lower than the isolation, then all of the transmitted signal is removed from the received signal, and the relay operation proceeds smoothly.

Our goal therefore is to maximize the isolation from the TX to the RX. We turn to recent work on self-interference cancellation for full duplex radios [41, 40] to provide the isolation between RX and TX signals. These techniques enable a radio to almost completely cancel the transmitted signal and enable clean reception of the received signal. However there is a catch which prevents us from being able to directly apply the cancellation techniques, the self-interference cancellation has to be performed with as little latency as possible (e.g. much smaller than 400ns for WiFi signals). Self-interference cancellation in the prior work has two components, an analog and a digital cancellation stage. Analog cancellation has negligible delay (around 10ns). However the digital cancellation stage (including the ADC and DAC delays) has a delay of nearly 400ns which would put us out of range for the relay requirements for WiFi. The ADCs and the DACs contribute around 50ns of delay, hence the digital cancellation stage adds nearly 350ns of delay.

We invent a novel self-interference cancellation technique that performs the cancellation with a near-zero delay (excluding than the latency of implementation, which is a few ns or lesser). In prior work on cancellation, the delay is primarily due to the fact that digital cancellation is non-causal [41]. In other words, the digital cancellation filters like to peek ahead into the future of the signal and use that information to cancel the signal at the present. In this relay, we could do this by buffering the received signal, so when we are canceling the self-interference signal at any instant, we know

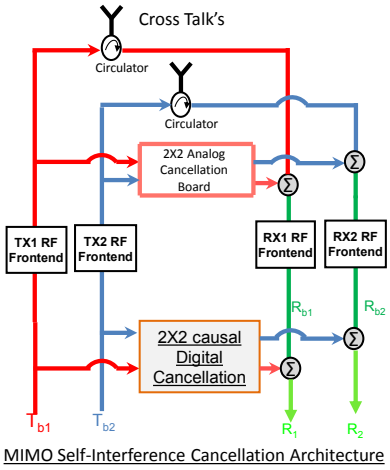

Figure 4.8: Self-interference cancellation architecture for a 2×2 MIMO FF relay.

the future of the transmitted signal is going to be. However buffering means delay, for example buffering even 5 digital IQ samples at a 100Msps sampling rate means a delay of 50ns. Hence in FF, we invent a digital cancellation technique that is causal, i.e. it only uses information about what has been already transmitted to cancel the self-interference and does not do any buffering of the received signal before transmission. So received samples are passed in a streaming fashion to the transmit side without any delay. However causal cancellation results in digital cancellation filters which are slightly longer, they need to use more taps to recreate the self-interference for cancellation. However these taps do not add delay, they are for signal samples that have already been transmitted.

Fig. 4.8 shows the cancellation architecture for a 2×2 MIMO FF relay. Analog cancellation is implemented as discussed in prior work [41, 40] using a tunable FIR analog filter. Digital cancellation is slightly different, it uses a FIR filter like before but there is no buffering and delay, it is a causal filter as shown in Fig. 4.9.a. The samples that are used for cancellation are only the samples that are currently being or have already been transmitted, indicating causality.

The coefficients for both the analog and digital cancellation filter are dynamically tuned to maximize cancellation. However, dynamically tuning cancellation in a full duplex relay is more complex than standard full duplex. The reason is that the signal that is being transmitted is a slightly delayed version of the signal being received. To see how this impacts the tuning algorithm, we can look at what happens during analog cancellation. The cancellation problem is given by:

$$\begin{aligned} y(t) &= x_R(t) + h(t) * x_T(t) - \widehat{h}(t) * x_T(t) \\ &= x_T(t+\tau) + h(t) * x_T(t) - \widehat{h}(t) * x_T(t) \end{aligned}$$

where $x_R(t)$ is the signal relay is receiving from the source, $x_T(t)$ is the signal the relay is transmitting to the destination, $h(t)$ is the time domain transformation applied by the channel before the

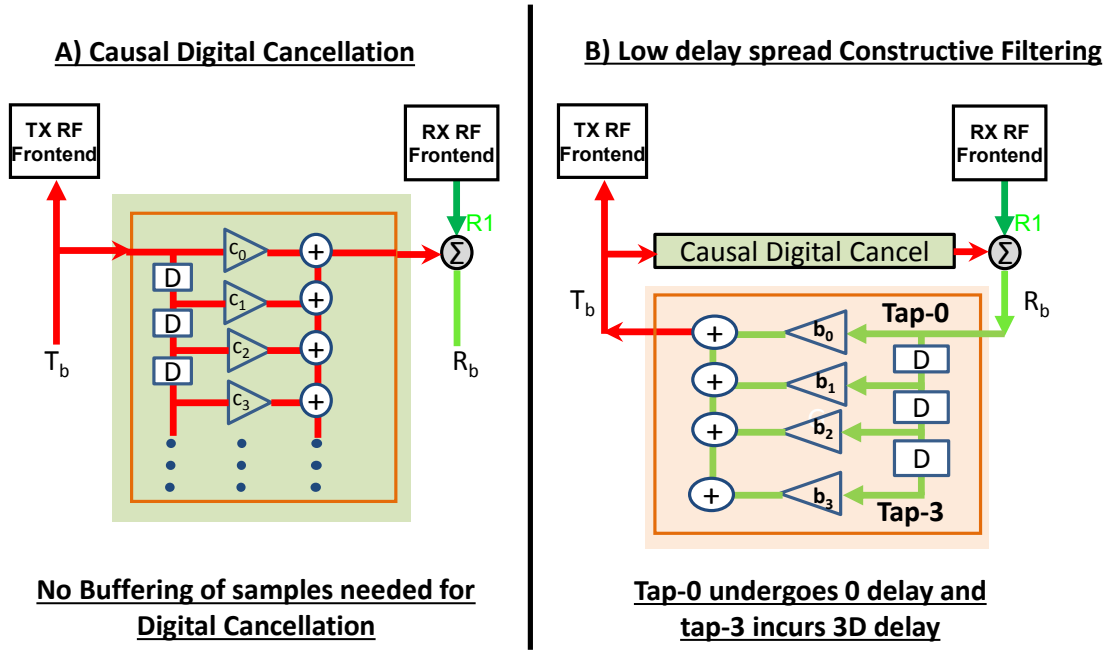

Figure 4.9: a) Digital cancellation in FF is causal, i.e. the cancellation is performed only using the current and past transmitted samples. No buffering of received samples is performed, which minimizes processing delay through the relay. b) The larger the number of tap delay, higher is the probability that it would cause inter-symbol interference at the destination.

transmitted signal from the relay causes self-interference to the received signal, $\widehat{h}(t)$ represents the filter that is being used by the analog cancellation block to approximate H and implement cancellation, and $y(t)$ is the combined signal that is being received by the relay. Clearly cancellation is maximized when $h(t) = \widehat{h}(t)$. In the second part of the above equation we substitute $x_R(t)$ with $x_T(t+\tau)$ because the relayed signal is a future version of the received signal, where the delay is represented by τ.

Prior work on analog cancellation solve the above estimation problem in the frequency domain. So the above problem can be rewritten in the frequency domain as:

$$\begin{aligned} Y(f) &= X_R(f) + H(f)X_T(f) - \widehat{H}(f)X_T(f) \\ &= \alpha(f)X_T(f) + H(f)X_T(f) - \widehat{H}(f)X_T(f) \\ &= \{\alpha(f) + H(f)\}X_T(f) - \widehat{H}(f)X_T(f) \end{aligned}$$

Where $\alpha(f) = exp(j2\pi f\tau)$ The above equation shows why correlation is a problem, in effect its quite likely that the tuning algorithm adapts $\widehat{H}(f)$ to approximate $\alpha(f) + H(f)$ which will end up canceling the received signal from the source too! We may end up with no received signal at the relay in this case.

To solve this challenge, we invent a novel cancellation tuning mechanism: we artificially inject Gaussian noise at a very low power, which is similar to the transmitter noise of the transmission, only this is known to us (30dB below the transmitted signal or 80dB above the noise floor in the worst case). Gaussian noise only undergoes the channel $H(f)$, as it is not part of received signal. Hence to figure out the response $H(f)$, i.e. the tuning parameters, we compute the correlation of the received signal with the Gaussian noise that was transmitted, and estimate the self-interference channel parameters. However once cancellation is tuned, we know that analog cancellation provides around 70dB of cancellation, and digital cancellation takes both the transmitted signal and Gaussian noise as input to eliminate all the remaining self-interference. So as soon as the cancellation is turned on, all of the Gaussian noise is canceled and is not left over in the canceled signal. Finally this injected noise doesn't affect the client data rate (since the maximum SNR required is 28dB for the highest data rate) and very likely by the time the relayed signal reaches it, the injected noise is quite likely attenuated to below the receiver noise floor of the client's receiver.

Experimental Results: We prototype the above cancellation design using WARP software radios with setup similar to the one used in [40], which is used in the evaluation Sec. 4.5. We experimentally evaluated the amount of cancellation when the FF relay node is placed at different locations in our indoor testbed, while its receiving the signal from another and re-transmitting the same signal after the constructive filtering. We observe that our design consistently achieves between 108-110dB of cancellation. Note that the maximum cancellation expected is 110dB, since the maximum transmit power is 20dBm and the noise floor is -90dBm.

4.3.4 FF: Low-delay Constructive Filter

As noted before, the relay can apply a filter such that the relayed signal adds up constructively at the receiver, as seen in Sec. 4.3.2. A typical implementation of this filter consists of a series of delay lines, each with its own gain, as shown in Fig. 4.9.b. Note that the signal at Tap-N of the filter ($N = 3$ in Fig. 4.9.b.) has an ND extra delay with respect to the signal at Tap-0, where D is the delay introduced by each tap. It is important to note that we have a constraint on the number of taps we can employ in our filter because the filter delay ND (which dominantly dictates the maximum delay at the relay) needs to be such that the relayed signal does not fall outside the cyclic prefix at the destination. This section describes how the ideal filter H_c can be implemented with as less filter delay as possible.

Recall that the basic intuition behind this filter is to rotate (i.e., change the phase of) the relayed signal such that it aligns with the direct signal at the destination, as we saw in Fig. 4.5. For example, to rotate a relayed signal at 2.45GHz by 90 degrees, the constructive filter needs to introduce a 100ps delay (400 ps is the time period of one wave at 2.45GHz which corresponds to 360 degrees, hence 100ps corresponds to 90 degrees). It is extremely hard to implement such fine-grained delays on the order of a hundred picoseconds in the digital domain. For example, if we have a sampling

bandwidth of 100MHz, successive digital IQ samples are spaced 10ns apart, in other words two orders of magnitude greater than the delay resolution desired. Figuring out the minute variation in the signal that is delayed by 100ps (which is an intermediate point between two consecutive digital samples) is possible, but extremely complex [118, 78] and defeats our filter delay requirement. The reason is that figuring out the value that an analog signal will take at an intermediate point between digital samples requires us to use sinc interpolation that spans many more future and past digital samples. Using a large number of past digital samples implies that our filter needs to have a large number of taps, which in turn increases the filter delay and thus increases the chances that the relayed signal falls outside the OFDM CP at the destination.

To tackle this problem, FF therefore designs a programmable analog filter that can provide the fine-grained delay adjustment constructive filtering needs as seen in Fig.4.10, without introducing significantly delay multi-path. We design a tunable analog FIR filter structure with four fixed delays and tunable gains on each delay. The delays are spaced 100 picoseconds apart (quarter wavelength of center frequency). To delay a signal by some intermediate value (between 0 to 400ps), the signal is split and passed through all the taps and the gains applied on each tap are adjusted such that the eventual signal has the right phase. Fig. 4.10 shows the basic idea with four delay lines separated by 100 ps and tunable gains on each line. The incoming signal is at 2.45GHz, hence the two copies of the signal after going through the filter have a relative phase shift of 90 degrees. Now, by adjusting the gain on each delay line, we can rotate the vector to any intermediate phase between 0 and 90 degrees. FF's constructive analog filter applies the same idea using 4 delay lines and spans the entire 360 degrees.

However the above discussion applies to only a single subcarrier, but the signals we are relaying are wider band and have multiple subcarriers. The challenge is that typically each subcarrier needs a different phase shift because channels are frequency selective. The analog filter applies the same delays to all subcarriers, so almost all of them will be rotated by the different phase shift and which wont lead to constructive filtering on all the subcarriers.

To tackle this challenge, we use a pre-filter that is implemented in the digital domain as seen in block diagram Fig. 4.3 (called as CNF Digital Filter). The intuition is that this pre-filter pre-rotates the phase in each subcarrier by different amounts such that after the analog rotation occurs, all the subcarrier phases are almost lined up for constructive relaying. Note that this rotation in digital is coarse on the order of a few nanoseconds and hence is much less complex to implement, the analog CNF filter is still responsible for the fine-grained rotation necessary for constructive filtering.

However the pre-filter is limited in the number of taps it can use because each tap adds delay (e.g. for a 80Msps sampling rate, each extra tap adds 12.5ns of delay). To build a reasonable low-delay spread filter, we therefore allow only a delay budget of 50ns which would imply a 4-tap filter at 80Msps. To compute the optimal values of the coefficients for this limited pre-filter, we solve the following optimization problem:

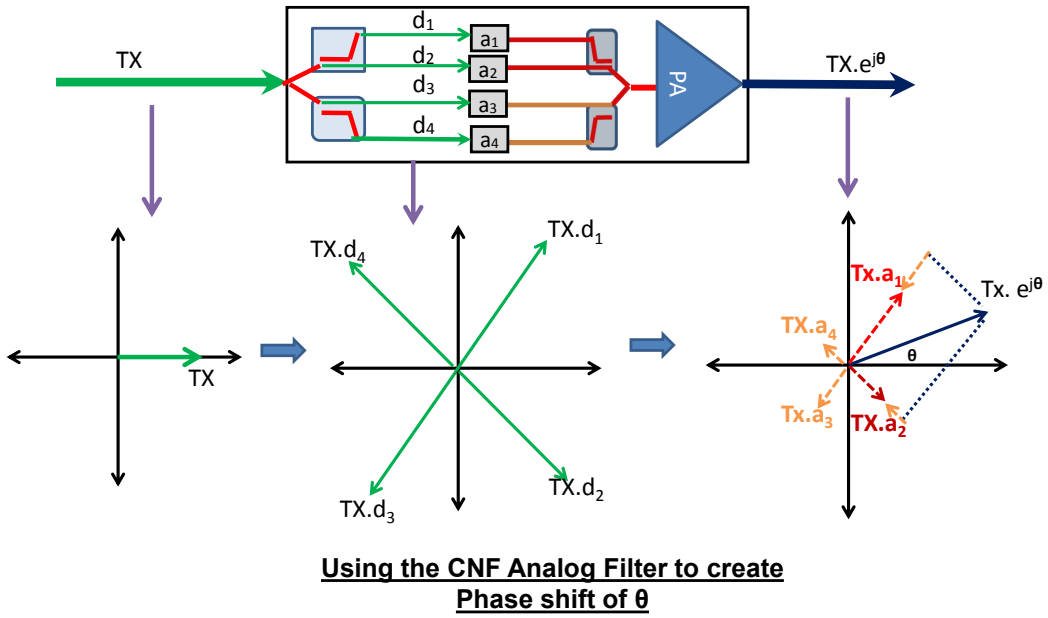

Figure 4.10: FF's constructive analog filter. The filter enables us to rotate the input signal TX by a fixed angle θ by appropriately adjusting the gains on the four taps of the analog filter. The four taps are placed 90 degrees apart, which at 2.45GHz implies that the tap delays are in increasing multiples of 100 picoseconds.

$$\min_{h_D(n),H_A(f_i)} \left| H_A(f_i).\left\{ \sum_{n=0}^{4} h_D(n)e^{j2\pi f_i n} \right\} - H_c(f_i) \right|^2$$

where, $H_A(f_i)$ is the response of the analog constructive filter, $h_D(n)$ represents the pre-filter as described above and $H_c(f_i)$ is the desired overall constructive filter response as computed in Sec. 4.3.2. The above problem is essentially trying to divide up the work of rotation for constructive filtering between the digital and analog CNF filter stages in an optimal manner so as to best approximate the desired constructive filtering response. We omit the details of how to solve this optimization problem for brevity, we use a standard convex optimization technique called sequential convex programming (SCP) to solve it [23].

At this point the constructive filtering is complete. We incur a 50ns delay in the digital domain, and a negligible delay (3ns) in the analog constructive filter. Fig. 4.3 shows the overall block diagram.

4.3.5 Does the relay amplify noise?

A natural concern is whether the relay amplifies noise. For example, let's say the relay is receiving a signal at 20dB SNR. If the actual noise floor is -90dBm, the signal received is at -70dBm. Lets say it applies the 90dB amplification and transmits a 20dBm signal, in that 20dBm signal, noise is at 0dBm. If the path from the relay to the destination attenuates the signal by 80dB, then even at

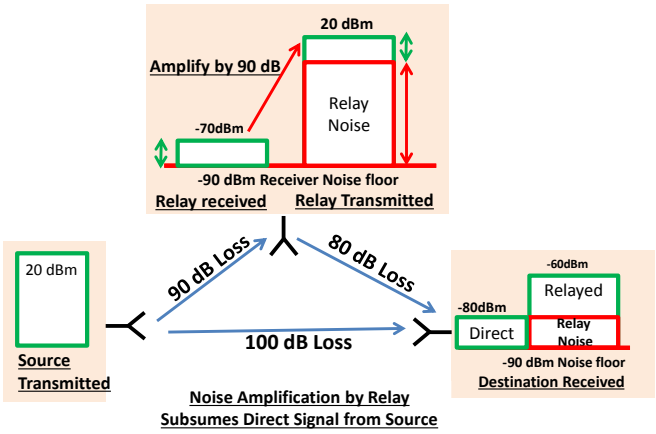

Figure 4.11: Naive amplification at relay can amplify and relay noise to the destination, which can subsume the direct signal from the source to the destination and negate the benefits of construct-and-forward relaying.

the receiver the noise from the relay is at -80dBm. This can overwhelm any signal directly received by the destination from the source if the SNR on that direct link is less than 10dB. So in effect the direct signal from the source is drowned out by the noise that is amplified by the relay. Fig. 4.11 shows how this visually.

Our key insight is that this can be prevented by smartly leveraging the relay's knowledge of the channels. The idea is to compute the amplification factor that ensures that the noise from the relay, by the time it is attenuated by the relay-destination channel, is well below the destination's noise floor. To accomplish this, let's say the attenuation applied by the channel from the relay to the destination is a dB, the maximum amplification factor is given by $(a-3)$ dB (the 3dB is extra margin for safety). In other words amplification is dictated by how much the signal is attenuated from the relay to the destination, the higher the attenuation, the higher the amplification that can be applied. Remember however that amplification is limited at the top by the amount of cancellation achievable.

In the above example where the relay-destination channel attenuation is 80dB, if we use a maximum amplification of 77dB, the relay would transmit a 7dBm signal, and noise would be at -13dBm. This signal after being attenuated by the channel would be received at the destination at -73dBm and noise would be -93dBm. Since the destination's own noise floor is at -90dBm, higher than the noise received in the relayed signal, it doesn't hurt performance. Now the direct signal from the source is not washed out, and assuming constructive and forward filtering has been applied, it should add up to provide a SNR gain.

4.4 Implementation

A full design of FF has to grorange with several engineering challenges, we describe a few prominent ones below. Note that we defer the discussion of how the relay knows the identity of the source and destination of the packet it is relaying to Sec. 4.6, it needs this information to use the right CNF filter. For now, we assume that the FF relay knows the identities of the source and destination to simplify description.

4.4.1 Carrier Frequency Offset and other issues

As with any radio, inevitably there is a carrier frequency offset between the radios at the source and the destination. Relaying should not introduce another carrier offset into the relayed signal, this would break the assumption of it being another multi-path from the source and can confuse the receiver's CFO correction algorithms. Ideally to avoid confusion, the receiver should get the relayed signal also with the same CFO as the signal it is receiving from the source. So the relay should in fact try to relay the signal such that the original CFO offset from the source is preserved.

This would be easy to achieve if the relay itself did not need to process the signal. However for the relay's own processing and constructive filtering, the CFO w.r.t the source has to be removed. Hence the relay applies the following trick: It computes its CFO wrt to the source. When it receives a signal from the source, it first corrects for that CFO [88]. After that it performs its processing, including digital cancellation and constructive filtering. Before transmission however, it applies the reverse of the CFO correction it applied earlier. In effect it restores the CFO that existed in the signal from the source.

4.4.2 How does the relay know the channels for construct and forward relaying?

For construct and forward relaying, the relay needs to know the channel from the source to the destination which it cannot directly measure, as well as channels from the source to itself and from itself to the destination. The channel between itself and the source can be easily measured using received signals, and the channel from the destination to the relay can be measured by snooping on ACK packets and estimating the channel. However the direct channel between the source and destination cannot be measured by the relay, it needs to be explicitly informed of it.

