Experimental Insights on Software-based Real-Time SI Cancellation for In-Band Full Duplex DF Relays

Muhammad Sohaib Amjad and Falko Dressler
Heinz Nixdorf Institute and Dept. of Computer Science, Paderborn University, Germany
{amjad,dressler}@ccs-labs.org

***Abstract*—In the last decade, infrastructure relays have been adopted by wireless standards, such as WiMAX and LTE, to substantially enhance the coverage and performance of wireless systems. These relays operate in a half-duplex mode, which not only increases the overall latency but also causes spectral losses. In contrast, an in-band full-duplex relay could effectively cope with these issues by simultaneously receiving packets from source and forwarding them towards destination. This becomes particularly interesting given the recent advancement in self-interference suppression techniques. In this work, we present an SDR-based real-time implementation of a full-duplex decode and forward relay in GNU Radio. Based on open-source and programmable hardware and software, the implementation is completely transparent and can be studied in all details and modified if needed. With an extensive set of experiments, we validate the practical performance of the proposed relay system, and measure achievable throughput gains. Our results demonstrate and underline the huge advantage of switching from the classical half-duplex relaying to full-duplex relay systems.**

I. Introduction

The degradation of a wireless signal due to the highly complex and unpredictable channel conditions has a significant impact on the performance of a wireless system. The decoding errors at the destination highly depend on the amount by which a Signal-of-Interest (SoI) is degraded while traveling from source to destination – this not only affects the data rate but also the coverage area of a wireless system. For instance, in a highly degrading wireless channel, we may need to reduce the coverage region of a wireless system (reducing the cell size) to maintain the higher data rates which means more equipment, or increase the coverage area at the cost of lower data rates along with the possible risk of losing the communication entirely. In recent years, to overcome this performance vs. coverage dilemma, infrastructure relays have been used and even incorporated in wireless standards like 3GPP LTE [1] and WiMAX [2], as they can greatly enhance the system performance and expand the coverage of a wireless network at the same time.

Nevertheless, these relays operate in Half-Duplex (HD) mode, which means they require additional resources typically in time domain for reliable communication. As illustrated in Figure 1, a standard two hop Half-Duplex Relay (HDR) with Time Division Duplexing (TDD) receives the data from a source in time slot T_α, and then waits to retransmit the data towards a destination in the next available time slot T_β, where the waiting time depends on the implemented relaying protocol. The deployment of such relays in a network not only causes spectral losses but also increases the end-to-end delay.

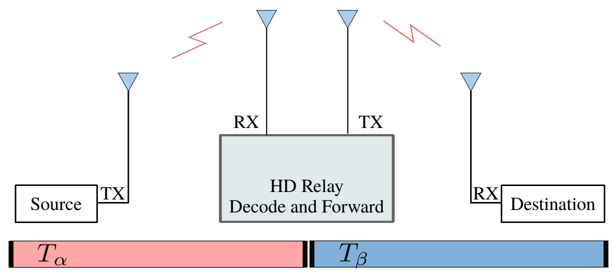

Figure 1. A typical two hop relay system operating in Half-Duplex mode.

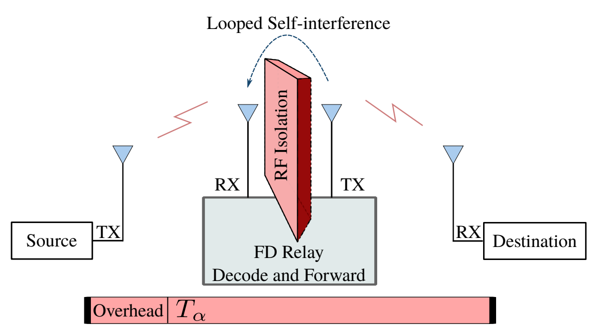

Figure 2. A two hop relay system operating in Full-Duplex mode.

In the past few years, a substantial amount of research has been done on in-band Full-Duplex (FD) wireless systems [3]. Several works [4]–[9], presented different techniques and architectures to address the prime factor impeding FD wireless communications, namely the Self-Interference (SI), which primarily arises due to radio's own transmission at the same time and frequency. As depicted in Figure 2, a Full-Duplex Relay (FDR) system can simultaneously receive from source and transmit towards destination. This not only improves the spectral efficiency of the relay system but also reduces the network latency considerably. In addition, depending on the implemented relaying scheme, there can be a marginal increase in the latency due to additional processing at the relay node. However, this is still significantly smaller compared to what HDR systems offer.

