ScatterMIMO: Enabling Virtual MIMO with Smart Surfaces

Manideep Dunna, Chi Zhang, Daniel Sievenpiper, Dinesh Bharadia
University of California, San Diego

ABSTRACT

In the last decade, the bandwidth expansion and MIMO spatial multiplexing have promised to increase data throughput by orders of magnitude. However, we are yet to enjoy such improvement in real-world environments, as they lack rich scattering and preclude effective MIMO spatial multiplexing. In this paper, we present ScatterMIMO, which uses *smart surface* to increase the scattering in the environment, to provide MIMO spatial multiplexing gain. Specifically, *smart surface* pairs up with a wireless transmitter device say an active AP and re-radiates the same amount of power as any active access point (AP), thereby creating virtual passive APs. ScatterMIMO avoids the synchronization, interference, and power requirements of conventional distributed MIMO systems by leveraging virtual passive APs, allowing its *smart surface* to provide spatial multiplexing gain, which can be deployed at a very low cost. We show that with optimal placement, these virtual APs can provide signals to their clients with power comparable to real active APs, and can increase the coverage of an AP. Furthermore, we design algorithms to optimize ScatterMIMO's *smart surface* for each client with minimal measurement overhead and to overcome random per-packet phase offsets during the measurement. Our evaluations show that with commercial off-the-shelf MIMO WiFi (11ac) AP and unmodified clients, ScatterMIMO provides a median throughput improvement of 2 × over the active AP alone.

CCS CONCEPTS

• **Hardware → Wireless devices**; • **Networks → Physical links**; **Wireless local area networks**.

KEYWORDS

Smart walls, Smart Surfaces, Virtual MIMO, Smart surfaces, WiFi Backscatter, Antenna design

ACM Reference Format:
Manideep Dunna, Chi Zhang, Daniel Sievenpiper, Dinesh Bharadia. 2020. ScatterMIMO: Enabling Virtual MIMO with Smart Surfaces . In *The 26th Annual International Conference on Mobile Computing and Networking (MobiCom '20), September 21–25, 2020, London, United Kingdom.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3372224.3380887

1 INTRODUCTION

MIMO has been the core technology for multi-Gbps wireless connectivity for the last two decades. The 802.11ac standard released in 2011 already promises a peak data rate of 4.33 Gbps with 4 × 4 MIMO[1], sufficient for even high-quality wireless VR [34, 39]. However, 8 years down the line, we rarely see any Wi-Fi connection achieving this performance. Even with densely deployed Wi-Fi APs, we still suffer from poor throughput while being well within the coverage area. The primary reason is the lack of multiple strong paths in the environment. Typically a client observes a single dominant path, while the other paths are 10 – 20 dB weaker. As a consequence, the MIMO spatial diversity gain is lower, which in turn reduces the throughput. This problem is evident in the literature and has been approached extensively [13, 17, 23, 24, 36, 44, 52–54].

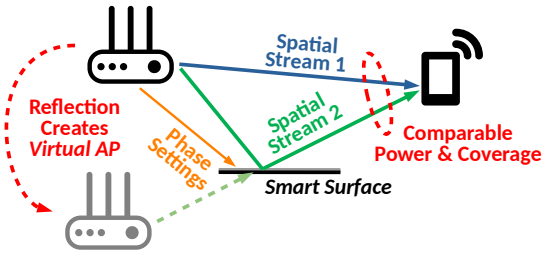


Figure 1: ScatterMIMO creating *virtual AP* for MIMO streams, via phase-shifted reflection and its optimization algorithms.

A large body of literature considers either multiple active APs or relays to enrich spatial diversity [13, 21, 23]. For instance, MegaMIMO [23] creates multiple independent spatial streams at the client by sharing the transmitted data and coordinating transmissions from multiple APs. However, it requires wireless carrier-level synchronization between multiple active APs which is shown to be challenging to achieve [23]. Furthermore, a recent full-duplex active relaying approach achieves additional spatial streams without the need for sharing the transmitted data [13], but needs to be wall-powered. On the other hand, novel passive approaches have been proposed to improve the throughput [14, 30, 55], however, they fall short of providing improvements as compared to active approaches. For example, a recent work [30] improves throughput by a fraction (20%) and is limited to a short range (8 meters).

In this paper we present ScatterMIMO, a passive *smart surface* that pairs up with a real active AP, creates a *virtual AP* by reflecting its signal in a controlled and optimized manner, and can even generate additional MIMO streams to double the throughput, as shown in Figure 1. ScatterMIMO works by creating a controlled and directional reflected path for each client, while making it as strong as the direct path signal. In other words, the signal reflected by *smart surface* has comparable power as the signal originating from the active AP, as if there were an active *virtual AP* placed and not a passive surface. ScatterMIMO achieves *virtual AP* by backscattering the AP's signal with multiple antennas inspired by [13, 19, 30], and ensures the constructive combination of the reflected signals at the client's location. To do so, each of its antennas applies an extra

[1] 5-GHz channels with 1024-QAM (MCS 11) and 160 MHz bandwidth, as supported by chip vendors such as Quantenna [40].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MobiCom '20, September 21–25, 2020, London, United Kingdom
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7085-1/20/09…$15.00
https://doi.org/10.1145/3372224.3380887

delay (*i.e.* phase shift) with a programmable phase-shifting circuit, to match the reflected phase at the client's location. ScatterMIMO's passive design makes it simple, low-cost, compact, and low-power, which allows it to be deployed in arbitrary location (such as hidden under paintings) and in large quantities.

Can we achieve reflected path power from the *smart surface* that is comparable to the direct path power from an AP? Intuitively, the signal in the reflected path is first absorbed at the *smart surface* and then re-radiated, and thus suffers from near-field attenuation [22] twice. Having large number of antennas on the smart surface can combat this attenuation loss if the reflected signals from the *smart surface* interfere constructively. A natural question here is, how many antennas would be sufficient to make reflected power comparable to direct path power? We observe that the required number of antennas vary depending on the distances between the AP, the *smart surface*, and the client. To model this dependency, we build an analytical model to calculate the number of antennas required as a function of distances. Our calculation shows that with reasonable placement, the *smart surface* requires fewer than 50 antennas, which can be packed within a 30 cm × 10 cm area as a 2-D 5-GHz antenna array.

However, to combine constructively the reflected signal from each antenna on the *smart surface*, ScatterMIMO needs to know the correct phase offsets to apply. A naive way is to derive the phases from the client's exact location, but it requires at least sub-centimeter-level location accuracy which cannot be reliably achieved with radio or even optical localization [29, 50, 56]. Alternatively, ScatterMIMO could exhaustively try all possible phase combinations and derive the optimal setting from client feedback, but the search space is too large and the process takes longer to complete than the channel coherence time. To solve this problem, we design a novel algorithm that can determine the optimal *smart surface* phase configuration with just 3 uplink channel measurements, by applying 3 special *smart surface* configurations and decomposing the phase observed by the client. This allows ScatterMIMO to optimize the *smart surface* within 180 us, which is well below the channel coherence time of 10 msec [22].

The above algorithm requires us to measure the change in phase of the reflected signal accurately across the 3 special *smart surface* configurations. Unfortunately, practical Wi-Fi radios incur random phase offsets and sampling time differences for each packet, which cannot be directly measured and would render the phase measurements useless. The key observation of ScatterMIMO is that the phase of the direct path should not change across measurements, and once we separate the direct and reflected path, we can compensate for the phase offset by correcting the direct path phase to the same value. In order to separate the direct path from the composite channel (direct path and reflected), ScatterMIMO leverages the insight that direct path arrives at the AP before any other path. By leveraging this property, ScatterMIMO uses a space-time technique to separate the direct from the composite channel and match the random phase offsets and sampling time offsets across the 3 special *smart surface* configuration packets.

Our contributions can be summarized as follows:

(1) We present the design of ScatterMIMO, which creates *virtual AP*, enriches spatial diversity for MIMO, and improves throughput with just a passive *smart surface*. We model the signal strength of the reflected path and derive a way to make it always comparable to the direct path, regardless of the location of the client;
(2) We design a fast optimization algorithm that requires just 3 packets to determine the phase offsets that achieve the best reflected power at the client, which is much faster than previous methods [30] and greatly improves the channel efficiency;
(3) We design a space-time filtering technique to match the per-packet random phase offset across packets on commercial, off-the-shelf(COTS) radios, which enables integrating ScatterMIMO in existing Wi-Fi access points and clients;
(4) We implement our system with COTS Wi-Fi APs and clients, and prototype ScatterMIMO's *smart surface* with low-cost PCB substrate; we characterize and evaluate ScatterMIMO thoroughly. ScatterMIMO improves the median throughput by 2× for both stationary and mobile clients, and improves coverage to 45 meters with just a passive 48-antenna *smart surface* placed 50 cm away from the active AP.

The rest of the paper is organized as follows: First, we discuss the prior work related to smart surfaces and advances made by ScatterMIMO (Sec. 7). Next, we present a feasibility analysis of ScatterMIMO by building a mathematical framework that can be used to design *smart surface* as desired by the user. ScatterMIMO uses the framework to achieve *virtual AP* while being practical and compact and providing necessary spatial diversity (Sec. 2). (Sec. 3) presents low-overhead algorithms for optimizing ScatterMIMO's *smart surface*,which works with COTS WiFi devices by overcoming the hardware impairments when working with COTS WiFi. We present the hardware design of ScatterMIMO to work with practical phase shifters (Sec. 4). Finally, we show our implementation (Sec. 5) and evaluation (Sec. 6) results in different scenarios, followed by limitations (Sec. 8) and conclusion (Sec. 9).

2 FEASIBILITY OF PASSIVE VIRTUAL AP

ScatterMIMO's goal is to use passive *smart surface* to create *virtual AP*, which in turn re-radiates the same power as the actual AP. Therefore, *virtual AP* creates additional spatial streams with comparable data rates, while improving the overall signal to noise ratio. ScatterMIMO builds on top of the observations of previous works [7, 30, 55], *i.e.* by combining multiple reflecting antennas, a *smart surface* can act as a concave mirror, collect signal power that would otherwise be lost in the space, and refocus it to the client. Such an effect requires adding phase-shifting elements to each antenna, so all the reflected signals can combine constructively at the client.

ScatterMIMO goal sets it apart from the other works, which is to enable virtual MIMO with one key prerequisite that *the reflected path must provide a signal power comparable to the direct path between the AP and the client.* However, since ScatterMIMO aims at using passive reflectors to avoid downsides of amplification and re-transmission (Sec. 1), providing sufficient power in the reflected path is extremely challenging. Here we explore how ScatterMIMO can achieve the aforementioned condition with just passive reflectors. Our key findings are:

- By placing the *smart surface* close to the AP but still far enough to create spatial diversity, the power reflected to the client can be comparable to the direct path with less than 50 antennas.

