Gaussian Approximation Based Interpolation for Channel Matrix Inversion in MIMO-OFDM Systems

Jian A. Zhang, Senior Member, IEEE, Xiaojing Huang, Senior Member, IEEE, Hajime Suzuki, Member, IEEE, and Zhuo Chen, Member, IEEE

Abstract—Channel matrix inversion, which requires significant hardware resource and computational power, is a very challenging problem in MIMO-OFDM systems. Casting the frequency-domain channel matrix into a polynomial matrix, interpolation-based matrix inversion provides a promising solution to this problem. In this paper, we propose novel algorithms for interpolation based matrix inversion, which require little prior information of the channel matrix and enable the use of simple low-complexity interpolators such as spline and low pass filter interpolators. By invoking the central limit theorem, we show that a Gaussian approximation function well characterizes the power of the polynomial coefficients. Some low-complexity and efficient schemes are then proposed to estimate the parameters of the Gaussian function. With these estimated parameters, we introduce phase shifted interpolation and propose two algorithms which can achieve good interpolation accuracy using general low-complexity interpolators. Simulation results show that up to 85% complexity saving can be achieved with small performance degradation.

Index Terms—Interpolation, Matrix Inversion, MIMO-OFDM

I. INTRODUCTION

In a multiple-input multiple-output (MIMO) orthogonal frequency division multiplexing (OFDM) system, channel matrix inversion is a very complex component consuming significant hardware resource and computational power [1]. Matrix inversion can be implemented by, e.g., direct inversion via computing adjugate matrix and determinant or QR decomposition [1], [2]. For an $M \times M$ system with $N$ subcarriers, $N$ $M \times M$ channel matrices need to be inverted, where the complexity of each $M \times M$ matrix inversion is in the order of $M^3$. Although channel matrix inversion is avoided in, for example, 3GPP long term evolution (LTE) systems, by using orthogonal codebook for channel quantization and user selection, it is still essential in many other systems, in particular, where the number of users is small or the channel quantization error is not tolerated.

Interpolation is an efficient way of reducing the complexity and cost for matrix inversion. Exploiting the correlation of frequency-domain channel coefficients, interpolation techniques can be used to interpolate not only frequency-domain channel coefficients, but also the inverted channel matrix. With known values at some base points/subcarriers, the values at other points/subcarriers can be interpolated with a low-complexity interpolator such as polynomial, spline and low-pass filter interpolators [3], [4]. Interpolation for frequency-domain channel coefficients has been well studied [5], [6], however, interpolation for inverted channel matrix remains as an open research problem. The main reason is that interpolation is only applicable to smooth curves which can be well approximated by lower-order polynomials, whereas each element in the inverted channel matrix or QR coefficients is the quotient of two polynomials, and is generally not smooth enough for interpolation to be applied directly.

By casting the frequency-domain channel matrix as a polynomial matrix [7], Borgmann and Bölcskei proposed interpolation-based direct matrix inversion by separately interpolating the adjugate matrix and determinant [1], each of which can be represented as a polynomial matrix. The adjugate matrix and determinant are computed directly for some base points, and they can then be interpolated for other subcarriers. The number of base points, $K$, needs to be larger than the order of the polynomial matrix, $L_p$. The idea is extended to implementing interpolation for QR decomposition [2] and other decompositions [8]. For convenience, we denote these schemes as Bölcskei schemes hereafter. In [9], a modified interpolation-based QR decomposition algorithm is proposed to avoid square root and division operations by computing Hermitian channel matrix, and a CMOS implementation of the scheme is also presented. Although the Bölcskei schemes may achieve great complexity reduction and good interpolation accuracy, they have three major limitations which are yet to be improved: 1) The maximum multipath delay spread $L$ needs to be known in advance; 2) A relatively complex filter, as an approximated sinc function, needs to be used to deal with possibly rapid phase transition in interpolating complex rather than real signals [2]; and 3) the determined number of the base points is more of an upper bound, and no efficient way is provided for determining how many base points are actually needed in each real-time implementation.

In this paper, we present new schemes for channel matrix inversion in MIMO-OFDM systems, based on characterizing the statistical distribution of the coefficients of the polynomial matrix. We show that the power distribution of the polynomial coefficients, as a function of the polynomial order, has the shape of a bell, which can be well approximated by a Gaussian function under general statistical assumptions. This finding motivates new schemes with at least two significant improvements to the Bölcskei schemes: 1) Estimating the number of required base points without requiring any statistical

Paper approved by J. Coon. Manuscript received July 13, 2012; revised October 16, 2012; accepted December 03, 2012.

The authors are with the wireless and networking technologies laboratory, ICT Centre, CSIRO, Sydney, Australian. Email: {Andrew.Zhang; Xiaojing.Huang; Hajime.Suzuki; Zhuo.Chen}@csiro.au.

Part of the work in this paper is presented in IEEE International Conference on Communications, 2012.
information of the channels; and 2) Enabling the use of low-complexity interpolators with greatly improved interpolation accuracy by introducing phase-shifted interpolation. Referring to the form of direct matrix inversion, this paper investigates these improvements, and delivers three major contributions as summarized below.

Firstly, we invoke Lyapunov central limit theory to show that the power of the polynomial coefficients for the adjugate matrix can be well approximated by a Gaussian function. For channels with different distribution and power delay profile (PDF), we also show that the parameters of the Gaussian approximation function may vary significantly. In particular, if only significant polynomial coefficients are of interest, the number of required base points could be very different even if the orders of these polynomials are the same.

Secondly, we develop simple and efficient methods to estimate the parameters of the Gaussian approximation function. In the simplest form, the computation only involves 2K extra multiplications, in addition to the complexity associated with computing the adjugate matrix and determinant at the K base points. With these estimated parameters, we can accurately characterize the polynomial coefficients, and determine the order of the polynomial matrix, as well as the number of required base point for each channel realization.

Thirdly, we introduce phase shifted interpolation to matrix inversion, which enables the use of general and low-complexity interpolators and avoids complex ones such as those proposed in [2]. With good knowledge of the polynomial coefficients obtained via the Gaussian approximation, now we can also better apply complex interpolators such as the one used in [2], in order to reduce the number of used base points and/or improve the interpolation accuracy.

Focusing on direct matrix inversion and low complexity interpolators, the rest of the paper is organized as follows. In Section II, the channel matrix inversion problem is formulated, and the Bölcskei scheme for interpolating the adjugate matrix is briefly described. In Section III, we show that the coefficients of the polynomial matrix can be well approximated by a Gaussian function, both analytically and numerically. In Section IV, schemes for estimating the parameters of the Gaussian approximation function are developed, and algorithms for Gaussian approximation based interpolation are proposed. In Section V, simulation results for the proposed algorithms are presented for 4 × 4 and 8 × 8 MIMO-OFDM systems. Finally, conclusions are provided in Section VI.