To achieve optimal performance with FDRs, the mitigation of Looped Self-Interference (LSI) is the fundamental requirement. For maximal diminution of LSI, usually both passive suppression and active cancellation techniques are employed. Any residual LSI basically reduces the Signal-to-Interference-plus-Noise Ratio (SINR) of SoI, which consequently degrades the overall system performance and decreases the achievable throughput gain.

In this work, we show a first implementation of a General Purpose Processor (GPP)-based Decode and Forward (DF)-FDR in GNU Radio[1] for use with Software Defined Radios (SDRs) as well as in simulation mode; and compare its practical performance with conventional half-duplex DF relays. The comparison studies both the Packet Delivery Ratio (PDR), and the achievable throughput gains. Our results demonstrate and underline the huge advantage of switching from the classical half-duplex relaying to full-duplex relay systems.

Our main contributions can be summarized as follows:

- We present a real-time Orthogonal Frequency Division Multiplexing (OFDM)-based Decode and Forward FDR implementation, which allows to monitor the real-time LSI cancellation in both time and frequency domains.
- We show that when LSI is fully suppressed, the throughput gain of FDR (including the overhead) is nearly twice compared to classical HDR systems.
- We investigate the impact of residual LSI in real-time on the FDR performance and throughput, the noise floor for SoI, and the transmit power requirement of the source.
- Our open-source software solution for FD relaying utilizes GNU Radio for signal processing. This makes the implementation accessible to fellow researchers and allows easy modifications for the testing of new concepts.

II. Related Work

In recent years, full-duplex relaying has been studied in great detail, after-all, the implications of such relaying systems are qualitatively beneficial in terms of both spectral efficiency and network latency. However, most of the research conducted in the domain have presented their analytical findings and considered theoretical approaches to state the gains of FD relaying. For instance, in [10], the authors considered an Amplify and Forward (AF) relaying system with low resolution Analog-to-Digital Converter (ADC); and did analytical modeling of LSI and quantization noise to analyze the achievable spectral efficiency. Similarly, in [11] an analytical model has been employed based on Markov chain modeling to analyze the outage probability in FD multi-relay channels. Likewise, in [12] the optimal power allocation in DF based FDRs to effectively handle the residual LSI has been discussed. Other such works include [13], where the RF impairment effects such as nonlinear behavior of power amplifier has been analyzed; and [14], in which the impact of looped-back channel estimation error on the performance of AF based FD relaying is studied.

In [15], a complete FDR design, implementation, and performance evaluation has been presented. The work introduced an intelligent class of AF relays and named it as Construct and Forward (CF) relaying, which unlike the naïve forwarding done by a typical AF relay, forwards the relayed signal in such a way that it constructively adds up with the direct signal (coming from source) at the destination. In order to work effectively, the constructive filter used at the relay node requires the Channel State Information (CSI) of all four paths, i.e., S-R, R-R, R-D, and S-D, which is a complex task. Also, the proposed design is still based on AF relaying, and although CF avoids noise amplification by efficiently choosing the amplification factor, but this also reduces the power levels of the relayed signal and compromises the system performance.

Contrary to the mentioned works, this paper presents real-time GNU Radio based implementation of an FDR with DF relaying scheme, which basically eliminates the noise amplification limitation of AF and CF based FD relays. Additionally, the existing works are Field-Programmable Gate Arrays (FPGA)-based such as WARP Mango board [15], and while these FPGA-based SDRs offer deterministic timing and low latency, nevertheless, they are rather inflexible, and it is often challenging to implement complex signal processing algorithms in them. In-contrast, our proposed FDR implementation is GPP-based, build upon open-source platform GNU Radio, which is easily accessible and most importantly, the signal processing is done in software, with high-level programming languages C++ and Python. Thus, making it particularly easy to use, modify, and debug.

III. System Model

We consider a two hop relay system with a source node, a destination node, and a relay node with DF relaying scheme. In our framework, the entire relay system operates in non-cooperative manners, and the packets from source cannot reach destination directly. Also, the relay node in-between source and destination can operate either in HD or FD mode. When the considered system operates in HD relaying mode, the relay node simply receives a packet from source in time slot T_α, decodes it, then re-encodes and forwards it to destination in time slot T_β. However, when operating in full-duplex mode, the relay node first needs to suppress the LSI before moving towards the decoding part.

A. Looped Self-Interference Suppression

In our system, the task of LSI suppression is achieved in two stages: first, a passive suppression stage and, second, an active digital cancellation stage, which eliminates the LSI (including multi-path components) in baseband via signal processing.

