

RESEARCH

Open Access



# Fast compression of OFDM channel state information with constant frequency sinusoidal approximation

Avishek Mukherjee and Zhenghao Zhang\*

## Abstract

We propose CSIAPx, a very fast and lightweight method to compress the channel state information (CSI) of Wi-Fi networks. CSIAPx approximates the CSI vector as the linear combination of a small number of base sinusoids on constant frequencies and uses the complex coefficients of the base sinusoids as the compressed CSI. While it is well-known that the CSI vector can be represented as the linear combination of sinusoids, fixing the frequencies of the sinusoids is the key novelty of CSIAPx, which is guided by our mathematical finding that almost any sinusoid can be approximated by a set of base sinusoids on constant frequencies. CSIAPx enjoys very low computation complexity, because key steps in the compression can be pre-computed due to the constant frequencies of the base sinusoids. We extensively test CSIAPx with both experimental and synthesized Wi-Fi channel data, and the results confirm that CSIAPx can achieve very good compression ratio with little loss of accuracy.

**Keywords:** Channel state information, Compression, Approximation

## 1 Introduction

In Wi-Fi, the channel state information (CSI) for an antenna pair is a vector of complex numbers, representing the channel coefficients of the orthogonal frequency division multiplexing (OFDM) subcarriers. The CSI is needed to calculate the modulation parameters for techniques such as multi-user multiple-input-multiple-output (MU-MIMO). In Wi-Fi, the CSI is typically measured at the receiver and is transmitted back to the sender, which requires significant overhead. For example, on a 20-MHz channel with 64 subcarriers, the full CSI for a single antenna pair has 64 complex numbers, and for 9 antenna pairs, 576. Although Wi-Fi does not use all subcarriers, the feedback for 9 antenna pairs still may exceed 1000 bytes. The Wi-Fi standard [1] defines options to compress the CSI, such as reducing the quantization accuracy or the number of subcarriers in the feedback or using the Givens rotation on the  $V$  matrix after the singular value decomposition (SVD) of the CSI matrix. However, these methods either reduce the accuracy of the CSI or only achieve modest compression ratios. For example, a 3 by

3 complex  $V$  matrix can only be compressed into 6 real numbers, at a compression ratio of 3.

In this paper, we propose CSIAPx, a novel CSI compression method for Wi-Fi networks with the following key features: (a) *high compression ratio*, e.g., capable of compressing the CSI of 40 subcarriers in most of our Wi-Fi experiments into just 6 or less complex numbers; (b) *little loss of accuracy*, e.g., the decompressed CSI is very close to the measured CSI; and (c) *very low computation complexity*, suitable for hardware implementation. Compared to the Givens rotation, CSIAPx achieves higher compression ratio, e.g., more than 2.5 times higher with our experimental data. Compared to other recent solutions in the literature [2–4], CSIAPx has much lower computation complexity, because its main computation is simply the dot products between the CSI and a small number of constant vectors.

CSIAPx is based on the well-known fact that the CSI is the linear combination of sinusoids [3, 4], where each sinusoid is the result of a physical path in the channel. Previous approaches solve complex optimization problems to find the parameters of the physical paths, i.e., the frequencies, the phases, and the amplitudes of the sinusoids. Radically different from such solutions, CSIAPx

\*Correspondence: [zzhang@cs.fsu.edu](mailto:zzhang@cs.fsu.edu)  
Department of Computer Science, Florida State University, Tallahassee, FL, USA

does not attempt to find the parameters of the physical paths. Instead, CSIApx uses the linear combination of a set of *base sinusoids*, which are on *fixed frequencies*, to *approximate* the CSI, and our results show that the approximation usually achieves very high accuracy. This approach is guided by our mathematical finding that *the linear combination of sinusoids on constant frequencies can approximate any given sinusoid very well*, which is explained in more details in Section 3. Roughly speaking, as each individual sinusoid can be approximated by just a small number of base sinusoids, the entire CSI, which is the summation of the individual sinusoids, can also be approximated, *regardless of the number and characteristics of actual paths in the channel*. Working with base sinusoids on fixed frequencies has two major benefits. First, it avoids solving complex optimization problems and dramatically reduces the computation complexity. Second, as the frequency values of the base sinusoids are constants, they do not need to be transmitted, further improving the compression ratio. While CSIApx is primarily designed for compressing the CSI of Wi-Fi channels, it will theoretically work on any OFDM-based system that measures the channel state information, as long as the delay spread is not too large.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 explains the theoretical foundations. Section 4 describes CSIApx. Section 5 describes the evaluation. Section 6 concludes the paper.

## 2 Related work

CSI compression has been a major topic of interest due to its practical importance in wireless networks. The differences between CSIApx and the existing methods in Wi-Fi have been discussed in Section 1. Some early approaches [5–7] use quantization and general purpose compression techniques like the Huffman coding, with typical compression ratios around 3:1, lower than that with CSIApx. Another popular approach is to reduce the frequency or the amount of CSI feedback in slow-varying channels similar to those in [8, 9], which complements compression techniques such as CSIApx. AFC [10] chooses from existing compression options for the least degradation of link throughput, which can be complemented by CSIApx as an additional compression option. The problem of consolidating CSI from a small number of settings to predict the CSI under other settings has been studied in [11], which is different from CSI compression.

Sparsity in certain wireless channels has been well-known [3, 4, 12–14] and has been exploited in applications such as CSI compression and CSI estimation. Unlike existing work that usually still attempts to find the actual paths by solving optimization problems, CSIApx focuses on *approximating* the CSI with *constant* frequency sinusoids, which leads to much lower computation complexity than

the existing algorithms such as CTDP [3, 4]. CSIApx also has much lower complexity than CSIFit [2], which uses the Levenberg-Marquardt (LM) algorithm to compress the CSI of MIMO channels, because the LM algorithm needs to solve a set of linear equations in each iteration. R2-F2 [15] uses the CSI measured on one direction of the wireless link to predict that of the other, which requires much more computation than CSIApx; in addition, R2-F2 depends on *channel reciprocity*, which may not be true depending on the hardware circuitry, and may also need periodical calibrations [16], which is why Wi-Fi defines explicit CSI feedback and does not solely depend on channel reciprocity to obtain the CSI.

We have presented an initial version of CSIApx in [17] and have obtained a patent [18]. Compared to the conference version, this paper contains significant improvements in the theoretical foundation and the compression ratio in practice.

## 3 Theoretical foundation

We prove that a *target sinusoid* on frequency  $g$  can be approximated as the linear combination of  $P$  *base sinusoids* and the approximation error decays exponentially fast as the number of base sinusoids increases, where the frequencies of the base sinusoids are constants. Therefore, the summation of many sinusoids, such as the CSI vector, can still be approximated as the linear combination of only  $P$  base sinusoids.

### 3.1 Approximating real sinusoids

We begin with a theorem on the approximation of real sinusoids.

**Theorem 1** *Consider  $P$  base sinusoids on evenly spaced frequencies:  $\{\cos(0x), \cos(\delta x), \dots, \cos((P-1)\delta x)\}$ . Suppose  $\cos(gx)$  is to be approximated as the linear combination of the base sinusoids for  $x \in [0, X]$ , where  $0 \leq g \leq (P-1)\delta$ . If  $X\delta < 1$ , there exists an approximation with error decaying exponentially fast as  $P$  increases.*

*Proof* Consider any fixed time instant  $x_0 \in [0, X]$  and the function  $G(g) = \cos(x_0 g)$  where  $g$  is the variable. Let

$$F(g) = \sum_{k=1}^P \gamma_k \cos[x_0(k-1)\delta],$$

where

$$\gamma_k = \prod_{h=1, h \neq k}^P \frac{(h-1)\delta - g}{(h-k)\delta}.$$

and is called the *coefficient* of base sinusoid  $k$ .  $F(g)$  is the Lagrange interpolation of function  $G()$  at  $g$  based on the values of  $G()$  at  $0, \delta, \dots, (P-1)\delta$ . As function

$G()$  is infinitely differentiable, the Lagrange interpolation guarantees that

$$|F(g) - G(g)| \leq \left| \frac{G^{P+1}(\xi) \prod_{k=1}^P [(k-1)\delta - g]}{(P+1)!} \right|,$$

for some  $\xi$  within  $[0, X]$ . Clearly,

$$|G^{P+1}(\xi)| \leq X^{P+1},$$

and we claim that

$$\left| \prod_{k=1}^P [(k-1)\delta - g] \right| \leq \delta^P (P-1)!$$

To see this, suppose  $g \in [(k-1)\delta, k\delta]$  for some  $k$ . Clearly,

$$\begin{aligned} \left| \prod_{k=1}^P [(k-1)\delta - g] \right| &\leq \prod_{h=1}^k [k\delta - (h-1)\delta] \prod_{h=k+1}^P [(h-1)\delta - (k-1)\delta] \\ &\leq \delta^P k! (P-k)! \\ &\leq \delta^P (P-1)! \end{aligned}$$

Therefore,

$$|F(g) - G(g)| \leq \frac{X^{P+1} \delta^P}{P^2 + P}.$$

Clearly, if  $X\delta < 1$ , the interpolation error decays exponentially fast as  $P$  increases. Finally, note that the argument is true for any point in  $[0, X]$ .  $\square$

### 3.2 Extensions

We now discuss a few extensions.