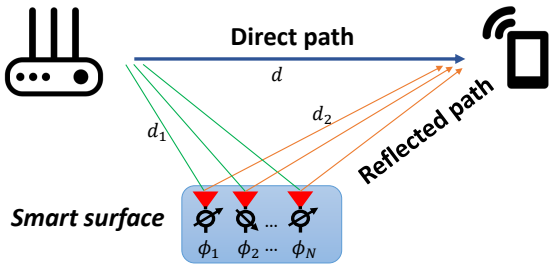


Figure 2: ScatterMIMO's multi-antenna *smart surface* with phase-matching capability.

- The same placement can ensure the reflected power is comparable to the direct path regardless of the client's location, thus giving ScatterMIMO the same coverage as the active AP.

2.1 Modeling Reflected Power

To derive the exact power of the reflected signal that ScatterMIMO can achieve, we first consider the scenario where there is just 1 AP, 1 *smart surface* with multiple antennas, and 1 client, as shown in Figure 2. We assume the client's location is known, and the passive smart reflect can reflect the signal in any desired direction with negligible internal loss. Later in Section 3 we will discuss how practical *smart surface* can achieve these requirements. We then denote the distance between the AP and the *smart surface* as d_1, and the distance between the *smart surface* and the client as d_2, while the distance of the direct path as d. By applying the Friis equation, we can get the power of the direct path:

$$P_R = P_T \frac{G_T G_R \lambda^2}{(4\pi d)^2} \tag{1}$$

where P_R is the power received by the client, P_T is the power transmitted by the AP, and G_T, G_R are the antenna gains of the AP and the client, respectively, while λ is the wavelength and d is the distance between the TX and the RX. Here we assume Omni-directional antennas, so G_T and G_R do not change across directions. Meanwhile, for the reflected path, the same process happens twice (before and after reflection), hence we have:

$$P'_R = \left[P_T \frac{G_T G_r \lambda^2}{(4\pi d_1)^2}\right] \frac{G_r G_R \lambda^2}{(4\pi d_2)^2} = P_T \frac{G_T G_r^2 G_R \lambda^4}{(16\pi^2 d_1 d_2)^2} \tag{2}$$

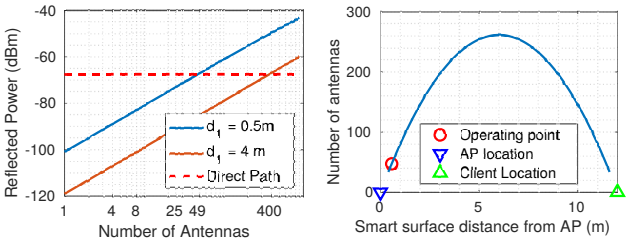
where G_r is the antenna gain of a single reflecting antenna, which also captures the internal losses of the reflector.

Now imagine that we put N antennas on the *smart surface* instead of just one, and that ScatterMIMO's *smart surface* can change the phase of the reflected signal, as shown in Figure 2. In such a case, we can phase-match all the reflected paths with phase shifters (Sec. 3), hence making the *amplitudes* of the signals add up constructively at the client's location. Assume the *smart surface* is small enough and the variation in d_1 and d_2 across antennas are negligible. The total *power* of the reflected paths then should become N^2 times stronger:

$$\sum P'_R = N^2 P_T \frac{G_T G_r^2 G_R \lambda^4}{(16\pi^2 d_1 d_2)^2} \tag{3}$$

The ratio between the power of the 2 paths is:

$$\frac{\sum P'_R}{P_R} = \left(\frac{N G_r \lambda d}{4\pi d_1 d_2}\right)^2 \tag{4}$$



(a) Power vs #antenna elements **(b) Optimizing operating point**

Figure 3: Number of antennas needed to match the direct path power for different *smart surface* placement.

It is clear that for given channel conditions (λ, d_1, d_2, d) and *smart surface*'s antenna gain G_r, we can manipulate this ratio by varying N. The goal of ScatterMIMO is to *make this ratio close to 1*, so a MIMO receiver can get the best SNR for all the streams within its dynamic range.

2.2 Optimizing Placement and Number of Antennas

One question that naturally arises from the foregoing analysis is how many antennas should ScatterMIMO's *smart surface* have, *i.e.* how large should N be. Besides, we need to decide how to place the *smart surface*, *i.e.* how to choose d_1. To answer these questions, we picked a typical antenna design with G_r = 4dB, and computed the reflected power for various d_1, while d = 12 m and $d_2 \approx d - d_1$ (the AP, *smart surface*, and the client are roughly on the same line). We then compute the expected power $\sum P'_R$ and compare it with the direct path power in Figure 3a. The result clearly shows that we need a prohibitively large number of antennas when the *smart surface* is placed far-away from the AP, while when the distance is 0.5 meters, only 49 antennas would be required.

From the above results, it seems that we can set d_1 to 0 to further reduce N. However, placing the *smart surface* too close to the AP will reduce the spatial diversity and diminish the gain for MIMO. To study how should we trade-off d_1 and N, we again vary d_1 in the previous setup, from 0 (*smart surface* co-located with the AP) to d (*smart surface* co-located with the client), while computing the number of antennas N required for $\sum P'_R / P_R = 1$. From the results shown in Figure 3b, we see that when the *smart surface* is close to either the AP or the client (*i.e.* $d_1/d \to 0$ or 1), N is relatively small and manageable, but when the *smart surface* is in the middle of the AP and the client (*i.e.* $d_1/d \approx 0.5$), N gets very large and can get unpreferable to implement in practice.

Since the client is mobile, we cannot guarantee that $d_1/d \to 1$. However, we can place the *smart surface* close to the stationary AP. In this case, *i.e.* $d_1/d \approx 0$, and $d_2 \approx d$. Considering that we want $\sum P'_R / P_R = 1$:

$$\frac{\sum P'_R}{P_R} \approx \left(\frac{N G_r \lambda}{4\pi d_1}\right)^2 = 1 \quad , i.e. \quad d_1 = \frac{N G_r \lambda}{4\pi} \tag{5}$$

observe that the performance becomes independent to client location d or d_2, *i.e. ScatterMIMO has the same coverage as the active AP* when its *smart surface* is placed close to the AP. Further, for a given *smart surface* (with known N and G_r), the optimal distance from the AP can be computed with Eq. (5). For example, for a *smart surface* with N = 48 antennas and G_r = 2.5 = 4 dB antenna gain, the

optimal distance between the AP and the *smart surface* is $d_1 = 55$ cm. Such d_1 can still provide substantial spatial diversity for MIMO, while it does not require too many antennas for achieving equal power from the direct and reflected path.

2.3 Improving MIMO diversity and multiplexing gain

In the previous section, we have discussed the requirements for reflected power equal to that of the direct path. In this section, we explain how our approach improves MIMO diversity and multiplexing gains. Diversity gain is obtained by having multiple copies of the same transmitted signal that combine coherently at the receiver. Multiplexing gain is obtained by transmitting multiple streams by precoding the data on the orthogonal basis of the channel matrix.

To understand how ScatterMIMO improves these gains, we consider a simple scenario. Consider a setup with a direct path and first order reflection from the *smart surface*. Without loss of generality, we assume the two physical paths constitute an orthogonal basis for MIMO streams. Since the first-order reflection is orthogonal to the direct path, maximizing the reflected power from the *smart surface* will surely add a new spatial stream. To improve the diversity gain, the reflected path and the direct path have to add up constructively at the receiver. We explain this constructive combining in Sec. 3.

In general, the reflected path and the direct path may not constitute an orthogonal basis. To find an orthogonal basis in such a case, we assume the direct path corresponds to a MIMO stream. Now, the reflected path has two components, one component along the direct path, and the other component orthogonal to the direct path, which is the second MIMO stream. Hence our approach to maximizing the reflected path power from the *smart surface* increases the power of both the MIMO streams, thus improving the multiplexing gain along with the diversity gain. In the next section, we explain how to maximize the reflected power from the *smart surface* and constructively combine it with the direct path to achieve diversity gain.

3 FAST CONSTRUCTIVE COMBINATION

In the last section, we showed how ScatterMIMO's *smart surface* could act like a *virtual AP* and generate reflected power comparable to the direct path over the entire coverage area of the AP, by leveraging the constructive combination from multiple antennas. An inherent assumption made by the previous section was that ScatterMIMO can achieve such a constructive combination by configuring its phase shifters properly, *i.e.* it knows the correct phase shifts, so all the reflected paths arrive at the client with the same phase. Since the client is mobile, *such phase values must be derived in real-time.*

Intuitively, these phase values can be derived from the location of the client, since the location of the AP and the *smart surface* is fixed and known. However, reasonable phase accuracy requires sub-centimeter-level localization, which cannot be reliably obtained by current Wi-Fi localization techniques [29, 50].

Alternatively, one can try different phase shift values by configuring the *smart surface* and letting the AP to receive a packet from the client to check how well the phase shift values work. When the *smart surface* contains N antennas equipped with 2-bit phase shifters, there are 4^N phase configurations to be tried. However, it takes years to iterate with merely 20 antennas even when using the shortest packet in the Wi-Fi protocol that takes around 60 µs, which gives the minimum time per measurement.

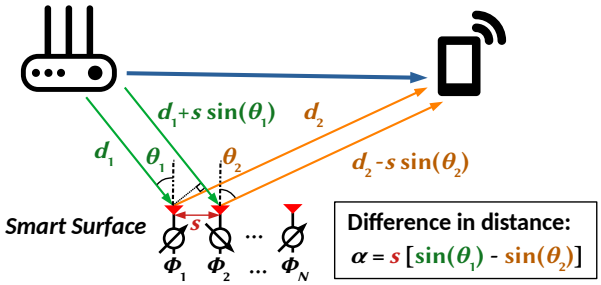


Figure 4: Path length difference for different antennas.

To overcome the exorbitant time complexity, prior work [55] proposed a greedy algorithm, which uses just $2N$ packets. With $N = 50$ as assumed in Sec. 2, the measurement time is 6 ms. However, as the channel coherence time is around 10 ms [22], the measurements need to be repeated every 10 ms, leaving merely 40% of the channel time to the data traffic. This greatly degrades throughput and defeats ScatterMIMO's purpose. Here we explore how ScatterMIMO can avoid such degradation.