Direct Channel: In cellular systems such as LTE, clients measure the channel from the basestation to themselves and feed it back explicitly to help with scheduling [20], our relay can snoop on this feedback and learn the channel. However WiFi has historically been passive, there is no explicit channel feedback from the receiver to the source. To obtain this information for WiFi at the relay, we use recent enhancements in the WiFi standards. Specifically 802.11n/ac implements an explicit channel sounding phase [99, 1, 15] where the AP sends a pre-defined packet which each client uses

to measure their channels from the AP. The clients respond with the compressed channel state measurement later when polled by the AP. This is known as the Very High Throughput (VHT) beacon packet [1] in the 802.11ac standard. When FF relays are deployed, we make the corresponding AP send out the HT sounding packet every 50ms.

FF relays then take advantage of this mechanism to obtain the channel estimates from the source to each destination in the network. We make the FF relay spoof the AP and send a polling packet to all clients in the network periodically (every 50ms). The relay then listens to the replies from the clients which contain the channel estimates of the channel from the AP to themselves. Further the relay uses these packets to also measure the channel between the relay and each client in the network. The relay also keeps track of the channel between itself and the AP whenever it receives a packet from the AP.

Note that once the relay computes the constructive filter to use in the downlink direction for a particular AP-client pair, it can use the same filter in the uplink direction for the same client-AP pair. The reason is that by reciprocity the environment between the AP and the client is the same in the reverse direction. Further, the cumulative effect of the channel from the AP to the relay, the constructive filter and from the relay to the client is the same even if the order of channels and filter is permuted and multiplied in a different order by commutativity. Hence the same constructive filter can be used in both directions[1].

4.4.3 Hardware Prototype

We have built a prototype of the FF relay using the WARP software radio boards [28]. We build on prior full duplex radio implementations [40], but modify them appropriately to implement the relaying functionality. For all our experiments, we have built a MIMO full duplex 2×2 FF relay building on the self-interference cancellation design from [41, 40]. The prototype has 2 antennas and uses the MIMO analog cancellation design described in recent work [40]. The analog cancellation circuit has 8 taps that are spaced around 100-200ps apart as well as taps for canceling the cross-talk between MIMO antennas. Each tap has tunable digital step attenuators [41] which can be adjusted in increments of 0.25dB from 0 to 31.75dB. The couplers get a copy of the signal from the transmit side, and couple it back in to cancel it on the receive side as seen in Fig.4.8. The cancellation circuit is tuned from baseband after observing the residual using the algorithm described in Sec. 4.3.3.

The baseband implementation is relatively simple. We implement a 4 tap digital construct and forward filter, as well as implement digital cancellation using a 120 tap causal filter. Further CFO correction and re-distortion blocks are also located in baseband. The overall extra delay introduced by baseband process is less than 100ns in our prototype, of which nearly 50ns is from the digital CNF pre-filter and the rest are ADC and DAC delays, the digital cancellation itself doesn't introduce any

[1]Note that, the amplification applied is different in both direction, as the noise introduced at the relay receiver, is asymmetric in uplink and downlink directions.

delay because its causal. Finally, the signature technique identifying the source destination discussed in Sec. 4.6 is implemented which lets the relay know the source and the destination of the packet and allows the right constructive filter to use for relaying.

The ADC and DAC used in WARP software radios aren't optimized for the group delay as one would expect, they have significant group delay. The relay needs to receive the signal and transmit it within 100 nsec, which is not possible with warp software defined radios. So, in essence with this limitation one cannot use the hardware to relay in same time slot. We circumvent this issue by using two time slot. In slot 1, source is transmitting packet 1 and relay is also transmitting the packet 1 simulating the correlation effect of the cancellation tuning, in next time slot the same packet is transmitted from source and relay however relay packet is 100 ns delayed, this time slot is used to calculate the throughput of the system. Thus for every packet we use 2 slots, which also creates the effects of imperfect cancellation impact on the received packet, using the received packet to re-transmit after construct and forward. We built a custom hardware solution to prove feasibilty of low latency loop back transceiver.

Achieving a Low Latency of the transceiver Loop: The ADC and DAC itself can have a lot of pipeline latency, to support different interpolation or downsampling mode, building a low latency transceiver needs ADC and DAC for single sampling rate, further increasing the clock rate can allow achieve very low latency. We use [11] a ADC and DAC converter board has 7ns latency on ADC [4] and DAC has [6] latency of 76 ns, a total of 83 ns, a significant delay on the Kintex KC705 [27]. This particular DAC has high latency, hence, we use a very low latency DAC [5] which has 3 ns latency, a total of 10 ns latency for ADC and DAC. Further, we need to implement a 4 tap filter here would need a few clock cycles to multiply and add on FPGA (at 1Ghz, 40 ns group delay and 9 ns implementation latency , a total of 59 ns). Analog receive and transmit chain can easily achieved with a budget of 41 ns cycles for a Direct conversion receiver is sufficient. We use the Direct conversion receiver subsystems in [7] has major group delay in the low pass filter (LTM9004-AD) has 5ns.

All of our experiments are run with a standard 20MHz OFDM PHY that is based on the WiFi PHY. The PHY uses 56 subcarriers and a 400ns cyclic prefix interval (this is the faster version of WiFi which uses a smaller CP). The numbers we report in our evaluation are all PHY layer throughputs and do not include MAC layer effects. Since the relay is operating at Layer 1, we are orthogonal to any MAC layer effects, so we expect the relative gains should carry over.

4.5 Evaluation

We evaluate the performance of FF using experiments in an indoor setting. We place the AP and a FF relay in *various different indoor* settings, those are, open wide office space, L-shaped corridor and a wide room, two large wide room and including the one shown in the Fig. 4.1. The AP is a 2×2 MIMO AP, and the relay and the client are also equipped with two antennas can 2×2 MIMO. We

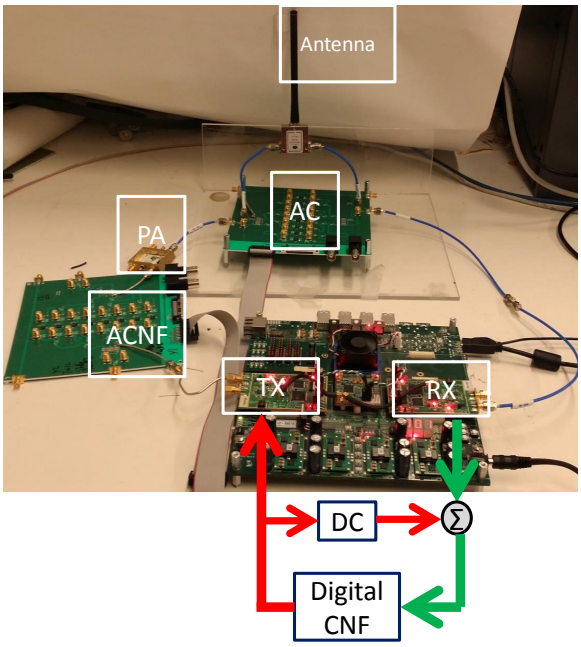

Figure 4.12: FF Prototype

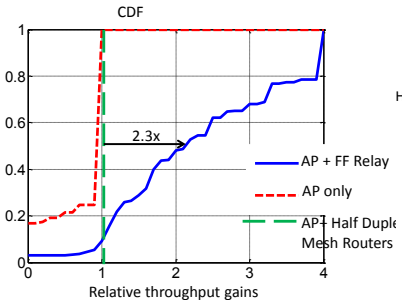

Figure 4.13: FF's overall throughput gains. FF provides a 3× increase in median throughput, and nearly a 4× gain in dead spot scenarios. Further, it significantly outperforms half duplex mesh routers, almost by a factor of 2.3×.

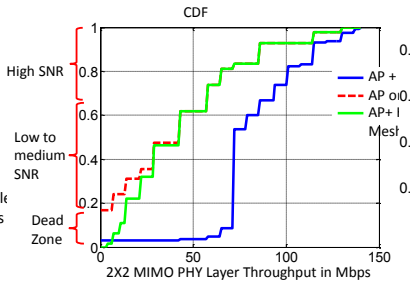

Figure 4.14: PHY Layer absolute throughputs achieved by different schemes. FF provides a significant throughput for nodes that were previously almost getting no connectivity or very low throughput.

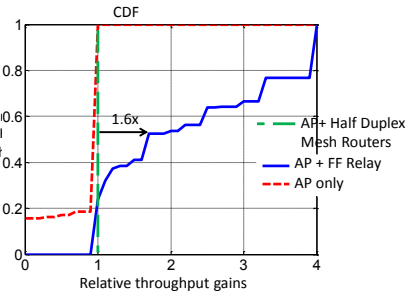

Figure 4.15: FF's throughput gains due to SNR amplification from construct-and-forward relaying for a SISO AP, FF relay and client. FF provides a median gain if 1.6× even without the benefit of MIMO rank expansion.

were limited to 2-antenna devices primarily because of the availability of analog cancellation boards at the relay, we require four of them for implementing MIMO full duplex. We also require four RF analog construct-and-forward boards. However the qualitative conclusions from the experiments apply to any MIMO setup since the constructive filtering technique for improving SNR and MIMO rank is independent of the number of antennas. We assume relay knows the source and destination

for every transmission.

We compare the following three approaches:

- **AP only**: In this approach we only assume that an AP is deployed without any relays.
- **AP + Half-Duplex Mesh Routers**: This is akin to the approach where we have an AP and a half duplex router such as the Apple Airport Express. To make sure the gains are reported correctly, we assume that the AP and the mesh router are perfectly synchronized and transmit in alternative time slots to eliminate any MAC layer contention effects. Hence the numbers reported are PHY layer throughputs assuming perfect MAC coordination. The half-duplex mesh router also has two antennas. Also, AP is smart enough to figure out when it should use the half-duplex router and when not to use it.
- **AP + FF Relay**: This is the design proposed in this chapter. We place it at the same location as the half duplex mesh node. Here too we pick the optimal bitrate to use at the AP assuming the construct-and-forward relaying is in place.

The metric we use is PHY layer throughput which is defined as the optimal bitrate that can be used at any location given the SNR and the MIMO rank. Hence we eliminate any impact of bitrate adaptation algorithms, MAC layer artifacts etc and the experiments purely quantify the impact of relaying. Further to make relative comparisons across the compared approaches, we use a relative throughput gain metric where the baseline scenario is the AP and the half duplex mesh router case. We do not use the AP only scenario because we have dead spots in this scenario where the throughput is zero and we cannot compute relative gain. So all relative gain numbers are wrt to the throughput achieved by using the AP and half duplex mesh router.

Our experiments show that:

- FF provides a median throughput gain of 3× in our experiments wrt to the AP only case. For the bottom 20th percentile of the locations, the throughput gain is as high as 4×.
- FF's gains from MIMO rank increase and SNR gains affect different nodes. For clients that had a decent SNR but low MIMO rank, the majority of the gains are from the addition of a separate independent MIMO path. For clients that are located in dead spots or with very low SNRs, the big gains are from the SNR gain.
- Construct-and-forward relaying has significant benefits, especially for clients with low SNRs. An amplify-and-forward relay without FF's constructive filtering capability performs worse, the median gain wrt to the AP only scenario drops to 1.5×.
- Low latency cancellation and constructive filtering are critical, without them relaying can actually hurt overall performance due to inter-symbol interference, in some cases the performance is worse than no relaying.

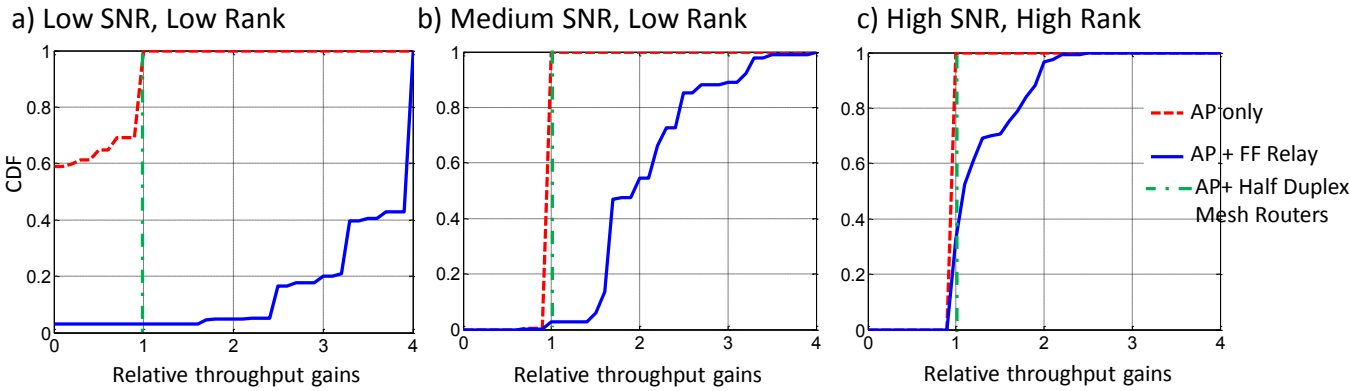

Figure 4.16: FF's performance gains in different scenarios. In low SNR and low MIMO rank scenario (figure a) the gains are significant because FF provides both a SNR gain as well as MIMO rank expansion, leading to a 4× increase in throughput. FF's performance gains in medium SNR and low MIMO rank scenarios (figure b) leads to a 1.7× increase in throughput. FF's gains in the scenarios where the clients already had high SNR and good MIMO rank (figure c) are minor as expected.

4.5.1 Overall Performance Gains

We begin with the basic question: how much does the FF relay help in improving throughput and coverage. We conduct the experiment as follows. We place clients at different locations in the testbed relative to the AP and relay placement as shown before. We measure the channels from the AP to the client and feed it to the relay. We also measure the channel between the AP to the relay and from the relay to the destination. These measurements are all made available to the relay and the measurements are repeated every 50ms. The relay uses these measurements to compute the construct-and-forward filter. We then conduct an experiment where the AP transmits directly to the client without any assistance from the relay. We then repeat this experiment assuming the relay is a half duplex mesh router, and then with the FF relay. We compute the relative throughput gain and plot the two CDFs in Fig. 4.13.

The FF relay provides a 3× increase in median throughput over the AP alone, and a 2.3× increase over half duplex relays. The reasons are as expected, the SNR gain we get from construct-and-forward relaying, as well as the increase in MIMO rank due to the additional independent path from the relay. Consequently the AP is able to use very high bitrates. Further at the edge of the coverage area where performance is typically poor, a FF relay improves performance by a factor of 4×. Compared to the half duplex router, the gains are primarily because a full duplex relay does not need an additional time slot to relay. The half duplex relay definitely helps in the edge of the coverage area, where the direct channel from the AP to the client is so poor, that it is better to take the extra hop with the half duplex mesh node.

A natural question is how much of the gains are coming from the SNR gain due to construct and forward, and how much are due to MIMO rank enhancement. We evaluate this question next.

4.5.2 Performance gain with SISO

We conduct this experiment by using a SISO WiFi AP, a SISO client and a SISO relay (both for the HD and the FF cases). The rest of the experiment is conducted the same way as above. We plot the CDFs of the relative throughput gains in Fig. 4.15. The gains in this experiment should be purely from the SNR gain from construct-and-forward relaying since there is no MIMO. As we can see, the median gain is 1.6× and the gain at the tail is 4×. The experiment demonstrates the fact that in this case the clients at the edge of the coverage area benefit the most. This is expected, since without the AP these clients probably have an SNR in the range of 2-8dB. The relay significantly improves the SNR to about 15-20dB. This translates to allowing the AP to use a 64-256QAM modulation compared to BPSK or QAM before, leading to a $3-4\times$ increase in throughput. On the other hand clients that had medium to high SNR with the AP already don't benefit as much, the gains for them are marginal. The reason is that going from 64QAM to 256QAM doesn't help much, it only increases the bitrate by 33%. The intuitive reason is that the Shannon capacity curve is concave with SNR, there are diminishing returns in terms of capacity as SNR increases. For example, going from 64QAM to 256QAM requires a 6dB increase in SNR, but it only increases the bitrate by 33%.

4.5.3 Performance gains due to MIMO rank expansion

Next we turn to evaluating the impact of FF's ability to expand MIMO rank. We conduct the same experiment as in Sec. 4.5.1. However we divide the results into three classes according to how the MIMO channel matrix looked between the AP and the client without any relaying. The first category is when the SNR and the MIMO channel rank are both low, this corresponds to clients at the edge of the coverage area where both propagation losses and MIMO rank degradation are severe. The second category is when the SNR is medium to good, but the MIMO channel rank is low. This corresponds to clients which are suffering from the pinhole effect, they only have one strong path to the AP which reduces MIMO rank but the SNR is still decent. Finally, the last category is high SNR and full MIMO rank, this of course corresponds to clients which are close to the AP and enjoy strong, multiple independent links to the AP. Fig. 4.16 plots the CDFs of throughput gains in those categories.

Fig. 4.16.c shows that the benefits from FF for the last scenario (high rank, strong SNR) are small, only around 15%. This is as expected, since FF cant increase rank any more and benefits from SNR gains are small. For the second category Fig. 4.16.b, where there is good SNR but low rank due to pinholes, the benefits are substantial from using the FF relay. In effect these relays end up providing an additional strong MIMO path and increase the rank to full rank for MIMO, thus providing close to a 1.7× increase in throughput.

The last category shows the (Fig. 4.16.a) maximum gains, because the relays end up providing a rank of at least two between the AP and the client, as well as enhancing SNR. Given the low baseline these clients are starting from, the gains are therefore significant, showing a 4× increase in

throughput.

4.5.4 Impact of Processing Latency

As we discussed earlier, processing latency at the FF relay has a significant impact. In this experiment we quantify the impact. We artificially introduce some buffering to vary the processing delay at the FF relay from 100ns to 400ns. We then repeat the same throughput experiments as before and plot the median throughput gain as a function of processing latency at the relay in Fig. 4.17. As we can see, the median throughput gains drop significantly and is in fact worse than having no relay when the processing latency exceeds 300ns. The reason is as expected, above a certain latency we hinder OFDM's ability to absorb highly delayed multipath reflections into the current symbol and avoid inter-symbol interference.

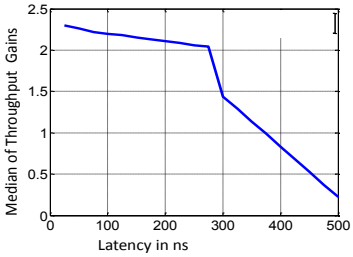

Figure 4.17: Relaying performance suffers as processing latency increases at the relay. Higher latency means that the relayed OFDM symbol falls outside the CP of the quickest arriving OFDM symbol at the destination, leading to inter-symbol interference and poor performance.

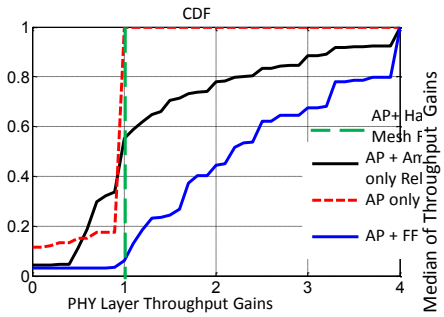

Figure 4.18: FF's construct-and-forward relaying is crucial for obtaining good performance. If we disable it and implement simple amplify-and-forward relaying, sometimes the performance is worse than no relaying because noise gets amplified.

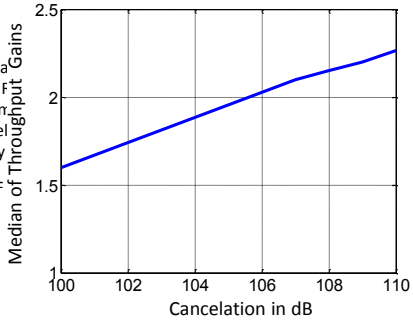

Figure 4.19: Reduced cancellation means reduced amplification, which leads to significantly reduced throughput gains for FF relays.

4.5.5 Impact of No Construct-and-Forward Relaying

In this experiment, we turn off construct-and-forward filtering at the relay and let it simply amplify the received signal to the maximum extent, i.e. as much as the amount of cancellation we obtain. The rest of the throughput experiment is the same as before. We plot the CDF in Fig. 4.18. As we can see, there are still significant gains at the tail. These correspond to client which were at the edge of the coverage area of the AP, and benefit significantly from the amplified relaying. However the median gain is small to non-existent. This is because for the clients that have medium to good SNRs, blind amplification ends up amplifying noise and washing out the direct signal from the AP to the client. Hence the gains are limited and in some cases are worse than before because of the enhanced noise.

4.5.6 Impact of Reduced Cancellation

We conduct an experiment where we vary the amount of cancellation at the relay. Remember that cancellation sets an upper limit on the amount of amplification that the relay could use. We plot the median throughput gain as a function of the amount of cancellation obtained in Fig. 4.19. As expected, with reduced cancellation, overall median throughput gains drop significantly. The reason is that at the edge of the coverage area, being able to use high amplification factors is crucial. A reduced amount of cancellation means the relay's amplification factor is reduced and consequently clients in dead spots see reduced throughput.