*Extension to  $\sin(gx)$*  : With exactly the same arguments, it can be proved that  $\sin(gx)$  can be approximated as the linear combination of the base sinusoids. Clearly, for the approximation of  $\sin(gx)$ , exactly the same coefficients as the coefficients for the approximation of  $\cos(gx)$  can be used, because the coefficients are determined only by  $g$  and the frequencies of the base sinusoids.

*Extension to  $[-X, X]$*  : The extension to  $[-X, X]$  is immediate. That is, if the approximation matches  $\cos(gx)$  in  $[0, X]$ , multiplying the base sinusoids by the same coefficients for  $x$  in  $[-X, 0]$  should also result in a match, because the target sinusoid and the base sinusoids have the same parity.

*Extension to  $e^{igx}$*  : The extension to complex wave  $e^{igx}$  is clearly

$$\sum_{k=1}^P \gamma_k \cos(f_k x) + i \sum_{k=1}^P \gamma_k \sin(f_k x) = \sum_{k=1}^P \gamma_k e^{if_k x}.$$

### 3.3 The summation of many sinusoids

So far, we considered approximating one target sinusoid. The CSI vector, on the other hand, may be the

summation of many sinusoids. However, if the base sinusoids are selected correctly, any individual sinusoid in the CSI vector can be approximated, and therefore, the summation can also be approximated with the same set of base sinusoids. The deterministic maximum approximation error will be the summation of all individual approximation errors and may be large. However, in practice, the sinusoids in the CSI have random phases and the errors almost never add up constructively. A probabilistic bound therefore is more suitable; however, it depends on the assumptions of the path delay and power distribution. We instead choose to use both the real-world data and synthesized data to evaluate CSIApx and the results show that CSIApx approximates the CSI very well.

## 4 CSIApx

CSIApx is a fast compression method based on our theoretical findings. We begin with an overview.

### 4.1 Overview

According to our theoretical findings, the CSI can be approximated as the linear combination of the base sinusoids. The coefficients of the base sinusoids in the linear combination can be found by minimizing the *fit residual*, defined as the total squared error between the CSI vector and the approximation. The approximation is therefore called an *MSE Fit* and requires very low computation complexity, mainly because the linear system to be solved in the optimization problem is defined by a constant matrix. As any sinusoid can be approximated in this manner, the simplest approach would be to select just one set of base sinusoids to be used for all CSI. However, different types of channels have different delay spreads, which translate to different frequency ranges of the sinusoids in the CSI, the larger the delay spread, the higher the frequency. As sinusoids on lower frequencies can be approximated with fewer base sinusoids, to further improve the compression ratio, a small number of *configurations* are used in CSIApx which have different *orders*, where the order refers to the number of the base sinusoids. CSIApx finds the MSE Fit coefficients for all configurations and selects a configuration, considering both the compression ratio and the fit residual. The computation complexity is also reduced by sharing certain base sinusoids among multiple configurations.

### 4.2 The MSE Fit

#### 4.2.1 Finding the MSE Fit

The core of CSIApx is to find the coefficients of the MSE Fit, denoted as a vector  $\Gamma$ .

To minimize the squared error is to select coefficients to minimize

$$J = \sum_{j=1}^N \left| \left( \sum_{k=1}^P \gamma_k e^{ijf_k} \right) - y_j \right|^2,$$

where  $P$  is the order;  $\gamma_k$  and  $f_k$  are the coefficient and frequency of base sinusoid  $k$ , respectively;  $y_j$  is element  $j$  in the CSI vector  $Y$ ; and  $N$  is the length of the CSI vector. By taking the derivatives of  $J$  with respect to the coefficients and setting them to 0,  $\Gamma$  that minimizes  $J$  is the solution to a linear system  $Q\Gamma = S$ , where:

- $Q$  is a  $P$  by  $P$  matrix, in which  $q_{k,h} = \sum_{j=1}^N e^{i(f_h - f_k)j}$ ,
- $S$  is a  $P$  by 1 vector, in which  $s_k = \sum_{j=1}^N e^{-if_k j} y_j$ .

It can be seen that  $S$  is determined by the CSI vector, but is just the dot products of the CSI vector and the conjugate of the base sinusoids. On the other hand, as the frequency values are constants,  $Q^{-1}$  is a constant matrix and can be pre-computed. Therefore, after  $S$  is obtained,  $\Gamma$  can be found by simply multiplying the constant matrix  $Q^{-1}$  with  $S$ .

### 4.3 The compression process of CSIApx

CSIApx has  $U$  configurations. Configuration  $u$  is defined by the frequencies of its  $P_u$  base sinusoids, denoted as  $f_{u,k}$  for  $k = 1, 2, \dots, P_u$ , where the frequency is the amount of angular rotation between neighboring points in the CSI vector. For each configuration, CSIApx solves the constant linear system in Section 4 to get the coefficients of the MSE Fit. To evaluate the quality of the fits, for each configuration, CSIApx evaluates the MSE Fit on  $L$  evenly spaced sample points, where  $L = \frac{N}{4}$ , and finds the total fit residual on the sampled points, denoted as  $\eta_u$  for configuration  $u$ . The MSE Fit is not evaluated on all points to reduce the computation complexity. CSIApx selects the fit coefficients of configuration  $u$  as the compressed CSI, if  $u$  is the smallest index satisfying  $\eta_u < \zeta \min\{\eta_1, \eta_2, \dots, \eta_U\}$ , where  $\zeta$  is a constant, empirically chosen as 1.75 and 4 for CSI with 64 and 40 subcarriers, respectively.

### 4.4 The configurations and the base sinusoids

The configurations and base sinusoids should be selected considering compression ratio, accuracy, implementation cost, and the range of the fit coefficients. As Wi-Fi has a fixed number of subcarriers and well-studied types of channels [19], we select a configuration for each type of channel.

#### 4.4.1 Finding parameters for a given channel type

For a given type of channel, the parameters to be determined include the number of base sinusoids, denoted here as  $P$ , and the frequencies of each base sinusoid, denoted here as  $f_1, f_2, \dots, f_P$ . Note that  $f_1$  is actually always 0 to be able to cover the lowest frequency. Our approach is to first solve the problem of determining the frequencies of the

base sinusoids for a given  $P$ , then conduct a linear scan on  $P$  to find the best  $P$ . As the linear scan is straightforward, in the following, we focus on the first problem.

As a channel has multiple paths, while paths of different delays may not have equal power, the selection of the base sinusoids will have to be based on the power profile of the channel. For example, if 90% of the power is concentrated within a certain small delay range, more base sinusoids should be allocated to the corresponding range of frequencies. The power profile is generated using the delay taps and relative power values for the given channel model. It also takes into account possible imperfect symbol-level synchronization, which is assumed to be uniformly within 0 to 50 ns, because such synchronization error will increase the frequencies of the sinusoids.