The baseband digital samples at the input of full-duplex relaying node can be written as

$$y[n] = x_s[n] * h_{s-r}[n] + I_r[n] + w[n], \tag{1}$$

where $y[n]$ are the received samples, $x_s[n]$ are the transmitted samples from source, $h_{s-r}[n]$ are the channel coefficients from source to relay node, $I_r[n]$ are the LSI samples, and $w[n]$ are the Additive White Gaussian Noise (AWGN) samples. Our goal is to eliminate the looped SI $I_r[n]$, which is obtained as

$$I_r[n] = x_r[n] * h_{r-r}[n], \tag{2}$$

where $x_r[n]$ are the retransmitted samples generated after re-encoding, and $h_{r-r}[n]$ are the channel coefficients between relay transmitting and receiving ends. Since $x_r[n]$ are already known at the relay node so by obtaining an estimate of $h_{r-r}[n]$, approximate looped SI samples are generated as

$$\hat{I}_r[n] = x_r[n] * \hat{h}_{r-r}[n]. \tag{3}$$

[1]https://www.gnuradio.org/

Subtracting Equations (1) and (3) yields

$$y[n] - \hat{I}_r[n] = x_s[n] * h_{s-r}[n] + I_r[n] - \hat{I}_r[n] + w[n], \quad (4)$$

which can be further simplified as

$$y[n] - \hat{I}_r[n] = x_s[n] * h_{s-r}[n] + e[n] + w[n]. \quad (5)$$

In Equation (5), $e[n]$ represent the error due to the difference in actual received self-interference $I_r[n]$, and regenerated SI $\hat{I}_r[n]$. Note that if the error is negligible, i.e., $e_n[n] \approx 0$, the residual LSI is completely eliminated and the expression is reduced to

$$y[n] - \hat{I}_r[n] = x_s[n] * h_{s-r}[n] + w[n]. \quad (6)$$

In Equation (6), the right hand side is same as for received samples in a typical receiver operating in HD mode.

In practice, $e[n]$ can be reduced to significantly small numbers but it is never zero. This is certainly due to the inaccuracies in channel estimate, non-linear behavior of the amplifier, and oscillator phase noise at the retransmitting relay node. To keep the design simple and less complex, the latter two are not modeled in our system and left as potential future work. This paper primarily focuses on the implementation of linear LSI cancellation in digital domain. The impact of ignoring the other two parameters is discussed and shown in Section V.

B. Estimation of Looped SI Channel

To estimate the SI channel, we employed the time domain least square estimation approach. It basically computes the Channel Impulse Response (CIR) estimate $\hat{h}_{r-r}[n]$ through the Long Training Sequence (LTS) symbol embedded in the preamble of our OFDM packet. The estimation process is completed during the training transmissions, i.e., $x_s[n] = 0$.

From Equations (1) and (2), the received samples $y[n]$ during training transmissions are obtained as

$$y[n] = x_r[n] * h_{r-r}[n] + w[n], \quad (7)$$

i.e., only looped-back self-interference samples $I_r[n]$. Based on Equation (7), the received LTS samples can be written as

$$y^N_{LTS} = x^N_{LTS} * h^P_{r-r} + w^N_{LTS}, \quad (8)$$

where N represents the length of LTS samples and P indicates the number of channel taps, which typically corresponds to Cyclic Prefix (CP). For fixed and predefined x^N_{LTS} samples, the time domain convolution in Equation (8) can be expressed as a matrix multiplication, i.e.,

$$y^N_{LTS} = \mathbf{X}^{N\text{x}P} \cdot h^P_{r-r} + w^N_{LTS}. \quad (9)$$

Here, $\mathbf{X}^{N\text{x}P}$ is the Toeplitz matrix of order $N \times P$, formed using the known transmitted LTS samples [16]. Also, since the LTS samples are fixed and known in advance, the matrix $\mathbf{X^{N\times P}}$ can be precomputed and stored prior to the beginning of training transmissions.

The time domain least square estimate can thus be obtained as

$$\hat{h}^P_{r-r} = \mathbf{X}^{N\times P^\dagger} \cdot y^N_{LTS}, \quad (10)$$

where $\mathbf{X}^{N\times P^\dagger}$ is the Moore-Penrose (pseudo) inverse of $\mathbf{X}^{N\times P}$ and y^N_{LTS} are the received LTS samples.

