

A Multidimensional Phase-Locked Loop for Blind Equalization of Multi-Input Multi-Output Channels

John R. Barry and Anuj Batra

School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, Georgia 30332-0250

Abstract — We present a new technique for blind equalization of multiple-input multiple-output (MIMO) communication channels. In scalar channels it is common to follow a blind equalizer adapted according to the constant-modulus algorithm (CMA) by a phase-locked loop to compensate for carrier frequency offset. We generalize this structure to multiple dimensions, and propose to follow a blind MIMO equalizer adapted according to the vector CMA by a multidimensional phase-locked loop. We show through simulations that the new technique can converge much faster than prior techniques.

I. INTRODUCTION

A multiple-input, multiple-output (MIMO) channel model arises whenever a receiver observes signals from multiple transmitters through multiple sensors. Depending on the application, the interference among the transmitted signals may be called co-channel interference, crosstalk, or multiuser interference, and the mitigation of this interference has motivated extensive research [1–9].

A simple MIMO channel model is shown in Fig. 1. The channel input \mathbf{x}_k is a sequence of $n \times 1$ vectors whose components represent the symbol sequences of n different users. The channel transfer function is $\mathbf{H}(z)$. For simplicity we neglect noise, and we assume that $\mathbf{H}(z)$ is square, so that the dimension of the channel output \mathbf{r}_k is also n .

The objective of a MIMO equalizer, as shown in Fig. 1, is to mitigate both intersymbol interference and co-channel interference. Most prior work in MIMO equalization theory has assumed that the channel responses are known or that a training sequence is available. Only recently has the problem of blind MIMO equalization (without training) been addressed, and several blind algorithms based on vector generalizations of the constant-modulus algorithm (CMA) have been proposed [10,11].

Just as the scalar CMA is invariant to phase rotations, the vector CMA [11] is invariant to unitary transformations. In [10,11], this unitary rotation was avoided by modifying the

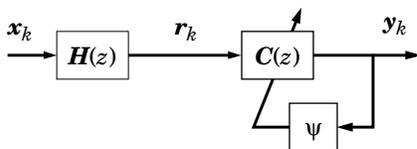


Fig. 1. A MIMO channel and a blind MIMO equalizer.

This research was supported by the Center for Research in Applied Signal Processing.

vector CMA cost function in an attempt to force each component of the vector-valued signal to have a constant modulus. This pointwise technique is directly analogous to the scalar algorithm of [12]. In contrast, in this paper we propose to identify the unitary rotation using a multidimensional phase-locked loop (PLL). This approach is a multidimensional generalization of the common scalar receiver consisting of a CMA equalizer followed by a decision-directed carrier loop [13].

In Sect. II we review two blind equalization algorithms for MIMO channels: the vector CMA [11] and the combination CMA [10]. In Sect. III we review the structure of a scalar PLL, and in Sect. IV we derive the multidimensional PLL. In Sect. V we present numerical examples.

II. BLIND EQUALIZATION FOR MIMO CHANNELS

Consider the adaptive MIMO linear equalizer structure shown in Fig. 1; it is a transversal filter with L matrix-valued taps described by the matrix $\mathbf{C}_k^T = [\mathbf{C}_{0,k} \mathbf{C}_{1,k} \dots \mathbf{C}_{L-1,k}]$. Let $\mathbf{R}_k^T = [\mathbf{r}_k^T \mathbf{r}_{k-1}^T \dots \mathbf{r}_{k-L+1}^T]$ be a vector of equalizer inputs, so that the equalizer output at time k is $\mathbf{y}_k = \mathbf{C}_k^T \mathbf{R}_k$.