**Notation:** Lower-case bold face variables (a, b, ... ) indicate vectors, and upper-case bold face variables (A, B, ...) indicate matrices. Frequency domain variables are denoted with tilde above (â, Ā, ... ). I denotes the identity matrix, E[·] denotes expectation, and (·)T, conj(·), (·)*, (·)−1 and (·)† denote transposition, conjugate, conjugate transposition, inverse and pseudo-inverse, respectively.

**II. PROBLEM FORMULATION**

Consider a MIMO-OFDM system with Mt transmit antennas, Mr receive antennas, and N subcarriers. Let  

\[ h_{r,t}(n), n = 0, \cdots, M_{r,t} - 1 \]

denote the nth multipath between the rth receive antenna and the tth transmit antenna. Note that  

\[ h_{r,t}(L_{r,t} - 1) \neq 0 \]

Although some of the proposed schemes can tolerate small timing offset, and some are robust to any timing offset, we assume that the multipath taps  

\[ h_{r,t}(0) \neq 0 \]

and are aligned for all r and t for the simplicity of notation.

Let  

\[ L = \max_{r,t} L_{r,t} \]

Construct  

\[ M_r \times M_t \text{ time domain channel matrices } H(n), n = 0, \cdots, L - 1, \] with its \{r, t\}-th element being  

\[ h_{r,t}(n) \]

For  

\[ n \geq L_{r,t}, h_{r,t}(n) = 0 \]

The N-point discrete Fourier transform (DFT) of  

\[ H(n), H(k) = \{ \tilde{h}_{r,t} \}, \]

can be represented as

\[ \tilde{H}(k) = \sum_{n=0}^{L-1} H(n)s_k^n, k = 0, 1, \cdots, N - 1, \]

where  

\[ s_k = \exp(-j2\pi k/N) \]

Such a matrix  

\[ \tilde{H}(k) \]

is known as a polynomial matrix of order L [7].

In applications where receive antennas are more than transmit antennas, we obtain a tall matrix where  

\[ M_r > M_t \]

For a tall rectangular matrix  

\[ X, \] its pseudo inverse is  

\[ X^\dagger = (X^*X)^{-1}X^* \]

and interpolation is performed for  

\[ (X^*X)^{-1}. \]

Hence we only consider square matrices with  

\[ M_r = M_t = M \]

It is noted that, however, the order of the polynomial matrix can change from  

\[ L \]

to  

\[ 2L - 1, \]

and the symmetric structure of such matrices may enable further improvement of the proposed schemes.

The direct matrix inversion of  

\[ \tilde{H}(k) \]

is given by its adjugate matrix, normalized by its determinant

\[ \tilde{H}^{-1}(k) = \frac{\text{adj} \tilde{H}(k)}{\det(\tilde{H}(k))}, \]

where the submatrix  

\[ A(k)_{r,t} \]

is obtained by deleting the rth row and tth column of  

\[ H(k) \]

For such a polynomial matrix  

\[ \tilde{H}(k), \] its determinant and each element in the adjugate matrix are polynomial functions with variable  

\[ s_k, \]

and interpolation can be applied to them separately [1]. For ease of understanding, referring to the conventional channel interpolation problem where frequency domain samples are interpolated from a few known base points for a band-limited time-domain signal, for adjugate matrix and determinant interpolation, we can interpret polynomial coefficients as time-domain samples, and base points and interpolation output as frequency-domain samples.

The \( (t, r) \)-th element in \( \text{adj} \tilde{H}(k) \), denoted as  

\[ \alpha_{t,r}(k), \]

is given by

\[ \alpha_{t,r}(k) = (-1)^{r+t} \det(\tilde{A}(k)_{r,t}), \]

and its highest possible order is  

\[ L_p = \max(M - 1)(L - 1) \]

Therefore interpolation for the adjugate matrix needs at least  

\[ L_p + 1 \]

base points. Similarly, it can be determined that the highest order of the determinant  

\[ \det(\tilde{H}(k)) \]

is  

\[ M(L - 1) \]

However, as shown in [1], the determinant can also be computed via (3) once all the adjugate matrices are obtained by interpolation. To make the presentation focused, we only present interpolation for the adjugate matrix in this paper. We also concentrate on using comb base points for interpolation. Let the number of
actually used base points be $K, K \geq L_p+1, T_d = N/K$ be the downsampling rate, and $Q_K$ denote the set of $K$ base points. The steps of applying Algorithm II-A in [1] to interpolate the adjugate matrix of $\mathbf{H}(k)$ is briefly summarized below:

1) Compute $\mathbf{H}(k), k \in Q_K$;
2) Directly compute $\text{adj} \mathbf{H}(k)$ for base points $k \in Q_K$ using (4);
3) Interpolate $\text{adj} \mathbf{H}(k), k \in Q_K$ to obtain the adjugate matrix for all the desired subcarriers.

The ratio of the complexity between the Bölskei schemes and computing matrix inversion for every subcarrier approximately equals to that between the number of used base points and $N$, and detailed analysis considering interpolation complexity and hardware implementation structure is available from [1], [2].

III. COEFFICIENT DISTRIBUTION OF POLYNOMIAL MATRIX

Since the main complexity is associated with the computation of the adjugate matrix and the proposed algorithm is not specific to any element in the adjugate matrix, we use $\text{det}(\mathbf{A}(k)_{M,M})$ as an example in this section, and the notation differentiating different elements in the adjugate matrix will be omitted.

Instead of using the Laplace formula as in (3), we can use the Leibniz formula to represent $\text{det}(\mathbf{A}(k))$ in (4) as

$$\text{det}(\mathbf{A}(k)) = \sum_{\zeta \in S_{M-1}} \text{sign}(\zeta) \prod_{i=1}^{M-1} \tilde{h}_{i,\zeta}(k), \quad (5)$$

where the sum is computed over all $(M-1)!$ permutations $\zeta$, the permutation $\zeta$ is a function that reorders a set of integers $\{1, 2, \cdots, M-1\}$, $S_{M-1}$ denotes the set of all such permutations, and $\zeta_i$ denotes the position of the $i$-th element after the reordering. With the Leibniz formula, we can see that $\text{det}(\mathbf{A}(k))$ equals to the $\{1, -1\}$-weighted sum of $(M-1)!$ terms, each being the product of $(M-1)$ channel coefficients from $\mathbf{H}(k)$ with different row and column indices.