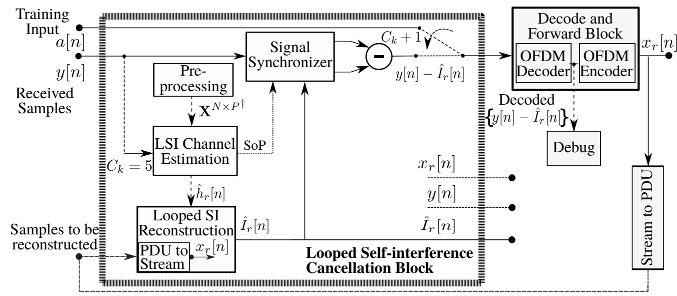

Figure 3. Detailed baseband level block diagram of our novel LSI suppression module for full-duplex relay implementation.

C. Reconstruction of Looped SI

Reconstruction of the looped SI is similar to the equalization process but instead of equalizing the received samples, the known retransmitted samples $x_r[n]$ are equalized with the acquired channel estimate $\hat{h}_{r-r}$[n]. In order to apply the channel impairment effects on reconstructed LSI samples, the estimated CIR is convolved with the known samples $x_r[n]$, shown with Equation (3). As a result, the reconstructed self-interference samples $\hat{I}_r[n]$ innate the same channel properties as that carried by the received LSI samples $I_r[n]$.

IV. Implementation Details

For performance evaluation, we implemented both HDR and FDR with DF relaying scheme in GNU Radio. We choose GNU Radio as the implementation platform because of its wide-spread use as a real-time signal processing framework and its ability to do rapid prototyping. Moreover, the GNU Radio Companion (GRC), a graphical tool for creating flow graph, allows to monitor the real-time received/processed samples through visualization scopes in both time and frequency domains.

For the implementation of DF relaying scheme, we used GNU Radio's OFDM blocks in the GRC with key parameters listed in Table I. The design of DF based half-duplex relay is rather simple as it just needs to receive the packet from source, decode it, then re-encode and forward the packet to destination. However, for FDR, we have implemented a novel core block for the cancellation of looped-back self-interference in GNU Radio framework. It is important to mention here that the GRC does not allow direct feedback of the streaming samples in a flow graph; which is the key requirement in FD relaying, necessary for the reconstruction of LSI. For this reason, all the re-encoded samples $x_r[n]$ are first converted

Table I
Key parameters of the employed GNU Radio's OFDM block.

Modulation	Q-PSK
Number of Sub-Carriers	64
Pilots	4
Data Carriers	48
CP Length	16
FFT/IFFT Size (N)	64 points
Packet Preamble (STS + LTS) Symbols	$(1+1)$ OFDM Symbol
Packet Header (3 B)	1 OFDM Symbol

into a Protocol Data Unit (PDU) message, and then fed back to the looped SI cancellation block as illustrated in Figure 3.

A. *Looped SI Cancellation Block*

The looped SI cancellation block first forwards the $C_k + 1$ training packets within the DF relaying node for the estimation of SI channel, and for stabilizing the sub-blocks such as signal synchronizer. In Figure 3, C represents the number of training packets and k is the process repetition interval. During the forwarding of training packets, the transmissions from source are turned off until the relay switches to full-duplex relaying mode, as shown in Figure 3.

1) Preprocessing: The preprocessing block first performs the Inverse Fast Fourier Transform (IFFT) on LTS symbol enclosed in the packet preamble, hence converting it into time domain samples. Afterwards, the obtained time domain samples are used to create Toeplitz matrix $\mathbf{X}^{N \text{x} P}$, and finally to calculate the Moore-Penrose (pseudo) inverse $\mathbf{X}^{N\times P^{\dagger}}$ of the Toeplitz matrix $\mathbf{X}^{N\times P}$, which is later used with received LTS samples y^{N}_{LTS} to compute the estimate of SI channel. Here, N is same as number of IFFT points and P is set to be half of CP, the values of each are listed in Tables I and II, respectively.

2) SI Channel Estimation: The estimation block operates only during the training transmissions. It first correlates the received samples $y[n]$ with the known LTS samples x^{N}_{LTS} to determine the Start-of-Packet (SoP). Once SoP is determined, it then extracts the received LTS samples y^{N}_{LTS} and uses them with $\mathbf{X}^{N\times P^{\dagger}}$ to compute the SI channel estimate $\hat{h}_{r-r}[n]$.

3) LSI Reconstruction: The reconstruction block first converts the PDU message containing re-encoded samples $x_r[n]$ into streaming samples and then convolves them with the obtained SI channel estimate $\hat{h}_{r-r}[n]$ to produce approximate looped SI samples $\hat{I}_r[n]$.