The problem of finding the base sinusoid frequencies for given  $P$  is solved using a standard solver like Levenberg-Marquardt. First, from 0 to the maximum frequency in the power profile,  $H$  evenly spaced frequencies are selected, denoted as  $g_1$  to  $g_H$ . The objective function passed to the solver basically minimizes the total fit residual of all sinusoid on such frequencies, which are weighted according to the power profile. By doing so, the solver should find a set of base sinusoids that is expected to minimize the fit residual of this type of channel. To be more specific, the objective function is defined as follows with  $f_2, f_3, \dots, f_P$  as parameters:

$$\sum_{s=1}^H F_{f_2, f_3, \dots, f_P}(g_s) W(g_s),$$

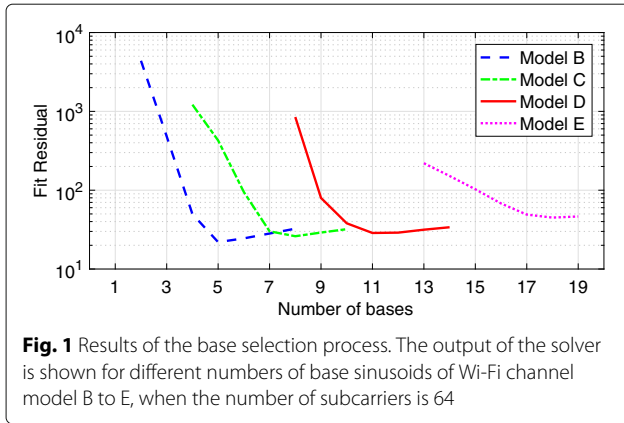
where  $F_{f_2, f_3, \dots, f_P}(g_s)$  is the MSE Fit residual of a pure sinusoid of frequency  $g_s$  when the base sinusoid frequencies are  $0, f_2, f_3, \dots, f_P$ , and  $W(g_s)$  is the power profile weight of frequency  $g_s$ . It should be noted that some noise is added to the pure sinusoids such that the signal-to-noise ratio (SNR) is 20 dB, to emulate a real-world scenario.

#### 4.4.2 Results

Figure 1 shows, in log scale, the output of the solver for different numbers of base sinusoids of Wi-Fi channel model B to E, when the number of subcarriers is 64. It can be seen that the fit residual decreases exponentially as the number of bases increases up to a point, after which there is very little change to the final fit residual. In fact in some cases, the fit residual increases slightly with more bases. This is due to over-fitting, i.e., the MSE Fit trying to follow the noise in the data. Based on this result, the selected numbers of bases for each configuration are 5, 7, 11, and 17, respectively.

#### 4.4.3 Manual tuning

It should be noted that two types of channel models, namely model A and model E, are excluded from this study. Channel model A is an ideal case, i.e., modeling a



channel with a single path, which we have covered with an additional simple configuration with three base sinusoids. Channel model F is excluded because it requires a large number of base sinusoids to approximate and as such does not bring in much in terms of compression. Moreover, based on our experience, model F rarely occurs in indoor Wi-Fi networks.

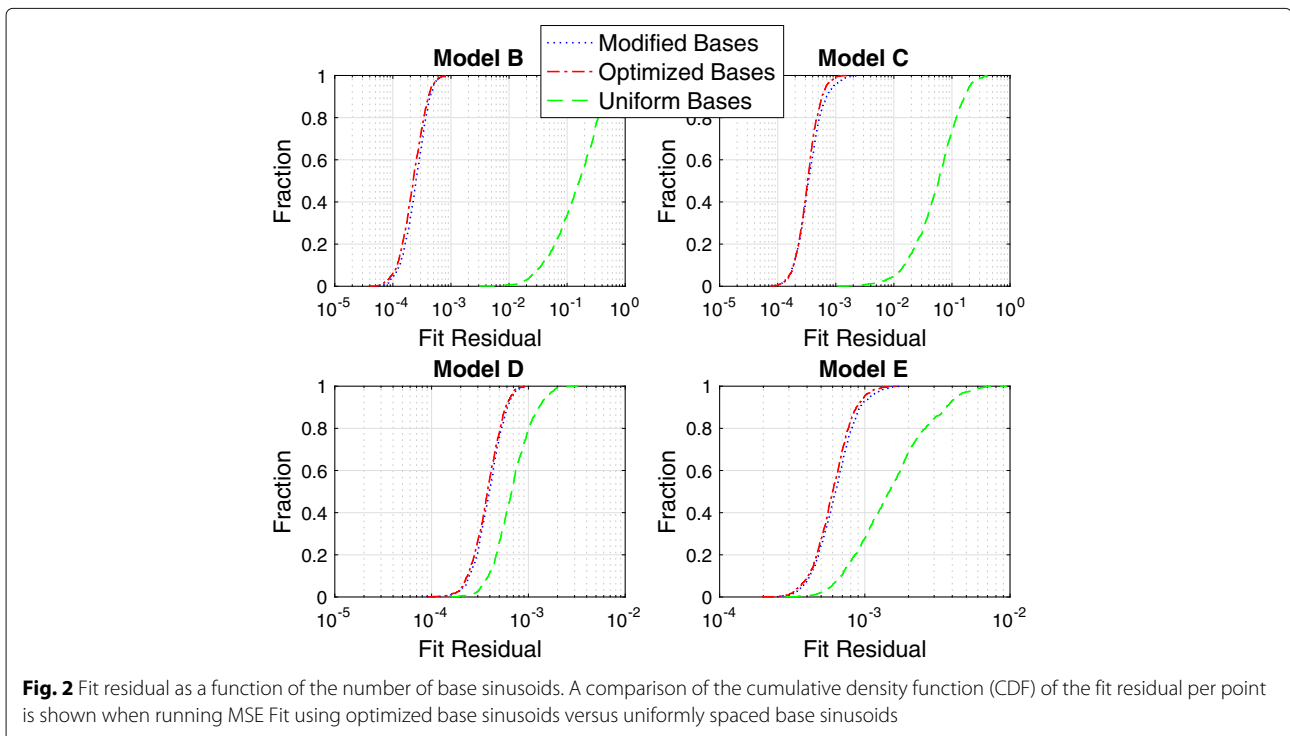
Figure 2 shows, in log scale, a comparison of the cumulative density function (CDF) of the fit residual per point when running MSE Fit using these optimized base sinusoids as opposed to using uniformly spaced base sinusoids. These tests were performed on 1000 CSI vectors with 20-dB SNR for each channel model. It can be seen that the optimized base sinusoids often resulted in an

order of magnitude or greater reduction in the approximation error.

We note that the main computation cost in CSIApx is actually finding the dot products between the base sinusoids and the CSI. If a base sinusoid used in one configuration can be used by another, the computation cost can be reduced.

Therefore, we slightly modified the optimized base sinusoid frequencies, to allow one base sinusoid to be used for more than one configuration when possible. For 64 subcarriers, the selected base frequencies for the five configurations are {0, 0.06, 0.12}, {0, 0.05, 0.1, 0.15, 0.25}, {0, 0.06, 0.12, 0.18, 0.24, 0.3, 0.42}, {0, 0.06, 0.12, 0.18, 0.24, 0.3, 0.36, 0.42, 0.525, 0.6375, 0.75}, and {0, 0.075, 0.15, 0.225, 0.3, 0.375, 0.45, 0.525, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3}, respectively. In total, there are 27 unique base sinusoids. Figure 1 also shows the CDF of the fit residual when running MSE Fit with this modified set of base sinusoids. It can be seen that there is a small difference under 10% when compared against the optimized base frequencies. However, these modified configurations cut down the complexity of CSIApx considerably. It should also be noted that for channel model E, the number of base sinusoids was reduced from 17 to 16, as there is very little loss in accuracy at 20 dB, but leads to a higher compression ratio.

We repeated the same process also for CSI vectors of 40 subcarriers, which is used in our experiments. The five configurations are {0, 0.05, 0.10}, {0, 0.06, 0.12, 0.2},



{0, 0.075, 0.15, 0.225, 0.3, 0.45}, {0, 0.075, 0.15, 0.225, 0.3, 0.375, 0.525, 0.675, 0.825, 0.975}, and {0, 0.09, 0.18, 0.27, 0.36, 0.45, 0.575, 0.7, 0.825, 0.95, 1.075, 1.2, 1.325, 1.45}, respectively. In total, there are 27 unique base sinusoids.