For the simplicity of notation, we consider the following product term in (5):

$$\lambda_1(k) \triangleq \prod_{i=1}^{M-1} \tilde{h}_{i,1}(k) \overset{(a)}{=} \prod_{i=1}^{M-1} \left( \sum_{n=0}^{L-1} \tilde{h}_{i,1}(n)s_k^k \right) \triangleq \lambda_1(s_k), \quad (6)$$

where (a) follows from (1).

From the viewpoint of a polynomial of $s_k$, (6) shows that $\lambda_1(s_k)$ is a polynomial function of $s_k$ of order $L_p - 1$ where $L_p = \sum_{i=1}^{M-1} L_i$. As the product of $M-1$ polynomials, the polynomial coefficients of $\lambda_1(s_k)$ equal to the repeated convolution of $M-1$ sequences $h_{i,1} \triangleq \{h_{i,1}(0), h_{i,1}(1), \cdots, h_{i,1}(L_i-1)\}, i = 1, 2, \cdots, M-1$. Denote the set of polynomial coefficients of $\lambda_1(s_k)$ as $b_1 = \{b_1(0), b_1(1), \cdots, b_1(L_p-1)\}$, which is given by

$$b_1 = \bigotimes_{i=1}^{M-1} h_{i,1}, \quad (7)$$

where $\bigotimes$ represents repeated convolution.

Intuitively, each element except for $b_1(0)$ and $b_1(L_p-1)$ in $b_1$ equals to the sum of two or more products of channel taps, and the terms in the sum monotonically increases for elements from $b_1(1)$ to $b_1(floor(L_p/2))$ and from $b_1(L_p-2)$ to $b_1(ceil(L_p/2))$. Let $|b_1|^2$ be a vector where each element represents the power of the corresponding element in $b_1$. When each $h_{i,i}(n)$ is independently generated, we can expect that the averaged waveform of $|b_1|^2$ resembles a bell-shaped curve according to the property of repeated convolution. Next, we show that when channels $h_{i,i}(n)$ are independent random variables, the averaged waveform of $|b_1|^2$ converges to a Gaussian function with $M$ approaching infinity.

A. Gaussian Approximation

With its Leibniz expression in (5), the mean power of $\text{det}(\mathbf{A}(k))$ can be represented as

$$E[|\text{det}(\mathbf{A}(k))|^2] = E\left[ \sum_{\zeta \in S_{M-1}} \prod_{i=1}^{M-1} \tilde{h}_{i,\zeta}(k) \right]^2, \quad (8)$$

where $E[\cdot]$ denotes the expectation operation, with respect to the channel $h_{i,i}(n)$ here. Although $\prod_{i=1}^{M-1} \tilde{h}_{i,\zeta}(k)$ with different $\zeta$ may share a few common factors, at least one out of $M-1$ factors is different between any two permutations. Since $\tilde{h}_{i,\zeta}(k)$ are mutually independent random variables with zero mean, it can be easily verified that (8) can be simplified as

$$E[|\text{det}(\mathbf{A}(k))|^2] = \sum_{\zeta \in S_{M-1}} E\left[ \prod_{i=1}^{M-1} \tilde{h}_{i,\zeta}(k) \right]^2. \quad (9)$$

Without loss of generality, we consider $E[|\lambda_1(k)|^2]$ and its polynomial representation $E[|b_1|^2]$. As derived in Appendix A, $E[|b_1|^2]$ is given by

$$E[|b_1|^2] = \sum_{i=1}^{M-1} \sigma_i^2(L_i - 1). \quad (10)$$

For a uniform PDF with $\sigma_i(\ell)$ being a constant for any $\ell$, (10) resembles the B-spline basis functions in [10], where it is shown that the waveform of the B-spline basis function converges to a Gaussian function. It is also shown that for a low-order cubic B-spline basis functions, equivalent to $E[|b_1|^2]$ with $M = 4$ in (10), the Gaussian approximation is already quite good. For more general scenarios, the following theorem, which is proven in Appendix B, states the convergence of $E[|b_1|^2]$ to a Gaussian function:

Theorem 1: Normalized to its maximum value, the power waveform of the polynomial coefficients averaged over different channel realizations, $E[|b_1|^2] = \{E[|b_1(\ell)|^2]\}, \ell \in [0, L_p - 1]$, converges to a Gaussian function

$$g_1(\ell) = \exp\left( -\frac{(\ell - u_1)^2}{2\sigma_1^2} \right), \quad (11)$$

when $M$ approaches infinity.
for $\det(A(k))$. As the sum of $(M - 1)!$ permutations, the mean power of the polynomial coefficients corresponding to $\det(A(k))$ converges to the sum of multiple Gaussian functions. From the derivation process, we can see that the above results can be generalized to any $\det(A(k), r, t)$ with respective $u_n$ and $\varepsilon_n$ values.

Although the closed-form expression is derived under the conditions of $M \to \infty$, it is a good approximation to the actual $E[b(\ell)]^2$ for quite a few typical channels at moderate and even small $M$. Approximation with one single Gaussian function for each element in the adjugate matrix is also found sufficient. Next we illustrate the approximation effect for some channels using numerical examples. The parameters $u_n$ and $\varepsilon_n$ of the Gaussian approximation may be determined via channel statistics, however, we will show in Section IV that they can effectively be estimated in real time using comb base points.

B. Numerical Examples

Consider MIMO-OFDM systems with 1024 subcarriers, where the multipath taps $h_{r,t}(n)$ are generated using independently and identically distributed (i.i.d.) complex Gaussian distribution with zero mean and variance 1, and shaped by a power delay profile (PDP) function. Channels between different antennas are allowed to be different to reflect real situations: The number of non-zero taps is randomly generated according to a uniform distribution over $[2, 6]$, and they are assigned randomly over $[1, 40]$ via a uniform random distribution. Two PDP functions, including an exponential PDP $\exp(-n/s)$ with $s$ being a random number uniformly distributed over $[10, 19]$ and a uniform PDP, are tested. Parameters for Gaussian approximation functions are computed using algorithms to be proposed in Section IV. The modulation scheme is 16QAM.

Fig. 1 shows one exemplified plot for the power of the polynomial coefficients and its Gaussian approximation, where $M = 8$ and the PDP function is exponential. The solid lines marked with stars in the figure show the normalized power of the polynomial coefficients of a randomly chosen element in the adjugate matrix. The dashed curve represents a Gaussian function $g(\ell) = \exp(-(\ell - u)^2/(2\varepsilon^2))$ with $u = 90$ and $\varepsilon = 26.7$, $\ell = 0, \cdots, L_p - 1$, which approximates the envelope of the solid lines. If we rely on the statistics to determine the order of polynomials, $L_p$ will be about $40 \times 7 = 280$. However, from the figure we can see that there are only about 150 non-zero polynomial coefficients.