4) Signal Synchronizer: The synchronizer block synchronizes the reconstructed LSI samples $\hat{I}_r[n]$ with the received samples $y[n]$ during training transmissions. It calculates the delay introduced by the relay's front ends, i.e., from Tx to Rx. Since the fed back known samples $x_r[n]$ arrive earlier compared to the received LSI samples, the synchronizer starts buffering the reconstructed samples and waits for an SoP indicator to release them. Also, the synchronizer block computes the required buffer length during training session, i.e., no transmissions from source. Once the buffer length is determined it does not changes because the delay from relay Tx to Rx end remains the same.

After synchronization, the reconstructed samples are subtracted from received samples and forwarded to the DF relaying block provided that the training transmission period is over.

Table II
KEY PARAMETERS OF FD RELAY NODE.

Training Packets (C)	5
Samples per Packet	3520
Number of Estimated Channel Taps (P)	8
Sampling Frequency	17.6 MHz
OFDM Symbol Duration	4.5 µs
CP Duration	910 ns
Estimable SI Channel Impulse Response	455 ns

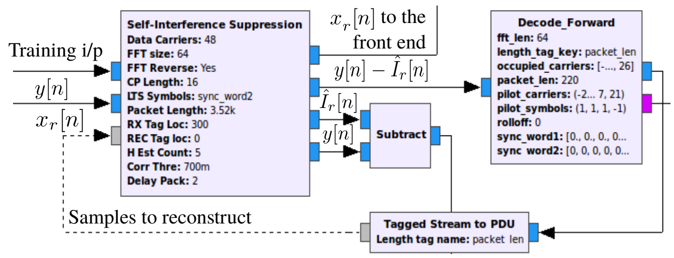

Figure 4. A screen shot of the most relevant blocks of our Full-Duplex Relay implementation in GNU Radio Companion.

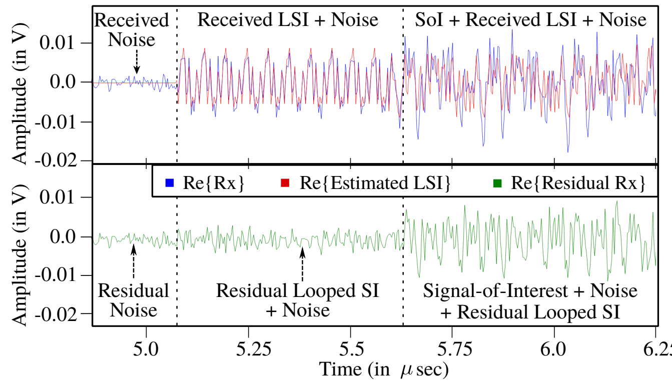

Figure 5. Snapshot of real-time LSI cancellation performance of our relay node under full-duplex mode.

The decoded output of DF block is also fed to a debugger, to check whether a packet is correctly decoded.

Figure 4 shows the implemented LSI cancellation block and DF module in GNU Radio. A screen shot of real-time looped SI cancellation with 0 dBm transmit power level of the relay node is shown in Figure 5. For the sake of clarity, the figure only shows the cancellation performance with real samples, i.e. the in-phase component. The signals in blue, red and green are received (LSI & SoI), reconstructed LSI and (residual LSI & SoI), respectively.

B. *Passive Suppression*

In our FDR systems, passive suppression is employed to suppress the direct/leaked SI signal, shown in Figure 2. As both Tx and Rx front ends are quite close, the looped-back SI signal is significantly stronger then the SoI arriving from a distant source, and if not suppressed to an extent, it can occupy the whole dynamic range of ADCs in the received signal process path. Therefore, the passive suppression stage is quite crucial. Different designs have been proposed for passive suppression [17]–[19], where a Radio Frequency (RF) isolation of up to 73 dB is shown to be achieved.

In this work, we used a very basic RF isolation approach, which provides a passive suppression of approx. 52 dB. We placed a Balsa foam wrapped with aluminum foil between the transmit and receive antennas. Even though the approach does not sound efficient, but, considering the available resources it has worked well enough to test, validate and evaluate the performance of our GNU Radio-based FDR implementation.