3.1 Deriving Optimal Phase

Can we come up with a new way to measure the channel and derive the optimal phase in *constant time*, thus improving the channel time efficiency? ScatterMIMO's key insight is that *we only need to know θ_2, the reflection angle between the smart surface and the client, to compute the correct phase shifts.* To prove that, we present a mathematical formulation of the channel, shown in Figure 4. For simplicity, *we assume the placement meets the far-field condition [26], i.e.* the size of the *smart surface* is much smaller than the distance between it and the AP or the client. Hence the different reflected paths can be seen traveling parallel to each other. Denoting the incidental and reflection angles as θ_1 and θ_2, respectively, and s as the separation between 2 adjacent antennas. When there is no additional phase shift, the total path length r_n, r_{n+1} for the n-th and $(n+1)$-th antennas, and their path length difference α are:

$$r_n = d_1 + d_2 \quad r_{n+1} = [d_1 + s\sin(\theta_1)] + [d_2 - s\sin(\theta_2)]$$
$$\alpha = r_{n+1} - r_n = s[\sin(\theta_1) - \sin(\theta_2)] \tag{6}$$

Observe that s and θ_1 are the properties of the *smart surface*, thus fixed and known. Hence, the α value solely depends on θ_2. We now see how α translates to correct phase shift values.

To make the signals reflected from every antenna to add constructively, their phases at the client need to be the same. In other words, *the difference in their path lengths should be an integer multiple of the wavelength.* However, α does not necessarily satisfy such condition, since the value of θ_2 is arbitrary, depending on the location of the client[2]. That's where the additional phase shift of the reflector comes into play. It corrects α so it becomes an integer multiple of wavelength λ:

$$\alpha + \lambda\frac{\phi_{n+1} - \phi_n}{2\pi} = K\lambda \tag{7}$$

[2] θ_1 is fixed since the AP and the *smart surface* have fixed location, and s is fixed when the *smart surface* is manufactured, which is usually $\lambda/2$.

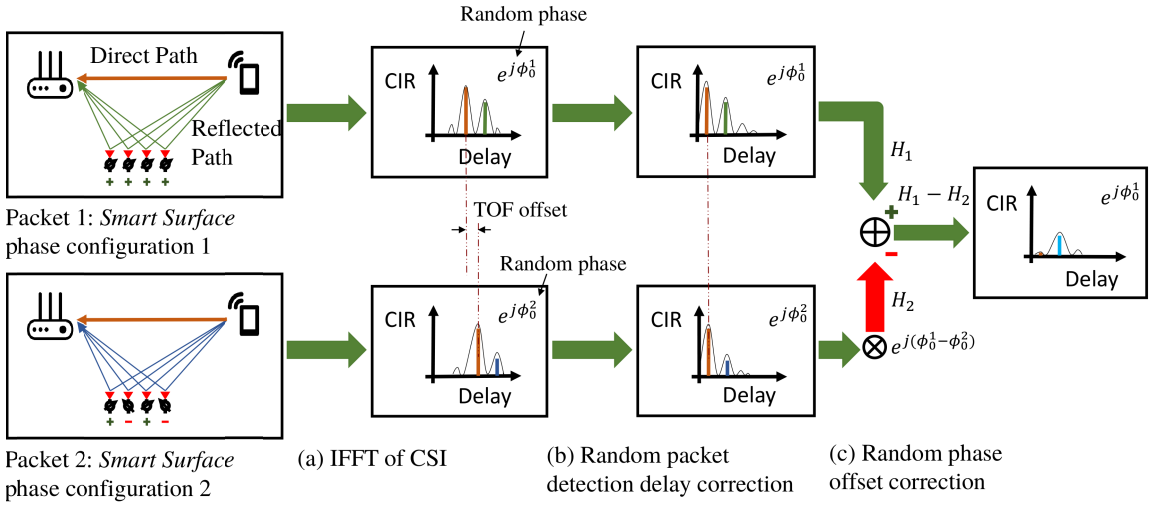


Figure 5: Illustration of ScatterMIMO's algorithm to compensate for packet detection delay and carrier phase offset using 1D IFFT.

where ϕ_n is the additional round-trip phase-shift added by the n-th reflector, and K is an integer that satisfies the equation. Let $\Delta\phi = \phi_{n+1} - \phi_n$ and rearrange Eq. (7), we have:

$$\Delta\phi = \frac{2\pi}{\lambda}(K\lambda - \alpha) \tag{8}$$

As long as $\Delta\phi$ satisfies Eq. (8), the lengths of all the reflected path will differ by multiples of wavelengths, thus the phase at the client will be the same and the signal amplitude will add up constructively. Here we see that we only need to know α to derive the phase difference $\Delta\phi$ between antennas. The optimal value for phase shift ϕ_n of antenna n is hence:

$$\phi_n = \phi_0 + n\Delta\phi = \phi_0 + \frac{2\pi}{\lambda}n(K\lambda - \alpha) \tag{9}$$

where ϕ_0 is an arbitrary phase offset that does not affect constructive combination, and K is an arbitrary integer, which can be set to any value to meet the range of the phase shifter. Later in Sec. 3.4 we will see how ϕ_0 comes into play when we need to support SISO links.

3.2 Measuring α in Constant Time

It is now clear that ScatterMIMO can optimize the phase shifters once the path length difference α is known. However, we still need to somehow measure α. Can we solve for α in one packet channel measurement at the AP, by inspecting the channel state information (CSI)? Turns out we cannot, the challenge is the AP would always receive a direct signal from the client, together with the reflected signal from the *smart surface*:

$$\begin{aligned} H_{r,n} &= |H_{d1}||H_{d2}|e^{-j\phi_n}e^{-jk(d_1+d_2+n\alpha)} \\ H &= H_d + \textstyle\sum_{n\in\{1,\ldots,N\}} H_{r,n} \end{aligned} \tag{10}$$

where H is the overall channel observed by the AP[3], H_d is the channel of the direct path, $H_{r,n}$ is the reflected channel between the AP and the client via *smart surface* antenna n, $k = 2\pi/\lambda$ (wavenumber), while $|H_{d1}|$ and $|H_{d2}|$ represents the path-loss of the signal from the AP to the *smart surface* and from the *smart surface* to the client, respectively (assuming path-loss caused by α is negligible due to far-field condition). Even if we can somehow isolate the total reflected signal, the α is embedded deep into the reflected signal as modeled in Eq. (10). Therefore, we cannot use just one channel measurement to solve for α.

Our key insight is to extract the reflected channel via differential measurement, by varying the phase configuration on the *smart surface* and measure the overall channel H, while keeping the direct-path channel the same across the measurements. We design a set of special phase configurations to minimize the number of measurements required, which requires just 3 measurements, instead of $2N$ in previous works.

To simplify the problem, we will explain our algorithm assuming $N = 4$. We let the client send 3 consecutive packets to the AP, while the AP measures the overall channel H_1, H_2 and H_3, for each packet. In the first packet, we set all the antenna phases to $\phi_n = 0$, measuring the baseline channel induced by the *smart surface*. For the second packet, we alternate phase shift ϕ_n of every antenna between 0 and π, *i.e.*:

$$e^{-j\phi_n} = +1, \forall n \in \text{even} \quad e^{-j\phi_n} = -1, \forall n \in \text{odd}$$

Finally, we send the third packet with:

$$e^{-j\phi_n} = -1, \forall n \in \text{even} \quad e^{-j\phi_n} = +1, \forall n \in \text{odd}$$

The channel estimates for the three packets are as follows:

$$\begin{aligned} H_1 &= H_d + |H_{d1}||H_{d2}|(1 + e^{-jk\alpha} + e^{-2jk\alpha} + \cdots)e^{-jk(d_1+d_2)} \\ H_2 &= H_d + |H_{d1}||H_{d2}|(1 - e^{-jk\alpha} + e^{-2jk\alpha} - \cdots)e^{-jk(d_1+d_2)} \\ H_3 &= H_d + |H_{d1}||H_{d2}|(-1 + e^{-jk\alpha} - e^{-2jk\alpha} + \cdots)e^{-jk(d_1+d_2)} \end{aligned} \tag{11}$$

Now subtract channel estimates from each 2 packets, resulting:

$$\begin{aligned} H_1 - H_2 &= 2|H_{d1}||H_{d2}|e^{-jk(d_1+d_2)}\textstyle\sum_{\forall n\in\text{odd}} e^{-njk\alpha} \\ H_1 - H_3 &= 2|H_{d1}||H_{d2}|e^{-jk(d_1+d_2)}\textstyle\sum_{\forall n\in\text{even}} e^{-njk\alpha} \end{aligned} \tag{12}$$

Now we take the angle of $(H_1 - H_3)(H_1 - H_2)^*$:

$$\angle[(H_1 - H_3)(H_1 - H_2)^*] = k\alpha = \frac{2\pi}{\lambda}\alpha \tag{13}$$

given that channel responses H_d and path-loss $|\tilde{H}| = |H_{d1}||H_{d2}|$ remains constant for these three packets. This means that we can

[3]Note we do not need to maximize $|H|$, since MIMO AP and clients have multiple antennas and can separate components in H.

conjugate multiply the two subtractions to derive α with just 3 packets, all without changing the existing Wi-Fi protocol.

In the above analysis, we assumed that the channel measurement has no random packet detection delay and no random phase offsets. The presence of any such random variables will make the cancellation impossible. Unfortunately, in practical Wi-Fi hardware, such random factors always exist. Sec. 3.3 will present how ScatterMIMO deals with hardware distortions to enable the cancellation on commercial, off-the-shelf (COTS) Wi-Fi devices, and thereby the calculation of the α.

3.3 Eliminating Hardware Distortions for COTS WiFi

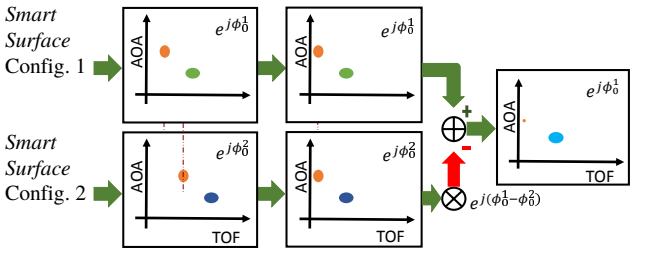
ScatterMIMO's goal is to enable the working of the *smart surface* with commercial off-the-shelf (COTS) WiFi devices. The previous sub-section presented the algorithm in a simplistic way to get the overall algorithm, and its working clearly explained. In this section, we would elaborate on our findings to overcome the practical issues such as phase error, packet detection delay, and multi-path and their impact on the fast constructive combining algorithm.