4.6 How can we deploy FF?

A final implementation question is how selective is the FF relay. Should it relay any packet it detects? Further, which construct and forward filter should it apply? If it did just an amplify and forward, the FF device might relay packets from a different network and AP (neighbor's WiFi for example) and cause destructive multi-path. Even within the network, if the channel between source and destination is strong, relaying may hurt performance by adding noise. Hence we make a conscious design decision that FF should only constructively relay the packets from its own network.

Further to achieve construct and forward filtering, the relay needs to learn the identities of the source and destination pair to apply the correct constructive filter. In the last section, relay knew the identity of source and destination to know whether to relay or not, and which filter to apply. In scheduled systems such as LTE, this information is known in advance to the AP and can be communicated to the relay explicitly, hence this isn't an issue in LTE. However systems such as WiFi are random access and at any point of time any of the clients or the AP could be transmitting.

One approach could be for the relay to just decode the MAC header (as seen in Fig. 4.20)it is receiving, identify the source and destination and use that to then apply the right constructive filter. However this won't work in practice, because the MAC header is after the PHY header and channel estimation at the destination is performed using the PHY header. Hence the destination would use an incorrect channel estimate in decoding, if relay waits for MAC header to start construct and forward relaying. Therefore in WiFi, we need to find a mechanism for the relay to start applying the right constructive filter before the PHY header itself.

To do so, we make each AP explicitly prepend a pseudo-random sequence of length $4\mu s$, repeated twice, to each packet they transmit. A separate pseudo-random sequence is used for each associated client, and these sequences are learned by the relay on the fly, as AP transmits packets to these clients. The relay continuously looks for these sequences via simple correlation as seen in Fig. 4.21, and whenever it finds a match, picks up the right constructive filter and applies it to the rest of the packet. The pseudo-random sequence at the start of the packet does not affect the client since its decoding kicks in only after it recognizes the standard WiFi preamble. Fig. 4.20 shows the structure

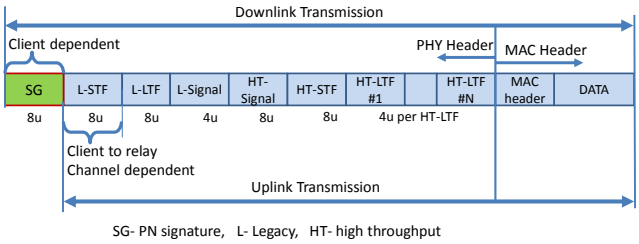

Figure 4.20: WiFi Header with the amendment on the downlink and for uplink we use standard WiFi header, but we use the STF to find the source of the uplink.

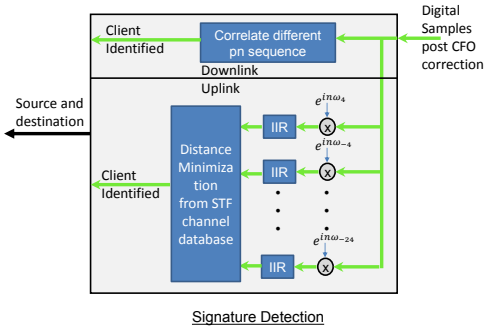

Figure 4.21: Signature Detection technique showing both uplink and Downlink. For Downlink we use correlation based client identifier. For uplink we extract the 10 subcarriers of STF (using the complex exponent and low latency IIR filters) to run distance minimizing on the database of client estimation, which is simply finding minimum distance vector with a phase compensation.

of the downlink packet.

The above technique clearly requires the APs to change, and we believe thats reasonable to expect since its relatively easier to upgrade them. However, we cannot use this technique at the clients, since it will be far harder to expect them all to be upgraded with this new feature. Therefore the above technique only works in the downlink. So, what could we do about the uplink?

We make a key observation here, unlike the downlink, on the uplink the identity of the destination is fixed, its the AP. All we need to do is identify the source, i.e. the transmitting client. To do so, we design a fingerprinting technique as seen in Fig. 4.21. The idea is that every WiFi packet has a short preamble at the start of the packet that is known in advance and when it is transmitted it undergoes a transformation governed by the channel between the client and the relay. Remember that the relay already knows the channel between every client in the network to itself, so it can try and match the received preamble to a set of pre-transformed preambles corresponding to all the clients. This is once again similar to the pseudo-random sequence correlation idea used in the downlink, but in this case we are simply using the transformed standard preambles itself as the sequences to correlate against. Note that this technique does not require any changes to the clients.

This technique will have false positives , since the WiFi preamble even after transformations

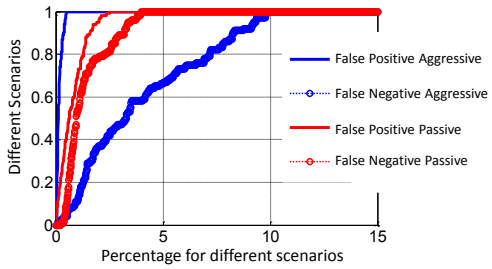

Figure 4.22: Performance of two channel fingerprinting technique, the aggressive one is more suitable.

corresponding to different channels does not have the same differentiating properties as a set of carefully designed pseudo-random sequences. In Sec. 4.5 we evaluate the false negative and positive rates of this technique. A false negative is relatively harmless, since it just means that no constructive filtering will be applied and the situation will be no worse than a standard WiFi network. A false positive (defined as mistaking one client for another) could in some cases worsen the SNR by applying the wrong filter. Hence we tune our identification thresholds to have nearly a zero false positive rate at the expense of a higher false negative rate.

4.6.1 Sender Identity from Channel Fingerprints

We evaluate how well our correlation based technique to identify the identity of the sender in the uplink direction as described in Sec. 4.6 works in practice. We place 4 different client in 100 different locations in our testbed, and for each location calculate the accuracy of sender identification over a time period of five minutes and atleast 1000 packets per client. The extended time period allows us to also account for any channel fluctuations over time. Fig. 4.22 plots the CDF of false positive and negative rates. A false negative means that no sender is identified, whereas a false positive means that some other sender from the actual sender is identified. We see that the technique does have a 5% false negative rate, but essentially a zero false positive rate. The reason for the false negatives is the aggressive threshold applied for identification, sometimes legitimate senders are missed because of these stringent requirement. The conservative trade-off does ensure a zero false positive rate however and prevents the relay from doing any harm.

4.7 Conclusion

This chapter demonstrated how we can design powerful yet simple relaying techniques that can greatly improve throughput and coverage, yet are minimally invasive and do not require sophisticated changes to clients. FF operates within the framework of the current network architecture and design, and we believe can be easily deployed.

Chapter 5

Application: BackFi, low power high throughput WiFi Backscatter

5.1 Introduction

Embedded and connected gadgets - colloquially referred to as the Internet-of-things (IoT) - are increasingly making it possible to continuously monitor our bodies, personal lives and surroundings to improve health, energy usage, security and so on. These gadgets (e.g. wearable, fitness/health trackers, security cameras/microphones, thermostats [96]) integrate with cheaply available sensing technology to continuously measure physical variables such as temperature, heart rate, ambient sounds, etc. and upload them via wireless links to the cloud. Analytics applications then analyze such data to implement useful functionality such as fitness monitoring, intruder detection, regulating HVAC, etc. The future is likely to bring many more such devices helping us instrument more parts of our lives and surroundings, and enable us to measure and analyze almost every aspect of our lives.

We will refer to these IoT gadgets as either IoT sensors, or tags, or simply sensors in the remaining of our chapter depending on the context. To widely realize the IoT vision, we believe that the wireless connectivity on these devices needs to satisfy three key requirements:

- **R1: Sufficient throughput & range**: A typical such gadget produces anywhere between a few Kbps (e.g. temperature sensors measuring every 100 ms) to a few Mbps (e.g., security microphones/cameras recording audio/video), and can be placed anywhere in the home or on the body. So the wireless link from the gadget to the wired gateway connected to the Internet should provide at least a *few Mbps of uplink throughput and 1-5 meters of range.*
- **R2: Very low power design**: These gadgets need to be able to operate for a long time without requiring battery replacements, or ideally without batteries at all. Recent work has demonstrated the possibility of powering these devices primarily using power harvesting from

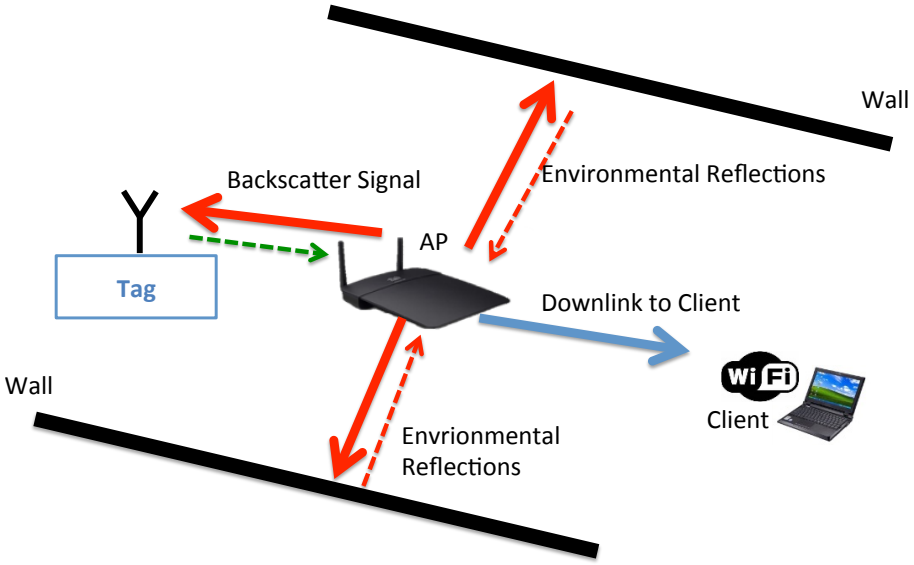

Figure 5.1: Overview of BackFi backscatter system : The AP transmits packet that is meant for the WiFi client (in blue), the transmitted signal (in red) is also reflected by reflectors in the environments like walls. The IoT sensor also receives these transmissions, and modulates its data on it and backscatters the signal to the AP (in green).

ambient RF sources such as TV and cellular signals. A typical RF powered device can harvest upto 100 microwatts of power [120, 110, 86] from TV signals. Hence, ideally the gadget's radio should provide the necessary throughput and range using a *few tens of microwatts of power* to be operable without batteries. If feasible this would eliminate the need for dedicated powering infrastructure such as RFID readers.

- **R3: Reuse ambient signals**: Ideally the IoT sensors should be able to piggyback their data on ambient, widely prevalent communication signals such as WiFi, Bluetooth etc. For example, while a WiFi AP is transmitting a packet to a standard WiFi client, an IoT sensor should be able to modulate its own information on the ambient WiFi signal and communicate its own data back to the AP. However this should not interfere with the normal WiFi communication from the AP to the client. If such a capability is feasible, then one can imagine being able to provide connectivity to IoT sensors using infrastructure that is already being widely deployed for standard wireless communication, thus reducing complexity and cost.

To the best of our knowledge, no current systems satisfies all three requirements. Recent work on WiFi backscatter [74, 70] is the closest, but it does not satisfy **R1**, it only provides around 0.5 Kbps of uplink-throughput and a range of 1 meter which is insufficient for many applications. RFID-based systems satisfy **R1** [127, 55, 116] and some of them satisfy **R2**, but not **R3**. They would require the widespread deployment of dedicated RFID reader infrastructure as well as require their own spectrum band of operation in the unlicensed band. Standard communication radios such as WiFi or Blue-tooth Low Power would satisfy **R1** and **R3**, but clearly cannot satisfy **R2**, they require

between $30 - 50$ mW (Blue-tooth) to several hundred mW (WiFi) of power to operate.

Our goal is to design a radio uplink for IoT sensors that satisfies all the above requirements. We present BackFi, a novel communication link design between backscatter IoT sensors and WiFi radios. The key contributions are a IoT sensor design for backscattering WiFi signals, and a novel radio circuit and algorithm design at the WiFi AP which doubles up as the (AP) reader decoding the backscatter signals from the IoT sensor. The AP reader operates while it is sending a standard WiFi packet to a standard WiFi client as seen in Fig. 5.1. The design satisfies the throughput and range requirements described above, it delivers at least 1 Mbps of throughput even at a range of 5m and much higher throughputs upto 6.67 Mbps at shorter ranges of a meter, To put these performance numbers in context, they are between one to three orders of magnitude better than the best known WiFi backscatter system [74, 70].

BackFi's design makes three key technical contributions:

- First, we design a novel low power IoT sensor that can backscatter standard WiFi signals while being able to sustain high data rates of around 5 Mbps. The IoT sensor consists of a low power design for phase modulations ranging from BPSK to 16-PSK as well as mechanism for detecting WiFi transmission on which IoT sensor data can be modulated and backscattered.
- Second, a novel design of the WiFi AP radio such that it can receive the backscatter signals even while it is simultaneously transmitting a WiFi packet to a standard WiFi client. We leverage recent work on self-interference cancelation for full-duplex radios to enable the backscatter signal to be received while the WiFi device is transmitting [51, 57, 109, 41, 34, 39, 36, 34, 71, 48, 101, 38]. Specifically, the backscatter signal is a modulated version of the transmitted signal itself. Hence self-interference cancelation has to be modified to ensure that the backscatter signal itself does not get canceled. We design novel self-interference estimation techniques that protect the backscatter signal from any degradation due to cancelation.
- Third, we invent novel demodulation and decoding algorithms that can estimate fine-grained changes in the backscatter signal to decode the IoT sensor data. Specifically, we show that WiFi backscatter can be modeled as a channel that is linear but time-varying modifying the IoT sensor data. BackFi incorporates novel decoding algorithms that can continuously track the time-varying channel and use standard diversity combining techniques such as Maximal Ratio Combining (MRC) to deliver a reliable, high throughput link [44].

We prototype BackFi and show that it can provide 5 Mbps of throughput at 1 m range and at least 1 Mbps at 5 m range. In comparison the best performing prior WiFi backscatter system [74, 70] provides a throughput of up to 1 Kbps, a range of less than a meter. We also show that BackFi has minimal impact on the operation of the standard WiFi network whose ambient signals it is piggybacking on to backscatter its own data.

We also note that the focus of this chapter is on the uplink from the IoT sensor to the BackFi AP. The reason is that the IoT applications that we are designing for are bottle-necked on the uplink.

These gadgets (such as fitness trackers, home sensors, wearables, etc) are collecting a lot of sensor data and need to upload them to the cloud and downlink often isn't needed, or if it is, very low throughput of a few Kbps suffice [96, 125]. Hence in the rest of the chapter we will focus on the uplink, but note that prior work has already demonstrated WiFi backscatter designs (which can be used with BackFi too) for the downlink that can provide upto 20 Kbps [74]. We further note that although we have chosen WiFi signaling for the description and implementation of BackFi, the system is applicable for other types of communication signals like Bluetooth, Zigbee, etc., as well.

5.2 Related Work

BackFi is most closely related to recent work on WiFi backscatter [74, 70]. The prior design also uses ambient WiFi transmissions to backscatter data. Specifically, IoT sensors encode data in binary decisions of whether or not to backscatter the received packet transmission which is detected as changes in RSSI/CSI at a nearby helper WiFi device that is also receiving the packet from the AP. The design needs a helper device because the prior design doesn't have self-interference cancellation, hence the transmitting AP cannot detect changes in RSSI/CSI while it is transmitting due to large self-interference. Since information is encoded in binary decisions that span an entire packet, the information rate is only 1 bit per WiFi packet. The range is also low (less than a meter) because the WiFi helper needs the IoT sensors to be close to detect changes in RSSI/CSI. The reason is that the helper device needs to detect the changes in RSSI/CSI while it is receiving the strong WiFi transmission from the AP. This WiFi transmission essentially acts as interference to the detection of weak changes in RSSI/CSI induced by the tag's decision to backscatter or not, and thus limits range.

BackFi on the other hand does not have any of these limitations. Because it modulates information by changing the phase of the received WiFi signal at a much faster rate throughout the WiFi packet, it achieves three orders of magnitude higher throughput. Its range is an order of magnitude higher because self-interference cancellation enables the reader to completely clean out the effect of the ambient WiFi transmission and detect fine-grained changes in the backscatter signal. Finally BackFi provides a framework to analyze energy/bit, which is independent of platform (FPGA, ASIC, discrete) and the technology choice for implementation. However we note that the prior WiFi-backscatter system required no changes to the WiFi AP. BackFi does require the addition of self-interference cancellation hardware. So the trade-off is increased hardware complexity for a much higher throughput and range.

BackFi is related to a large body of work on RFID systems [55, 127, 119, 116, 63, 62, 124, 111, 112, 126, 46, 9], which use dedicated, powered reader infrastructure to supply power as well as receive data from the RFID IoT sensors [89]. The IoT sensors themselves are designed to be low power and may or may not have batteries. However the cost of deploying and maintaining dedicated reader

infrastructure has tempered the adoption of these systems. BackFi and other WiFi backscatter systems [74] use ambient WiFi signals for communicating backscatter data, hence deployment is easier.

BackFi is also related to recent work on ambient backscatter communication [82, 94] that enables two RF powered devices to communicate with each other. However these systems do not provide connectivity to the Internet which is BackFi's primary focus. BackFi is also related but complementary to recent work on harvesting power from RF sources such as TV signals [82, 114, 89], cellular transmissions [95] and WiFi [74, 61, 70, 91]. These systems have demonstrated the ability to harvest around $60 - 100\mu$W from ambient sources such as TV signals [111, 120, 110, 86] which is sufficient power to provide a high throughput battery-less IoT sensor. Hence with BackFi's high throughput, long range, and low power WiFi backscatter connectivity combined with the ability to harvest power from ubiquitous RF sources, we believe we are closer to the vision of RF powered, batteryless IoT sensors ubiquitously deployed and connected to the Internet.

BackFi advances the state-of-the-art in backscatter communication by being able to provide the following:

Improved backscatter decoder: BackFi's decoder presents a first formal framework to decode backscatter on wide-band signals. All the prior backscatter systems use tone as the excitation signals, whereas BackFi uses wideband signals. Further this framework can improve the decoding of the tone based backscatter systems too. The reason is that the silent mode of BackFi eliminates all the backscattered signal by the rest of the environment (including the structural mode of antenna). This allows use of the information on the tone (excitation signal) for decoding, instead of nulling it as in most RFID decoders.

Effective backscatter protocol: BackFi presents a protocol for backscatter devices which allows an efficient decoding for backscatter system. For high order modulation, this design choice becomes imperative to provide good throughput and SNR.

Spectral Efficiency: BackFi presents a high throughput system by piggybacking on the existing data signaling like WiFi or Bluetooth. BackFi capability to reuse existing signaling makes it spectrally efficient and easy to seamlessly deploy. Moreover, since WiFi can be deployed in 900 MHz band too, deploying BackFi is much more effective than deploying RFID readers.

5.3 Overview

BackFi's basic mode of operation is shown in Fig. 5.1. A BackFi capable WiFi AP transmits a WiFi packet to a standard WiFi client. The IoT device with the BackFi tag backscatters the WiFi transmission back to the WiFi AP, and modulates its data on the backscatter signal. The AP decodes the backscatter signal to recover the data from the IoT gadget.

At a high level, the above description also applies to a RFID reader and RFID tag. So why can't we just reuse the RFID design to build WiFi backscatter systems? We argue why but start with a

brief primer on standard RFID backscatter first.

5.3.1 How does traditional RFID work?

In traditional RFID systems, communication happens by the reader first transmitting an excitation signal which is typically a single frequency tone (a sinusoid) in the 900 MHz band. The tag receives this excitation signal and then backscatters (reflects) it after appropriately modifying the phase of the excitation signal. The data that the tag wishes to transmit is modulated on these phase changes. The tag design at a conceptual level is very simple, it is an antenna connected to an array of switches which are turned on and off appropriately to control the phase of the reflected signal from the tag. The array of switches is controlled by logic that reads the information bits, and computes the on-off routine that needs to be implemented on the switch to create the phase difference that encodes the information bits. The backscattered signal is then received by the reader whose goal is then to demodulate the signal by first detecting the phase changes introduced by the tag and then recovering the original data. The design of the tag is fairly standard and is not the focus of this chapter, we refer the reader to a large body of literature [116, 89] on the circuit level details of implementing tags.