#### 4.5 Complexity of CSI<sub>Apx</sub>

Overall, let  $W$  be the total number of unique base sinusoids in all configurations; the complexity of CSI<sub>Apx</sub>, measured by the number of complex multiplications, includes only:

- $WN$ : for computing the dot products between the base sinusoids and the CSI
- $\sum_{u=1}^U P_u^2$ : for finding the fit coefficients of all configurations
- $\sum_{u=1}^U (P_u + 1)L$ : for computing the fits at sampled points and the sampled fit residuals

#### 4.6 Coping with shift frequency

In practice, the raw measured CSI often has a *shift frequency*, which is a frequency value added to the frequencies of all sinusoids, caused by the sample timing offset to the OFDM symbol boundary. The shift frequency needs to be removed before running CSI<sub>Apx</sub>, because it may force CSI<sub>Apx</sub> to choose higher configurations to approximate sinusoids on higher frequencies and reduce the compression ratio. This can easily be achieved by multiplying the CSI with a sinusoid on the negative of the shift frequency, a process we call *rotation*. The value of the shift frequency is known to the wireless receiver, because it selects the OFDM symbol boundary. The frequency used in the rotation can also be slightly adjusted to make sure that the sinusoids in the CSI are still on positive frequencies after the rotation.

### 5 Results and discussion

We report our evaluation of CSI<sub>Apx</sub> on both real-world and synthesized CSI data in this section.

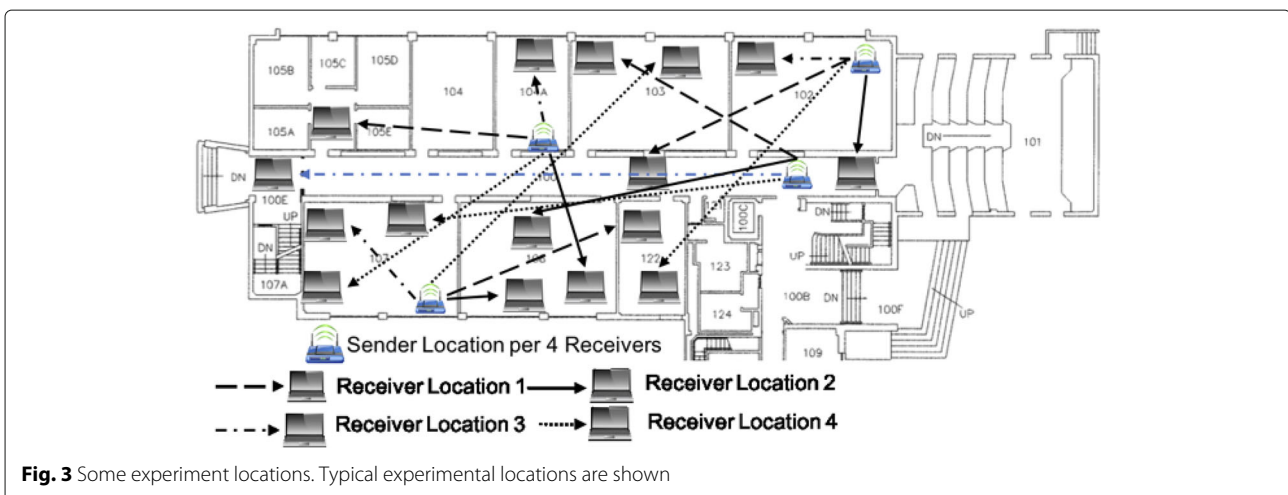
#### 5.1 Evaluation with the experimental CSI data

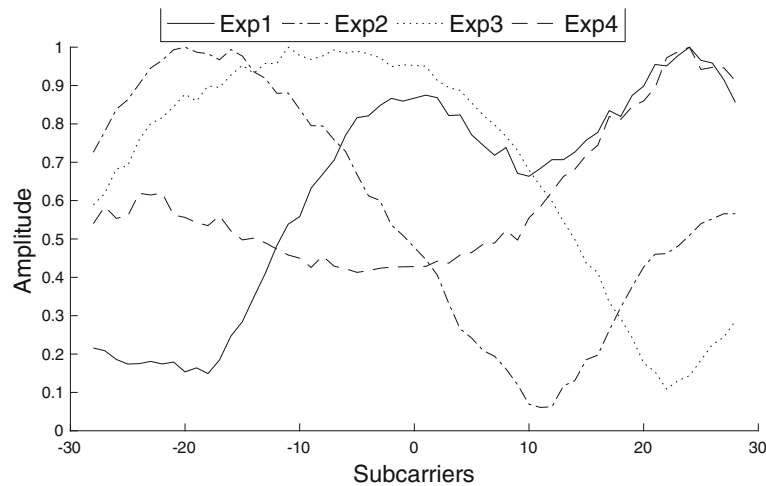
We first discuss our evaluation of CSI<sub>Apx</sub> with the real-world experimental CSI data.

##### 5.1.1 Data collection and preprocessing

CSI data was collected using the Atheros CSITool [20] installed on two laptops with the Atheros AR9462 wireless card with two antennas on 20-MHz channels. A total of 100 experiments in various location settings were conducted, which include typical environments like office buildings, apartment complexes, and large hallways. The experiments include both line of sight and non-line of sight cases as well as varying channel conditions due to human movements near the machines. Some of the experiment locations are shown in Fig. 3.

The CSITool reports the CSI on 56 selected subcarriers for four antenna pairs. Figure 4 shows the absolute values of some raw CSI vectors, where it can be seen that the data has some level of noise. A few preprocessing steps were taken before passing the data for compression. Firstly, as the signal always seems to be attenuated at both ends of the spectrum, caused most likely by additional filtering in hardware, not representing the characteristics of the actual channel, 8 subcarriers on both ends are removed, with only the middle 40 subcarriers kept. Secondly, as some of the experiments have very weak signals, measurements with RSSI lower than 30 dB were not used. Thirdly, the CSI data of all antenna pairs is normalized by a common factor such that the maximum amplitude is 1. Lastly, as explained before, each CSI vector is rotated to remove the shift frequency. As the shift frequency value is not reported by our current device to the driver level, a simple estimation method is used, which basically keeps rotating the CSI from the same transmitting antenna incrementally, until most energy appears to occupy only a spectrum starting from 0 up to some frequency for both receiving antennas. As it may over-rotate the CSI and lead to negative frequencies, when running CSI<sub>Apx</sub>, the CSI for all





**Fig. 4** Absolute values of some measured CSI. The absolute value of the CSI across 64 subcarriers is shown for the strongest antenna pair in four experiments

antenna pairs are multiplied by a sinusoid with a positive frequency of 0.0491 to move most sinusoids in the CSI to positive frequency.

### 5.1.2 Compared methods

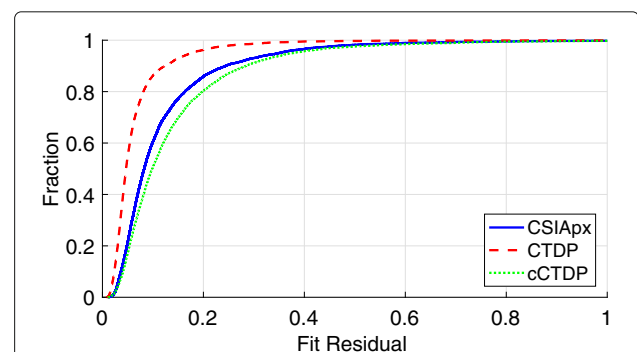
For comparison, we also implement the CTDTP extraction algorithm according to [4], referred to as CTDTP, which iteratively selects a sinusoid that best matches the current residual signal, until the power of the selected sinusoid is below a threshold. CTDTP is chosen because it is one of the more recent methods and has a good performance. As CTDTP requires the noise power value, which needs to be estimated with the experimental data, we use the fit residual found by CSIApx as the total noise power, which should be very close. The frequency range of sinusoids scanned in CTDTP is  $[-0.785, 1.57]$ , which should cover all frequencies in the CSI. Another constrained version of CTDTP, referred to as cCTDP, is also evaluated, with which the fit residuals of CTDTP and CSIApx can be compared when using similar number of sinusoids. That is, with cCTDP, the number of sinusoids used is the smallest upper bound of the average number of sinusoids used by CSIApx for the same CSI measurement, noting that CSIApx may use different configurations for the different antenna pairs. It should also be mentioned that as CTDTP has to solve an optimization problem to select the frequency of a sinusoid in each iteration, it has much higher implementation complexity than CSIApx, because CSIApx avoids this problem altogether by using constant frequencies.