Recall that measuring α relies on the ability to cancel the channel while changing the phase configuration on the *smart surface*. However, an inherent assumption made was that the transmitter and the receiver clocks are synchronized for such cancellation to work. However, in real-world deployments, the transmitter and the receiver are not synchronized and have carrier frequency offset and sampling frequency offsets, which leads to variation in both the phase and packet detection delay (sampling time offset) from packet to packet. A natural question is how is the cancellation impacted by the phase offset and sampling time offset. To understand that, let us build a mathematical model. The phase offset and sampling time offset for the measured channel $H_{measured}$ can be written as:

$$H_{measured}(f) = e^{(j\phi_e - j2\pi f \tau_e)} H_{true}(f) = e^{j\beta(f)} H_{true}(f)$$

where H_{true}, represents the underlying channel, ϕ_e and τ_e represents the phase offset and the sampling time offset, which are represented in a single variable $\beta(f)$. f represents the sub-carriers of the OFDM system. Note that error caused by both the effects leads to an additional phase error for each sub-carrier and changes with every packet. Therefore, even if the underlying channel is the same, the estimated channel for two consecutive packets would have different phase error as the phase offset and sampling time offset would be different, In other words, even if we subtracted them, they wouldn't cancel each other.

The variation in the phase and packet detection delay renders the channel subtraction in-effective at learning α. The above problem has been at the heart of many different problems, including prior work on smart surfaces. The previous work[30] has used 4 msec long packet intervals to measure the CFO accurately and then used the radio's sampling clock to calculate the difference in time between two packets and compensate for the random phase offset due to the packet detection delay. Furthermore, to make matters worse, the radio's sampling clock time is not reported by off the shelf WiFi devices, which is necessary to estimate the accurate CFO and to compensate for the packet detection delay. So we cannot use the 4msec long optimization to optimize ScatterMIMO.



(a) Applying 2D-MUSIC algorithm (b) Correcting packet detection delay (c) Correcting random phase offset

Figure 6: Illustration of ScatterMIMO's algorithm for correcting packet detection delay and random phase offset using 2D-MUSIC.

Our goal is to overcome the phase error and random packet detection delay with only relying on the CSI provided from the COTS radios (without access to time-stamp from radios sampling clock) without any additional channel air time overhead, beyond the three packets measured. To provide insights for our solution, let us recap the process of learning α. Recall that we measure two channels with two different configurations of *smart surface* and subtract them, with the goal that direct path would cancel out, and the leftover channel would be the relative reflected signal from two configurations. The relative reflected signal for special configurations leads to the estimate α. To be specific, our goal is to measure the relative reflected signal from the entire process of the cancellation. Mathematically shown as,

$$H1_{measured} = e^{j\beta_1(f)} H1_{true} = e^{j\beta_1(f)}(H_{direct} + H1_{refl})$$
$$H2_{measured} = e^{j\beta_2(f)} H2_{true} = e^{j\beta_2(f)}(H_{direct} + H2_{refl})$$

Our key observation here is that the direct path doesn't change for the channel measurement of all three packets with the *smart surface* in a special configuration. Note that both the direct path and reflected signal from the ScatterMIMO undergo the same phase error, so if we can match the phase error for all three special channels, we can cancel them to achieve the relative reflected signal. Our key insight is to use the channel corresponding to the direct path to match the phase error for all 3 packets. Specifically, if we can match the $\beta_1(f) = \beta_2(f)$, we can achieve the cancellation. The challenge is that channel measured is a composite of the direct and reflected signal, so how can we isolate the direct path?

Our next insight to separate the direct signal from the reflected signal is that the direct path would be the first signal to arrive at the AP from the client [4] and quite often would be the strongest signal. Even when the client is in the non-line of sight compared to the AP, the signal from the client which reaches the AP first would be part of the direct channel, the reflected signal from the *smart surface* would arrive later.

Inspired by this observation, we can take IFFT of the $H1_{measured}(f)$ to obtain the time domain response and isolate the first peak signal, which would be corresponding to the direct signal as illustrated in Figure 5. We can repeat the process for all three packets and measure the amplitude corresponding to the first peak. Note that even with the sampling time offset changing for each packet, the first path would be represented by the first peak, just changing

[4] levearged in the context of localization; however, we are using it uniquely to match the phase error

the time index corresponding to it. We can align all the channels first peak to common index say 0 delay and then take the complex amplitude value for different packet at the peak *i.e.* 0 delay and match the phase error across all three packets and subtract them, thereby achieving much needed cancellation.

Instead of IFFT we use 1D-music to isolate all the multi-path, we use MUSIC [41] algorithm which is known to be more accurate. Once we resolve the multi-path delays and align the 0 delays with the shortest delay. Then using the complex amplitude of the shortest delay, we match the phase and perform the cancellation. The signal that is left post-cancellation is proportional to the relative reflection signal.

The above process, which seems quite robust, however, provides a resolution that is inversely proportional to the bandwidth of the signal transmitted by the client. We further improve the robustness of the above process by extending the algorithm to 2D *i.e.* space and time instead of just 1D-time. The two paths would quite likely have different space (angle of arrival) and time representation, therefore separating the direct and reflected signal more robustly. The idea is that most APs today have multiple antennas, and therefore, they measure CSI on multiple antennas for all three special packets without any additional channel airtime. We leverage the CSI collected on multiple antennas and multiple sub-carriers ($N_{ant} \times N_{sub}$) to perform the 2D-music as outlined in spotFi [29].

We apply the same process as described for the 1D-IFFT based approach. We perform the 2D-MUSIC, to extract all the multi-path with their delays and angle of arrival information. We align the first peak of each channel ToF-AoA to zero delays, thereby eliminating the sampling time offset. Then, we isolate the complex amplitude corresponding to the first path and measure the phase offset and match the phase offset for all the three packets. Then we can cancel the packets to achieve the relative reflected signal, which is then used to calculate the value of α. Removing this phase drift from consecutive packets allows us to measure α by subtracting the new channel estimates from consecutive packets, as explained in section 3.2. The above process is shown in the Figure 6. The 2D-Music is resilient to the multi-path which would act as interference in identifying direct path and therefore the algorithm is robust to multipath.

Upon solving the above, we set the *smart surface* to maximize the power in the the direction of θ_2, which can deliver the maximum power to the client. Note that even if the client is in non-line-of-sight, the algorithm figures out the signal direction which leads to the client, which is sufficient to maximize the power at the client. Therefore, the algorithm even works for non-line-of-sight clients.

To summarize, we used only three channel measurements to measure the direction of the client from *smart surface*, with off the shelf WiFi access point *i.e.* a constant number of packets. In contrast, smart wall proposed a solution which optimizes each antenna and therefore require linear time to optimize the design [30, 55]. Prior work like [7] used client location as it was connected to VR headset to optimize the relay design in their work.

3.4 Improving Diversity gain

In a practical network, both MIMO and SISO clients can associate with the same AP. Although ScatterMIMO cannot create additional spatial streams for SISO clients, it can still improve the SNR of the SISO link, thus improving the throughput and coverage. However, the optimization goal is to maximize $|H|$ instead of $|H_{r,n}|$, *i.e.* now the reflected paths need to phase-match with the direct path, rather than within itself. Fortunately, we still have an additional parameter ϕ_0, *i.e.* the base phase shift of the antennas, which can be optimized to make sure that the reflected paths add constructively to the direct path.

Discrete phase levels	360	8	4	2
Reflected power [dB] relative to direct path	-18	-6	-5	-6

Table 1: Reflected power vs. phase shifter configuration.

To optimize for SISO links, we rewrite Eq. (10) for the channel during the i-th packet and at each sub-carrier f_j:

$$\begin{aligned} H(f_j)^i &= H_d(f_j) + \textstyle\sum_{n\in\{1,\ldots,N\}} H^i_{r,n}(f_j) \\ &= H_d(f_j) + e^{-j\phi_0^i}|H_{d1}(f_j)||H_{d2}(f_j)|\textstyle\sum_{n\in\{1,\ldots,N\}} e^{-jk(d_1+d_2+n\alpha)} \\ &= H_d(f_j) + e^{-j\phi_0^i}H_r(f_j) \end{aligned} \tag{14}$$

where ϕ_0^i is the base phase shift for i-th packet, and H_r is the reflected channel without the base phase shift. As you can see, the second term in this complex addition is rotated based on value of ϕ_0^i. Thus once all the phases within the reflected path are optimized, ϕ_0^i determines whether the reflected path adds constructively to the direct path.

To determine the phase rotation required on the reflected path, ScatterMIMO lets the client send two more packets, each with the *smart surface* set to a different base phase ϕ_0^i, say 0 and π, so that we can measure $H_d(f_j)$ and $H^i_{r,n}(f_j)$, by subtracting these channel measurements. Now that we know the two complex terms in Eq. (14), we can choose ϕ_0 among phase values that would maximize the average SNR over all sub-carriers:

$$\max_{\phi_0}\left(\textstyle\sum_{f_j}\left\|H_d(f_j) + e^{j\phi_0}H_r(f_j)\right\|^2\right) \tag{15}$$

where $\phi_0 \in (0, 2\pi)$ can take a few discrete choices to approximate infinite continuous phase values. Later in Sec. 4.1 we will show that in a practical implementation, $\phi_0 + k\alpha$ (phase shift for the first antenna) and thus ϕ_0 can only have a few discrete values. This allows ScatterMIMO to maximize the SISO throughput quickly by iteratively computing the average SNR across sub-carriers for all available phase shift values.

4 PRACTICAL SMART SURFACE DESIGN

Previous sections are based on theoretical *smart surface* with ideal phase shifters. In this section, we describe real-world hardware constraints, their impact on the performance of the *smart surface*, and designs for the communication between the ScatterMIMO AP and the *smart surface*.