To further quantify the accuracy of the Gaussian approximation clearly, the cumulative energy ratios (CERS) for the polynomial coefficients and corresponding Gaussian approximation functions are plotted in Fig. 2 and Fig. 3, for the exponential and uniform PDPs, respectively. For better illustration, only the segments of curves for high energy ratio $[0.5, 1]$ are shown. Each curve in the two figures are generated from results averaged over 10 realizations. The CER, denoted as $\chi(\ell, c)$, is defined as the ratio between the energy of $2\ell, c + 1$ coefficients with indices centered about the estimated $u$ and the total energy. That is,

$$\chi(\ell,c) = \frac{\sum_{\ell=0}^{\ell,c} |b(u - \ell + 1)|^2 + |b(u + \ell)|^2}{\sum_{\ell=0}^{N-1} |b(\ell)|^2}$$

for the polynomial coefficients, and

$$\chi(\ell) = \frac{\sum_{\ell=0}^{\ell,c} (g(u - \ell + 1) + g(u + \ell))}{\sum_{\ell=0}^{N-1} g(\ell)}$$

for the Gaussian function $g(\ell)$. Using CER instead of conventional cumulative distribution function can clearly demonstrate how the energy is concentrated within those significant coefficients approximately symmetric about $u$.

From the two figures, we can see that the CER curves of Gaussian approximation functions and the power of polynomial coefficients match very well, for various $M$ values and varying channels. Therefore, by estimating the parameters $u$ and $\varepsilon$, we can fully characterize the distribution of the power of polynomial coefficients. Comparing Fig. 2 with Fig. 3, we
can also see that the exponential PDP greatly reduces the number of significant polynomial coefficients. It implies that when insignificant polynomial coefficients can be ignored, we may use different numbers of base points for different channels even if their maximum delay spreads are the same.

IV. INTERPOLATION BASED ON GAUSSIAN APPROXIMATION

Based on the Gaussian approximation, at least three options are available for improving the Bölcskei schemes: 1) The polynomial order can be determined once the parameter \(u\) is known by assuming that the polynomial distribution is symmetric about \(u\); 2) Phase shifted interpolation can be applied to improve the interpolation accuracy; and 3) The number of base points for interpolation may be reduced by ignoring small polynomial coefficients. In this section, we develop novel schemes by investigating these options.

Consider any \((t, r)\)-th element in the adjugate matrix \(\text{adj}(H(k))\): \(\alpha = \{\alpha(k)\}, k = 0, \cdots, N-1\). The \(K\) comb base points are \(\beta = \{\beta(pN/K)\}, p = 0, \cdots, K-1\), where \(T_d = N/K\) is the downsampling rate. Let \(\alpha\)'s corresponding \(L_p\) polynomial coefficients be \(b = (b(0), \cdots, b(L_p - 1))\), and its zero-padded \(K\)-point vector be \(\beta = (\beta(0), \cdots, \beta(K-1))\), with \(\beta(\ell) = b(\ell)\) for \(\ell \in [0, L_p - 1]\) and \(\beta(\ell) = 0\) for \(\ell \in [L_p, K-1]\). Then \(\beta\) is the \(K\)-point DFT of \(\beta\) and \(\beta = \beta F_K\) where \(F_K\) denotes the \(K\)-point DFT matrix. The sequence \(|b(\ell)|^2\) can be well approximated by a Gaussian function \(g(\ell) = \exp(-|\ell - u|^2/(2\sigma^2))\).

A. Interpolation with Phase Shifting

It has been shown in [4], [11] that introducing a phase shift term to the interpolation for complex signals can significantly improve the interpolation accuracy. Since low-complexity interpolators operate on the real and imaginary components separately for a complex signal, signals with small phase transitions between adjacent samples lead to better interpolation accuracy. Recast \(b\) as \(L_p\) time-domain samples and \(\alpha\) as \(b\)'s \(N\)-point DFT coefficients. The multiplicative phase shifting terms in the frequency domain, \(\exp(j2\pi k\tau/N)\) with \(\tau\) being an integer, correspond to a circular time shift \(\tau\) of the sequence consisted of \(b\) and \(N-L_p\) zeros. Such a phase shift is used to reduce the variation of the real and imaginary components. Interpolation is then applied to the phase-shifted base points \(\alpha(pN/K)\exp(j2\pi pt/K),\) and \(\exp(-j2\pi k\tau/N)\) is multiplied to the output of the interpolator at the \(k\)-th subcarrier to get the final signal.

The optimal phase shifts proposed in [4], [11] are derived for specific interpolators and not easy to compute in real-time. In [12], we propose to find the optimal phase shift by minimizing the phase transition between consecutive samples \(\alpha(k)\) for \(k = 0, \cdots, N-1\), which is equivalent to jointly minimizing the variation between adjacent real and imaginary components of \(\alpha(k)\). The optimal time shift \(\tau_{op}\) is formed as [12]

\[
\tau_{op} = -\frac{N}{2\pi} \arg \left( \sum_{\ell=0}^{L_p-1} |b(\ell)|^2 e^{-j2\pi \ell/N} \right),
\]

which links \(\tau_{op}\) to the angle of the first DFT coefficient of the polynomial coefficients \(|b(\ell)|^2\).

For any signal \(x(n)\) symmetric about \(u\), that is, \(x(n) = x(2u-n)\) for \(0 \leq n \leq u-1\), we have

\[
\sum_{n=0}^{u} |x(n)|^2 e^{-j2\pi n/N} + \sum_{n=u+1}^{N} |x(n)|^2 e^{-j2\pi n/N} = \sum_{n=0}^{u-1} |x(n)|^2 (e^{-j2\pi n/N} + e^{-j2\pi(2u-n)/N}) = e^{-j2\pi u/N} \sum_{n=0}^{u-1} |x(n)|^2 (e^{j2\pi(u-n)/N} + e^{-j2\pi(u-n)/N}) \equiv \lambda
\]

where \(\lambda\) is real. As shown in Section III, we can approximate \(|b|^2\) as a Gaussian function with parameter \(u\) and \(\varepsilon^2\), and \(|b(\ell)|^2\) is symmetric about \(u\). Therefore, (14) becomes

\[
\tau_{op} = -\frac{N}{2\pi} \arg \left( (|b(u)|^2 + \lambda) e^{-j2\pi u/N} \right) = u,
\]

which shows that the optimal phase shift corresponds to a timing offset value equal to the center of symmetric signals.