It is useful to construct a model of the signal that the reader receives after the tag backscatters the signal. If $x(t)$ is the excitation signal transmitted by the reader, it undergoes four distortions before it arrives back at the reader again after reflections and backscatter. First, the signal gets reflected by objects in the environment other than the RFID tag and arrives back at the reader, we model this environmental distortion as h_{env}. The other portion of the signal is the one that first goes to the tag, has its phase changed to modulate data, and then comes back to the reader, i.e, the backscatter signal. We represent the forward channel between the reader and tag as h_f, the phase modulation at the tag is simply a multiplication of the received signal by $e^{j\theta(t)}$ and the backward channel is represented by h_b. The phase $\theta(t)$ is changed at the tag according to the data that is being modulated, for example, if DQPSK is being used, the phase w.r.t. the previous symbols phase is shifted by the appropriate multiple of 90 degrees. Note that $\theta(t)$ is changing at the rate of the symbol period at the tag. So the overall signal received back at the reader is given by:

$$y_{rx}(t) = \underbrace{x(t) * h_{env}(t)}_{environment} + \underbrace{\{(x(t) * h_f(t)).e^{j\theta(t)}\} * h_b(t)}_{backscatter} \tag{5.1}$$

The goal for the reader of course is to estimate $\theta(t)$ and thus demodulate the tag data. As the above equation shows, there are two challenges in accomplishing that. First is the environmental term; it contains no useful information and therefore acts as interference. This self-interference (because its generated by the reader's own transmission) is likely quite strong relative to the backscatter signal because it consists of direct leakage from the reader's transmitter to the receiver as well as reflections from nearby objects. In many cases, the self-interference and the backscatter signal

can be separated by more than the dynamic range of the reader's receiver chain, which would end up completely drowning the backscatter signal. Second, if the environmental interference can be eliminated, the challenge is to estimate h_f and h_b, and then given that we know $x(t)$ it is simple to recover $\theta(t)$ and demodulate the tag data. The above two challenges are true for any backscatter system, we describe how current RFID systems handle them and why we cannot use that design for BackFi next.

Decoding Standard RFID Backscatter

In standard RFID based backscatter, the excitation signal is a sinusoid. So $x(t)$ in the above equation is $e^{j\omega_c t}$, where $\omega_c = 2\pi f_c$ is angular frequency and f_c is the carrier frequency (typically in the 900MHz ISM band). This simple fact ends up making both the interference cancellation and demodulation problem easier.

First, self-interference cancellation is simple because with a tone as the excitation signal, the interference term $x(t) * h_{env}(t)$ is simplified to $H_{env}(\omega_c)e^{j\omega_c t}$, where $H_{env}(\omega_c)$ is the frequency domain channel response corresponding to $h_{env}(t)$ and is evaluated at the tone frequency ω_c. In other words the original excitation signal is modified by a single complex number, essentially a single attenuation value and a phase shift. This is a special property of sinusoidal inputs to LTI channels, convolution simply becomes multiplication with the frequency domain channel response's value at the tone's frequency for tone inputs. This simplification does not apply to wideband signals such as WiFi. Hence to implement interference cancellation, all we need is a tunable phase shifter and attenuator, which is programmed dynamically to emulate $H_{env}(\omega_c)$. The cancellation circuit would get a copy of the transmitted excitation signal as input, pass it through the phase shifter and attenuator which have been tuned to $\angle H_{env}(\omega_c)$ and $|H_{env}(\omega_c)|$ respectively. Finally, the design subtracts it from the received signal at the reader to eliminate the self-interference. Note that this is a well known technique and is implemented in commercial readers today [45, 24].

Similarly, recovering $e^{j\theta(t)}$ becomes easy because $x(t)$ is a simple tone. To see why consider the following mathematical simplification after substituting $x(t)$ with a tone, $e^{j\omega_c t}$:

$$\{(x(t) * h_f(t))e^{j\theta(t)}\} * h_b(t) = H_f(\omega_c)\{e^{j\omega_c t}e^{j\theta(t)}\} * h_b(t)$$

Further simplification happens after down-conversion to baseband at the reader:

$$y_{\text{tag}}(t) = H_f(\omega_c)h_b(t) * e^{j\theta(t)}, \tag{5.2}$$

which is a standard decoding problem on a linear time invariant system with channel $H_f(\omega_c)h_b(t)$ and input $e^{j\theta(t)}$. Hence standard phase demodulation and decoding techniques [100] can be applied to recover the original phase modulated data.

5.3.2 Why can't we reuse the above design for BackFi?

The key difference between BackFi and conventional RFID backscatter is that BackFi aims to use standard WiFi signals as the excitation signal. So none of the above simplifications that came about because the excitation signal was a simple tone apply. In fact the self-interference cancellation and demodulation problems become significantly harder as we show below.

First, self-interference cancellation now has to eliminate a relatively wider band signal, not just a tone. The implication is that the self-interference cannot be modeled as a simple attenuation and phase shift applied to the original excitation signal. For WiFi signals that typically span 20-40 MHz or even more wider bandwidths, the frequency domain representation of the distortion introduced by the environment, h_{env} is quite frequency selective. The practical implication is that a simple cancellation circuit that uses a programmable attenuator and phase shifter is not enough to cancel, in fact we need more sophisticated designs that can model the attenuation and phase shifts that happen over the entire bandwidth of the WiFi signal. Hence the traditional reader design for eliminating self-interference doesn't apply.

Second, and more importantly, the decoding problem no longer reduces to a standard demodulation problem at the reader like it did with a tone. To see why, the reader is now trying to recover the phase $\theta(t)$ from the following received signal at the reader after down-conversion:

$$y_{\text{tag}}(t) = (e^{j\omega_c t}x(t) * h_f(t)).e^{j\theta(t)}\} * h_b(t) \tag{5.3}$$

The above equation represents a **time variant** channel that transports the input $e^{j\theta(t)}$ into the output $y_{\text{tag}}(t)$, and the information that we are trying to decode is buried inside this time variant channel. The reason the channel term is time varying is because the WiFi signal $x(t)$ is also acting as a channel distortion that is modifying $e^{j\theta(t)}$. Consequently standard decoding techniques designed for linear time invariant systems cannot be applied.

The main contributions of this chapter are the design of self-interference cancellation and decoding techniques that can work when WiFi signals are used for backscatter. We also describe how BackFi ensures that it does not interfere with standard WiFi communication which the WiFi signal was originally created for.

5.4 Design

BackFi uses ambient WiFi transmissions that are being sent by a WiFi AP to a standard WiFi client as the excitation signal. The tag receives the WiFi signal, modulates data on the received WiFi signal, and backscatters the signal to the AP. The architectural design of the BackFi tag is shown in Fig. 5.2. IoT sensor consists of BackFi tag and a sensor populating the data in the tag data memory unit.

5.4.1 The BackFi Link Layer Protocol

First, we describe how a BackFi AP activates and gets a BackFi tag to backscatter information. The protocol proceeds in two stages as described below.

How is the tag activated?

Whenever a BackFi AP transmits, if it is willing to receive backscatter communication, it follows a special protocol before transmitting the WiFi packet. Specifically, like in prior work [74], it transmits a *CTS_to_SELF* packet to force other WiFi devices to keep silent. Next it transmits a series of short pulses to encode a pseudo-random preamble sequence. If the preamble bit is one, then a pulse is transmitted and if its zero, no pulse is transmitted. The preamble is 16 bits long and each bit period lasts for a 1 μs. The preamble is meant to be the signal to the BackFi tag that the BackFi AP is willing to listen to backscatter transmissions. Note that a preamble can be unique to a particular BackFi tag that is connected to this BackFi AP and can be used to select which BackFi tag gets to backscatter at that instant. In such cases, a tag only backscatters when it detects the preamble meant for it.

A BackFi tag by default is in an energy saving sleep mode if it has no data to transmit. If it has sufficient data to transmit (potentially after a sensor has collected enough data), the tag wakes up and listens for its preamble from the BackFi AP. To listen and detect the preamble, the tag uses an energy efficient detector circuit. To build our preamble detector we leverage a large body of work done in low power wake up radio design [104, 54, 98, 97]. These detectors work at power consumption between $98nW$ [104] to 7.5 μW [54], and can detect input signals with power between -41 dBm and -56 dBm. The design has an envelope detector, a peak finder, a set-threshold circuit and a comparator. The envelope detector removes the 2.4 GHz carrier frequency from the received signal and the peak detector detects and holds the peak amplitude of the received signal after envelope detection. The set-threshold circuit obtains the output of the peak detector and outputs half the amplitude as the threshold. Finally the comparator compares the signal after the output of the envelope detector with the threshold and outputs one bit whenever the received signal is greater than the threshold value and a zero bit otherwise. The comparator outputs a bit decision every microsecond, corresponding to the bit period in the preamble. Finally digital logic on the BackFi tag correlates the detected 16-bit long sequence over sliding windows with the known preamble associated with that tag, and if there is a match it activates the rest of the backscatter circuitry to begin modulation of its data.

How does the tag modulate its data?

Fig. 5.4 shows the various timing events and packet format used by the BackFi tag. We will describe their functionality in detail later in this section, here we give a brief overview. Once the excitation energy is detected and the reader is identified (which lasts 16 μs), the tag goes into a *silent period* that lasts for another 16 μs. During this time the tag will suppress any backscatter transmission, which allows the reader to estimate the channels needed for self-interference cancellation as described

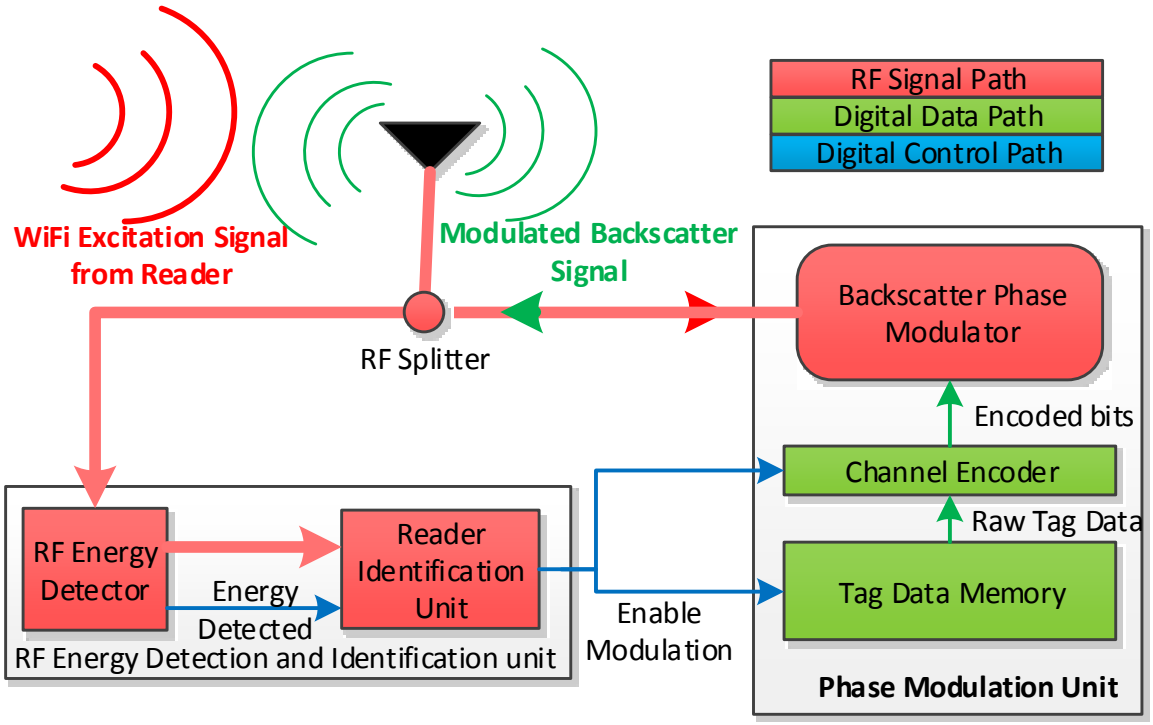

Figure 5.2: Architecture of the tag used in BackFi: Once the tag senses the WiFi excitation signal from the reader, it wakes up the modulation subsystem. The tag then reads the data to be uploaded and modulates it on the excitation signal by selecting discrete phase using the *Backscatter Phase Modulator*.

in Sec. 5.4.2. After that the tag transmits its own *preamble* sequence for 32 μs that is known at the BackFi reader. Using this sequence the reader can estimate the channels it needs for decoding the backscatter data. This sequence is a pseudo random with very high auto-correlation, and is used by the reader to find the symbol timing from the tag.

The tag then sends its data payload by phase modulating the received signal. Specifically, let's say the tag is using QPSK modulation, hence there are four symbols $[e^{j\theta_1}, e^{j\theta_2}, e^{j\theta_3}, e^{j\theta_4}]$ in the constellation map separated by 90 degrees on which two bits of information can be modulated. The tag reads the data that needs to be transmitted, picks out two bits at a time, maps it to the appropriate QPSK symbol and then multiples the received excitation signal from the WiFi transmitter with the corresponding phase signal, $e^{j\theta_i}, i = 1 \ldots 4$ to modulate the data on to the WiFi signal. The specific circuit by which the phase modulation signal $e^{j\theta_i}$ is generated is a well studied problem and has been widely used in RFID tags [116]. Fig. 5.3 shows the detail of the RF phase modulator we use in the BackFi tag.

The phase modulator consists of several RF Single Pole Double Throw (SPDT) switches that are connected in a binary tree structure. These switches can pass incoming RF signal to one of the two ports. These switches can be controlled using digital signals and the tag uses the data to be

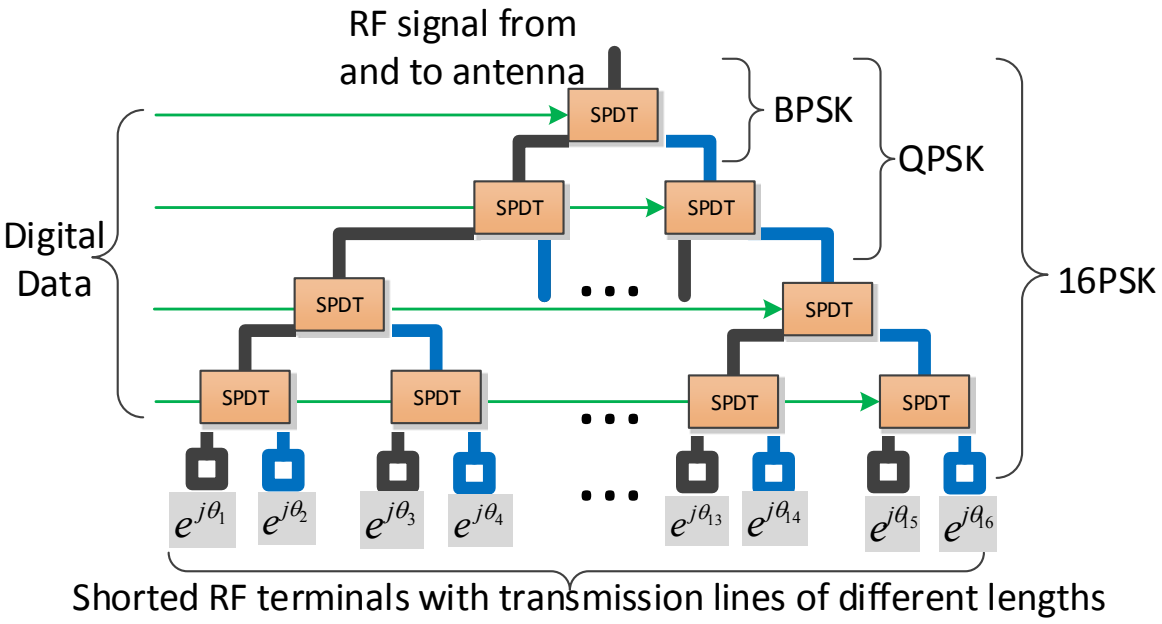

Figure 5.3: Structure of the backscatter phase modulator used in the tag of BackFi: The four digital signal can be used to select one of the 16 possible phases at the leaf of the tree. The incoming RF signal traverses from the top input port all the way to the selected leaf node and is reflected back from the short circuited terminals to the input RF port.

modulated as the control signal for these switches. At the leaf of the switch tree, different lengths of RF traces are connected. These trace lengths are designed specifically to achieve the discrete phase shift required for the supported constellation. The number of SPDT switches is determined by the number of constellation points that are supported. For example, for BPSK only one switch is needed, for QPSK three switches are needed and for 16-PSK 15 switches are needed. Also, if the tag can support higher modulations, then all the lower modulations can also be supported. For example, the design in Fig. 5.3 can support 16-PSK, QPSK, and BPSK, by appropriately preventing some of the switches from toggling as shown in the figure.

To improve the performance of the link the tag also employs simple channel encoding using convolutional codes. The convolutional codes are powerful error correcting codes yet their encoders are very easy to implement using few standard digital components which incurs small energy penalty on the tag. For example, a rate $\frac{1}{2}$ convolutional encoders with constraint length of 7, will require 6 shift registers and 8 XOR gates.

Tag Symbol Rate: The BackFi tag also has a choice on the rate at which it will generate the phase modulation symbols by controlling the switching frequencies on the SPDT switches. The trade-off here is that higher frequencies consume more power and energy, hence the actual rate to use is a function of how much energy is available either via batteries or harvesting. In BackFi tags, this is a configurable parameter ranging from 0.01 megasymbols/second (MSPS) to 2.5 MSPS.

Next, the BackFi AP after receiving the phase modulated, backscattered signal proceeds to

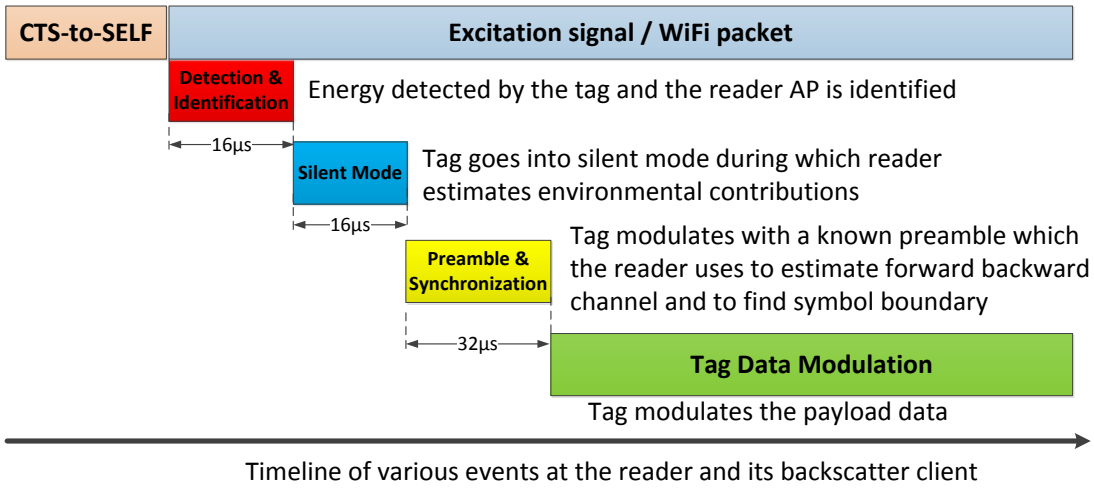

Figure 5.4: The BackFi AP first sends out the CTS-to-SELF to force other WiFi into silent mode. It then sends out the energy detection and identification data to its backscatter client. Once the WiFi excitation signal is received by the tag, it goes through sequence of operations shown above before modulating its data on the excitation signal. The excitation signal is in fact a WiFi packet meant for a regular WiFi client which receives and decodes the WiFi packet without ever noticing the presence of the backscatter communication that is happening simultaneously.

decode the tag's data. As discussed in the previous section, the two key challenges here are wideband self-interference cancellation and time-varying decoding. We describe how BackFi addresses these challenges next. Note that the channel model of the signal received back at the reader with BackFi is exactly the same as standard RFID backscatter and has been derived in Eq. 5.3, the only difference of course is that $x(t)$ is the WiFi OFDM signal instead of a tone.