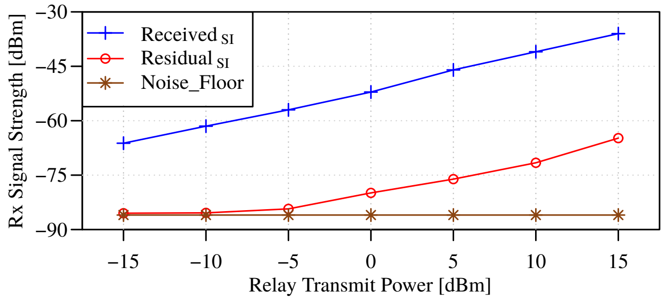

Figure 6. Looped SI suppression performance of the implemented FDR in digital domain with increasing transmit power levels of the relay node.

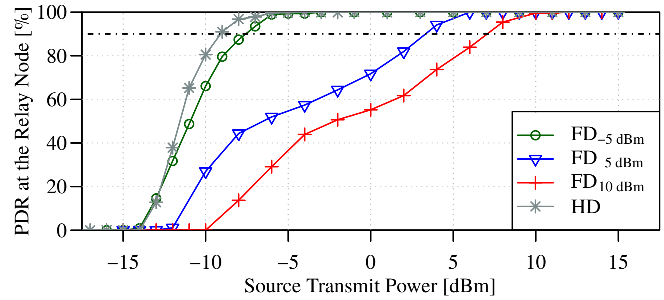

Figure 7. Experimentally measured PDR performances of DF based HDR and FDR implementations.

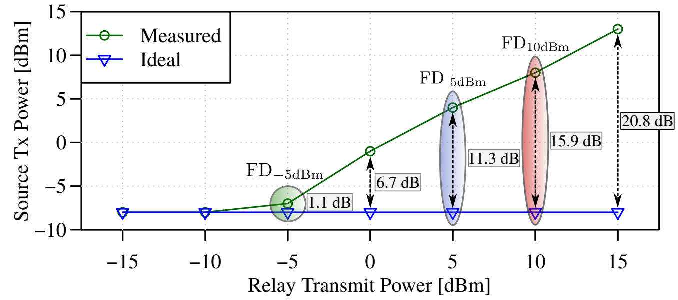

Figure 8. Required source transmit power levels to maintain PDR of 90 % for a given relay transmit power. The circled points highlights the 90 % PDR values of Figure 7.

V. Performance Evaluation

For the performance evaluation of our DF based FDR, we conducted experiments in our radio lab. In our experimental setup, we used three B210 USRP SDRs as transmitting, relaying, and receiving nodes. The S-R and S-D distances are 15 m and 30 m, respectively. In each transmission 46 packets are transmitted from the source node, and the process is repeated 20 times for every considered power level. A single packet includes 44 OFDM symbols out of which 3 symbols contribute towards the overhead (Short Training Sequence (STS), LTS, and packet header). All relevant hardware specific parameters are listed in Table III. It is worth mentioning here that since decoding delay in DF relaying scheme is same regardless of HD or FD transmission mode, therefore, its impact is not studied in this work.

A. Looped SI Suppression Performance

Figure 6 shows the looped SI suppression achieved in digital domain for different transmit power level of the relay node. The measured noise floor of B210 USRPs operating at a sampling frequency of 17.6 MHz is −86 dBm. It can be seen in the figure that received LSI is suppressed to the receiver's noise floor for low transmit power level (up to −10 dBm). However, for higher transmit power levels, a gradual increase in the residual SI is observed due to the following reasons.

First, the obtained RF isolation is far from being perfect and with higher transmit power level insufficient isolation becomes more obvious. By employing more sophisticated RF suppression techniques such as dual-port dual polarized slot coupled antenna or antenna separation through RF absorber along with orthogonal polarization, RF isolation can be greatly improved. Secondly, the implemented LSI suppression block does not models the non-linear behavior of the amplifier in RF chain. For high transmit power levels, the non-linear factor added by the amplifier becomes more significant, resulting in increased levels of residual SI. By addressing the two mentioned factors, the residual SI can be further suppressed close to the receiver's noise floor, even at higher transmit powers of the relay node.