4.1 Working with Practical Phase Shifters

The previous analysis assumes that ϕ_n for each reflector can take any value, without any loss. This can be implemented by a continuous phase shifter, such as [5]. But in practice, controlling these phase shifters requires high analog voltage, which is hard to obtain in low-power designs, and more importantly, they incur a one-way loss of over 5 dB (hence > 10 dB round-trip), which significantly reduces the effectiveness of the *smart surface*. Another option is to

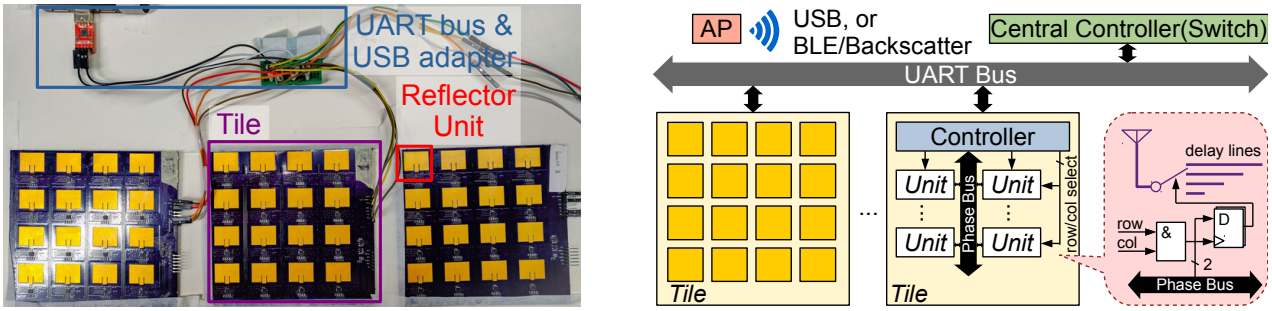


Figure 7: Hardware prototype and functional block diagram of ScatterMIMO *smart surface*.

use low-power, low-cost, and low-loss RF switches together with lossless delay lines to implement the phase shifters. The heuristic rule of thumb is that a M-bit RF switch (which produces 2^M discrete phase shifts) incurs M dB loss. For example, a 2-bit, quad-phase shifter would have 2 dB loss (late in Sec. 5 we will see the actual loss is close to 1 dB). However, these shifters can only support finite discrete phase values, which introduces a quantization error in the phase.

To understand how well these discrete phase shifters can approximate the ideal continuous and lossless ones, we compute the mean power reflected to a client for different M, across various random α values, for lossless and practical discrete phase shifters. We also compute the power with the ideal continuous and lossless phase shifters as a benchmark. Each time, we optimize the *smart surface*'s actual phase shift ϕ_n similar to what has been discussed in Sec. 3.4:

$$\max_{\phi_n}\left(\sum_{f_j}\left\|H_d(f_j)+|H_{d1}(f_j)||H_{d2}(f_j)|\sum_n e^{-j[k(d_1+d_2)+\phi_n]}\right\|^2\right),$$
$$\phi_n=\phi_0+nk\alpha\in\left\{0,2\pi\frac{1}{M},\ldots,2\pi\frac{(M-1)}{M}\right\} \quad (16)$$

and then calculate the SNR *w.r.t.* the ideal case benchmark.

Results in Table 1 suggests that on average, a 4-phase discrete phase shifter produces the best practical results, but more importantly, *it is just 5 dB worse than an ideal continuous phase shifter*. Hence, in our implementation we use the simple 4-phase shifter design for the *smart surface*.

4.2 Putting it All Together

In total, to learn the direction of the client, the ScatterMIMO AP sends explicit feedback packet transmission, which, when acknowledged, generates the 5 up-link packets needed to optimize the *smart surface*'s direction and the phase. Furthermore, the explicit feedback packets are short 60 µs packets, and it typically takes a total of 400 µs to optimize the *smart surface* for a particular client. The explicit feedback packets encode the channel in the acknowledgments, which is used to infer if the channel is coherent or changing too fast. Based on that, ScatterMIMO can decide whether to use the *smart surface*.

Eventually, the *smart surface* needs to know which client the packet is sent to or from. Since the AP is close enough to the smart reflector, we can use BLE packets or even backscatter communication [59] to send information about the use of the channel and potentially the client information. ScatterMIMO's *smart surface* then looks up the database to find the last known best phase setting for the particular client and programs the phase shifters to use those settings. Similarly, when the smart-phone or other devices intend to send a packet, they can choose to inform the *smart surface* over BLE about the use of channel and the client's identity. The advantage of a BLE control plane is its low-power and low latency (10–100 µs).

5 IMPLEMENTATION

5.1 ScatterMIMO Hardware Design

We hierarchically implement the ScatterMIMO *smart surface* to simplify the design and ensure scalability, as shown in Figure 7. Each *smart surface* can contain multiple *tiles*, which are coordinated by a *central controller* through a UART bus. The *central controller* communicated with the ScatterMIMO AP to obtain the appropriate phase information, and deliver them to each *tile*. Each of the *tiles* hosts a matrix of phase-shifted reflecting antennas, or namely, *reflector units*.

For our implementation, we put $4\times4=16$ *reflector units* on each *tile*. The 2-dimensional design allows us to steer the reflected beam both horizontally and vertically. However, since most indoor spaces have limited freedom in height, our evaluation will be focused on the horizontal case.

The *reflector unit* uses a patch antenna for maximum reflection. The antenna is then connected to a `HMC7992` 5-GHz RF switch, which further connects to 4 open-ended transmission lines. The transmission lines provide 0, $\pi/2$, π, and $3\pi/2$ round-trip phase shifts.

In each *tile*, we use a `STM32L053C6` low-power microcontroller to configure the RF switches. The two control signals of the switch can be directly wired to the microcontroller. In this case, the 4×4 *tile* would require $4\times4\times2=32$ separate signals. However, such design would create more traces than allowed by the space between the *reflector unit*, which is limited by the half-wavelength requirement. To make the design scalable, each *reflector unit* has its own addressable memory, which consists of two flip-flops that latch their input when the reflector is selected by row and column. The input of all reflector units can then be wired together to a bus. In such a way, a *tile* module can have an arbitrary number of *reflector units* by simply replicating the basic *reflector unit* design, as long as the central controller can provide the row/column signals. In our case, only $4+4+2=10$ signals are needed to address a 4×4 *tile*.

The microcontroller also takes commands from the UART bus. Each command sent on the bus contains the microcontroller's unique serial number, so only the desired *tile* takes the command. In such a way, all the *tiles* can share the same UART line, *i.e. tiles*

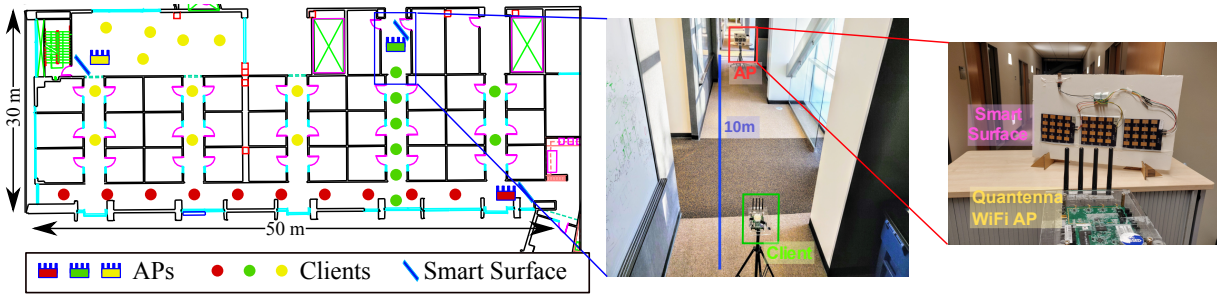


Figure 8: Left: The floor plan shows our deployment of the AP and *smart surface* at three different office locations (shown in red, green, and yellow color). For each AP location, we also show a few representational client positions. Right: Our setup showing 10 m link in a long corridor.

can be added as needed, without any dedicated wiring. For testing, we connect the UART bus to a computer running MATLAB via a USB adapter, which act as the *master controller*. Alternatively, the UART bus can be connected to a BLE adapter (*e.g.* [3]), allowing the *tiles* to receive commands from the active AP.

Each *tile* measures $12.2 \times 13.2 \times 0.3$ cm, which is thin and can be unobtrusively embedded into walls or hidden under paintings. The *tile* draws around 4.2 mA from 3.3 V, where 16 RF switches consume a total of 2.3 mA, and the rest 1.2 mA is from digital circuits. The total power consumption is thus less than 14 mW, or around 1/500 – 1/1000 of a typical active AP [1, 2, 4, 6]. Alternatively, if we use `SKY13575-639LF` RF switch that draws around 10 uA from 3.5 V, the power consumption will drop to 0.56 mW per tile.

We manufactured our *tiles* with OSH Park's 4-layer PCB, which uses Isola `FR408` substrate. From our measurement with a vector network analyzer [8], non-idealities of the low-cost PCB is evident at 5 GHz, and reflections (around -10 dB) in the RF path makes phase shifts deviate from their designed values. Fortunately, such error is smaller than the quantization error caused by discrete phase shifts (Sec. 4.1), which makes it tolerable in our system. Currently, it costs around $70 to manufacture a tile and $5 for each `HMC7992` RF Switch [18]. The total system cost includes $210 for 3 tiles and $240 for the IC components. However, system cost can be significantly reduced by using `SKY13575-639LF` RF Switch [43] that costs around $0.75 per unit. For large scale production of the PCBs, each PCB costs around $1.5 per unit. This would lower the system cost down to $40($35 for IC components + $5 for the 3 PCBs).

5.2 Client and AP Setup

We use commercial Wi-Fi access points to evaluate ScatterMIMO and demonstrate the improvements brought by ScatterMIMO in real-world office environments. we use 2 Quantenna `QHS840` 802.11ac development boards, each with 4×4 MIMO, one as an AP and one as the client. Each of them is connected to a separate PC via gigabit Ethernet. The development board's firmware allows us to collect channel state information (CSI) feedback from them. We then run `iperf3` on the PCs to measure the TCP and UDP throughput under various configurations.

Due to antenna limitations, we use Wi-Fi channel 36 (5.18 GHz) and 80 MHz bandwidth. However, the results should scale to other channels. In our experimental setup, we mainly test the downlink traffic, *i.e.*, from AP to client. Since, in practical scenarios, most of the traffic is downlink (e.g., for mobile VR and video streaming), the setup is sufficient to demonstrate ScatterMIMO's overall benefits.

6 EVALUATION

In this section, we first present microbenchmarks to demonstrate the working of the *smart surface*, and then we evaluate the end-to-end performance of ScatterMIMO. We compare the performance of ScatterMIMO in terms of throughput and SNR gains over a baseline, which does not make use of the *smart surface*. We conduct multiple experiments at three different placement of the AP in a 50 meter $\times$ 30 meter floor plan of our office building. The client is placed randomly in the arena at 100 locations to create both line-of-sight and non-line-of-sight scenarios, as shown in Figure 8. For each client location, we perform measurements for two cases, one with the presence of *smart surface* and the other without it as a baseline.

Our analysis is based on two different measurement studies. First, we collect TCP throughput data and show throughput gain, coverage, and optimization latency achieved by ScatterMIMO. Furthermore, we log CSI data at the AP to calculate the received SNRs. In order to compare the experimental results analytically, we report the throughput gain defined as the throughput achieved by ScatterMIMO divided by the throughput achieved by the baseline. Similarly, we define the SNR gain (in dB) achieved by ScatterMIMO to compared against the baseline. We report the combined results across the three different AP placements as aggregate CDF plots.