Two other methods were also implemented, but the results are not shown in the figures in this section, since their performances are not comparable with CSIApx and CTDTP. One of the methods is CSIFit [2], the median of fit residual of which, for example, is more than six times

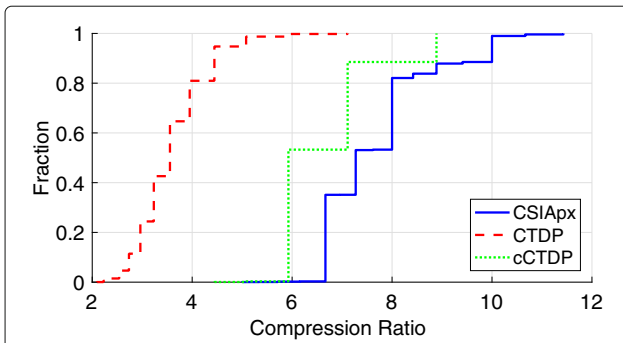
than that with CSIApx for the experimental data, while the compression ratio is about the same. The other maybe obvious approach, i.e., keeping only a small number of significant FFT coefficients, is also not included, because it usually has an order of magnitude higher fit residual than CSIApx even when using twice number of coefficients.

### 5.1.3 Fit accuracy and compression ratio

As the fit residual and the compression ratio are related, i.e., improving one is often at the cost of the other, they are jointly compared. Figure 5 shows the CDF of the total fit residual of all four antenna pairs in 7923 CSI measurements. Figure 6 shows the compression ratio, which is the number of real numbers in the CSI vector divided by that needed by a compression method to describe the sinusoids, noting that a complex number consists of two real numbers.



**Fig. 5** Fit residual with experimental data. A comparison of the cumulative density function (CDF) of the fit residual per point is shown when running CSIApx versus CTDTP. In addition, a comparison is also shown when running a constrained version of CTDTP (cCTDP)



**Fig. 6** Compression ratio with experimental data. A comparison of the cumulative density function (CDF) of the compression ratio achieved when running CSIApx versus CTDP versus CCTDP

It can be seen that the fit residual of CSIApx in most cases are very small with a median of 0.0828 for all antenna pairs, which translates to an error of 0.0005 per data point. The fit residual of CTDP is better with a median of 0.0467, however it is at a cost of a much lower compression ratio, as the average compression ratio of CSIApx against 40 subcarriers is 7.68, much better than CTDP, which is 3.59. By closely examining the fits, we found that CSIApx actually fits the signal very well, and its fit residual is mainly the quantization noise, such as those shown in Fig. 4, which cannot be eliminated unless more sinusoids are introduced to fit the noise. In this sense, CSIApx achieves a better tradeoff between fit residual and compression. The better performance with CSIApx can also be seen from the cCTDP results, as cCTDP actually has higher fit residual, at the same time slightly lower compression ratio.

#### 5.1.4 MU-MIMO rate

The end result of the compression method can be the MU-MIMO data rate of the users. We use the MU-MIMO rate program at [21], which first calculates the modulation parameters with the supplied imperfect CSI, then finds the achievable data rate when the selected parameters are used on the actual channel. We configure the program to use the greedy method for user selection and run at SNR of 20 dB. For each subcarrier, we run the program twice, feeding the compressed and the measured CSI to the program to obtain two values, representing total data rates to all users with imperfect and perfect CSI, respectively. The difference between the two, divided by the latter, is called the *normalized rate difference* and is used as the metric.

A total of 1500 tests are run, where each test has one sender and two receivers. In each test, we use the CSI collected from experiments where the sender was at a fixed location for four receivers, and randomly select two receivers from the four actual receivers. As the link is 2 by 2 but each MU-MIMO receiver has only one antenna, we

select the first antenna for each receiver. Figure 7 shows the CDF of the normalized rate difference, where we can see that the rate difference with CSIApx is usually very small, e.g., within  $-3\%$  and  $3\%$  in over 98.3% of the cases. CTDP performs better reporting 99.0%, but this comes at the cost of its compression ratio. At similar compression ratio, cCTDP performs worse than CSIApx at 95.7%. The rate difference in some very rare cases can also be positive, since the greedy method sometimes selects different sets of users when given the compressed and measured CSI.

#### 5.1.5 Parameter distribution

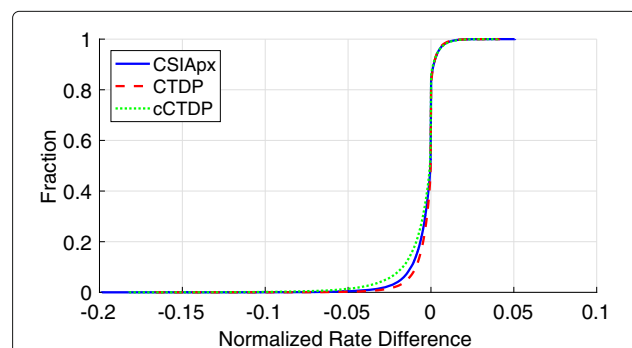
One of the nice features of CSIApx is that the fit coefficients stay in a small range, making it simple and inexpensive to quantize and transmit the coefficients as the compressed CSI. Figure 8 shows the distribution of the real and imaginary parts of the coefficients found by CSIApx for the strongest antenna pair in each test case, because the distributions for other antenna pairs should just be its scaled versions. We can see that all numbers resides in a small range with smooth density.

## 5.2 Evaluation with synthesized CSI data

We test CSIApx with synthesized CSI data, which complements the experimental evaluation by challenging CSIApx with more channel types and testing CSIApx under controllable settings like the SNR level.

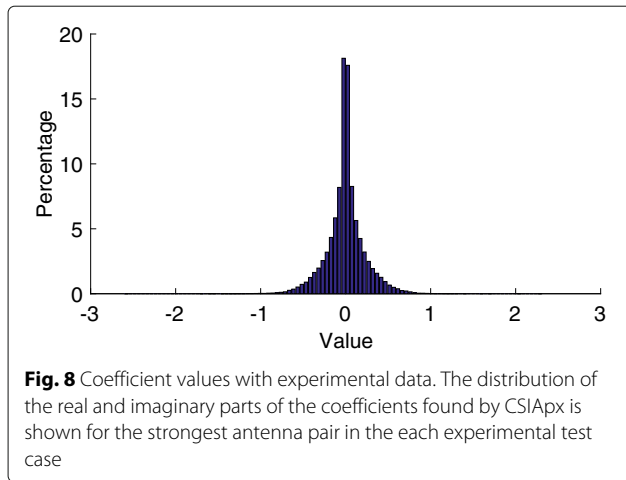
### 5.2.1 CSI generation

We use the channel model code at [19] to generate CSI for all 64 subcarriers in Wi-Fi on 20-MHz channels for 3 by 3 links with nine antenna pairs. Four cases, referred to as model B, model C, model D, and model E, are used, which represent typical indoor environments with around 100-ns, 200-ns, 400-ns, and 800-ns delay spread, respectively, and should cover the majority of typical Wi-Fi channels. The maximum amplitude is again normalized to 1. White Gaussian noise is added to the



**Fig. 7** Normalized rate difference with experimental data. A comparison of the cumulative density function (CDF) of the normalized rate difference is shown when running CSIApx versus CTDP versus CCTDP





CSI vector, and a total of 1000 test cases are performed for each SNR level. To simulate imperfect rotation, the CSI is further multiplied by a sinusoid with random frequency, which translates to within 0 to 50 ns of timing error.