Table III
Hardware specific parameters

Carrier Frequency	868 MHz
Receiver Noise Floor	−86 dBm
Source-Relay Distance	15 m
Source-Destination Distance	30 m
RF Looped SI Isolation	≈ 52 dB
Digital SI Suppression	up-to 32 dB

B. Packet Delivery Ratio at the Relay Node: FDR vs HDR

In Figure 7, the achieved PDR at relay node operating in both FD and HD mode is plotted for increasing transmit power level of the source node. Here, packet-delivery ratio (PDR) 100% means that all packets have been correctly detected and decoded at the receiver. The three FD mode curves in the plot represent the PDR obtained at different transmit power levels of the relay node (see legends subscript). It can be seen that the PDR with both $FD_{-5\,dBm}$ and HD is relatively similar. There is a roughly 1 dB difference in the performance certainly due to non-negligible residual LSI. Also, the PDR performances with $FD_{5\,dBm}$ and $FD_{10\,dBm}$ is much worse, both achieve 100 % PDR at higher source transmit power levels. This is due to the reason that when more transmitter gain is applied at relay node, i.e., $FD_{5\,dBm}$ and $FD_{10\,dBm}$, the increased residual LSI as a result, raises the noise-plus-interference level for SoI arriving from source node, hence more power is needed from source to overcome this increased noise floor, and to maintain 100 % PDR. By simply employing a better RF isolation technique, such as in [18] with 70 dB RF suppression compared to 52 dB of isolation achieved in this work, the PDR performance can be greatly improved even at higher relay transmit power levels.

Figure 8 demonstrates the required source transmit power levels for a given relay transmit power in order to maintain a PDR of 90 % at the relay node. Ideally, this plot should have been a straight horizontal line, however, a ramp like graph here

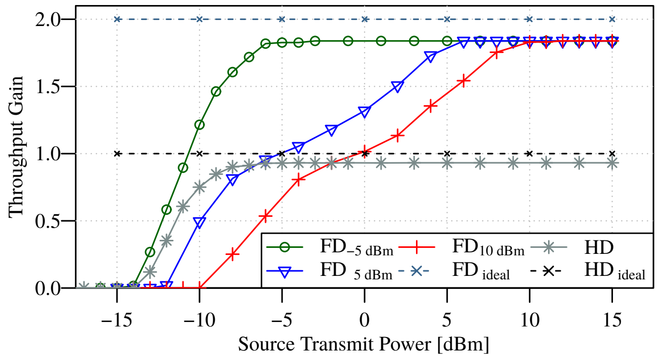

Figure 9. Achieved throughput gains with FD relaying over HDR at different transmit power levels of the relay node.

is due to the increasing levels of residual LSI at higher relay transmit power. This as a result, raises the overall noise-plus-interference level for SoI and more transmit power is required from source to retain the desired 90 % PDR.

C. Throughput Gain: FDR vs HDR

Figure 9 depicts the throughput gain of FD relaying over HDR system in our described experimental setup. To keep the training transmission overhead to a minimum, the training packets C are fixed to 5. Ideally, the throughput gain with FDR should be twice of HDR, however, after considering both packet and training transmission overheads, a maximum throughput gain of $1.8\times$ is measured with $\text{FD}_{-5\,\text{dBm}}$. This is still a nearly two fold increase in throughput gain with FDR over HD relaying. The plot also demonstrates that at high transmit power level of the relay node, which results in residual LSI, reduces the throughput gain considerably as compared to the throughput gain achieved at low transmit power level at which the residual LSI is almost eliminated. These results clearly highlight the strict requirement of residual LSI suppression to the receiver's noise floor in FDRs, to achieve maximum throughput gains.

VI. Conclusion and Discussion

In this paper, we presented a novel SDR-based real-time Full-Duplex (FD) Decode and Forward (DF) relay implementation in GNU Radio. To the best of our knowledge, this is the first lab-ready GPP-based FD relaying system. In a series of experiments, we were able to show that when Looped Self-Interference (LSI) is fully suppressed, the throughput gain with our FDR implementation (including the overhead) is nearly twice compared to an HDR. We furthermore investigated the effects of residual LSI, and showed its impact on the FDR performance, the noise floor for Signal-of-Interest (SoI), and the transmit power requirement of the source node. Our FDR implementation prototype is based on open-source GNU Radio framework, and with slight modifications it can be extended to work with any OFDM based wireless system.

As a concluding remark, the work presented here only models the linear looped-back Self-Interference (SI) component, and employs a basic RF isolation method. The real-time LSI cancellation performance can be significantly improved with sophisticated RF isolation techniques, and can be further bettered with analog cancellation circuits. Additionally, for non-linear LSI modeling, coefficients computation along with filtering of higher order components is required (see [8]). The solution of aforementioned possible improvements lies outside the scope of our proposed software-based implementation, and it is therefore left as potential future work.

References

[1] 3GPP, "Further advancements for E-UTRA physical layer aspects," 3GPP, TR 36.814 V9.0.0 (2010-03), Mar. 2010.

[2] "Local and metropolitan area networks," IEEE, Standard P802.16j-06/026r4, May 2007.