The summary of the evaluation of ScatterMIMO and the key findings are as follows:

- ScatterMIMO improves median throughput gain by 2$\times$ compared to the baseline.
- ScatterMIMO can improve the SNR of spatial streams by 2 dB for SISO and can improve the overall MIMO second spatial stream by 5 dB while increasing the first spatial stream by 1 dB.
- ScatterMIMO increases the coverage from 30 meters to 45 meters in a typical office environment.
- ScatterMIMO's beam steering accuracy is within 0.5 dB of an ideal continuously tunable phased array.

6.1 Smart surface controllability

We first evaluate how a tile of antennas with different phase configurations affects the received signal power at the client. To verify this, we set up the AP, client, and a tile of the *smart surface* to be at the same height. For microbenchmarks, we use a four antenna AP and a single antenna client and place them 3 m apart in direct line-of-sight. The *smart surface* is placed 20 cm behind the AP facing towards both AP and client as shown in the setup of figure 8. We record the CSI for every *smart surface* configuration and compute the average SNR across all subcarriers. We also collect CSI when the *smart surface* is removed from the environment. Figure 9 plots the

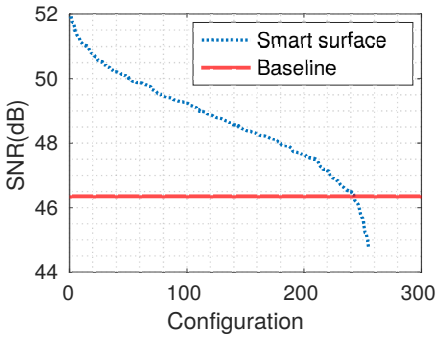


Figure 9: SNR at client for different *smart surface* configurations.

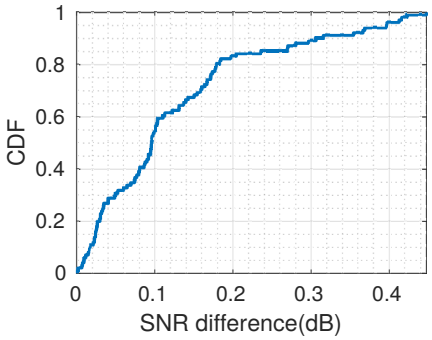


Figure 10: SNR difference between predicted best configuration and brute-force search

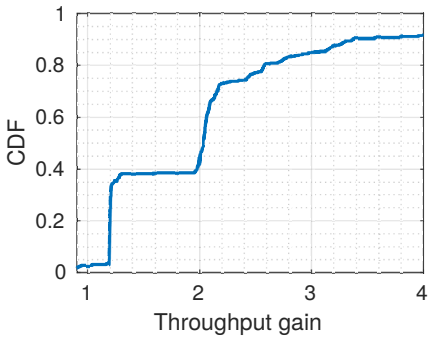


Figure 11: End-to-end TCP throughput gain over the baseline in the coverage area.

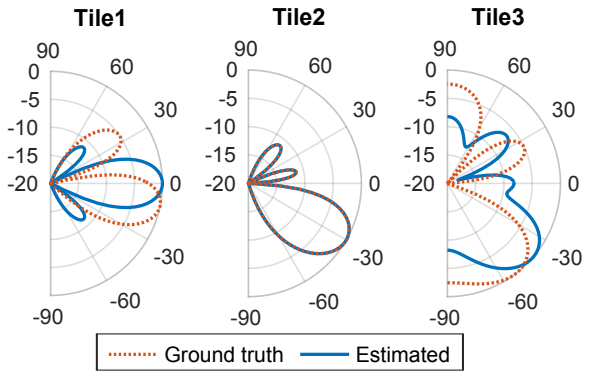


Figure 12: BeamPatterns of the three tiles of the *smart surface*.

SNR at the client for 256 *smart surface* configurations (one tile with 4 horizontal antennas each with 4 possible phase configurations) and compares it with the baseline (without *smart surface*). We observe that the presence of *smart surface* causes 7dB variations in the SNR across different configurations while outperforming baseline by 5.5 dB SNR gain. Thus, the best configuration can achieve SNR gain close to thereotical maximum of 6 dB. The results also validate that *smart surface* controls the amount of reflected power that reaches the client.

6.2 Direction sensing

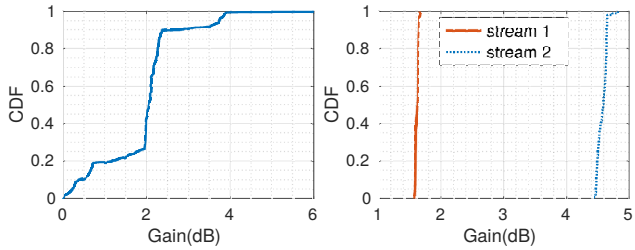
ScatterMIMO's key algorithmic component is to find parameter α that captures the relative direction of the client from the AP via the *smart surface*. ScatterMIMO needs to estimate α accurately to obtain the best phase configuration at the *smart surface*. Here we will show how well the array can deduce the client's direction and focus the power towards it. The test setup is similar to the one mentioned in the subsection 6.1 except that we now use all three tiles at the *smart surface* (4×12 antenna array). We collect CSI by setting the *smart surface* in different configurations as explained in section 3 and estimate α. We also obtain the true α by manually measuring the angular location of the AP and the client via the *smart surface*. The value of α is used to calculate the *smart surface*'s phase configuration that gives the directional beampatterns at the *smart surface*. Figure 12 plots the beampatterns that correspond to both true and estimated α for all three tiles. The ground truth beampatterns for each tile indicates the direction towards which the *smart surface* must rotate the reflected beam. We observe the peaks of the estimated beampatterns are within 10 degrees of the ground truth beampatterns. Notably, in the case of tile2, the estimated beampattern matches the ground truth beampattern.

6.3 ScatterMIMO's algorithm accuracy

Next, we will quantify how the accuracy of α affects ScatterMIMO's performance. We reuse the test setup from subsection 6.1, and probe CSI for over 100 trials. In each trial, we calculate the SNR at the client and report its cumulative distribution function (CDF). We compare ScatterMIMO's phase configuration algorithm with a brute-force approach. Recall that ScatterMIMO needs to try only three arrangements to estimate the best phase configuration at the *smart surface*. On the other hand, the brute-force search runs over 256 possibilities. We emphasize here that an exhaustive search over 16 antenna elements (each with 4 possible phases) in a tile would require 4^{16} probes, which would take months to run, and this complexity would increase exponentially with the addition of another tile. Therefore, we restrict the AP, client, and the *smart surface* to the same height, which reduces the number of trials to $4^4 = 256$ possibilities. In such a 2D setup, all the vertical antennas share the same phase. We observe in Figure 10 that the proposed algorithm performs as good as the brute-force method with less than 0.5 dB of the performance gap. The analysis substantiates our claim that ScatterMIMO achieves the required accuracy with high efficiency and low latency.

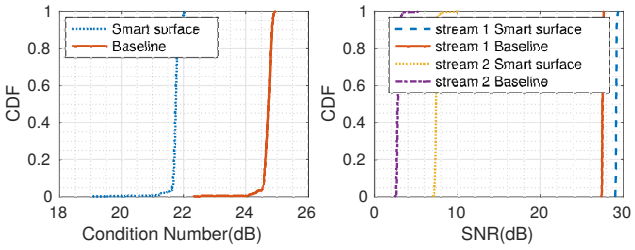
6.4 Throughput Gain

We now proceed to full system evaluation in terms of end-to-end throughput gain achieved by ScatterMIMO over the baseline. The AP always uses four antennas while the client can use either one or two antennas representing most commercial WiFi router and smartphone devices. We aggregate all the measurements across different scenarios and report a CDF plot of throughput gain in Figure 11. To show only the fair throughput gains, we consider the throughput measurements which fall under the coverage area and have an active data stream. Markedly, we notice a median throughput gain of 2× compared to the baseline (without the presence of *smart surface*). This evidence suggests that the *smart surface* has created a second path enabling an additional data stream effectively doubling the throughput. For the case of a single antenna client, the throughput improvements come because of the higher diversity gain of the single spatial stream to the client.



(a) SISO case (b) MIMO case

Figure 13: SNR gain over the baseline for two cases.



(a) Condition Number (b) Singular values

Figure 14: Singular Value and Condition number.

Additionaly, ScatterMIMO improves 95th-percentile throughput by 4×. The reason for this very high performance can be explained by the scenarios when the SNR is low on both the streams (notably, the non-line-of-sight). Introducing the *smart surface* in such a situation improves the SNR on both the streams realizing more than 2× gains, as shown in Figure 11. To shed more light on the SNR improvement of individual streams, we do the following CSI measurement study.

6.5 SNR Improvement

We will first investigate SNR improvement for a single antenna client (single stream or SISO case) and then explain for the client with two antennas (two spatial streams or MIMO case).

6.5.1 SISO SNR Improvement. From Figure 13a, we observe around 2 dB median SNR improvement over the baseline for a single stream between AP and client. The reason is that the *smart surface* created an additional strong path towards the client, which combines coherently with the direct path. Ideally, a reflected path that is as strong as the direct path would double the received signal amplitude, thus improving the SNR by at most 6 dB. However, we note that SNR gain starts saturating at 4 dB. The saturation happens because of the 2-bit discrete phase-shifters we use in a reflector unit. Quantized phase-shifters cannot always align the reflected path's phase with that of the direct path. Hence, the direct and reflected paths do not always add up constructively resulting in SNR gains less than 6 dB.

6.5.2 MIMO SNR Improvement. While the goal with a single antenna client was to improve the diversity gain, a MIMO system can essentially provide both diversity and multiplexing gain. We show that ScatterMIMO can improve upon both metrics by creating additional coherent paths for each stream in a MIMO system. Our results with two antenna client with two MIMO streams in Figure 13b shows an improvement of 1.5 dB for the first stream and 4.5 dB for the second stream. This suggests that the reflected path from *smart surface* improved the SNR of both the streams. The different amount of SNR improvement across two streams suggests that the rank of the channel is improved along with the diversity gain, which contributes to the overall 2× throughput gain, as we have seen in the last section. The result shows that ScatterMIMO can improve MIMO channel rank regardless of the client's location or orientation. Creating such a spatially diverse channel is a significant step towards achieving full MIMO capacity in realistic environments.