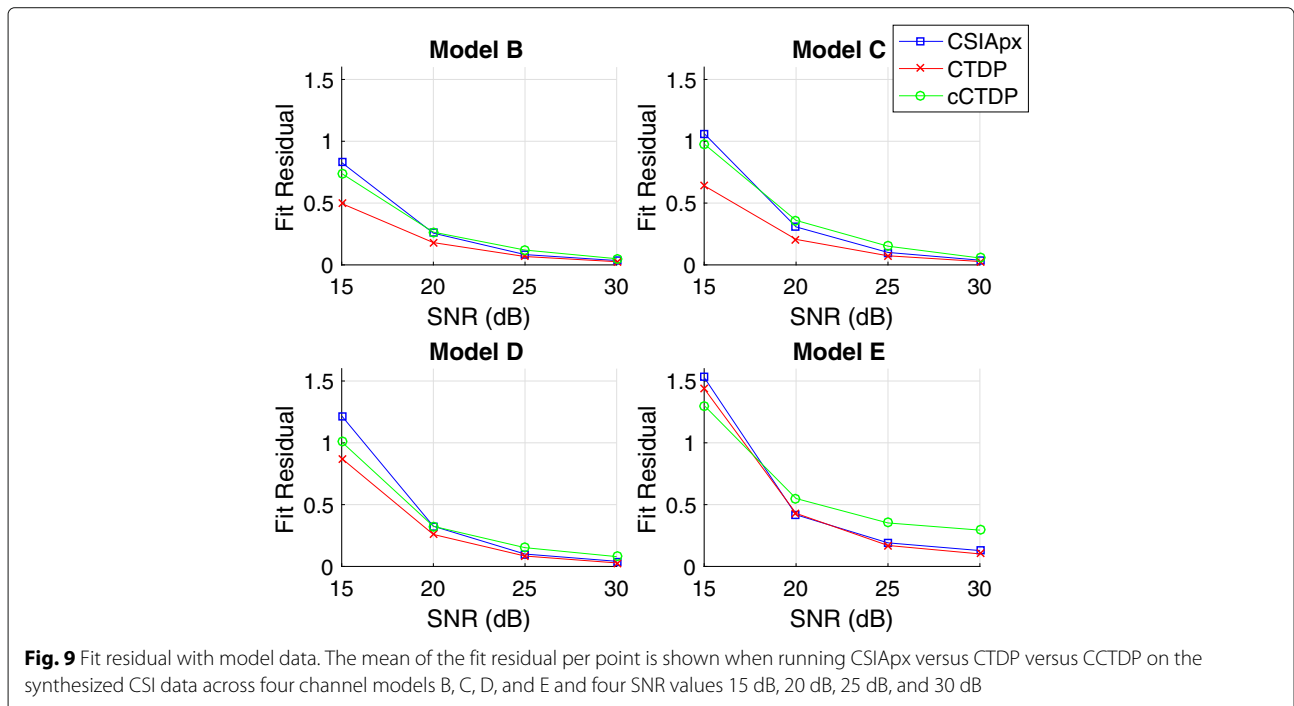
**5.2.2 Fit accuracy and compression ratio**

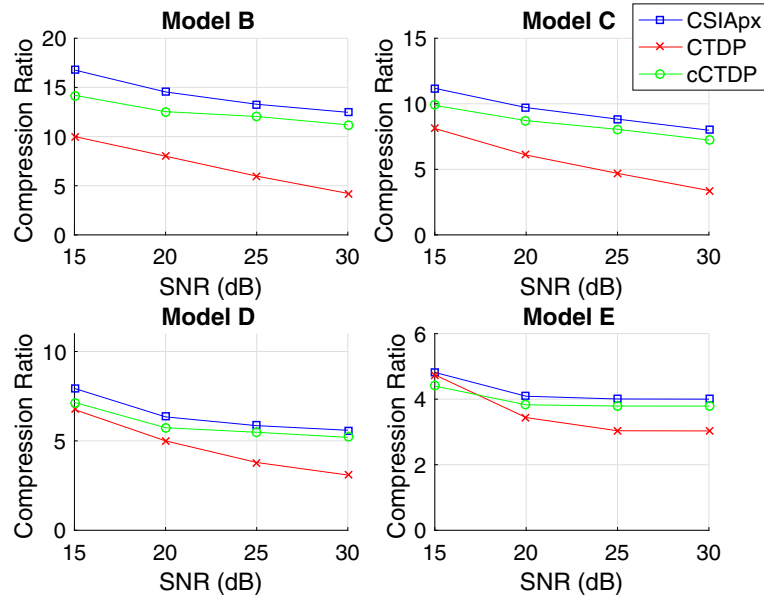
As the clean CSI is available, when calculating the final fit residual, the fit is compared with the clean CSI; all prior steps are still based on the noisy CSI. Figure 9 shows the mean of the total fit residual of all antenna pairs in various settings. The fit residual of CSIApx is usually very small,

such as about 0.0007 or lower per point at 20 dB or above. In addition, as the noise level reduces by 5 dB, the fit residual in most cases also reduces by roughly 5 dB, suggesting that the fit residual is mostly noise. Figure 10 shows the average compression ratios. It can be seen that CSIApx achieves very high compression ratios in many cases, i.e., above 12.4:1, 7.9:1, 5.5:1, and 4.0:1 against 64 subcarriers for models B, C, D, E, respectively, when the SNR is 20 dB or above. More complicated channel conditions do pose a challenge to CSIApx as it has to use higher configurations. Also, although CSIApx may have slightly larger fit residual, it has much higher compression ratios than CTDP in all cases. In addition, the compression ratio CSIApx is more stable than CTDP for each model when the SNR is 20 dB or higher, suggesting the CSIApx is better at capturing the actual signal and less susceptible to the influence of noise. cCTDP has higher fit residual and lower compression ratios in almost all cases when the SNR is 20 dB or higher.

**5.2.3 MU-MIMO rate**

MU-MIMO rate is also tested in a similar manner as with the experimental data. Figure 11 shows the percentage of cases that the normalized rate differences are above 3% or lower than -3%, where we can see that the fraction is very low for CSIApx when the SNR is 25 dB or higher, and still reasonably small at 20 dB except for model E which is the most complicated.





**Fig. 10** Compression ratio with model data. The mean of the fit residual per point is shown when running CSIApx versus CTDP versus CCTDP on the synthesized CSI data across four channel models B, C, D, and E and four SNR values 15 dB, 20 dB, 25 dB, and 30 dB

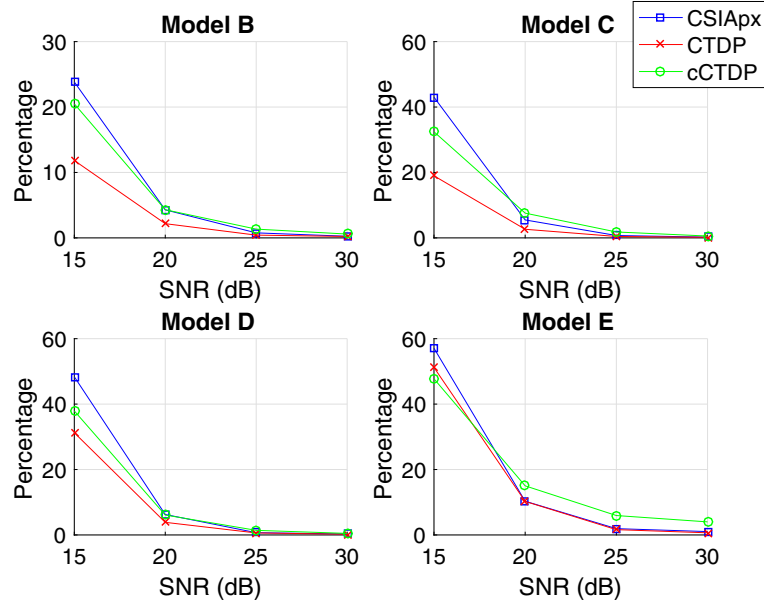
**5.2.4 Parameter distribution**

Figure 12 shows the distribution of the fit coefficients by CSIApx for the strongest antenna pair, which is similar to that with the experimental data.