5.4.2 Self-Interference Cancellation

Like conventional RFID systems, the tag's backscatter signal in BackFi is buried under strong self-interference. This interference stems from two sources: direct leakage from the AP's transmit chain to the receive chain and from reflections of the WiFi transmission by non-tag objects in the environment. But unlike the single tone excitation signal in RFID, BackFi's excitation signal is a wideband WiFi OFDM signal. Because of the wideband nature, scaling the excitation signal by a single attenuation and phase shift is not sufficient to model the self-interference. This is because different frequency components of the WiFi signal add constructively or destructively due to the multi-path effect which results in frequency dependent scaling and phase shifts. However, this problem has been studied extensively in recent years for designing full-duplex radios [41] where self-interference needs to be suppressed to be able to simultaneously listen to weak signals that are being received. The difference in BackFi from those scenarios is that the backscatter signal (which corresponds to the weak signal we want to receive) is a modified version of the transmitted signal, whereas in standard full duplex that is a completely independent signal originating from another

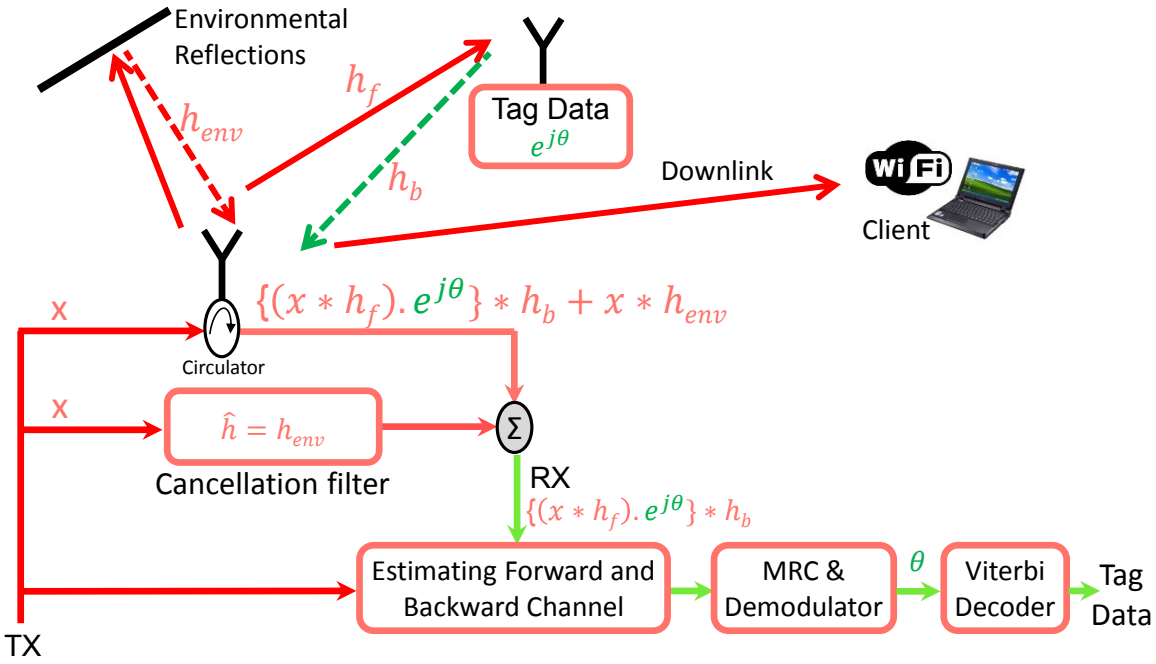

Figure 5.5: Architecture of the reader used in BackFi: The reader transmits the excitation signal x which is actually a WiFi packet meant for a client. This signal is reflected by the environment, which the reader cancels using *cancellation filter*. The residual signal after cancellation is used to estimate the forward and backward channel from and to the tag. The reader then applies MRC to estimate the tag data $\hat{\theta}$, which is further improved by passing it through Viterbi decoder.

sender. So BackFi leverages the recent work on full duplex, but modifies it appropriately to handle the fact that backscatter signals are highly correlated with the self-interference signal.

We briefly review the design of self-interference cancellation systems for completeness, but refer the reader to prior work [41] for a complete description. Self-interference cancellation systems first estimate the channel $h_{\text{env}}(t)$ that the leaked and reflected signal have gone through before reaching back at the receiver. This estimated channel distortion is applied to a copy of the transmitted WiFi signal to recreate the self-interference accurately, and the distorted signal is then subtracted from the received signal to eliminate self-interference. The distortion application and subtraction happens in two stages, analog and digital. Analog cancellation is necessary to ensure that the receiver's ADC is not saturated by self-interference which would drown out the weak backscatter signal before being received in baseband. Analog cancellation is implemented using a combination of RF FIR filters and couplers [41], but cannot completely eliminate self-interference due to the imprecision of analog components. Hence a second digital cancellation stage is employed after the signal is sampled by the receiver's ADC to eliminate the residual self-interference. Digital cancellation is implemented via digital FIR filters. Fig. 5.5 shows the design.

If we directly apply the prior design, it will end up canceling parts of the backscatter signal too. The reason is that prior design aims to accurately estimate the non-linear transfer function that

captures the relationship between the transmitted signal and the received signal [41]. But as we have shown in the previous section, the backscatter signal is actually a non-linear transformation of the transmitted signal. If naively applied, prior designs would end up canceling the backscatter signal too which would reduce the SNR and throughput of tag's transmissions back to the reader.

To tackle this, BackFi's link layer design ensures that during the channel estimation phase of self-interference cancellation, there is no backscatter transmission. Specifically, when a BackFi tag is excited by a WiFi transmission, they do not instantly start backscatter. Instead they employ a *silent period* of 16 μs as shown in Fig. 5.4, during which they do not backscatter, and only then start modulating their data on to the received signal and performing backscatter. We show experimentally that this small silent period is sufficient for the reader/AP to estimate the self-interference channel and perform cancellation for the rest of the WiFi packet. Since there is no backscatter during the channel estimation phase, self-interference cancellation does not model the backscatter reflections and therefore they are not affected by cancellation.

At this stage, the reader/AP is left with just the non-linear backscatter reflection from the tag, and its goal is to decode the data. We describe this step next.

5.4.3 Decoder Design of BackFi

As reviewed before, since the WiFi signal $x(t)$ is wideband, the excitation signal received at the tag $z(t) = x(t) * h_f(t)$ cannot be considered as simple scaled and phase shifted version of $x(t)$ as with standard RFIDs. Hence after the removal of the self-interference, the residual signal at the reader after down-conversion to baseband is given by

$$y_{\text{tag}}(t) = \underbrace{\Big[(x(t) * h_f(t))e^{j\theta(t)}\Big]}_{\text{tag signal}} * h_b(t)\,. \tag{5.4}$$

Here, the signal $x(t)$ is the WiFi transmission that the reader is sending. This signal is wideband and varying but known to the reader. The channels h_f and h_b are the forward and the backward channels. These channels can be considered time invariant for the duration of the tag packet but are unknown. The goal is of course to recover the tag signal $e^{j\theta(t)}$ from the above equation. This is challenging because the tag signal is being modified by a time varying unknown channel, namely $x(t) * h_f(t)$. Contrast this with standard RFID decoding at the reader in Eq. 5.2, where the tag signal is being modified by a time-invariant channel since both $h_f(t)$ and $h_b(t)$ are time-invariant for the duration of the tag packet. We describe how BackFi tackles this time-varying decoding problem next.

Estimating the forward/backward channels

First, the BackFi AP estimates the forward and backward channels, $h_f(t)$ and $h_b(t)$. We can assume these channels to be time invariant for the duration of the tag packet, hence to estimate them we

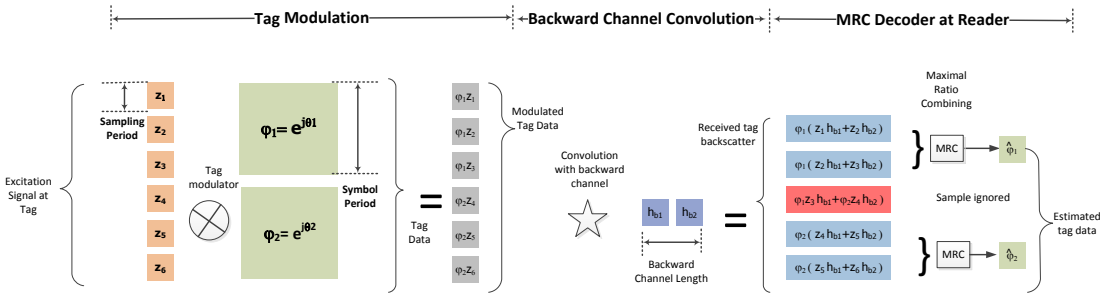

Figure 5.6: Discrete time representation of the design of BackFi: The samples of the WiFi excitation signal z is multiplied by the data ϕ at the tag.The modulated signal then passes through the backward channel h_b. The sampling period of WiFi is much smaller than the symbol period of the tag. This results in multiple copies of the tag data over several sampling period at the reader. These multiple copes are combined optimally by the MRC to estimate the tag data $\hat{\phi}$.

use a standard communication technique: a preamble. Specifically, after the tag detects that it should backscatter and stays quiet for the *silent period*, it modulates a constant phase $e^{j\theta_{pre}}$ on the backscatter signal for a fixed period of $32\mu S$. Thus during the *preamble interval* the received tag signal at the reader is given by

$$y_{pre}(t) = x(t) * [h_f(t) * h_b(t)] .$$

Now since $x(t)$ is known, this becomes a standard channel estimation problem encountered in every communication system. We omit the details of the channel estimation technique and refer the reader to the vast amount of literature on this topic [100]. The channel estimation algorithm thus calculates the value of $h_f(t) * h_b(t)$.

Note that the above procedure only provides an estimate of the combined forward-backward channel, but not the individual channels. Hence the decoding step has to work only with the combined channel estimate.

Decoding the Tag Data

The final step is decoding the tag data itself. BackFi's key insight here is the fact that symbol times for tag data are quite long due to the need to conserve energy at the tag. Specifically the tag modulates data by changing the phase term in $e^{j\theta(t)}$. Changing the phase is implemented by switching a transistor as described in Sec. 5.4.1. Transistor energy consumption scales linearly with switching frequency, hence tags use low rate switching frequencies. Typical transistor switching frequencies in tags are on the order of $0.5 - 2.5$ MHz in tags, hence the symbol period in a tag is between $500 - 2000$ ns. How can we exploit this insight to decode the data?

Our observation is that the delay spread in a typical channel between the reader and the tag is far smaller than 500 ns. In other words the length of the channel is far smaller. Intuitively this is because typical distances between a reader and a tag are around 10 m, so even accounting for

reflections the extra multipath delay spread is small. Hence a channel usually lasts for $50 - 80$ ns. But the symbol period from the tag is much longer at 500 ns, hence for the duration of the channel, we can consider the tag signal to be an unknown constant $e^{j\theta_c}$. BackFi leverages this insight to decode, it looks at the part of the symbol period (with some guard periods at the start and end of the symbol time as shown in Fig. 5.6 and tries to find the value of the constant phase within that period.

Specifically, with a constant phase from time $t_1 - t_2$, we can rewrite the decoding equation at the reader as:

$$y(t) = (x(t) * [h_f(t) * h_b(t)])e^{j\theta_c} + N; t_1 \leqslant t < t_2 \tag{5.5}$$

Note that all the terms except $e^{j\theta_c}$ are known in the above equation. A natural next step might be to divide $y(t)$ by $x(t) * [h_f(t) * h_b(t)]$ but this works poorly because it will also divide the noise term in the above equation and in many scenarios amplify it.

To tackle this, we turn to an old trick in communication theory: maximal ratio combining. To see how this works it helps to write the above equation in the discrete domain (the representations are equivalent assuming sufficient sampling rate) as follows:

$$y_{\text{tag}}[n] = e^{j\theta[n_1]}\mathbf{x^T_{n,L+M}}\mathbf{h_{fb}} \qquad \forall n \in [n_1, n_2] \tag{5.6}$$

Here L is the length of the forward channel, and M is the length of the backward channel. The $\mathbf{h_{fb}}$ is the length $L+M$ vector of the combined forward-backward channel and $\mathbf{x_{n,L+M}} = [x_n \dots x_{n+L+M-1}]^T$ is a vector of length $L + M$ constructed using the excitation data $x[n]$, and we have assumed that the tag signal is constant for the period $[n1, n2]$. The above equation is simply a discrete version of Eq. 5.5 with the discrete convolution operation represented as dot product of vectors $\mathbf{x_{n,L+M}}$ and $\mathbf{h_{fb}}$.

Notice that the tag signal is expressed in terms of the forward-backward channel that we have estimated earlier, and therefore individual estimates of the forward and the backward channel are not needed. Also note that the tag modulation is constant for $n_2 - n_1 + 1$ interval which is larger than $L + M$, this is restating the same insight that length of the forward and backward channels is much smaller than the symbol period of the tag. So we will have $n_1 + n_2 - L_M$ different values of $y_{\text{tag}}[n]$ which contains information of the unknown but constant tag signal $e^{j\theta_c}$. We can leverage this fact to combine all these values to obtain the most likely value of θ_c that could have produced those sequence of observations of $y_{\text{tag}}[n]$ over the period $[n1, n2]$ using maximal ratio combining (essentially the same as temporal diversity combining). Specifically MRC would use the following formula to estimate θ_c,

$$\hat{\theta}_c = \frac{\sum_{n=n_1}^{n_2} \hat{y}_{\text{tag}}[n]^{\complement} y_{\text{tag}}[n]}{\sum |\hat{y}_{\text{tag}}[n]|^2}, \tag{5.7}$$

where $\hat{y}_{\text{tag}}[n]$ is the expected tag backscatter signal without the modulation and can be computed

as

$$\hat{y}_{\text{tag}}[n] = \mathbf{x}^{\mathbf{T}}_{\mathbf{n,L+M}}\mathbf{h_{fb}} \qquad \forall n \in [n_1, n_2],$$

and $^{\mathbb{C}}$ is the complex conjugate operator. Essentially the different measurements of y over that interval are weighted appropriately and combined to produce the most likely estimate of θ_c.

At this point, we have a robust estimate of the tag data for that symbol. The algorithm is repeated for all the symbols in the tag packet. There may still be decoding errors of the n-PSK symbols, which we can correct by using a standard channel code on top. In BackFi, we use a convolutional code at the tag to improve the link performance. The coding provides additional robustness and is decoded using a standard Viterbi decoder [100], we omit the details for brevity.

5.5 Implementation

We build a prototype of both the AP and tag of BackFi system. We describe their implementation details below.

5.5.1 BackFi AP

The BackFi AP is implemented using WARP software radios. The WARP incorporates a standard 20 MHz WiFi baseband operating in the 2.4 GHz range. We also use the same implementation on a WARP board to work as a WiFi client in our experiments. Further the decoding logic for backscatter signals is also implemented in the WARP FPGA [28]. For self-interference cancellation, we reproduce the recent design on single antenna cancellation [41, 39].

5.5.2 BackFi Tag

The IoT sensor is designed to operate across the 2.4 GHz WiFi channels. The prototype uses a 2.4 GHz omni-directional antenna that can receive and backscatter WiFi signals and has a gain of 3 dB. In our current prototype, logic implemented on a Kintex Kc705 board [27] supplies the data to be transmitted and configures the backscatter circuitry. This can be replaced with custom ASIC in a full design which would consumes significantly lower energy.

The backscatter circuitry implements two components on the uplink: *the detector*, and *the modulator*. The modulator implements BPSK, QPSK and 16-PSK modulation. The phase modulation is implemented using SP4T switches. We chose phase modulation instead of n-QAM because this will result in the least amount of RF signal degradation during the backscatter modulation.

Energy consumption efficiency metric

In order to compare various implementation choices for IoT sensor, traditionally Energy per Bit (EPB) measured in average joules of energy required to transmit one bit of information has been used as a metric for energy efficiency. However, EPB varies significantly with the implementation

platform. For example, the EPB for an IoT sensor implemented using off-the-shelf discrete components can be orders of magnitude larger than the EPB for IoT sensor implemented in a sub-micron ASIC design. Even for the sub-micron ASIC designs, EPB varies significantly depending on the technology node chosen (say 65-nm CMOS node vs 45-nm CMOS node) and the design choices (low power sub-threshold CMOS design vs traditional strong inversion CMOS design). BackFi's contribution is in showing how the EPB of an IoT sensor are related to each other for various communication parameters on a particular implementation platform. For example, if an IoT sensor can choose BPSK or QPSK for communication, an interesting question may be, what is the relationship between the EPB of these two cases. While to the first order the EPB of these two cases should be the same and only the throughput should double going from BPSK to QPSK, a more detailed analysis shows that EPB is not the same for these two cases.

To understand why, let us refer to the architecture of the RF modulator as shown in Fig. 5.3. While BPSK requires only one SPDT switch, the QPSK requires three SPDT switches with double the throughput, therefore the EPB of the modulator goes up by a factor of $\frac{3}{2}$. Likewise, for 16-PSK we need 15 SPDT switches, but the data rate improvement is only 4 times compared to the BPSK, therefore the relative EPB for modulator increases by a factor of $\frac{15}{4}$. Also, power consumption in IoT sensor has two major components, the first one is dynamic power resulting from the charging and discharging of capacitors in various sub-systems of the IoT sensor as digital logic is computed, and the second is static power which is either due to leakage, or due to constant current required by some of the analog components in the IoT sensor. Because of the static power, the EPB is also effected by the symbol rate of the IoT sensor as the device takes longer time to transmit the same amount of data. For example, an IoT sensor can reduce the symbol rate which results in the improved SNR at the BackFi from MRC, but at the same time the static power consumption of the circuits will increase thereby increasing the overall EPB.

In order to show the energy efficiency trade-offs associated with the various choices offered by BackFi and to decouple them from the energy efficiency gained from actual choice of the implementation platform, we will present the remainder of the results using unit-less Relative Energy per Bit (REPB). We will first describe how energy consumption is modeled for our exemplary IoT sensor as shown in Fig. 5.2 and then show how we can compute its REPB for different parameter choices.

We have modeled the EPB of the tag by identifying the major power consumption modules of the IoT sensor architecture shown in Fig. 5.2. The three major contributors for EPB of this design are: *the RF modulator, the channel encoder and the memory.* As discussed earlier, the EPB of RF modulator varies depending on the chosen modulation index because the ratio of bit rate to the number of SPDT switches varies as we change the modulation index. In our current energy model we have computed the static and dynamic EPB of RF modulation unit by appropriately scaling the data provided for an industry standard modulator, the Analog Devices ADG904 [2].

BackFi uses a convolutional encoder to reduce bit error rates (BER). The exact EPB contributed

by the encoder circuit is a very small fraction of the total EPB required for communication because convolutional encoders with moderate constraint length (7 in BackFi) require only 6 shift registers and a few XOR gates to encode the IoT sensor data. But the major EPB contribution comes from the coding rate associated with the convolutional encoder. For example, a $\frac{1}{2}$ rate code will essentially double the EPB of the RF modulator because the IoT sensor will transmit twice the actual amount of data on the channel. Likewise, a rate $\frac{2}{3}$ code will bump the RF modulator's EPB by a factor of $\frac{3}{2}$ and so on.

And finally BackFi also models the EPB associated with the memory read of the data in the IoT sensor. Because memory reads are performed for the sole purpose of backscattering the data to the BackFi's reader, we believe it is very important to include the read energy associated with the memory element as part of the overall EPB. In our current energy model we have computed the static and dynamic EPB of the memory read by using data provided for Cypress Semiconductor CY62146EV30 [3].

Using the above energy modeling technique we can now compute the EPB required for a particular choice of communication parameters: *channel code rate, symbol switching rate, modulation index*

$$\text{EPB} = \text{EPB}_{\text{mem}} + \text{EPB}_{\text{mod}} + \text{EPB}_{\text{enc}}. \tag{5.8}$$

Here EPB_{mem} is the EPB associated with the memory read inside the IoT sensor. This has two parts, the dynamic EPB that is dependent on the number of read operations per bit of data of IoT sensor, and the static part that is dependent on the symbol switching rate T_s,

$$\text{EPB}_{\text{mem}} = \text{EPB}_{\text{mem,read}} + \text{P}_{\text{mem,static}} \times T_s.$$

Similarly, we can express the EPB associated with the convolutional encoder EPB_{enc} and the modulator EPB_{mod} with their constituent dynamic and static EPB.

In order to obtain the unit-less REPB, we use EPB for one set of such communication parameters as a reference and then divide the EPB for all the other choices with this reference EPB. In our current evaluation, we use $\frac{1}{2}$ rate code with BPSK modulation with symbol switching rate of 1 Mbps as reference communication parameters to compute the reference EPB. Based on the datasheets of the referred parts we computed the EPB for this reference case to be 3.15 pJ/bit.

Also, we have excluded the EPB associated with the energy detection logic as we believe their contribution to the overall EPB will be insignificant. The energy detector is based on prior work on wake up radio [104, 90]. The power consumption of this detector is around 100 nW. The energy detection needs to be done once for every backscatter packet and lasts for 16 μs. A typical backscatter packet will have 1000 bits of information in it. Based on these information the EPB contributed by the detection logic is in femtojoules per bits which is practically negligible. The wake up radio can detect input signals as weak as -41 dBm, which provides sufficient range to wake up

the tag radio even at a distance of 5 m from the AP. The same detection circuitry can be used to implement the downlink communication to the tag from the AP reader. The protocol for downlink communication has been described in prior work [74]. BackFi reuses this design for the downlink and provides similar throughputs of 20 Kbps. Since our focus in this work is on the uplink design we will evaluate it in detail in the next section by using REPB given by Fig. 5.8 as one of the metrics.

5.6 Evaluation

We evaluate BackFi's design in an indoor environment in our lab with rich multi-path reflections and dense WiFi deployment. Our evaluation reveals the following:

- BackFi provides three orders of magnitude higher throughput, an order of magnitude higher range compared to the best known WiFi backscatter system [74, 70]. Specifically BackFi can provide a throughput of 5 Mbps at 1m range and a throughput of 1 Mbps at 5 m range from the BackFi AP.
- BackFi's throughput and range are comparable to traditional RFID platforms such as Ekhonet [127]. The key benefit of course is that BackFi is a WiFi back-scatter system and does not need dedicated reader infrastructure or frequency spectrum.
- BackFi has negligible (less than 5%) impact on the standard WiFi network's throughput even when the IoT sensor is concurrently backscattering WiFi signals.

5.6.1 Throughput, Range, and REPB

First, we evaluate the trade-off between throughput, distance, and REPB for BackFi. For any given distance, BackFi can deliver a set of throughputs by picking the appropriate combination of symbol switching rate, modulation, and coding rate. Each choice of symbol switching rate and modulation has a different throughput as well as different REPB as described in Section 5.5.2. Fig. 5.7 shows the REPB and throughput for every combination of symbol switching rate, modulation, and coding rate. The EPB for each of the these entries can be calculated simply by multiplying REPB and EPB of the reference parameters (BPSK, 1/2 rate with symbol switching rate of 1 MHz).