6.6 MIMO Condition Number Improvement

To further understand MIMO spatial multiplexing gains, we analyze the condition number of the MIMO channel. The condition number is defined as the ratio of the largest singular value to the smallest singular value for a narrowband system. For a wideband system, we take an average of singular values across all the subcarriers before taking the ratio. An ideal singular value of 0 dB means all the streams are equally strong. We observe from Figure 14a, a median improvement of 3 dB in condition number when compared to the baseline. This shows that the channel matrix has become well-conditioned as compared to the baseline. Now, to understand the improvement in condition number, we plot the average of singular values corresponding to stream 1 and stream 2 in Figure 14b. The median of stream 1's singular values has increased by 1.5 dB, whereas stream 2's median value has increased by 4.5 dB. This observation suggests that the *smart surface* has contributed to the improvement of SNR of both the orthogonal data streams and hence improved the MIMO multiplexing gain, as explained in Section 2.3.

6.7 Coverage & Latency

We define coverage as the region up to which the *smart surface* can provide throughput improvement over existing throughput. All our experiments are conducted on a large floor in an office building shown in Figure 8, and we classified the range at which we can provide improvement in throughput as the coverage span. We observe that we can provide the throughput improvement up to 45 meters of range with the *smart surface*. The baseline without the *smart surface* works up to only 30 meters of range. Finally, we qualitatively contrast with the recent works [9, 30] in Table 2 below, on the coverage and latency of optimization of the *smart surface*.

Properties	**ScatterMIMO**	**LAIA** [30]	**RFocus** [9]
# antenna elements	48	36	3720
Algorithm Latency	3 packets	72 packets	4000 packets
Coverage	upto 45m	8 m	30 m

Table 2: Comparing ScatterMIMO with LAIA [30] and RFocus [9]

7 RELATED WORK

Our work presents a unique direction where we enable a *virtual AP* using a *smart surface*. Our work is closely related to multiple avenues as follows:

Passive Surfaces based Network Improvements: In recent years, there has been a new line of research to improve wireless

networks using passive surfaces [19, 31, 33], smart cloths [14, 35, 37], smart walls [30, 55]. LAIA [30] is closely related to ScatterMIMO and is the first work to use an array of antennas to manipulate the radio environment. We build on top of LAIA to improve the channel but differ in the following aspects.

First, LAIA enables smart walls to improve channel by facilitating penetration of wireless signals across the walls. LAIA improves the channel by deploying antennas on either side of a wall and connecting them via a phase shifter. In contrast, ScatterMIMO enables a *smart surface* that achieves passive distributed MIMO by creating a *virtual AP*, which provides power and coverage close to a physically deployed AP.

Second, LAIA requires the number of channel estimates to be of the order of the number of antennas. On the other hand, ScatterMIMO leverages the geometry of the *smart surface* (*i.e.* antenna placement relative to each other) to design special *smart surface* configurations and requires only a constant number of channel estimates to optimize for phases. ScatterMIMO presents a suite of novel algorithms and mathematical frameworks for optimization of the general *smart surface*s with antennas. Thus ScatterMIMO is scalable to a vast antenna array without any significant overhead. In such sense, it can be generalized to use multiple *smart surface*s with adjustable location and orientations.

Finally, LAIA demonstrates throughput improvement using software-defined radios, while ScatterMIMO has shown to work on commercial off-the-shelf Wi-Fi devices as well.

Active Distributed MIMO: Distributed MIMO has been considered the holy grail for wireless communication, and actively powered approaches for distributed MIMO have been widely explored. Achieving Distributed MIMO requires synchronization, and a backhaul link between the transmitters to share the data for spatial multiplexing. MegaMIMO [23] uses over-the-air synchronization and an Ethernet link to share the data to transmit coherently to the client. Recent work on full-duplex relaying demonstrates the ability to achieve both synchronization and backhaul wirelessly [11, 13]. In contrast to these works, ScatterMIMO achieves a similar function passively, with both synchronization and backhaul implicitly guaranteed by the nature of reflection. Since the *smart surface* is placed in far-field, it can be considered a form of distributed MIMO.

Active Relay/Reflectors: Amplify-and-forward relays use active devices to extend the coverage of wireless network [60, 61]. WiFi extenders are also trending, which typically employs a decode-and-forward relays [32, 47]. MoVR [7] uses active millimeter-wave reflectors to relay communications and achieve a reliable link. It further leverages virtual reality location assistance co-located on the client to optimize the reflector. On the other hand, ScatterMIMO is passive, and it does not require the location of the client.

Backscatter communication: ScatterMIMO's underlying design leverages the designs of backscatter tags [12, 25, 27, 28, 45, 51, 59]. Though ScatterMIMO is not built for backscatter communications, however, ScatterMIMO presents a novel analysis that can potentially be adapted for designing MIMO backscatter tags. The insights from ScatterMIMO can be extended to solve two significant challenges that backscatter devices face, namely, to increase the throughput and the range of backscatter communications.

Meta-surfaces: Metasurfaces [15] are artificial surfaces composed of sub wavelength periodic patterns with properties generally not occuring in nature. Metasurface manipulates the electromagnetic wave properties such as amplitude [16, 38], phase [46, 48, 49, 57], polarization [57] upon interaction with electromagnetic waves. [20] presents a metasurface that achieves perfect anamolous reflection *i.e.* reflection angle is different from the angle of incidence. But metasurface design in [20] cannot be tuned to control the angle of reflection. To enable tunability, [42, 49] suggest the use of varactors as phase shifters in their design to steer the reflected wave. In contrast, ScatterMIMO's *smart surface* uses delay lines as phase shifters which are less lossy compared to varactor diodes and also have broader bandwidth.

8 LIMITATIONS AND FUTURE WORK

ScatterMIMO presents a design of *smart surface*, which can mimic a *virtual AP* and improve the throughput with a single *smart surface*. We also provide a low complexity optimization of the *smart surface* and scalable design principle for *smart surface*s. However, there is significant scope for improvement to extend wireless networking further. We have listed some limitations of ScatterMIMO, and future work to build upon it.

Network of *smart surface*s: Our patch antennas have a 120-degree field-of-view which can be extended by deploying several of these reflectors at multiple locations. Multiple *smart surface*s can create more than two additional paths to the user, thus improving the spatial multiplexing capability.

Multi-user MIMO: We proposed ScatterMIMO to improve the throughput of a MIMO system with one user at a time. Multiplexing multiple users has been studied extensively in the context of MU-MIMO and Massive MIMO. Our idea to deploy multiple *smart surface*s has potential applications for MU-MIMO in WiFi 6 [10].

Ultra Low power design: The current *smart surface* design's power consumption is of the order of mW, which is $1000\times$ lower compared to an active AP while providing comparable throughput gains. The power consumption can further be reduced by replacing our switches with low power alternatives (e.g., [43]). Also, a very low power wake-up receiver can be incorporated into the *smart surface* design activating the *smart surface* only when there are active packet transmissions.

Dual band *smart surface* design: We designed patch antennas to work at 5GHz with a bandwidth of 100 MHz covering five WiFi channels. The design can be further improved to incorporate additional 5GHz channels as well as extended to 2.4 GHz WiFi using dual-band antenna design techniques [58].

9 CONCLUSION AND DISCUSSION

We present a novel mathematical framework to design smart reflectors, which can as Virtual AP's and can be designed to be used for different application. Using this mathematical framework we design smart reflector which can act as a virtual AP with same transmit power as traditional AP, therefore providing spatial multiplexing and SNR improvement for all the clients within the coverage. Furthermore, we present the design and implementation of smart reflector evaluated with off-the-shelf COTS WiFi 4x4 AP. We present novel algorithms to optimize the smart reflector while compliant with protocol. We believe this work paves a way forward for the smart surface based wireless communications and networking.

REFERENCES

[1] 2015. Linksys Business Wireless-AC Dual-Band Access Points. http://downloads.linksys.com/downloads/datasheet/en/LAPAC1200_LAPAC1750_English.PDF.

[2] 2017. Cisco Small Business 550/560 Wireless Access Points Data Sheet. https://www.cisco.com/c/en/us/products/collateral/wireless/small-business-500-series-wireless-access-points/data_sheet_c78-727995.html.

[3] 2019. Adafruit Bluefruit LE UART Friend - Bluetooth Low Energy (BLE). https://www.adafruit.com/product/2479.

[4] 2019. Aruba 303 Series Campus Access Points Data Sheet. https://www.arubanetworks.com/assets/ds/DS_AP303Series.pdf.

[5] 2019. HMC247 400° Analog Phase Shifter. https://www.analog.com/media/en/technical-documentation/data-sheets/hmc247.pdf.

[6] 2019. UniFi® AC APs Datasheet. https://dl.ubnt.com/datasheets/unifi/UniFi_AC_APs_DS.pdf.

[7] Omid Abari, Dinesh Bharadia, Austin Duffield, and Dina Katabi. 2016. Cutting the cord in virtual reality. In *Proceedings of the 15th ACM Workshop on Hot Topics in Networks*. ACM, 162–168.

[8] Agilent. [n.d.]. Network Analyzer.

[9] Venkat Arun and Hari Balakrishnan. 2019. RFocus: Practical Beamforming for Small Devices. arXiv:cs.NI/1905.05130

[10] Boris Bellalta. 2016. IEEE 802.11 ax: High-efficiency WLANs. *IEEE Wireless Communications* 23, 1 (2016), 38–46.

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

[12] Dinesh Bharadia, Kiran Raj Joshi, Manikanta Kotaru, and Sachin Katti. 2015. Backfi: High throughput wifi backscatter. *ACM SIGCOMM Computer Communication Review* 45, 4 (2015), 283–296.

[13] Dinesh Bharadia and Sachin Katti. 2015. Fastforward: Fast and constructive full duplex relays. *ACM SIGCOMM Computer Communication Review* 44, 4 (2015), 199–210.

[14] Justin Chan, Anran Wang, Vikram Iyer, and Shyamnath Gollakota. 2018. Surface MIMO: Using Conductive Surfaces For MIMO Between Small Devices. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MobiCom '18)*. ACM, New York, NY, USA, 3–18. https://doi.org/10.1145/3241539.3241562

[15] Hou-Tong Chen, Antoinette J Taylor, and Nanfang Yu. 2016. A review of metasurfaces: physics and applications. *Reports on progress in physics* 79, 7 (2016), 076401.

[16] Sasmita Dash, Christos Liaskos, Ian F Akyildiz, and Andreas Pitsillides. 2019. Wideband Perfect Absorption Polarization Insensitive Reconfigurable Graphene Metasurface for THz Wireless Environment. In *2019 IEEE Microwave Theory and Techniques in Wireless Communications (MTTW)*, Vol. 1. IEEE, 93–96.