B. Estimation of \(u\)

The task now is to get a good estimate for \(u\). Among a few approaches that are feasible for computing \(u\) in real-time, the following two methods based on using comb base points and consecutive base points are preferred. Both methods are robust to any timing offset between different antennas.

1) Estimation via Gaussian Approximation: Let \(\beta_{(q)} = \{\beta(\text{mod}(p+q, K))\}\) be a vector obtained by circularly shifting \(\beta\) by \(q\) samples, where \(\text{mod}(a, b)\) denotes \(a\) modulo \(b\). According to the circular shift property of DFT, \(\beta_{(q)} = \beta D F_K\) where \(D\) is a diagonal matrix with the \(p\)-th diagonal element being \(\exp(-j2\pi pq/K)\).
**Proposition 1:** Given $K$ known comb base points $\tilde{\beta}$, an estimate of $u$ is given by

$$
\hat{u} = -\frac{K}{2\pi} \cdot \angle \left( \text{conj}(\tilde{\beta}) \tilde{\beta}^T_{(q)} \right).
$$

(16)

**Proof:** The outer product of $\text{conj}(\tilde{\beta})$ and $\tilde{\beta}^T_{(q)}$ can be represented as

$$
\text{conj}(\tilde{\beta}) \tilde{\beta}^T_{(q)} = \text{conj}(\beta) F_K F_K D \beta^T
$$

(17)

where $L_p$ is obtained only when the number of base points $K$ is larger than the order of the polynomial $L_p$. This property will be exploited in Algorithm 1 to avoid the requirement of knowing $L_p$.

1) $L_p$ Approximated as $2u$: Since $|b(\ell)|^2, \ell = 0, 1, \ldots, L_p - 1$ is approximately symmetric about $u$, we can approximate $L_p$ as $2u$.

However, estimating $u$ through (16) requires to know $L_p$ due to the requirement of $K \geq L_p$. Violation of this condition will cause the estimate of $u$ to largely deviate from the center of the Gaussian function. Since $K \approx 2L_p$ is required for interpolation, we can develop the following simple rule to test the sufficiency of the base points:

**Rule 1:** When the difference of the two estimated values of $u$ between using $K$ and $2K$ base points is smaller than a pre-chosen small threshold, such as 10, it is decided that $K \geq L_p$ and $2K$ base points are regarded as sufficient for interpolation.

The initial $K$ can be set according to the desired hardware architecture and the channel statistics if it is available. A good choice is setting the initial downsampling rate $T_d$ as 8 or 16.

Using $L_p = 2u$ becomes less effective when, for example, there are some larger timing offset between different antennas. In this case, quite a few coefficients $b(\ell), \ell = 0, 1, \ldots, L_0$ become zeros. Thus $2u$ could become much larger than the number of actual non-zero coefficients.

2) Through $\varepsilon$ in Gaussian Approximation: An alternative way of determining the number of base points is through the parameter $\varepsilon$ in the Gaussian approximation function $g(\ell)$.

It is well known that for a Gaussian distribution with zero mean and variance $\varepsilon^2$, we can use a multiple of the standard derivation $\varepsilon$ to determine the confidence interval. Similarly, we can use $\chi(\varepsilon)$, which is defined in (13), to characterize the CER, i.e., the ratio of the energy between $2\varepsilon + 1$ samples centered around $u$ and all the samples, of a Gaussian waveform. For example, $\chi(2.57\varepsilon)$ and $\chi(3.1\varepsilon)$ correspond to CER values of 99% and 99.8%, respectively. The CER serves as a good indicator for the interpolation accuracy of the adjugate matrices. Thus with a pre-chosen $c$, $L_p$ can be determined as

$$
L_p = \min(2\varepsilon + 1, N).
$$

(20)

Since the inverted matrix is a ratio between the adjugate matrix and the determinant, $c$ generally needs to be larger than 3.5 according to our simulation results.

**Proposition 2:** When $|b|^2$ can be approximated as a Gaussian function, an estimate of $\varepsilon$ is given by

$$
\hat{\varepsilon} = \frac{K}{1.414\pi} \sqrt{-\ln \left( \frac{\text{abs}(\text{conj}(\beta) \tilde{\beta}^T_{(1)})}{\text{conj}(\beta) \tilde{\beta}^T_{(0)}} \right)}.
$$

(21)

**Proof:** Note that by taking absolute value and normalizing $\text{abs}(\text{conj}(\beta) \tilde{\beta}^T_{(1)})$ to $\text{conj}(\beta) \tilde{\beta}^T_{(0)}$ in (21), the effects of any scalar terms and timing offsets in a Gaussian function are removed from the estimate. Hence we can consider a simplified Gaussian function $g(\ell) = \exp(-\ell^2/(2\varepsilon^2))$.

As shown in Appendix C, the DFT of the discrete Gaussian function $g(\ell)$ is still a discrete Gaussian function, given by

$$
\hat{g}(q) = \exp \left( -\frac{2\pi^2 \varepsilon^2 q^2}{K^2} \right).
$$

(22)
whereas in Algorithm 2, 
\[ \frac{2}{\delta} \alpha \varepsilon \]
which can be easily satisfied with even a small number of 
which yields (21).

\[ \text{Algorithm 1: Phase-shifted Interpolation for adjugate ma-} \]
\[ \text{trix via } L_p \approx 2n. \]
\[ \text{Input: } H(k), k = 0, \cdots , N - 1, T_{t,r} = N/T_d; \text{ threshold } \delta; \]
\[ \text{for } t \leftarrow 1 \text{ to } M \text{ do} \]
\[ \text{for } r \leftarrow 1 \text{ to } M \text{ do} \]
\[ \text{while } K_{t,r} \leq N \text{ do} \]
\[ \text{Compute } \alpha_{t,r}(k) \text{ for } k \in Q_{2K_{t,r}}; \]
\[ \text{Estimate } u_1 \text{ and } u_2 \text{ using (16) with} \]
\[ K = K_{t,r} \text{ and } K = 2K_{t,r}, \text{ respectively;} \]
\[ \text{if } |u_1 - u_2| > \delta \text{ then} \]
\[ K_{t,r} \leftarrow 2K_{t,r}; \]
\[ \text{else} \]
\[ \text{quit while loop and go to next section} \]
\[ \text{Implement phase-shifted interpolation; } \]
\[ \text{end} \]
\[ \text{end} \]
\[ \text{end} \]
\[ \text{end} \]
\[ \text{end} \]

According to (17), \( \hat{g}(q) \approx \text{conj}(\hat{\beta})\beta_q^T \) as \( g(\ell) \) approxi-
mates \( |b(\ell)|^2 \). Therefore we can formulate the following metric
\[ \ln \left( \frac{\text{abs} (\text{conj}(\hat{\beta}))\beta_q^T}{\text{abs}(\hat{\beta})} \right) = \ln \left( \frac{\text{abs}(\hat{g}(1))}{\hat{g}(0)} \right) = -\frac{2\pi^2 \varepsilon^2}{K^2}, \]
which yields (21). The value of \( \varepsilon \) should be large enough to avoid aliasing in
digitizing a continuous signal. As shown in (25) in Appendix C, the condition for avoiding aliasing is
\[ \varepsilon \geq 0.4838, \quad (23) \]
which can be easily satisfied with even a small number of 
comb base points. Therefore only a few comb base points 
are needed to get a reliable estimate of \( \varepsilon \), and the number of 
required base points for interpolation can then be determined.