Note that while throughput monotonically increases from left to right in the table, REPB does not. For example, at an IoT sensor symbol switching rate of 1 MSPS, going from (QPSK, 1/2) to (QPSK, 2/3) results in a decrease in REPB. The reason is that energy needed to switch from 1/2 rate to 2/3 rate is not significant compared to the other energy contributions for this technology node and the increased throughput causes the REPB ratio to decrease. However, if at a certain range if the link SNR is such that both (QPSK, 1/2) and (QPSK, 2/3) encoded backscatter signals can be decoded at the reader, then BackFi would never use (QPSK, 1/2). The rate adaptation algorithm would always pick the modulation, coding rate and symbol switching rate combination with the

Symbol switching rate	Metric	BPSK, 1/2 rate	BPSK, 2/3 rate	QPSK, 1/2 rate	QPSK, 2/3 rate	16PSK, 1/2 rate	16PSK, 2/3 rate
10 KHz	REPB	29.2162	28.1984	31.2517	29.7250	40.4117	36.5951
	Thrput (Kbps)	5	6.67	10	13.33	20	26.66
100 KHz	REPB	3.5651	3.3333	4.0287	3.6810	6.1151	5.2458
	Thrput (Kbps)	50	66.7	100	133.3	200	266.6
500 KHz	REPB	1.2850	1.1231	1.6089	1.3660	3.0665	2.4592
	Thrput (Mbps)	.25	.33	.5	.67	1	1.33
1 MHz	REPB	1.0000	0.8468	1.3064	1.0766	2.6855	2.1109
	Thrput (Mbps)	.5	.67	1	1.33	2	2.67
2 MHz	REPB	0.8575	0.7086	1.1552	0.9319	2.4949	1.9367
	Thrput (Mbps)	1	1.33	2	2.67	4	5.33
2.5 MHz	REPB	0.8290	0.6810	1.1250	0.9030	2.4568	1.9019
	Thrput (Mbps)	1.25	1.67	2.5	3.33	5	6.67

Figure 5.7: Table provides BackFi tag's relative EPB and corresponding data rate for different choices of modulation, coding and tag symbol switching rate.

lowest REPB since the most precious resource here is energy, whether it comes from harvesting or batteries.

Next, we evaluate the throughout and range performance in our testbed. For these experiments we use our WARP based BackFi implementation for the BackFi AP to decode the IoT sensor's backscatter signals. The BackFi AP and the WiFi client are placed such that the maximum WiFi bit rate is 54 Mbps. They are configured to run on WiFi channel-6 in the 2.4 GHz range. The results for other WiFi channels are similar and not presented due to lack of space.

Impact of Range on Throughput: The BackFi's IoT sensor is placed at distances ranging from 0.5 m to 7 m. For each distance, we cycle the IoT sensor through all combinations of symbol switching rates and modulations, and then calculate throughput for combinations that can be decoded at the reader. In each iteration of the experiment, the BackFi's AP reader transmits 1 to 4 ms long packet at 24 Mbps bitrate including the backscatter start sequence as discussed in Sec. 5.4.1. The IoT sensor backscatters for the entire duration of the packet and stops when its detection logic signals the end of the transmission. We repeat the experiment 20 times at each combination of distance and BackFi throughput. Fig. 5.8 plots the maximum throughput achieved as a function of range for two different preamble duration of 32 μs and 96 μs.

Results: As we can see, BackFi is able to achieve a maximum throughout of around 6.67 Mbps at a distance of 50 cm. For more practical ranges, BackFi achieves a throughput of 1 Mbps at a distance of 5 m and around 5 Mbps at a distance of 1 m. This performance is three orders of

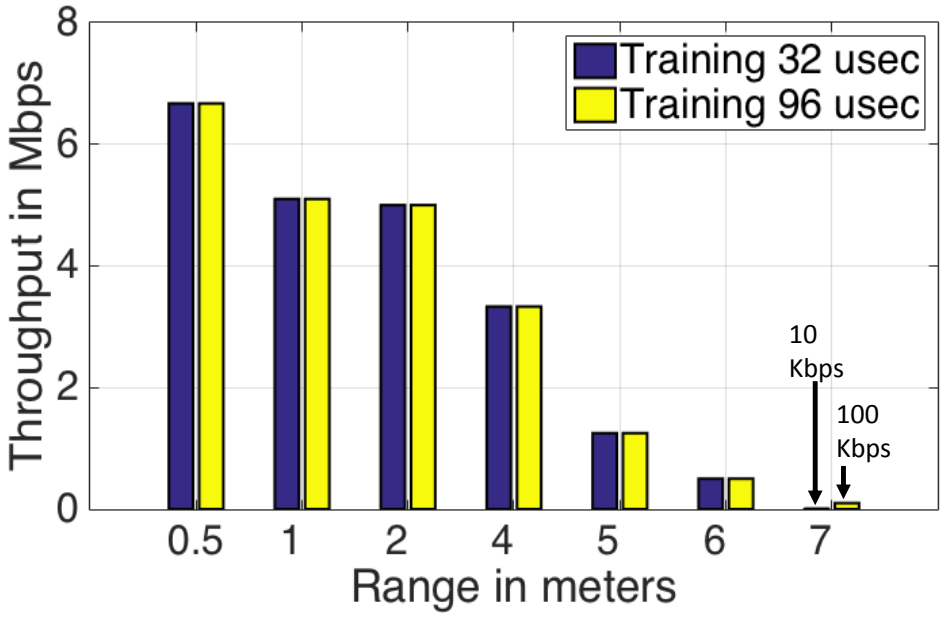

Figure 5.8: Relationship showing range of BackFi and maximum possible data rate possible for two different training times. At 7 meter, if we increase the preamble duration from 32 μsec to 96 μsec, it provides 10× improvement in the throughput.

magnitude better in throughput at the same range as compared to the best known WiFi backscatter system [74, 70]. Note, at 7 m the increased preamble duration of 96 μs shows a 10× increase in the throughput. This is due to the fact that a shorter preamble results in an inaccurate estimate of the forward-backward channel which limits the SNR of the backscattered signal. Hence, for 32 μs preamble, the IoT sensor compensates this loss of SNR by increasing the symbol period to 10×, which in turn reduces the throughput.

To analyze the energy efficiency that BackFi link achieves for different combinations of throughput and range, we plot REPB as a function of throughput achieved for different ranges in Fig. 5.9. To read this graph, note that for every value of range we studied (0.5 m, 1 m, 2 m, 4 m, 5 m), we have a different curve (with a different color). Now for each particular range, we check what combinations of tag symbol rate, modulation and coding rates employed at the tag can be successfully decoded at the BackFi AP. For each throughput, we look up all combination that achieve it, and their REPB from Table. 5.7 and choose a minimum REPB and plot a point. All the points for that particular range are now joined by lines to show the feasible points for each range.

Fig. 5.9 shows that for a given range, throughput increases are obtained by either increasing the symbol switching rate, moving to a denser modulation or higher coding rate or some combination of all three. Each one of these increases energy consumption as expected, which leads to the step increases in REPB. Of course certain throughputs simply cannot be supported at a given range because the link's SNR is not strong enough to decode the data. The vertical line indicates the

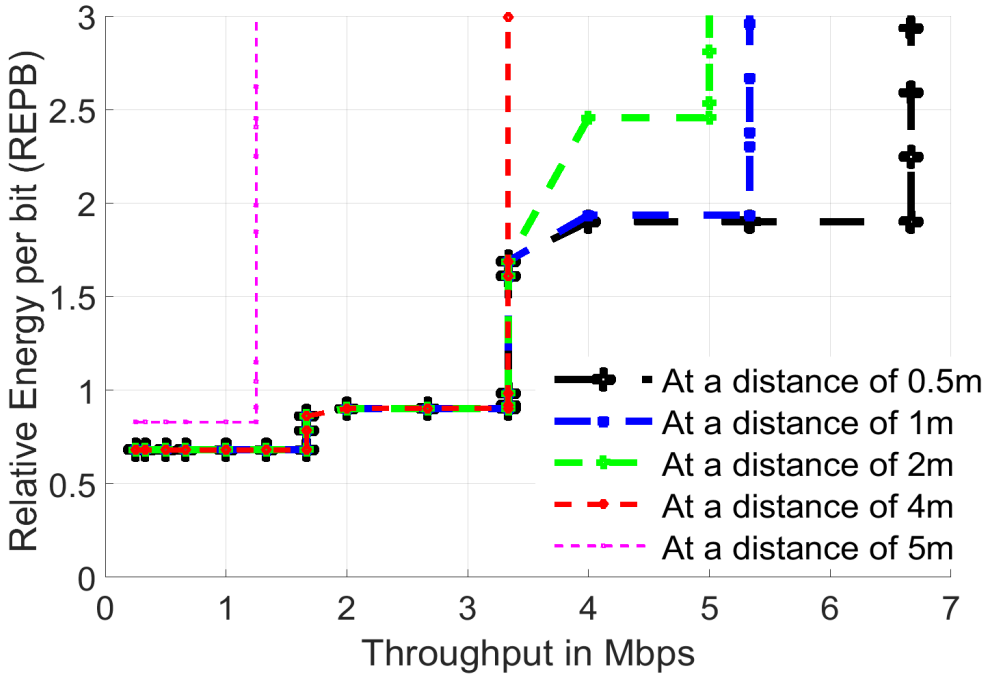

Figure 5.9: Each plot is BackFi's REPB for corresponding throughput achieved for the range varying between 0.5 m to 5 m. For example, we see that at a distance of 2 m to achieve 4 Mbps throughput we need to spend much more energy per bit than at a distance of 1m. Also, the vertical line indicates the maximum throughput that is achievable at a given distance between the tag and the reader.

maximum throughput that can be achieved for a given distance between the tag and the BackFi's reader. Hence we see the curves stopping after a certain throughput for different ranges. Overall REPB lies between 0.5 to 3 for most combinations.

Next, we plot how REPB changes as a function of range assuming we want the same throughput. For this experiment we pick two throughputs, 1.25 Mbps and 5 Mbps, for which we want to optimize the communication link. For each value of range, we pick the combination of tag symbol rate, modulation and coding rate that can achieve those throughputs if there are any. Among the possible combinations we pick the one with the lowest REPB and plot it for that range. Fig. 5.10 shows the REPB as function of range for these two throughputs.

Here we see expected results. For a fixed throughput, as we go to higher ranges we need to use lower coding rates. In our current design we only support two coding rates: 1/2 and 2/3. Hence for all these experiments we see the REPB change between two levels corresponding to the shift from higher coding rate to lower.

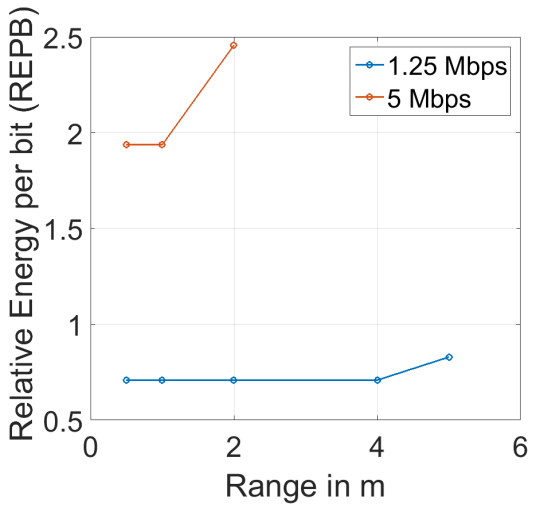

Figure 5.10: For achieving fixed throughput using BackFi for different distance, the tag needs to spend more energy as it goes far away. For achieving 1.25 Mbps we need to spend 2.5× more than power needed for reference modulation, coding and switching rate.

5.6.2 Reconstructing BackFi's performance

In this section, we aim to understand where do BackFi's benefits come from. As discussed before, BackFi's design has two key components: *self-interference cancellation and the decoding algorithm.* We try to shed light on the impact of each component on BackFi's performance.

Impact of self-interference cancellation: This component helps eliminate the unwanted leakage and environmental reflections from reducing the backscatter signal's SNR. Any uncanceled interference directly acts as noise to the backscatter signal and reduces throughput. To evaluate its impact we measure the SNR for the backscatter link at the reader and compare it to what the SNR would have been if cancellation was perfect. The experiment is conducted by placing the BackFi AP and the IoT sensor at 30 different locations in the testbed. For each location, we do ten runs where during each run we let the BackFi IoT sensor backscatter a known packet and measure the forward and backward channels from the tag using a vector network analyzer. In this scenario the VNA [108] acts as the BackFi AP and is being used so that we can measure the channels accurately for comparison. Next we perform the actual backscatter communication with a BackFi AP and decode the data after self-interference cancellation. We also compute the SNR of the demodulated phase modulated symbols from the tag and compare it to the SNR predicted by the channel measurement from the VNA. We plot these two SNR values for each run and each location as a scatter plot in Fig. 5.11(a). As we can see cancellation works well, the median degradation in SNR is less than 2.3 dB. This is consistent with earlier self-interference cancellation results from prior work [41, 39] which report a self-interference residue of 1.7 dB after cancellation.

Impact of Symbol Time and MRC: The second component of BackFi's decoder at the BackFi AP is the algorithm for dealing with the time-varying decoding problem. The algorithm has two key

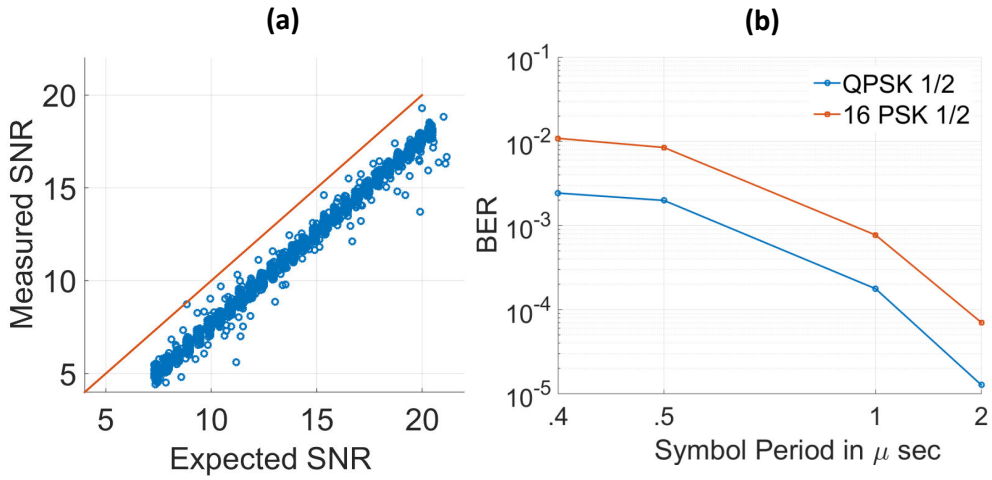

Figure 5.11: **(a)**Demonstrates the effect of imperfect cancellation on the degradation of the measured SNR vs the expected SNR at the reader of BackFi. When the cancellation is imperfect the environmental components are no longer completely removed and this acts as interference to the backscatter signal from the tag.**(b)** Demonstrates the diversity gains of MRC : as we increase the symbol time period, we have more samples for averaging, hence it improves the SNR. This increase in SNR results in lower bit error rate (BER) for a given modulation.

components: exploiting the larger symbol times from the tag packet to make an approximation that the channel can be converted into a simplified time invariant system and then apply MRC to solve it. MRC helps amplify the SNR of the signal by combining signal energy across time appropriately. Hence the key factor here is the tag symbol period which is inversely proportional to the tag symbol rate. To show the impact we plot the BER vs tag symbol rate for two modulations and a fixed coding rate of 1/2. The expectation is that as the tag symbol rate decreases and symbol time increases, the MRC gain will drive the BER down like a waterfall curve. Fig. 5.11(b) plots the BER as a function of decreasing tag symbol rate. As we can see, for this particular placement of AP and tag, at the highest tag symbol rate the BER is high between $10^{-2} - 10^{-3}$. As tag symbol rate decreases, the time diversity gain from MRC kicks in and BER drops down to between $10^{-4} - 10^{-5}$. This technique essentially points out the trade-off between throughput and range and why it exists.

5.6.3 Performance in typical WiFi Networks

BackFi tags only backscatter data when the WiFi reader is transmitting and they are activated by the reader with the activation sequence. The best candidate for the WiFi reader device is clearly the AP since it is likely the most dominant transmitter in a typical network. Nevertheless, in a typical network that is fully loaded (i.e. there is always outstanding traffic to transmit from the AP or a client), the AP will be transmitting a fraction of the time which would imply that the BackFi link would also be active for the same fraction. We evaluate the throughput BackFi can provide under such typical network conditions.

To conduct this experiment, we took traces from open source data [65, 113, 105]. The traces are

captured for a wide variety of scenarios for heavily loaded networks. If an AP is not loaded and there is a lot of idle channel time, then a BackFi AP can initiate backscatter communication anyway by sending dummy packets just for that purpose. Hence the interesting case is when the network is loaded and backscatter opportunities are limited due to contention.

Next, we filter the traces to only contain AP transmissions and replay the collected trace using our WARP based BackFi AP implementation to simulate the same traffic conditions. In other words, in our emulated experiment the WARP radio only transmits when the corresponding AP transmitted in the collected trace. We place a BackFi tag at a fixed distance of 2 m from the BackFi AP. We also activate the tag only at the times the AP is transmitting. We repeat this experiment for each AP we captured traces for, a total of 20 different APs. For each replay, we calculate the average throughput obtained by the BackFi link. Fig. 5.12(a) plots the CDF of these throughputs.

As we can see, in a loaded network, the BackFi link can obtain a median throughput of 4 Mbps. For a range of 2 m, the optimal throughput when the BackFi AP is continuously transmitting an excitation signal is 5 Mbps, hence this amounts 80% of the optimal throughput under realistic WiFi network conditions. The above number can be improved further if more WiFi devices have BackFi functionality. Specifically the above experiment assumed that only the AP has BackFi functionality. However if we can integrate the same into our laptops and smart-phones and turn them into gateways for BackFi links, then the BackFi link can be active for larger fractions of time.

5.6.4 Impact on the WiFi Network

A natural question is whether BackFi affects the performance of the WiFi network itself when the AP is doubling up as a WiFi backscatter reader. Specifically, one might imagine that the tags backscatter signals could propagate to the actual WiFi client which is the destination of the transmission from the AP and act as interference.

To quantify what impact one might see in a general WiFi network, we place the BackFi AP and ten clients at random locations in the testbed. Next we place the tag at increasing distances from the AP and calculate the WiFi throughput with and without an active BackFi tag. We repeat this experiment for 30 different configurations of the AP and the clients. We plot the throughputs with and without an active BackFi tag for different ranges between the BackFi AP and the tag in Fig. 5.12(b). The results confirm the previous benchmark, essentially when the tag and the AP reader are extremely close (between $0.25 - 0.5$m), there is a small impact on network throughput of less than 10%. Otherwise the normal WiFi network performance is negligibly affected since the backscatter signals are so weak.

5.6.5 Micro-benchmark Impact on WiFi

We now evaluate the worst case scenario for the WiFi client. This corresponds to the case where the tag is very close to the AP (at a distance of 0.25m) because in this case the backscatter signals

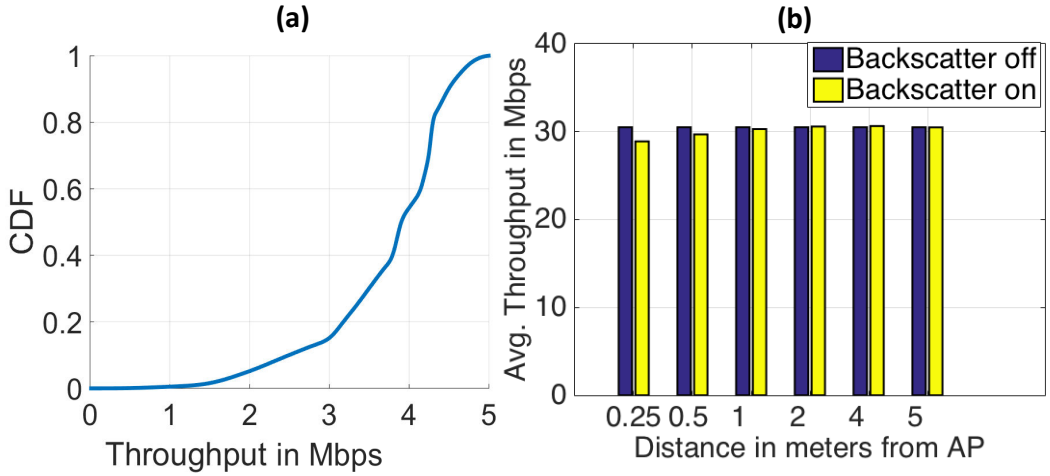

Figure 5.12: WiFi Deployments: **(a)** Throughput of BackFi's tag at a distance of 1m from the BackFi's reader under normal WiFi deployment. Note that BackFi tag is active only when the BackFi's reader is transmitting. Hence we achieve on an average 4 Mbps throughput vs the maximum throughput of 5Mbps. **(b)** Average throughput for all the clients at different locations as a function of distance of tag from the AP. As the tag moves away from the AP, it receives and radiates a smaller signal which will have smaller effect at the client. Hence, when the tag is at 0.25 m, we see a 10% throughput drop when tag is modulating. As the tag moves away from AP, we see no degradation in the average throughput.