[17] Aitor Del Coso, Umberto Spagnolini, and Christian Ibars. 2007. Cooperative distributed MIMO channels in wireless sensor networks. *IEEE Journal on Selected Areas in Communications* 25, 2 (2007), 402–414.

[18] Analog Devices. [n.d.]. HMC7992: Nonreflective, Silicon SP4T Switch, 0.1 GHz to 6.0 GHz.

[19] Marco Di Renzo, Merouane Debbah, Dinh-Thuy Phan-Huy, Alessio Zappone, Mohamed-Slim Alouini, Chau Yuen, Vincenzo Sciancalepore, George C Alexandropoulos, Jakob Hoydis, Haris Gacanin, et al. 2019. Smart radio environments empowered by reconfigurable AI meta-surfaces: an idea whose time has come. *EURASIP Journal on Wireless Communications and Networking* 2019, 1 (2019), 129.

[20] Ana Díaz-Rubio, Viktar S Asadchy, Amr Elsakka, and Sergei A Tretyakov. 2017. From the generalized reflection law to the realization of perfect anomalous reflectors. *Science advances* 3, 8 (2017), e1602714.

[21] Antonio Forenza, Robert W Heath Jr, Stephen G Perlman, Roger van der Laan, and John Speck. 2009. System and method for distributed input distributed output wireless communications. US Patent 7,599,420.

[22] Andrea Goldsmith. 2005. *Wireless communications*. Cambridge university press.

[23] Ezzeldin Hamed, Hariharan Rahul, Mohammed A Abdelghany, and Dina Katabi. 2016. Real-time distributed MIMO systems. In *SIGCOMM*.

[24] Christopher Husmann, Georgios Georgis, Konstantinos Nikitopoulos, and Kyle Jamieson. 2017. FlexCore: Massively Parallel and Flexible Processing for Large {MIMO} Access Points. In *14th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 17)*. 197–211.

[25] Vikram Iyer, Vamsi Talla, Bryce Kellogg, Shyamnath Gollakota, and Joshua Smith. 2016. Inter-technology backscatter: Towards internet connectivity for implanted devices. In *SIGCOMM*.

[26] Richard C Johnson, H Allen Ecker, and J Searcy Hollis. 1973. Determination of far-field antenna patterns from near-field measurements. *Proc. IEEE* 61, 12 (1973), 1668–1694.

[27] Bryce Kellogg, Aaron Parks, Shyamnath Gollakota, Joshua R Smith, and David Wetherall. 2014. Wi-Fi backscatter: Internet connectivity for RF-powered devices. In *ACM SIGCOMM Computer Communication Review*.

[28] Bryce Kellogg, Vamsi Talla, Shyamnath Gollakota, and Joshua R Smith. 2016. Passive Wi-Fi: Bringing Low Power to Wi-Fi Transmissions.. In *NSDI*.

[29] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. 2015. SpotFi: Decimeter Level Localization Using Wi-Fi *(SIGCOMM)*.

[30] Zhuqi Li, Yaxiong Xie, Longfei Shangguan, Rotman Ivan Zelaya, Jeremy Gummeson, Wenjun Hu, and Kyle Jamieson. 2019. Towards Programming the Radio Environment with Large Arrays of Inexpensive Antennas. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*.

[31] Christos Liaskos, Shuai Nie, Ageliki Tsioliaridou, Andreas Pitsillides, Sotiris Ioannidis, and Ian Akyildiz. 2019. A novel communication paradigm for high capacity and security via programmable indoor wireless environments in next generation wireless systems. *Ad Hoc Networks* 87 (2019), 1–16.

[32] Linksys. [n.d.]. AC750 Boost Range Extender.

[33] Fu Liu, Odysseas Tsilipakos, Alexandros Pitilakis, Anna C Tasolamprou, Mohammad Sajjad Mirmoosa, Nikolaos V Kantartzis, Do-Hoon Kwon, Maria Kafesaki, Costas M Soukoulis, and Sergei A Tretyakov. 2019. Intelligent metasurfaces with continuously tunable local surface impedance for multiple reconfigurable functions. *Physical Review Applied* 11, 4 (2019), 044024.

[34] Mattias Fridström. 2017. The bandwidth problem: 5 issues the VR industry must resolve. https://venturebeat.com/2017/05/06/the-bandwidth-problem-5-issues-the-vr-industry-must-resolve/.

[35] Jane McCann and David Bryson. 2009. *Smart clothes and wearable technology*. Elsevier.

[36] Konstantinos Nikitopoulos, Juan Zhou, Ben Congdon, and Kyle Jamieson. 2014. Geosphere: Consistently turning MIMO capacity into throughput. In *ACM SIGCOMM Computer Communication Review*, Vol. 44. ACM, 631–642.

[37] Alex Pentland. 1998. Smart rooms, smart clothes. In *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No. 98EX170)*, Vol. 2. IEEE, 949–953.

[38] George Perrakis, Odysseas Tsilipakos, George Kenanakis, Maria Kafesaki, Costas M Soukoulis, and Eleftherios N Economou. 2019. Perfect optical absorption with nanostructured metal films: design and experimental demonstration. *Optics express* 27, 5 (2019), 6842–6850.

[39] Qualcomm Technologies, Inc. 2018. VR and AR pushing connectivity limits. https://www.qualcomm.com/media/documents/files/vr-and-ar-pushing-connectivity-limits.pdf.

[40] Quantenna. [n.d.]. Quantenna 802.11ac WiFi Card. http://www.quantenna.com/.

[41] R. Schmidt. 1986. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation* 34, 3 (March 1986), 276–280. https://doi.org/10.1109/TAP.1986.1143830

[42] Daniel F Sievenpiper, James H Schaffner, H Jae Song, Robert Y Loo, and Gregory Tangonan. 2003. Two-dimensional beam steering using an electrically tunable impedance surface. *IEEE Transactions on antennas and propagation* 51, 10 (2003), 2713–2722.

[43] SkyWorks. [n.d.]. SKY13575-639LF: Dual-Band Matched SP4T Wi-Fi Switch.

[44] Sanjib Sur, Ioannis Pefkianakis, Xinyu Zhang, and Kyu-Han Kim. 2016. Practical MU-MIMO user selection on 802.11 ac commodity networks. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. ACM, 122–134.

[45] Vamsi Talla, Mehrdad Hessar, Bryce Kellogg, Ali Najafi, Joshua R Smith, and Shyamnath Gollakota. 2017. LoRa backscatter: Enabling the vision of ubiquitous connectivity. *IMWUT* (2017).

[46] SN Tcvetkova, VS Asadchy, A Díaz-Rubio, D-H Kwon, and SA Tretyakov. 2018. Multi-channel reflectors: Versatile performance experimentally tested. In *2017 11th International Congress on Engineered Materials Platforms for Novel Wave Phenomena (Metamaterials)*. IEEE, 346–348.

[47] TP-Link. [n.d.]. RE450 1750 Wi-Fi Range Extender.

[48] Odysseas Tsilipakos, Thomas Koschny, and Costas M Soukoulis. 2018. Antimatched electromagnetic metasurfaces for broadband arbitrary phase manipulation in reflection. *ACS photonics* 5, 3 (2018), 1101–1107.

[49] Odysseas Tsilipakos, Fu Liu, Alexandros Pitilakis, Anna C Tasolamprou, D-H Kwon, Mohammad Sajjad Mirmoosa, Nikolaos V Kantartzis, Eleftherios N Economou, Maria Kafesaki, Costas M Soukoulis, et al. 2018. Tunable perfect anomalous reflection in metasurfaces with capacitive lumped elements. In *2018 12th International Congress on Artificial Materials for Novel Wave Phenomena (Metamaterials)*. IEEE, 392–394.

[50] Deepak Vasisht, Swarun Kumar, and Dina Katabi. 2016. Decimeter-Level Localization with a Single Wi-Fi Access Point *(NSDI)*.

[51] Anran Wang, Vikram Iyer, Vamsi Talla, Joshua R Smith, and Shyamnath Gollakota. 2017. FM Backscatter: Enabling Connected Cities and Smart Fabrics.. In *NSDI*. 243–258.

[52] Dongming Wang, Jiangzhou Wang, Xiaohu You, Yan Wang, Ming Chen, and Xiaoyun Hou. 2013. Spectral efficiency of distributed MIMO systems. *IEEE Journal on Selected Areas in Communications* 31, 10 (2013), 2112–2127.

[53] Shu Wang, Vignesh Venkateswaran, and Xinyu Zhang. 2017. Fundamental analysis of full-duplex gains in wireless networks. *IEEE/ACM Transactions on Networking* 25, 3 (2017), 1401–1416.

[54] Teng Wei and Xinyu Zhang. 2016. Random access signaling for network MIMO uplink. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE, 1–9.

[55] Allen Welkie, Longfei Shangguan, Jeremy Gummeson, Wenjun Hu, and Kyle Jamieson. 2017. Programmable radio environments for smart spaces. In *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*. ACM, 36–42.

[56] Jie Xiong and Kyle Jamieson. 2013. ArrayTrack: A Fine-Grained Indoor Location System. In *Proceedings of the 10th USENIX conference on Networked Systems Design and Implementation (nsdi'13)*. USENIX Association, Berkeley, CA, USA, 71–84. http://dl.acm.org/citation.cfm?id=2482626.2482635

[57] Huanhuan Yang, Xiangyu Cao, Fan Yang, Jun Gao, Shenheng Xu, Maokun Li, Xibi Chen, Yi Zhao, Yuejun Zheng, and Sijia Li. 2016. A programmable metasurface with dynamic polarization, scattering and focusing control. *Scientific reports* 6 (2016), 35692.

[58] Jiachen Yang, Huanling Wang, Zhihan Lv, and Huihui Wang. 2016. Design of miniaturized dual-band microstrip antenna for WLAN application. *Sensors* 16, 7 (2016), 983.

[59] Pengyu Zhang, Dinesh Bharadia, Kiran Joshi, and Sachin Katti. 2016. Hitchhike: Practical backscatter using commodity wifi. In *SenSys*.

[60] Y. Zhao, R. Adve, and T. J. Lim. 2006. Improving Amplify-and-Forward Relay Networks: Optimal Power Allocation versus Selection. In *2006 IEEE International Symposium on Information Theory*. 1234–1238. https://doi.org/10.1109/ISIT.2006.262002

[61] Xia Zhou, Zengbin Zhang, Yibo Zhu, Yubo Li, Saipriya Kumar, Amin Vahdat, Ben Y Zhao, and Haitao Zheng. 2012. Mirror mirror on the ceiling: Flexible wireless links for data centers. *ACM SIGCOMM Computer Communication Review* 42, 4 (2012), 443–454.