D. Exemplified Algorithm

Based on the proposed methods above for determining the phase shift terms and the number of required comb base 
points for interpolation, we can develop various schemes for 
interpolation based matrix inversion. Here, two exemplified 
algorithms for interpolating the adjugate matrix are summa-
ried in Algorithm 1 and Algorithm 2, which use each of the 
two approaches proposed in Section IV-C for determining 
the length of base points, respectively. The step “Implement 
phase shifted interpolation” is referred to the following 
operations: Interpolate \( \eta_{t,r}(k) \triangleq \alpha_{t,r}(k) \exp(i2\pi ku/N) \) using 
any low-complexity interpolator, \( k \in Q_{2K_{t,r}} \), to obtain \( \eta_{t,r}(k), k = 0, \cdots , N - 1 \), and obtain \( \alpha_{t,r}(k), k = 0, \cdots , N - 1 \) via 
\( \alpha_{t,r}(k) = \eta_{t,r}(k) \exp(-j2\pi ku/N) \). Some suggested values
for constants used in the two algorithms are \( T_d = 16 \) or 8, 
\( \delta = 10 \) and \( c = 8 \).

It is noted that in Algorithm 1, \( K \) is preferable to be a power 
of 2 due to the requirement of extending \( K \) to \( 2K \) base points; 
whereas in Algorithm 2, \( K \) can be any integer in principle.

\[ \text{Algorithm 2: Phase-shifted Interpolation for adjugate ma-} \]
\[ \text{trix via estimating } \varepsilon. \]
\[ \text{Input: } H(k), k = 0, \cdots , N - 1, T_{t,r} = N/T_d; \]
\[ \text{for } t \leftarrow 1 \text{ to } M \text{ do} \]
\[ \text{for } r \leftarrow 1 \text{ to } M \text{ do} \]
\[ \text{Compute } \alpha_{t,r}(k) \text{ for } k \in Q_{K_{t,r}}; \]
\[ \text{Estimate } \varepsilon \text{ using (21);} \]
\[ \text{Set } K_{t,r} = \min(4\pi \varepsilon , N); \]
\[ \text{Compute } \alpha_{t,r}(k) \text{ for } k \in Q_{K_{t,r}}; \]
\[ \text{Estimate } u \text{ using (16);} \]
\[ \text{Implement phase-shifted interpolation; } \]
\[ \text{end} \]
\[ \text{end} \]

The extra complexity of the two proposed schemes, on

V. SIMULATION RESULTS

Here, we present some simulation results for \( 4 \times 4 \) and \( 8 \times 8 \)
MIMO-OFDM systems with 1024 subcarriers. Zero forcing
(ZF) equalization is used. The parameters are set as \( \delta = 10 \)
in Algorithm 1 and \( c = 4 \) in Algorithm 2.

Two channel models are used in the simulation. In Model
1, the channels are generated using the same configuration to
that in Section III-B, except for only exponential PDP being
used here. In Model 2, each channel between any transmitter
and receiver is independently generated following the WiMAX
SUI4 channel model [13], which is a Rayleigh channel with
two non-zero taps at \( \{0, 14, 36\} \) and PDP \( \{0, -4, -8\} \) dB.
The proposed schemes determine how many base points the
adjugate matrices need to be directly computed for before
interpolation, and they have no direct requirement on the
number and pattern of the pilots used for channel estimation.
The channel itself may be estimated in many ways, using either
comb pilots or full training preamble. Certainly, the accuracy
of the channel estimate largely affects the interpolation per-
tformance. To focus on matrix inversion, the frequency-domain
channel matrix \( \tilde{H}(k) \) is computed from time-domain matrix
for each base point without interpolation.

Different schemes for channel matrix inversion are simu-
lated and compared. The interpolation schemes use the same
number of base points, which is chosen as a power of 2, and
is determined by our proposed algorithms. These schemes
are explained in conjunction with the legends used in the figures
next:

- ItpComb - Proposed Algorithm 1 and 2 with phase
  shifting term estimated by (16);
- ItpLoc - Proposed Algorithm 1 and 2 with phase shifting
term estimated by (19);
Fig. 4. Normalized MSE of interpolated adjugate matrix for the $4 \times 4$ MIMO-OFDM system and channel model 1. Solid and dashed curves for Algorithm 1 and 2 respectively.

- w/o shift - Interpolation-based channel inversion without phase shifting;
- Direct Itp - Interpolation on inverted channel matrix by normalizing the adjugate matrix to the determinant;
- w/o Itp - Directly computed channel matrix inversion from $\mathbf{H}(k)$ for every subcarrier.

The simulation results for “w/o shift” can be regarded as the performance that can be achieved by the Bölcskei’s scheme in [1], at the similar complexity to our proposed schemes. This assumes that the Bölcskei’s scheme knows the required number of base points, which is the same as that determined by our schemes. In reality, the Bölcskei’s scheme can un-necessarily require much larger number of base points. Determined via the statistical maximum delay spread, these numbers could be 240 and 560 for the simulated $4 \times 4$ and $8 \times 8$ systems, corresponding to downsampling rates of 4.27 and 1.83, respectively.

We first present simulation results for channel model 1 and $4 \times 4$ MIMO-OFDM systems, where the spline interpolator (Matlab function `spline`) is used. The initial number of base points is set to 64, corresponding to a downsampling rate $T_d = 16$.

Fig. 4 shows the normalized mean square error (NMSE) of elements in the interpolated adjugate matrix, normalized to the mean power of these elements. The adaptively determined mean downsampling rates are 5.56 and 6.97 for Algorithm 1 and 2, respectively. With such significant complexity save, the proposed algorithms still achieve NMSE very close to the case without using interpolation. On the other hand, the performance of the scheme without using phase shifting is very poor.

It is known that direct matrix inversion has the numerically instable problem and is sometimes sensitive to perturbation. Therefore, instead of showing the averaged MSE for the inverted matrix, which may be biased by a few significant errors, we plot in Fig. 5 the cumulative distribution function (CDF) of the MSE of the inverted channel matrix, normalized to the power of each inverted matrix. The determinant of the matrix is computed via interpolated adjugate matrices according to (3).