[3] Z. Zhang, K. Long, A. V. Vasilakos, and L. Hanzo, "Full-duplex wireless communications: Challenges, solutions, and future research directions," *Proceedings of the IEEE*, vol. 104, no. 7, pp. 1369–1409, 2016.

[4] A. Sahai, G. Patel, and A. Sabharwal, "Pushing the limits of full-duplex: Design and real-time implementation," arXiv, Tech. Rep., Jul. 2011.

[5] D. Bharadia, E. McMilin, and S. Katti, "Full Duplex Radios," *ACM SIGCOMM Computer Communication Review (CCR)*, vol. 43, no. 4, pp. 375–386, Oct. 2013.

[6] E. Ahmed and A. M. Eltawil, "All-digital self-interference cancellation technique for full-duplex systems," *IEEE Transactions on Wireless Communications*, vol. 14, no. 7, pp. 3519–3532, 2015.

[7] H. Zhuang, J. Li, W. Geng, and X. Dai, "Duplexer design for full-duplex based wireless communications," *China Communications*, vol. 13, no. 11, pp. 1–13, 2016.

[8] M. S. Amjad, H. Nawaz, K. Ozsoy, O. Gurbuz, and I. Tekin, "A Low-Complexity Full-Duplex Radio Implementation With a Single Antenna," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 3, pp. 2206–2218, Mar. 2018.

[9] J. Zhou, N. Reiskarimian, J. Diakonikolas, T. Dinc, T. Chen, G. Zussman, and H. Krishnaswamy, "Integrated Full Duplex Radios," *IEEE Communications Magazine*, vol. 55, no. 4, pp. 142–151, 2017.

[10] J. Jiao, X. Jia, M. Zhou, M. Xie, and Q. Cao, "Full-duplex massive MIMO AF relaying system with low-resolution ADCs under ZFT/ZFR scheme," in *2nd IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC 2017)*. Chongqing, China: IEEE, Mar. 2017, pp. 2024–2028.

[11] Q. Li, S. Feng, X. Ge, G. Mao, and L. Hanzo, "On the Performance of Full-Duplex Multi-Relay Channels With DF Relays," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 10, pp. 9550–9554, Oct. 2017.

[12] L. Elsaid, M. Ranjbar, N. Raymondi, D. Nguyen, N. Tran, and A. Mahamadi, "Full-Duplex Decode-and-Forward Relaying: Secrecy Rates and Optimal Power Allocation," in *85th IEEE Vehicular Technology Conference (VTC2017-Spring)*. Sydney, Australia: IEEE, Jun. 2017.

[13] F. H. Gregorio, G. J. Gonzalez, J. Cousseau, T. Riihonen, and R. Wichman, "RF Front-End Implementation Challenges of In-band Full-Duplex Relay Transceivers," in *European Wireless Conference (EW 2016)*. Oulu, Finland: IEEE, May 2016.

[14] T.-C. Chen, Y.-C. Chen, M.-K. Chang, and G.-C. Yang, "The Impact of Loopback Channel Estimation Error on Performance of Full-Duplex Two-Way AF Relaying Communication Systems," in *85th IEEE VTC2017-Spring*. Sydney, Australia: IEEE, Jun. 2017.

[15] D. Bharadia and S. Katti, "Fastforward: Fast and constructive full duplex relays," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4, pp. 199–210, 2015.

[16] M. S. Amjad and O. Gurbuz, "Linear Digital Cancellation with Reduced Computational Complexity for Full-Duplex Radios," in *IEEE Wireless Communications and Networking Conference (WCNC 2017)*. San Francisco, CA: IEEE, Mar. 2017.

[17] M. Jain, J. I. Choi, T. Kim, D. Bharadia, S. Seth, K. Srinivasan, P. Levis, S. Katti, and P. Sinha, "Practical, Real-time, Full Duplex Wireless," in *17th ACM International Conference on Mobile Computing and Networking (MobiCom 2011)*. Las Vegas, Nevada: ACM, Sep. 2011, pp. 301–312.

[18] E. Everett, A. Sahai, and A. Sabharwal, "Passive self-interference suppression for full-duplex infrastructure nodes," *IEEE Transactions on Wireless Communications*, vol. 13, no. 2, pp. 680–694, 2014.

[19] H. Nawaz and I. Tekin, "Three dual polarized 2.4GHz microstrip patch antennas for active antenna and in-band full duplex applications," in *16th Mediterranean Microwave Symposium (MMS 2016)*. Abu Dhabi, UAE: IEEE, Nov. 2016.