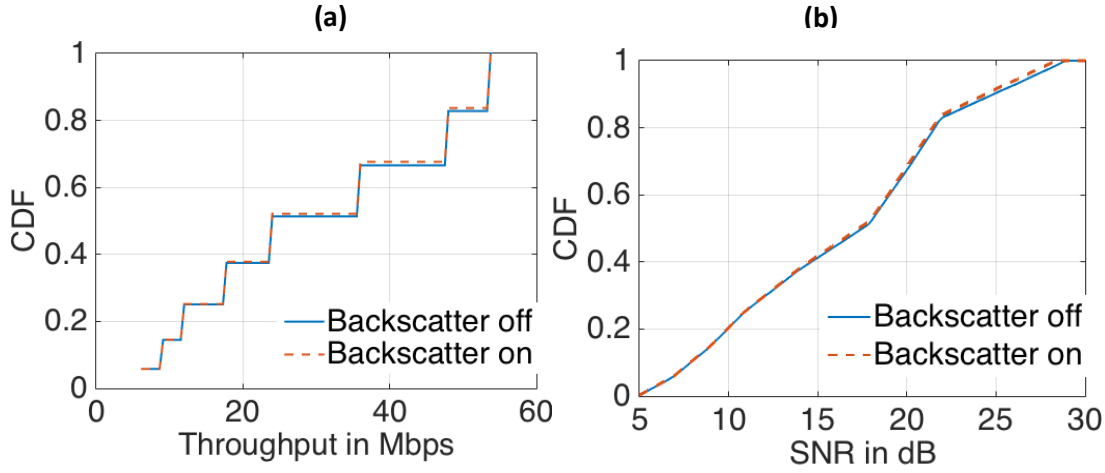

Figure 5.13: (a) Shows the CDF of the client throughput when the tag is placed at 0.25m from the AP. As seen, there is almost no degradation for lower bit rate of 6 Mbps, as client is farther from AP and the SNR required at the client to decode 6 Mbps is small. However, we observe noticeable difference at 54 Mbps, where both clients are closer to BackFi's AP and need higher SNR to decode data. (b) shows the degradation of SNR for tag on and tag off for each point for the plot on the left.

would be the strongest. Next we take a single WiFi client and place it at different distances so that we achieve each of the different rates of WiFi. Now for each WiFi bitrate, we evaluate the PHY layer throughput achieved with and without a BackFi tag being active. Fig. 5.13.a plots the CDF

of WiFi throughputs achieved for this link with and without a BackFi tag active. As we can see, the effect is minimal. The only case where there is a noticeable difference is when the WiFi AP and the client are using the highest bitrate of 54 Mbps where small decreases in SNR (as shown in Fig. 5.13.b) can force the WiFi AP to occasionally switch to lower bitrates.

Chapter 6

Discussion & Conclusion

In this thesis, we have presented full duplex MIMO radios, FastForward relay and BackFi backscatter communication system. This chapter would summarize the limitation and future work.

6.1 Full Duplex Radios

We presented the design and implementation of the first single antenna full duplex MIMO 2.4 GHz WiFi-PHY i.e. to achieve an m-chain MIMO transceiver we need only m antenna, as compared to prior needing $2 \times m$ antenna. We design novel cancellation algorithms and circuits that reduce all self-interference to the noise floor and enable full duplex MIMO PHY with almost no loss. The cancellation algorithms themselves are of independent interest and apply to many other interference problems in wireless, including coexistence [67], transmitting and receiving on arbitrary adjacent bands [66] etc. Below we discuss the current design's limitations, potential avenues of future work and then conclude.

Size of circuit: The current analog cancellation circuit is large, our prototype board measures 10×10 cm. Such a design is fine for APs and base-stations which is our initial focus, however this design is not implementable on phones and other portable devices where size is at a premium. To realize full duplex on such devices, we need to design an RFIC that is sufficiently small (at best $20 - 30$sq.mm for current phones). The key consumers of space on our circuit are delay lines, which we currently realize via traces on the board. For an RFIC we expect to use different techniques to realize the same delays, such as LC ladders and acoustic technologies such as SAW and BAW [80]. These techniques operate by slowing the speed of light, and thus true time delays are obtained in very short form factors that can be integrated on chip. However, the above discussion is speculative and is part of our future work.

LTE: Our current prototype targets WiFi frequencies in the 2.4GHz band. However our prototype can also be used for the 2.3GHz and 2.5GHz LTE bands found in Asia and Europe. However the

general design of our system is frequency independent, the dependence in our prototype comes from the fact that several analog components in our cancellation board work only in specific frequency ranges (our tunable attenuators and phase shifters operate only between 2-2.6GHz). However, the same design can be used for different frequencies with corresponding components that work in those frequency ranges. Further, unlike WiFi, LTE uses smaller channels, the widest channel is 20MHz and this makes the cancellation problem somewhat simpler. Hence we believe our current design can be adapted to work with LTE, and this remains future work.

Bandwidth: We have experimentally shown that the current design supports full duplex upto 80 MHz bandwidth. The more general observation is that if we can get 60dB of cancellation in analog, then we are almost always able to cancel the rest of the self-interference in digital. Hence the bottleneck in terms of bandwidth is analog cancellation. We believe our current design can support the widest WiFi bandwidth out there (160MHz), but at this point we have no way of testing it since we don't have software defined radio receivers that can sample 160MHz.

Antenna & Reflection: The current design works with a class of antennas, whose reflections (technically known as S_{11}) are less than 15dB, and has a smooth phase response that does not vary too abruptly. However certain directional antennas may not fit these criteria, making our design antenna independent is necessary for a infrastructure product like base-station. However, our simulations shows that we can build analog cancellation for high gain antennas as well.

Transmission Power: Our current design targets and works for power levels typically found in WiFi APs and LTE small cells (upto 30dBm TX power). Macro base-stations however use larger powers upto 50dBm, and our design doesn't yet work for such high power levels. The main challenge is transmitter noise (includes broadband noise) which is quite high when power amplifiers are being asked to output 100W of power. Some preliminary analysis suggests that for such power levels we would need nearly 90dB of analog cancellation itself to eliminate transmitter noise. Achieving such larger cancellation may be feasible if we can use more delay taps, which might be fine for macro base-stations since space is typically not a constraint.

6.2 FastForward Relay

FF relay presented has significant benefits. If given area to deploy WiFi network, which is large enough that one AP cannot cover the area (assuming we have single WiFi channel only to cover this area). However, 2 AP or the AP + FF relay can provide coverage, we expect that AP + FF relay would work better. The major benefit which isn't clearly demonstrated is that placing 2 AP cause more of interference in the network, collision of packets. Further, it increases the contention. However, in the case of FF relay as it wont participate for the channel contention, avoid packet

collision within the network. The FF relay allows AP to go to a centralized network.

ACK Improvement: ACK are crucial to improving the throughput in wireless system, as they provide end to end throughput improvement. The interesting feature here is that ACK are immediate packets after the transmitted packet so source and destination are know, which allows us to easily apply construct and forward. Since we can learn from last packet who is source and destination, and apply the correct construct and forward filter. Thus we believe ACK can relayed much more reliable than other packets. Further, another feature of modern WiFi standard, frame aggregation can also be supported as long as all the packets are headed to the same destination. This is due to the fact that we cannot apply multiple construct-and-forward filters on the same packet, as it would harm the channel estimation significantly. However, practically frames (MPDU and MSDU) [15], are combined for the same destination only. Further, construct and forward rely on three channels to maintain channel coherence while ongoing MPDU transmission. However, we expect that even though channel may change, the FF relay can still provide significant gains as the channel does not entirely decorrelate.

Standard Modification: Having a unique signature for every source and destination pair at the very beginning to the transmission of a packet (radio preamble) and the channel from source to destination as well would make this relay work seamlessly.

Achieving the transceiver Loopback with low latency: The ADC and DAC itself can have a lot of pipeline latency, to support different interpolation or downsampling mode, building a low latency transceiver needs ADC and DAC for single sampling rate, further increasing the clock rate can allow achieve very low latency, however this design would be expensive as well on account high sampling rate ADC and DAC. A slightly cheaper FF relay can be designed using analog only relay. A simple loop back analog only relay can be achieved which consist of analog only. We build analog only relay which has analog construct and forward and a analog switch [18] which can be used to switch of the output of the power amplifier. The maximum amplification used is dependent on the cancellation achieved in this design. We can achieve 60 dB of amplification to this analog relay, whenever relay determines to amplify.

Extra Power: The total power at the disposal of AP, FF relay and half duplex mesh relay is same i.e. 20 DBMS. However FF relay only uses max power for 10-15 % depending upon the environment. In these 10%, cases we use extra power per time slot as compared to the mesh relay.

Control Algorithm: The relay choice of whether to relay a packet or not. Whether to relay a packet from other network. These decision are the part of control algorithm. The Medium access control makes the control algorithm to better manage the interference in the network.

Relay placement: Much of the gains are dependent upon the placement of relay. The placement of relay is determined by the coverage area. The idea is to simulate the coverage area in [84], to provide us with channel information for each place in the coverage area. These channels can be used to decide a good placement for a given coverage area.

6.3 BackFi

Trade-off for 900 MHz BackFi AP or an RFID reader:
900 MHz band for RFID is an ISM band, hence both WiFi and RFID can use it. In light of this, using BackFi provides two benefits:

- Coexistence: BackFi can be deployed on any ISM band channels (900 MHz, 2.4 GHz or 5 GHz). However, many wireless systems including WiFi, Bluetooth, Zigbee, and cordless cellphones share these unlicensed ISM band channels. However, BackFi can co-exist and can be deployed seamlessly without affecting any of these systems. This is because the traditional wireless systems would perceive the weak backscatter signal as small noise compared to the relatively much stronger communication signal between the wireless devices, as demonstrated in Sec. 5.6.4. BackFi does not require any PHY or MAC layer modifications to the WiFi signaling and all the legacy WiFi devices can work without any modifications when BackFi is operating. Moreover, this also means that BackFi can be incrementally deployed in an environment where other WiFi APs have already been deployed.
- Coexistence: BackFi can be deployed on any ISM band channels (900MHz, 2.4GHz or 5GHz), will support both standard WiFi clients and IoT sensor. However, other systems also share the same ISM band channels as these are unlicensed bands (WiFi, Bluetooth, Zigbee, cordless cellphone etc), BackFi can piggyback on each of these signaling. Among many wireless systems that are deployed in these bands WiFi is by far the most ubiquitous. Therefore, we have demonstrated BackFi to seamlessly co-exist with all the WiFi devices and systems. BackFi achieves this coexistence by piggybacking on the backscatter resulting from the WiFi transmissions. BackFi requires neither PHY changes not MAC changes on the WiFi packets used for tag interrogation, therefore all the legacy WiFi devices can work without any modifications when BackFi is operating. Moreover, BackFi APs can co-exist without any modifications with other legacy WiFi APs. This means that BackFi can be incrementally deployed in an environment where other WiFi APs have already been deployed.
- Spectral efficiency : In addition to co-existing with legacy wireless devices, BackFi also achieves spectral efficiency by allowing the backscatter communication from tag to happen simultaneously while the WiFi packets are being delivered to a standard WiFi client on the same frequency band.

Integration of traditional RFID with WiFi AP violates either one or both of these two motivations of BackFi.

Power and Cost of IoT sensor: The actual cost of the IoT sensor depends on the implementation platform. Ultimately for very large volume such IoT sensor can be manufactured using sub-micron CMOS technology node where the total cost per tag could cost only few cents. For example, a SPDT switch is two transistor in CMOS, and array of 64 SPDT would be 128 transistor.

In 45 nm technology, the cost per 1000 transistors is a 1 cent for a million units, which is small. Absolute power consumed by the tag also depends on the implementation platform (e.g. 45 nm CMOS, 90 nm CMOS, discrete components, FPGA, MCU) and design techniques (e.g. sub-threshold design for CMOS) . For example, if we implement the IoT sensor using off-the-shelf discrete components the power consumption will be order of magnitude higher compared to cases when we implement the same tag using sub-micron CMOS technology nodes. On the other hand the power consumption, also depends on design techniques. For example, since the tag needs to operate at only up to 10s of MHz , it can be designed using sub-threshold design methodology which can further have 10x reduction in power consumption compared to traditional CMOS design techniques. The focus of this chapter is on systems design for the WiFi backscatter reader that can support multiple modulation index. However, each modulation and coding index has different energy and data rate associated with them. To offset the variation in power consumption based on implementation platform, we will present the energy-datarate tradeoff of BackFi using a relative EPB as the metric. In doing so we will compute EPB for one particular modulation using an exemplary implementation platform, all the other EPB will be computed relative to the EPB of this reference design. This way we can present BackFi's ability to tradeoff datarate-vs-EPB without having to talk about specific implementation detail of the tag.

Supporting multiple tags : BackFi presents a design which achieves a Mbps throughput at 5 meters. However, in some application scenarios like data upload from wearable sensors, there may be a need for data rates on the order of 100 Kbps, and there may be 10 such sensors. Hence we need a link of Mbps, but well distributed within the multiple tags. BackFi demonstrates the feasibility of such a link and exploring the MAC protocol to support multiple tags is part of future work.

MIMO BackFi AP : Multiple antennas at WiFi APs is becoming increasingly common to support higher data rates. We can infact use multiple antennas at the AP to our advantage to increase both the range and throughput of BackFi. Assuming a single antenna on the tag side, MIMO would provide us with diversity combining gain. Further, we can incorporate multiple antennas at the AP with minimal changes in BackFi's link layer protocol. Specifically, each transmit antenna would need a silent slot to eliminate the environmental component corresponding to that antenna. We could exploit existing WiFi MIMO packet structure for estimating the environment as it has slots where only one transmit antenna is active. At this point, we can perform MRC combining for the signals received across time from multiple WiFi packets and received across space from multiple antennas, providing BackFi with better SNR.

6.4 Summary

In summary, self-interference cancellation allows us to use radios in an entirely different fashion. Wireless radios can be transformed into a wireless camera, where each reflections received at the

WiFi AP as seen in the Fig. 6.1 is characterized by the amplitude, the angle of arrival, or the time of flight represented by α, θ, τ respectively. An example representation is shown in Fig. 6.1, the time of flight and angle of arrival and amplitude encoded in the color. Thus, wireless self-interferometry provides representation in $\alpha_i, \theta_i, \tau_i$ for i^{th} reflector. Using the representation of α, θ, τ, we can design the algorithm to achieve each of different applications. For example, human motion tracing or data from IoT sensors, many other applications.

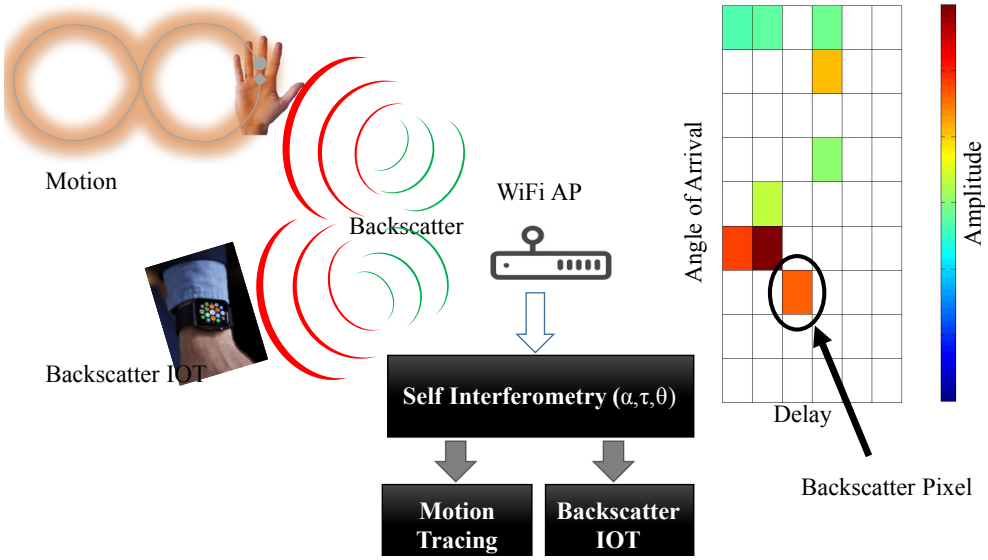

Figure 6.1: Shows a WiFi AP that transmits a red signal and the reflections are received as green signal back to the AP, which upon inference is represented by the figure on the right with time of flight, angle of arrival and amplitude of the reflection. This abstraction is referred as self-interferometry, which can be used as input to achieve applications as human motion tracing.

This abstraction to pixels for reflections is future work, we have achieved this at a lower update interval. This abstraction allows us to achieve [72]. However to achieve BackFi with this abstraction we need pixels to be generated more frequently, which is the current limitation of this work. Extending this framework to faster update interval is future work.

Bibliography

[1] *802.11ac: The Fifth Generation of Wi-Fi*. http://www.cisco.com/en/US/prod/collateral/wireless/ps5678/ps11983/white_paper_c11-713103.pdf.

[2] *Analog Devices ADG904 RF SP4T*. http://www.analog.com/media/en/technical-documentation/data-sheets/ADG904_904R.pdf.

[3] *Cypress Semiconductor CY62146EV30 SRAM*. http://www.cypress.com/?docID=48695.

[4] *Datasheet ADC-ADS5400*. http://www.ti.com/lit/ds/slas611b/slas611b.pdf.

[5] *Datasheet DAC-3162*. http://www.ti.com/lit/ds/symlink/dac3162.pdf.

[6] *Datasheet DAC 5681z*. http://www.ti.com/lit/ds/symlink/dac5681z.pdf.

[7] *Datasheet of LTM9004*. http://cds.linear.com/docs/en/datasheet/9004fa.pdf.

[8] *Decoupling of Static Channel*. http://snsg.stanford.edu/fullduplexmimo.pdf.

[9] *EPC Class-1 Gen-2 UHF RFID*. http://www.gs1.org/gsmp/kc/epcglobal/uhfc1g2/uhfc1g2_1_2_0-standard-20080511.pdf.

[10] *FCC Part 15.247 Rules Systems Using Digital Modulation*. http://www.semtech.com/images/datasheet/fcc_digital_modulation_systems_semtech.pdf.

[11] *FMC110- a dual channel ADC and dual channel DAC*. http://www.4dsp.com/pdf/FMC110_user_manual.pdf.

[12] *Huawei Sets Out Its 5G Stall*. http://www.lightreading.com/document.asp?doc_id=703466&init_gateway=true.

[13] *Kumu Networks funding*. https://www.crunchbase.com/organization/kumu-networks#/entity.

[14] *LTE Advanced Speeds*. http://en.wikipedia.org/wiki/4G#LTE_Advanced.

[15] *Meraki White Paper: 802.11n Technology.* https://meraki.cisco.com/lib/pdf/meraki_whitepaper_802_11n.pdf.

[16] *Modeling Indoor Propagation.* http://www.remcom.com/examples/modeling-indoor-propagation.html.

[17] *Next Generation Gigabit WiFi - 802.11ac.* http://www.netgear.com/landing/80211ac/images/wp_netgear_802_11ac_wifi.pdf.

[18] *PE 4251 Data-sheet.* www.psemi.com/pdf/datasheets/pe4251ds.pdf.

[19] *PE 47303 Data-sheet.* http://www.psemi.com/pdf/datasheets/pe43703ds.pdf.

[20] *Physical layer procedures(FDD).* http://www.qtc.jp/3GPP/Specs/25214-890.pdf.

[21] *Power Amplifier Data-sheet.* http://www.minicircuits.com/pages/npa/PGA-105+_NPA.pdf.

[22] *Power Amplifier Data-sheet.* http://datasheets.maximintegrated.com/en/ds/MAX2828-MAX2829.pdfi.

[23] *Sequential Convex Programming.* http://www.stanford.edu/class/ee364b/lectures/seq_slides.pdf.

[24] *ThingMagic. Mercury6e rfid reader module.* http://www.thingmagic.com/embedded-rfid-readers.

[25] *US Patent 5444864.* http://www.google.com/patents/US5444864.

[26] *US Patent 6539204.* http://www.google.com/patents/US6539204.

[27] *User Guide: KC705 Evaluation Board for the Kintex-7 FPGA.* http://www.xilinx.com/support/documentation/boards_and_kits/kc705/ug810_KC705_Eval_Bd.pdf.

[28] *WARP Project.* http://warpproject.org.

[29] *White paper by NI on Understanding Dynamic Hardware Specifications.* http://www.ni.com/white-paper/5529/en.