Only the results for schemes with the number of base points determined by Algorithm 2 are shown. From the figure we can see that similar to the results demonstrated by the MSE of the adjugate matrix, the two proposed algorithms perform closely to the scheme “w/o Itp”, which computes matrix inversion for each subcarrier, and outperform the schemes “Direct Itp” and “w/o shift”.

Fig. 6 shows the corresponding bit error rate (BER). Consistent with the observation from Fig. 4 and 5, the two proposed algorithms achieve much smaller BER than direct interpolation and the scheme without using phase shifting. Compared to the scheme “w/o Itp”, they cause a performance degradation of about 2 dB, however, achieve a complexity save of approximately 85%.

In Figs. 7 and 8, we demonstrate simulation results for channel model 1 and $8 \times 8$ MIMO-OFDM systems, where the low-pass filter interpolator (Matlab function `interp`) is used, and the initial downsampling value $T_d$ is set to 8. The means of the actual downsampling rates are 2.11 and 3.975 for Algorithm 1 and 2, respectively. In terms of both normalized
Fig. 7. Normalized MSE of interpolated adjugate matrix for the $8 \times 8$ system and channel model 1. Solid and dashed curves for Algorithm 1 and 2 respectively. The proposed schemes even outperform the case without interpolation because of the noise smoothing effect of interpolation.

Fig. 8. BER for the uncoded $8 \times 8$ system and channel model 1. Solid and dashed curves for Algorithm 1 and 2 respectively. MSE and BER, our two proposed algorithms achieve similar performance to the scheme “w/o Itp”, but their complexities are only about 50% and 25% of the latter’s, respectively.

Figs. 9 and 10 show the simulation results for channel model 2 and $4 \times 4$ MIMO-OFDM systems, where the spline interpolator is used. The mean of the downsampling rates are found to be 4.77 and 6.13 for Algorithm 1 and 2, respectively. Compared to the results in Fig. 4, a slightly larger NMSE gap between the proposed schemes and the case without interpolation is observed from Fig. 9, particularly for Algorithm 2. However, the actual values of the NMSE for Algorithm 1 remain similar, which demonstrates the stability of our algorithms. Fig. 10 presents the BER results for both uncoded and coded systems, where $1/2$-rate convolution and hard decision Viterbi decoding is applied in coded systems. For clarity, only the curves for base points determined by Algorithm 1 are plotted. A BER degradation of $2 - 3$ dB can be observed from the figure, with a complexity save of approximately 80%.

VI. CONCLUSIONS

We have shown that the polynomial coefficients of the frequency-domain channel matrix can be well approximated by a Gaussian function when projecting the frequency-domain channel matrix as a polynomial matrix. Low-complexity and efficient schemes are then developed to estimate the parameters of the Gaussian approximation function. With these known parameters, we can accurately characterize the distribution of the power of the polynomial coefficients and develop various interpolation algorithms for matrix inversion, without requiring any prior knowledge on the channel matrix. Two exemplified algorithms are presented, and they can determine the number of required base points adaptively and improve the interpolation accuracy significantly by introducing phase shifted interpolation. Simulation results show that the proposed algorithms achieve BER performance very close to the case when channel matrix inversion is computed for every subcarrier, with a complexity reduction up to 85%.

The principle presented in this paper can be extended to
other matrix inversion and decomposition approaches such as QR decomposition and LU decomposition, wherever polynomial matrices are formed during the matrix operation. For QR decomposition based matrix inversion, polynomial matrices can be generated by using the auxiliary variables as proposed in [2]. Since the order of polynomial matrices decreases with the indices of the columns in the Q and R matrices, the Gaussian approximation accuracy decreases with the column index decreasing. Hence the proposed interpolation scheme may be better applied only to columns with larger indices.

**APPENDIX A: DERIVATION OF E[|b₁|^2]**

Firstly, we will show that for M sets of random vectors \( x₁, x₂, ..., x_M \) where all random variables are mutually independent, the variance of their repeated convolution \( y = \bigotimes_{m=1}^{M} x_m \) equals to the repeated convolution of the variance of the M vectors.

We consider two vectors first. Let \( x₁ = \{X₁, g\}, g = 1, 2, ..., G₁ \) and \( x₂ = \{X₂, g\}, g = 1, 2, ..., G₂ \) be two random vectors, and \( E[X_n, g] = 0 \). We know that given G independent complex random variables \( X_g, g = 1, 2, ..., G \), the variance of their product \( X = \prod_{g=1}^{G} X_g \) is [14]

\[
\text{Var}[X] = \sum_{g=1}^{G} \text{Var}[X_g] + E[X_g](E[X_g])^* - \prod_{g=1}^{G} E[X_g](E[X_g])^*
\]

When the mean is zero, it becomes \( \text{Var}[X] = \prod_{g=1}^{G} \text{Var}[X_g] \).

The \( p \)-th element in the convolution output \( y = x₁ \otimes x₂ \) is given by

\[
Y_p = \sum_{g=1}^{\min(p, G₂)} X₁, p - g X₂, g,
\]

and \( \text{Var}[Y_p] \) can be computed as

\[
\text{Var}[Y_p] = \sum_{g=1}^{\min(p, G₂)} \text{Var}[X₁, p - g X₂, g] = \sum_{g=1}^{\min(p, G₂)} \text{Var}[X₁, p - g] \text{Var}[X₂, g],
\]

Therefore \( \text{Var}[y] = \text{Var}[x₁] \otimes \text{Var}[x₂] \).

Repeatedly applying this process to \( M \) vectors, we get

\[
\text{Var}[\bigotimes_{m=1}^{M} x_m] = \bigotimes_{m=1}^{M} \text{Var}[x_m].
\]

For \( E[|b₁|^2] \), using the computational formula for the variance, we have

\[
E[|b₁|^2] = E[|b₁|] + E[|b₁|](E[|b₁|])^* = \text{Var}[b₁],
\]

where the second equality is from \( E[|b₁|] = 0 \) which can be easily verified.