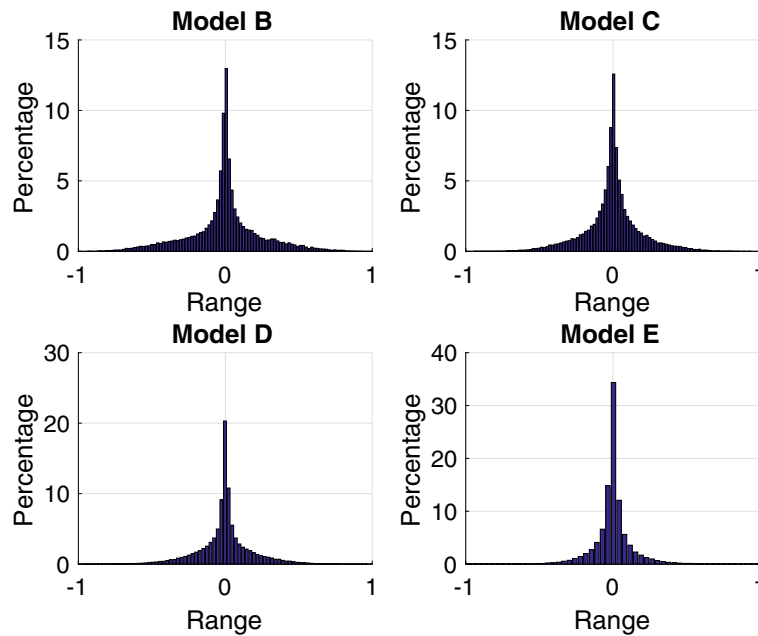
**5.3 More compression with Huffman coding**

Even higher compression ratio can be achieved for CSIApx by running Huffman coding on the coefficients,

taking advantage of the fact that the distribution of the real and imaginary parts of the coefficients, such as that in Fig. 13, is spiky and has low entropy. The process is explained for the experimental data in the following. We empirically choose 12 bits for quantization in range  $[-2.56, 2.56]$ , which results in negligible quantization error and includes all coefficients. A training set with 3962 experiments is randomly chosen from the

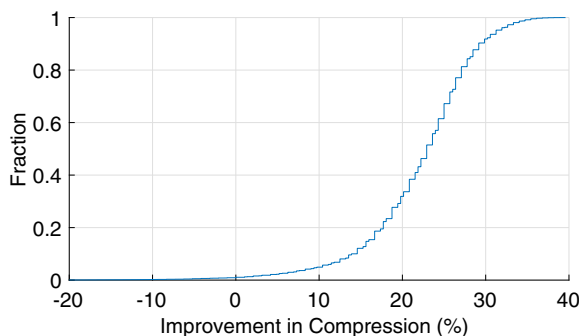


**Fig. 11** MU-MIMO rate difference with model data. The percentage of cases where the normalized rate differences are above 3% or lower than -3% is shown when running CSIApx versus CTDP versus CCTDP on the synthesized CSI data



**Fig. 12** Coefficient values with model data. The distribution of the fit coefficients found by CSIApx for the strongest antenna pair when evaluated on the synthesized CSI data

data to obtain the dictionary of the Huffman coding, which is then tested on the remaining data. Figure 13 shows the CDF of the compression ratio achieved by Huffman coding, where the ratio is calculated by subtracting the size of the raw coefficients by the size of the compressed coefficients then divided by the former. The average compression ratio is 22.1%, and the ratio is positive for over 98% of the cases. Separate Huffman coding dictionaries can also be built for each configuration; however, our results show that the improvement is marginal.



**Fig. 13** Improvement with Huffman coding. The improvement in compression ratio when running Huffman coding after CSIApx is shown for the experimental data

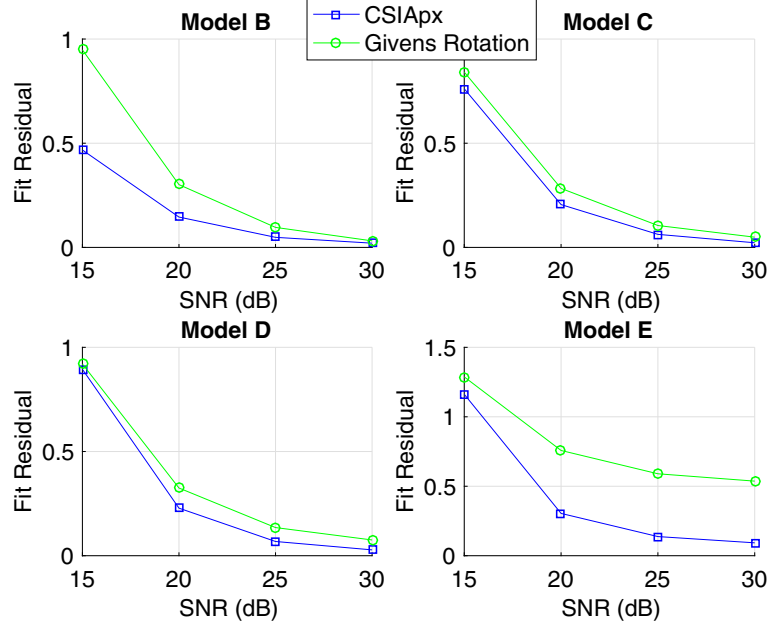
#### 5.4 Comparing with Givens rotation

The Wi-Fi standard includes an option to use the Givens rotation to compress CSI. That is, instead of sending the entire CSI, it calculates a compressed feedback matrix by zeroing out some elements, which is later reconstituted to obtain the full CSI. We provide a head-to-head comparison between CSIApx and the Givens rotation method and argue that CSIApx is a better alternative.

##### 5.4.1 Fit accuracy and compression ratio

Givens rotation is lossless in the sense that the other end of the communication link can exactly reproduce the measured CSI. It therefore appears that Givens rotation will have higher accuracy than CSIApx, as CSIApx is based on approximation. However, this is only true when the measured CSI is clean, i.e., without any noise. With measurement noise and quantization noise, we found that CSIApx actually achieves better accuracy than the Givens rotation, i.e., the CSI with CSIApx follows the shape of the actual CSI more closely than the Givens rotation. From a high level, this is because when fitting a CSI curve, CSIApx serves as a filter and filters out most of the noise, whereas Givens rotation will simply preserve the noise.

Figure 14 shows this comparison between CSIApx and Givens rotation on the model data, where we can see that CSIApx indeed achieves lower fit residual. The model data is used because the clean CSI is available. For a more fair



**Fig. 14** Fit residual comparison with Givens rotation. The comparison between the fit residual achieved by CSIAPx and Givens rotation is shown when running on the synthesized CSI data at four SNR values 15 dB, 20 dB, 25 dB, and 30 dB

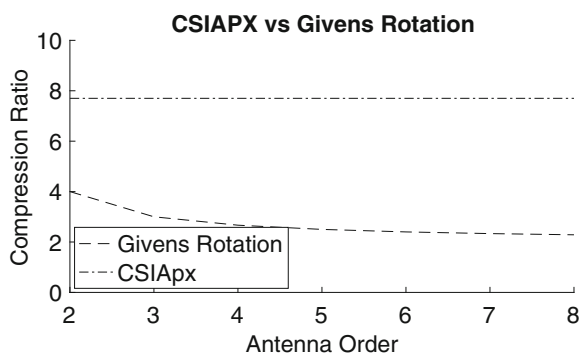
comparison, before running the Givens rotation, a low-pass filter is used in an attempt to filter out some noise, as it is expected that such filter will be used in practice. Due to the low-pass filter, only the results of the middle 50 subcarriers are used in this comparison. The performance of CSIAPx is better with the middle 50 subcarriers than with all subcarriers, because the subcarriers near the ends have less constraints in the fitting and have larger errors.

CSIAPx will also enjoy a higher compression ratio than the Givens rotation. Figure 15 shows the compression ratio for the experimental data. As mentioned earlier the average compression ratio achieved by CSIAPX for the  $2 \times 2$  system in a real-world setting was 7.68. Because

CSIAPx fits each antenna pair individually, hence, it can remain constant even when the number of pairs increases. The compression ratio achieved by Givens rotation on the other hand will keep decreasing as the antenna order increase is approaching 2.