[30] *Xilinx 7 Series FPGA Overview.* http://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf.

[31] *Xilinx Power Estimator Tool.* http://www.xilinx.com/products/design_tools/logic_design/xpe.htm.

[32] P. Almers, F. Tufvesson, and A.F. Molisch. Keyhole effect in mimo wireless channels: Measurements and theory. *Wireless Communications, IEEE Transactions on*, 5(12):3596–3604, December 2006.

[33] Efram Burlingame And. An analog cmos high-speed continuous-time fir filter, 2000.

[34] Ehsan Aryafar, Mohammad Amir Khojastepour, Karthikeyan Sundaresan, Sampath Rangarajan, and Mung Chiang. Midu: enabling mimo full duplex. In *Proceedings of the 18th annual international conference on Mobile computing and networking*, Mobicom '12, pages 257–268, New York, NY, USA, 2012. ACM.

[35] Joseph Bardwell. *Tech Report.* `http://www.connect802.com/download/techpubs/2005/commercial_radios_E0523-15.pdf`.

[36] Dinesh Bharadia, Kiran Raj Joshi, and Sachin Katti. Full duplex backscatter. In *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, page 4. ACM, 2013.

[37] Dinesh Bharadia, Kiran Raj Joshi, Manikanta Kotaru, and Sachin Katti. Backfi: High throughput wifi backscatter. *SIGCOMM Comput. Commun. Rev.*, 45(4):283–296, August 2015.

[38] Dinesh Bharadia and Sachin Katti. Fastforward: Fast and constructive full duplex relays. In *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, pages 199–210, New York, NY, USA, 2014. ACM.

[39] Dinesh Bharadia and Sachin Katti. Full duplex mimo radios. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14)*, pages 359–372, Seattle, WA, April 2014. USENIX Association.

[40] Dinesh Bharadia and Sachin Katti. Full duplex mimo radios. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14)*, pages 359–372, Seattle, WA, April 2014. USENIX Association.

[41] Dinesh Bharadia, Emily McMilin, and Sachin Katti. Full duplex radios. *SIGCOMM '13: To appear in the Proceedings of the ACM SIGCOMM 2013 conference*, 2013.

[42] D. W. Bliss, P. A. Parker, and A. R. Margetts. Simultaneous transmission and reception for improved wireless network performance. In *Proceeings of the 2007 IEEE Workshop on Statistical Signal Processing*, 2007.

[43] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[44] DG Brennan. Linear diversity combining techniques. *Proceedings of the IEEE*, 91(2):331–356, 2003.

[45] M. Buettner and D. Wetherall. A software radio-based uhf rfid reader for phy/mac experimentation. In *RFID (RFID), 2011 IEEE International Conference on*, pages 134–141, April 2011.

[46] Michael Buettner, Ben Greenstein, and David Wetherall. Dewdrop: An energy-aware runtime for computational rfid. In *Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation*, NSDI'11, pages 197–210, Berkeley, CA, USA, 2011. USENIX Association.

[47] A. Carleial. Multiple-access channels with different generalized feedback signals. *Information Theory, IEEE Transactions on*, 28(6):841–850, Nov 1982.

[48] Jung Il Choi, Mayank Jain, Kannan Srinivasan, Phil Levis, and Sachin Katti. Achieving single channel, full duplex wireless communication. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, MobiCom '10, pages 1–12, New York, NY, USA, 2010. ACM.

[49] T. Cover and A.E. Gamal. Capacity theorems for the relay channel. *Information Theory, IEEE Transactions on*, 25(5):572–584, Sep 1979.

[50] L. Ding. Digital predistortion of power amplifiers for wireless applications. 2004.

[51] Melissa Duarte, Chris Dick, and Ashutosh Sabharwal. Experiment-driven characterization of full-duplex wireless systems. *CoRR*, abs/1107.1276, 2011.

[52] Melissa Duarte and Ashutosh Sabharwal. Full-duplex wireless communications using off-the-shelf radios: Feasibility and first results. In *Forty-Fourth Asilomar Conference on Signals, Systems, and Components*, 2010.

[53] Melissa Duarte, Ayan Sengupta, Siddhartha Brahma, Christina Fragouli, and Suhas Diggavi. Quantize-map-forward (qmf) relaying: An experimental study. In *Proceedings of the Fourteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '13, pages 227–236, New York, NY, USA, 2013. ACM.

[54] M.S. Durante and S. Mahlknecht. An ultra low power wakeup receiver for wireless sensor nodes. In *Sensor Technologies and Applications, 2009. SENSORCOMM '09. Third International Conference on*, pages 167–170, June 2009.

[55] J.F. Ensworth and M.S. Reynolds. Every smart phone is a backscatter reader: Modulated backscatter compatibility with bluetooth 4.0 low energy (ble) devices. In *RFID (RFID), 2015 IEEE International Conference on*, pages 78–85, April 2015.

[56] E. Everett, M. Duarte, C. Dick, and A. Sabharwal. Empowering full-duplex wireless communication by exploiting directional diversity. In *Signals, Systems and Computers (ASILOMAR), 2011 Conference Record of the Forty Fifth Asilomar Conference on*, pages 2002 –2006, nov. 2011.

[57] Evan Everett, Achaleshwar Sahai, and Ashutosh Sabharwal. Passive self-interference suppression for full-duplex infrastructure nodes. *CoRR*, abs/1302.2185, 2013.

[58] E.C. Fear, S.C. Hagness, P.M. Meaney, M. Okoniewski, and M.A. Stuchly. Enhancing breast tumor detection with near-field imaging. *Microwave Magazine, IEEE*, 3(1):48–56, 2002.

[59] Andrea Goldsmith. *Wireless Communications*. Cambridge University Press, New York, NY, USA, 2005.

[60] Shyamnath Gollakota, Haitham Hassanieh, Benjamin Ransford, Dina Katabi, and Kevin Fu. They can hear your heartbeats: non-invasive security for implantable medical devices. *SIGCOMM Comput. Commun. Rev.*, 41(4), August 2011.

[61] Kenneth Gudan, Sergey Chemishkian, Jonathan J. Hull, Matthew S. Reynolds, and Stewart Thomas. Feasibility of wireless sensors using ambient 2.4ghz rf energy.

[62] Jeremy Gummeson, Shane S. Clark, Kevin Fu, and Deepak Ganesan. On the limits of effective hybrid micro-energy harvesting on mobile crfid sensors. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 195–208, New York, NY, USA, 2010. ACM.

[63] Jeremy Gummeson, Pengyu Zhang, and Deepak Ganesan. Flit: A bulk transmission protocol for rfid-scale sensors. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, MobiSys '12, pages 71–84, New York, NY, USA, 2012. ACM.

[64] D. Gunduz, A. Goldsmith, and H.V. Poor. Mimo two-way relay channel: Diversity-multiplexing tradeoff analysis. In *Signals, Systems and Computers, 2008 42nd Asilomar Conference on*, pages 1474–1478, Oct 2008.

[65] Arpit Gupta, Jeongki Min, and Injong Rhee. Wifox: Scaling wifi performance for large audience environments. In *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '12, pages 217–228, New York, NY, USA, 2012. ACM.

[66] Steven S. Hong, Jeffrey Mehlman, and Sachin Katti. Picasso: flexible rf and spectrum slicing. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, SIGCOMM '12, pages 37–48, New York, NY, USA, 2012. ACM.

[67] Steven Siying Hong and Sachin Rajsekhar Katti. Dof: a local wireless information plane. In *SIGCOMM '11: Proceedings of the ACM SIGCOMM 2011 conference*, pages 230–241, New York, NY, USA, 2011. ACM.

[68] Xuemin Hong, Cheng-Xiang Wang, J. Thompson, B. Allen, W.Q. Malik, and Xiaohu Ge. On space-frequency correlation of uwb mimo channels. *Vehicular Technology, IEEE Transactions on*, 59(9):4201–4213, Nov 2010.

[69] Yingbo Hua, Ping Liang, Yiming Ma, A.C. Cirik, and Qian Gao. A method for broadband full-duplex mimo radio. *Signal Processing Letters, IEEE*, 19(12):793 –796, dec. 2012.

[70] H. Ishizaki, H. Ikeda, Y. Yoshida, Tadashi Maeda, T. Kuroda, and M. Mizuno. A battery-less wifi-ber modulated data transmitter with ambient radio-wave energy harvesting. In *VLSI Circuits (VLSIC), 2011 Symposium on*, pages 162–163, June 2011.

[71] Mayank Jain, Jung Il Choi, Taemin Kim, Dinesh Bharadia, Siddharth Seth, Kannan Srinivasan, Philip Levis, Sachin Katti, and Prasun Sinha. Practical, real-time, full duplex wireless. MobiCom '11, pages 301–312, New York, NY, USA, 2011. ACM.

[72] Kiran Joshi, Dinesh Bharadia, Manikanta Kotaru, and Sachin Katti. Wideo: Fine-grained device-free motion tracing using rf backscatter. In *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*, pages 189–204, 2015.

[73] Sung-Chan Jung, Min-Su Kim, and Youngoo Yang. A reconfigurable carrier leakage canceler for uhf rfid reader front-ends. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 58(1):70 –76, jan. 2011.

[74] Bryce Kellogg, Aaron Parks, Shyamnath Gollakota, Joshua R. Smith, and David Wetherall. Wi-fi backscatter: Internet connectivity for rf-powered devices. In *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, pages 607–618, New York, NY, USA, 2014. ACM.

[75] J.P. Kermoal, L. Schumacher, K.I. Pedersen, P.E. Mogensen, and F. Frederiksen. A stochastic mimo radio channel model with experimental validation. *Selected Areas in Communications, IEEE Journal on*, 20(6):1211–1226, Aug 2002.

[76] Vu Van Khang and Nguyen Dinh Thong. Rank-deficiency in indoor mimo. In *TENCON 2007 - 2007 IEEE Region 10 Conference*, pages 1–4, Oct 2007.

[77] M.E. Knox. Single antenna full duplex communications using a common carrier. In *Wireless and Microwave Technology Conference (WAMICON), 2012 IEEE 13th Annual*, pages 1–6, 2012.

[78] T.I. Laakso, V. Valimaki, M. Karjalainen, and U.K. Laine. Splitting the unit delay [fir/all pass filters design]. *Signal Processing Magazine, IEEE*, 13(1):30–60, Jan 1996.

[79] Si-Hyeon Lee and Sae-Young Chung. When is compress-and-forward optimal? In *Information Theory and Applications Workshop (ITA), 2010*, pages 1–3, Jan 2010.

[80] T.H. Lee. *The Design of CMOS Radio-Frequency Integrated Circuits*. Cambridge University Press, 2004.

[81] Xu Duan Lin and Kyung Hi Chang. Optimal pn sequence design for quasisynchronous cdma communication systems. *Communications, IEEE Transactions on*, 45(2):221–226, 1997.

[82] Vincent Liu, Aaron Parks, Vamsi Talla, Shyamnath Gollakota, David Wetherall, and Joshua R. Smith. Ambient backscatter: Wireless communication out of thin air. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, SIGCOMM '13, pages 39–50, New York, NY, USA, 2013. ACM.

[83] A. Lo and Peng Guan. Performance of in-band full-duplex amplify-and-forward and decode-and-forward relays with spatial diversity for next-generation wireless broadband. In *Information Networking (ICOIN), 2011 International Conference on*, pages 290–294, Jan 2011.

[84] P. Mededovic, M. Veletic, and Z. Blagojevic. Wireless insite software verification via analysis and comparison of simulation and measurement results. In *MIPRO, 2012 Proceedings of the 35th International Convention*, pages 776–781, May 2012.

[85] Edward C. Van Der Meulen. Three-terminal communication channels. *Advances in Applied Probability*, 3(1):pp. 120–154, 1971.

[86] C. Mikeka, H. Arai, A. Georgiadis, and A. Collado. Dtv band micropower rf energy-harvesting circuit architecture and performance analysis. In *RFID-Technologies and Applications (RFID-TA), 2011 IEEE International Conference on*, pages 561–567, Sept 2011.

[87] D.R. Morgan, Zhengxiang Ma, Jaehyeong Kim, M.G. Zierdt, and J. Pastalan. A generalized memory polynomial model for digital predistortion of rf power amplifiers. *Signal Processing, IEEE Transactions on*, 54(10):3852–3860, Oct 2006.

[88] Patrick Murphy and Ashutosh Sabharwal. Design, implementation and characterization of a cooperative communications system. *CoRR*, abs/1102.0485, 2011.

[89] P.V. Nikitin and K.V.S. Rao. Theory and measurement of backscattering from rfid tags. *Antennas and Propagation Magazine, IEEE*, 48(6):212–218, Dec 2006.

[90] Seunghyun Oh, N.E. Roberts, and D.D. Wentzloff. A 116nw multi-band wake-up receiver with 31-bit correlator and interference rejection. In *Custom Integrated Circuits Conference (CICC), 2013 IEEE*, pages 1–4, Sept 2013.

[91] U. Olgun, C.-C. Chen, and J.L. Volakis. Design of an efficient ambient wifi energy harvesting system. *Microwaves, Antennas Propagation, IET*, 6(11):1200–1206, August 2012.

[92] Alan V. Oppenheim, Ronald W. Schafer, and John R. Buck. *Discrete-time signal processing (2nd ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1999.

[93] A. Ozgur and S. Diggavi. Approximately achieving gaussian relay network capacity with lattice codes. In *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on*, pages 669–673, June 2010.

[94] Aaron N. Parks, Angli Liu, Shyamnath Gollakota, and Joshua R. Smith. Turbocharging ambient backscatter communication. In *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, pages 619–630, New York, NY, USA, 2014. ACM.

[95] A.N. Parks and J.R. Smith. Sifting through the airwaves: Efficient and scalable multiband rf harvesting. In *RFID (IEEE RFID), 2014 IEEE International Conference on*, pages 74–81, April 2014.

[96] Shyamal Patel, Hyung Park, Paolo Bonato, Leighton Chan, and Mary Rodgers. A review of wearable sensors and systems with application in rehabilitation. *Journal of neuroengineering and rehabilitation*, 9(1):21, 2012.

[97] Nathan Pletcher and Jan M. Rabaey. *Ultra-Low Power Wake-Up Receivers for Wireless Sensor Networks*. PhD thesis, EECS Department, University of California, Berkeley, May 2008.

[98] N.M. Pletcher, S. Gambini, and J. Rabaey. A 52 micro w wake-up receiver with - 72 dbm sensitivity using an uncertain-if architecture. *Solid-State Circuits, IEEE Journal of*, 44(1):269–280, Jan 2009.

[99] R. Porat, E. Ojard, N. Jindal, M. Fischer, and V. Erceg. Improved mu-mimo performance for future 802.11 systems using differential feedback. In *Information Theory and Applications Workshop (ITA), 2013*, pages 1–5, Feb 2013.

[100] J.G. Proakis. *Digital Communications*. McGraw-Hill Series in Electrical and Computer Engineering. Computer Engineering. McGraw-Hill, 2001.

[101] Bozidar Radunovic, Dinan Gunawardena, Peter Key, Alexandre Proutiere, Nikhil Singh, Vlad Balan, and Gerald Dejean. Rethinking indoor wireless mesh design: Low power, low frequency, full-duplex. In *Wireless Mesh Networks (WIMESH 2010), 2010 Fifth IEEE Workshop on*, pages 1 –6, 2010.

[102] Bozidar Radunovic, Dinan Gunawardena, Alexandre Proutiere, Nikhil Singh, Vlad Balan, and Peter Key. Efficiency and fairness in distributed wireless networks through self-interference cancellation and scheduling. Technical Report MSR-TR-2009-27, Microsoft Research, 2009.

[103] B. Rankov and A. Wittneben. Achievable rate regions for the two-way relay channel. In *Information Theory, 2006 IEEE International Symposium on*, pages 1668–1672, July 2006.

[104] N.E. Roberts and D.D. Wentzloff. A 98nw wake-up radio for wireless body area networks. In *Radio Frequency Integrated Circuits Symposium (RFIC), 2012 IEEE*, pages 373–376, June 2012.

[105] Maya Rodrig, Charles Reis, Ratul Mahajan, David Wetherall, and John Zahorjan. Measurement-based characterization of 802.11 in a hotspot setting. In *Proceedings of the 2005 ACM SIGCOMM Workshop on Experimental Approaches to Wireless Network Design and Analysis*, E-WIND '05, pages 5–10, New York, NY, USA, 2005. ACM.

[106] Rohde & Schwarz. *Rohde & Schwarz FSW Signal and Spectrum Analyzer User Manual*, 2012.

[107] Rohde & Schwarz. *Rohde & Schwarz SMBV 100A Signal Generator User Manual*, 2012.

[108] Rohde & Schwarz. *Rohde & Schwarz Vector Network Analyzer User Manual (ZNB8 4 port)*, 2012.

[109] Achaleshwar Sahai, Gaurav Patel, Chris Dick, and Ashutosh Sabharwal. On the impact of phase noise on active cancellation in wireless full-duplex. *CoRR*, abs/1212.5462, 2012.

[110] A. Sample and J.R. Smith. Experimental results with two wireless power transfer systems. In *Radio and Wireless Symposium, 2009. RWS '09. IEEE*, pages 16–18, Jan 2009.

[111] Alanson P Sample, Aaron N Parks, Scott Southwood, and Joshua R Smith. Wireless ambient radio power. In *Wirelessly Powered Sensor Networks and Computational RFID*, pages 223–234. Springer, 2013.

[112] A.P. Sample, D.J. Yeager, P.S. Powledge, and J.R. Smith. Design of a passively-powered, programmable sensing platform for uhf rfid systems. In *RFID, 2007. IEEE International Conference on*, pages 149–156, March 2007.

[113] Aaron Schulman, Dave Levin, and Neil Spring. On the fidelity of 802.11 packet traces. In *Proceedings of the 9th International Conference on Passive and Active Network Measurement*, PAM'08, pages 132–141, Berlin, Heidelberg, 2008. Springer-Verlag.

[114] Ryo Shigeta, Tatsuya Sasaki, Duong Minh Quan, Yoshihiro Kawahara, Rushi J Vyas, Manos M Tentzeris, and Tohru Asami. Ambient rf energy harvesting sensor device with capacitor-leakage-aware duty cycle control. *Sensors Journal, IEEE*, 13(8):2973–2983, 2013.

[115] Sébastien Simoens, Olga Muñoz Medina, Josep Vidal, and Aitor Del Coso. Compress-and-forward cooperative mimo relaying with full channel state information. *Trans. Sig. Proc.*, 58(2):781–791, February 2010.

[116] S.J. Thomas and M.S. Reynolds. A 96 mbit/sec, 15.5 pj/bit 16-qam modulator for uhf backscatter communication. In *RFID (RFID), 2012 IEEE International Conference on*, pages 185–190, April 2012.

[117] David Tse and Pramod Viswanath. *Fundamentals of wireless communication*. Cambridge university press, 2005.

[118] V. Vlimki and T. I. Laakso. Principles of fractional delay filters. In *PROCEEDINGS OF THE IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING*, pages 5–9, 2000.

[119] Jue Wang, Haitham Hassanieh, Dina Katabi, and Piotr Indyk. Efficient and reliable low-power backscatter networks. *SIGCOMM Comput. Commun. Rev.*, 42(4):61–72, August 2012.

[120] D.J. Yeager, P.S. Powledge, R. Prasad, D. Wetherall, and J.R. Smith. Wirelessly-charged uhf tags for sensor data collection. In *RFID, 2008 IEEE International Conference on*, pages 320–327, April 2008.

[121] Kai Yu, M. Bengtsson, B. Ottersten, D. McNamara, P. Karlsson, and M. Beach. Modeling of wide-band mimo radio channels based on nlos indoor measurements. *Vehicular Technology, IEEE Transactions on*, 53(3):655–665, May 2004.

[122] M. Yuksel and E. Erkip. Diversity-multiplexing tradeoff in cooperative wireless systems. In *Information Sciences and Systems, 2006 40th Annual Conference on*, pages 1062–1067, March 2006.

[123] M. Yuksel and E. Erkip. Multiple-antenna cooperative wireless systems: A diversity x2013;multiplexing tradeoff perspective. *Information Theory, IEEE Transactions on*, 53(10):3371–3393, Oct 2007.

[124] Hong Zhang, Jeremy Gummeson, Benjamin Ransford, and Kevin Fu. Moo: A batteryless computational rfid and sensing platform. 2011.

[125] Jinyun Zhang, P.V. Orlik, Z. Sahinoglu, A.F. Molisch, and P. Kinney. Uwb systems for wireless sensor networks. *Proceedings of the IEEE*, 97(2):313–331, Feb 2009.

[126] Pengyu Zhang and Deepak Ganesan. Enabling bit-by-bit backscatter communication in severe energy harvesting environments. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14)*, pages 345–357, Seattle, WA, April 2014. USENIX Association.

[127] Pengyu Zhang, Pan Hu, Vijay Pasikanti, and Deepak Ganesan. Ekhonet: High speed ultra low-power backscatter for next generation sensors. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, MobiCom '14, pages 557–568, New York, NY, USA, 2014. ACM.