Hence we get

\[
E[|b₁|^2] = \bigotimes_{i=1}^{M-1} \text{Var}[b₁, i].
\]

**APPENDIX B: PROOF OF THEOREM 1**

**Proof:** Theorem 1 can be proven by linking the repeated convolution in (10) with the sum of random variables.

Let \( Z = \sum_{i=1}^{M-1} Z_i \) where \( Z_i, i = 1, \cdots, M - 1 \) are mutually independent discrete random variables with discrete samples \( 0, 1, \cdots, L_{ii} - 1 \) and probability mass function (PMF)

\[
p_{z, i} \triangleq \{\sigma^2_{i,i}(0), \sigma^2_{i,i}(1), \cdots, \sigma^2_{i,i}(L_{ii} - 1)\}/\sum_{\ell=0}^{L_{ii}-1} \sigma^2_{i,i}(\ell).
\]

The PMF function of \( Z \) is given by

\[
p_z = \bigotimes_{i=1}^{M-1} p_{z, i}.
\]

Comparing (24) with (10), we can see that \( E[|b₁|^2] \) and \( p_z \) have almost the same expressions except for a scalar.

Exact closed-form PMF of \( Z \) is generally intractable. Let \( u₁ = \sum_{i=1}^{M-1} E[Z_i] \), and \( ε^2₁ = \sum_{i=1}^{M-1} \text{Var}[Z_i] \). When \( M \) approaches infinity, we can apply the Lyapunov central limit theory [15] to show that \( Z \) is a Gaussian random variable and \( p_z \) converges to a Gaussian distribution with mean \( u₁ \) and variance \( ε^2₁ \). This establishes Theorem 1.

**APPENDIX C: DFT OF A DISCRETE GAUSSIAN WAVEFORM**

It is known that the Fourier transform of a continuous Gaussian waveform is still Gaussian. Since a timing offset in the time domain corresponds to a phase shift in the frequency domain and we do not care about the phase when estimating \( ε \), we will ignore the timing offset and phase shift in the derivation.

Consider a continuous Gaussian waveform \( x(t) = \exp(-aτ^2/2), a > 0 \). Its Fourier transform is given by \( \tilde{x}_a(f) = \exp(-π^2f^2/2a^2) \) [16]. The \(-10 \text{ dB bandwidth} B \) of \( \tilde{x}_a(f) \) can be determined via

\[
20 \log_{10} επ^2 B^2/a^2 = 10,
\]

which leads to \( B = 0.3415a \) Hz. Let the sampling rate be \( T_s \).

To avoid spectrum aliasing, we require

\[
T_s \leq 1/(2B) = 1.464/a.
\]

Let the sampled signal from \( x(t) \) be

\[
x(t) = \exp(-a^2T_s^2t^2), t = -K/2, \cdots, 0, \cdots, K/2 - 1,
\]

and we assume \( N \) is large enough so that \( x(kT_s), |k| > K/2 \) can be ignored. Denote the DFT of \( x(t) \) as \( \tilde{x}_d(k) \). According to the relationship between the Fourier transform of a continuous signal and the DFT of its sampled version [17], we get

\[
\tilde{x}_d(k) = \exp \left( -\frac{π^2k^2}{a(KT_s)^2} \right).
\]
REFERENCES


Jian (Andrew) Zhang (M’04, SM’11) received the B.S. degree from Xi’an JiaoTong University, China, in 1996, the M.Sc. degree from Nanjing University of Posts and Telecommunications, China, in 1999, and the Ph.D. degree from the Australian National University, in 2004. Currently, Dr. Zhang is a senior research scientist in the ICT Centre, CSIRO, Sydney, Australia. From 1999 to 2001, he worked as a system and hardware engineer in ZTE Corp, Nanjing, China. From 2004 to 2010, he was a researcher in the Networked Systems, NICTA, Canberra, Australia. He is an adjunct honorary fellow in the Macquarie University and University of South Australia. Dr. Zhang’s research interests are in the area of signal processing for wireless communications, with focus on MIMO, multicarrier, ultra wideband and sensor networks. He has published more than 60 papers on leading international Journals and conferences. He is a recipient of CSIRO Chairman’s Medal and the Australian Engineering Innovation Award in 2012 for exceptional research achievements in multi-gigabit wireless communications.

Xiaoqing Huang (M’99, SM’11) received his Bachelor of Engineering, Master of Engineering, and Ph.D. degrees from Shanghai Jiao Tong University, Shanghai, China, in 1983, 1986, and 1989, respectively, all in electronic engineering.

From 1989 to 1994, Dr. Huang worked in the Electronic Engineering Department of Shanghai Jiao Tong University, where he had been a Lecturer since 1989 and an Associate Professor since 1991. From 1994 to 1997, he was the Chief Engineer with Shanghai Yang Tian Science and Technology Corporation Limited, Shanghai, China. In 1998, he joined the Motorola Australian Research Centre, Sydney, Australia, as a Senior Research Engineer and had been a Principal Research Engineer since 2003. From 2004 to 2009, he was an Associate Professor in the School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, Wollongong, Australia. In March, 2009, he joined the ICT Centre, Commonwealth Scientific and Industrial Research Organization (CSIRO), Sydney, Australia, as a Principal Research Scientist. He is also an Adjunct Professor at Macquarie University, Sydney, Australia. His research interests are in communications theory, digital signal processing, and wireless communications networks. With over 23 years of professional experiences, Dr. Huang has published over 150 technical papers and filed 27 patents. He is currently the project leader of CSIRO’s microwave and mm-wave backhaul projects and a recipient of CSIRO Chairman’s Medal and the Australian Engineering Innovation Award in 2012 for exceptional research achievements in multi-gigabit wireless communications.

Hajime Suzuki (S’94-M’99) received the B.E. and M.E. degrees from the University of Electro-Communications, Tokyo, Japan, in 1993 and 1995, respectively, and the Ph.D. degree from the University of Technology, Sydney, Australia, in 1999.

In 1999, he joined the Division of Telecommunications and Industrial Physics, Commonwealth Scientific and Industrial Research Organisation (CSIRO), Marsfield, Australia, and was transferred to the CSIRO Information and Communication Technologies Centre in 2003. Currently he is the project leader of Rural Broadband Access project investigating the use of multi-user multiple-input multiple-output (MU-MIMO) for high capacity wireless communications in rural areas. He is an active participant of the Australian Radio Communication Study Group 3 and of the Study Group 3 (Bands), International Telecommunication Union Radiocommunications Sector (ITU-R), particularly of the Working Party 3K (point-to-area propagation).

Zhuo Chen (S’01-M’05) received the B.S. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 1997 and the M.S. and Ph.D. degrees from the School of Electrical and Information Engineering, University of Sydney, Sydney, Australia, in 2001 and 2004, respectively. He is currently a Senior Research Scientist with the Wireless and Networking Technologies Laboratory, Commonwealth Scientific and Industrial Research Organization (CSIRO), Sydney. He is the Team Leader for the Digital Communications Group, and a Project Leader for several industrial projects. His research interests include wireless communications, multiple-input-multiple-output systems, error control coding techniques, and wireless sensor networks.