#### 5.4.2 MU-MIMO rate

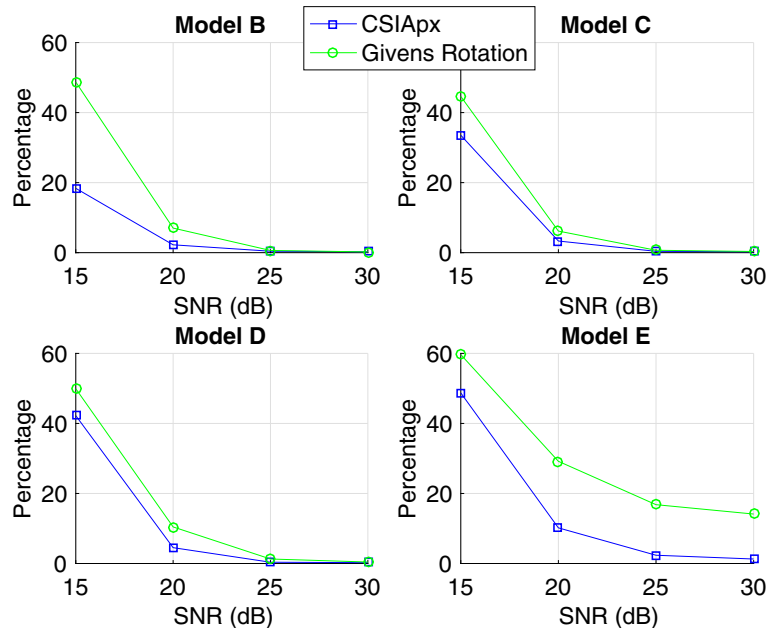
We further evaluate the fit accuracy by comparing the data rate achieved in a MU-MIMO setting from both CSIAPx and Givens rotation. Figure 16 shows the percentage of cases where the normalized rate differences are higher than 3% or lower than  $-3\%$  when compared against the clean signal. We see that CSIAPx outperforms Givens rotation.



**Fig. 15** Compression ratio comparison with Givens rotation. The comparison between compression ratio for CSIAPx versus Givens Rotation is shown for the experimental data

## 6 Conclusion

We propose CSIAPx, a fast and lightweight method for compressing the CSI of OFDM wireless links. We first prove that almost any sinusoid can be approximated as the linear combination of a small number of base sinusoids on constant frequencies. Exploiting the constant frequencies of the base sinusoids, CSIAPx pre-computes key steps and finds a minimum square fit of the CSI vector with very few computations. We evaluate CSIAPx with both experimental and synthesized CSI data, and the results show that CSIAPx achieves very good compression ratios and approximation accuracy. We therefore believe CSIAPx can be a very useful tool to be incorporated into the Wi-Fi protocol and will enable timely and accurate CSI feedback to improve the network performance.



**Fig. 16** MU-MIMO rate difference comparison with Givens rotation. The percentage of cases where the normalized rate differences are above 3% or lower than  $-3\%$  is shown when running CSIApx versus Givens rotation on the synthesized CSI data

### Abbreviations

CSI: Channel state information; OFDM: Orthogonal frequency division multiplexing; MU-MIMO: Multi-user multiple-input-multiple-output; SVD: Singular value decomposition

### Acknowledgments

The authors greatly appreciate the editor and reviewers for their work.

### Funding

This research work was supported by the US National Science Foundation under Grant 1618358.

### Availability of data and materials

The data in this work will be made available to the public for free through web access.

### Authors' contributions

Both authors contributed to the work. Both authors read and approved the final manuscript.

### Competing interests

The authors declare that they have no competing interests.

### Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 19 July 2018 Accepted: 19 March 2019

Published online: 03 April 2019

### References

- I. P. -T. G. AC, Status of Project IEEE 802.11ac (2013). [http://www.ieee802.org/11/Reports/tgac\\_update.htm](http://www.ieee802.org/11/Reports/tgac_update.htm). Accessed 15 Sept 2013
- A. Mukherjee, Z. Zhang, in *2016 13th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. Channel state information compression for MIMO systems based on curve fitting (IEEE, London, 2016)
- X. Wang, Channel feedback in OFDM systems (2014). Google Patents. US Patent No. 8,908,587. 9 Dec
- X. Wang, S. B. Wicker, in *2013 IEEE Global Communications Conference (GLOBECOM)*. Channel estimation and feedback with continuous time domain parameters (IEEE, Atlanta, 2013), pp. 4306–4312
- S. Ferguson, F. Labeau, A. M. Wyglinski, in *2010 IEEE 72nd Vehicular Technology Conference-Fall (VTC)*. Compression of channel state information for wireless OFDM transceivers (IEEE, Ottawa, 2010), pp. 1–5
- V. P. G. Jiménez, T. Eriksson, A. G. Armada, M. J. F. García, T. Ottosson, A. Svensson, Methods for compression of feedback in adaptive multi-carrier 4G schemes. *Wirel. Pers. Commun.* **47**(1), 101–112 (2008)
- U. K. Tadikonda, Adaptive bit allocation with reduced feedback for wireless multicarrier transceivers. PhD thesis (2007)
- V. Pohl, P. H. Nguyen, V. Jungnickel, C. V. Helmolt, in *2003 IEEE Global Telecommunications Conference (GLOBECOM)*. How often channel estimation is needed in MIMO systems, vol. 2 (IEEE, San Francisco, 2003), pp. 814–818
- K. Huang, R. W. Heath Jr., J. G. Andrews, Limited feedback beamforming over temporally-correlated channels. *IEEE Trans. Sig. Process.* **57**(5), 1959–1975 (2009)
- X. Xie, X. Zhang, K. Sundaresan, in *19th Annual International conference on Mobile computing & networking (MOBICOM 2003)*. Adaptive feedback compression for MIMO networks (ACM, San Diego, 2013), pp. 477–488
- R. Crepaldei, J. Lee, R. Etkin, S. Lee, R. Kravets, in *2012 Proceedings IEEE International Conference on Computer Communications INFOCOM*. CSI-SF: Estimating wireless channel state using CSI sampling & fusion (IEEE, Orlando, 2012), pp. 154–162
- C. Carbonelli, S. Vedantam, U. Mitra, Sparse channel estimation with zero tap detection. *IEEE Trans. Wirel. Commun.* **6**(5), 1743–1763 (2007)
- F. Wan, W. Zhu, M. Swamy, Semi-blind most significant tap detection for sparse channel estimation of OFDM systems. *IEEE Trans. Circuits Syst. I. Reg. Papers.* **57**(3), 703–713 (2010)
- M. Duarte, Y. Eldar, Structured compressed sensing: from theory to applications. *IEEE Trans. Signal Process.* **59**(9), 4053–4085 (2011)
- D. Vasishth, S. Kumar, H. Rahul, D. Katabi, in *Proceedings of the 2016 ACM SIGCOMM Conference*. Eliminating channel feedback in next-generation cellular networks (ACM, Florianopolis, 2016)

16. M. Guillaud, D. T. M. Slock, R. Knopp, in *2005 Proceedings of the Eighth International Symposium on Signal Processing and Its Applications, (ISSPA)*. A practical method for wireless channel reciprocity exploitation through relative calibration (IEEE, Sydney, 2005)
17. A. Mukherjee, Z. Zhang, in *2017 IEEE Global Communications Conference (GLOBECOM)*. Fast compression of OFDM channel state information with constant frequency sinusoidal approximation (IEEE, Singapore, 2017)
18. Z. Zhang, A. Mukherjee, System and method for fast compression of OFDM channel state information (CSI) based on constant frequency sinusoidal approximation (2017). Google Patents. US Patent No. 9838104. 5 Dec
19. L. Schumacher, Implementation of the IEEE 802.11 TGn Channel Model (2006). [http://www.info.fundp.ac.be/~lsc/Research/IEEE\\_80211\\_HTSG\\_CMSC](http://www.info.fundp.ac.be/~lsc/Research/IEEE_80211_HTSG_CMSC)
20. Y. Xie, Z. Li, M. Li, in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MOBICOM 2015)*. Precise power delay profiling with commodity WiFi (ACM, New York, 2015), pp. 53–64
21. N. Ravindran, N. Jindal, Multi-user diversity vs. accurate channel state information - MATLAB code (2009). [http://www.ece.umn.edu/~nihhar/mud\\_csi\\_code.html](http://www.ece.umn.edu/~nihhar/mud_csi_code.html). Accessed 23 July 2012

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](http://springeropen.com)

---