A. Vector Constant Modulus Algorithm

As described in [11], the vector CMA for blind equalization is a natural extension of the conventional CMA [14] to MIMO channels. The vector CMA cost function is:

$$J = \mathbf{E}[(\|\mathbf{y}_k\|^2 - M)^2], \quad (1)$$

where $M = \mathbf{E}[\|\mathbf{x}_k\|^4] / \mathbf{E}[\|\mathbf{x}_k\|^2]^2$, and the weight update equation for the vector CMA is:

$$\mathbf{C}_{k+1} = \mathbf{C}_k - \mu \mathbf{R}_k^* \mathbf{e}_k^T, \quad (2)$$

where μ is the step size and the error signal is defined as:

$$\mathbf{e}_k = \mathbf{y}_k (\|\mathbf{y}_k\|^2 - M). \quad (3)$$

The cost function (1) is invariant to unitary transformations. In other words, if $\mathbf{F}(z) = \mathbf{C}(z)\mathbf{H}(z)$ is the overall system transfer function from the channel input to the equalizer output, then the cost function will be minimized when \mathbf{F} is any unitary matrix.

If the components of \mathbf{x}_k are selected from an alphabet with constant modulus, then \mathbf{F} need not be unitary to minimize J . For example, a diagonal matrix $\mathbf{F} = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_n]$ will minimize J when $\sum |\lambda_i|^2 = n$ and the alphabet is constant modulus. In addition, if the k -th row of \mathbf{F} contains only one nonzero entry of the form $e^{j\theta_k}$ for some θ_k , such as $\begin{bmatrix} 1 & 0 \\ & 1 & 0 \end{bmatrix}$, then \mathbf{F} will also minimize J .

However, if the alphabet is not constant modulus and the channel is memoryless, then we hypothesize that only a unitary matrix will minimize the vector CMA cost function:

Conjecture. If the components of \mathbf{x} are independent and uniformly distributed over a discrete alphabet with non-constant modulus, then $\mathbf{y} = \mathbf{F}\mathbf{x}$ minimizes the cost function $J = \mathbb{E}[(\|\mathbf{y}\|^2 - M)^2]$ if and only if \mathbf{F} is unitary.

B. Combination of Vector and Pointwise CMA

To prevent the equalizer from converging to a unitary matrix rather than the identity matrix, Oda and Sato modified the vector cost function by adding a pointwise component, yielding the so-called *combination CMA* cost function [10]:

$$J_c = A\mathbb{E}[(\|\mathbf{y}_k\|^2 - M)^2] + B \sum_{i=1}^n \mathbb{E}[|y_k^{(i)}|^2 - M_i]^2, \quad (4)$$

where $M_i = \mathbb{E}[|x_k^{(i)}|^4]/\mathbb{E}[|x_k^{(i)}|^2]$ and A and B are positive constants. The purpose of the first term in (4) is to force \mathbf{y}_k to have a constant modulus, and the purpose of the second term is to force each component $y_k^{(i)}$ of \mathbf{y}_k to have a constant modulus. Although a unitary matrix will minimize the first term, it will not minimize the second term. The combination CMA reduces to the *pointwise CMA* when $A = 0$ [11].

Although originally defined for multidimensional systems, the combination CMA can be used in a complex-valued scalar system by viewing it as two-dimensional real-valued system. Indeed, this viewpoint is adopted in [12] to eliminate carrier phase offset. However, it is much more common to append a phase-locked loop after a conventional scalar CMA equalizer to eliminate carrier phase offset [13].

The simplicity and effectiveness of the CMA equalizer-PLL cascade in scalar systems has motivated us to propose a similar structure for MIMO systems, as described in Sect. IV. Before describing the MIMO PLL, however, we first review the components of a scalar PLL.

III. A SCALAR PHASE-LOCKED LOOP

Perhaps the most popular structure for blind equalization of scalar channels is a CMA equalizer followed by a decision-directed PLL to track phase error, as illustrated in Fig. 2-a [13]. The scalar CMA cost function is insensitive to the phase of the received signal, so that any carrier frequency offset between the receiver local oscillator and the transmitter oscillator will cause the constellation after the equalizer to rotate. Let θ_k be the phase offset at time k , so that after convergence the equalizer output y_k is related to the transmitted symbol x_k by $y_k = e^{j\theta_k}x_k$. The carrier recovery loop generates $U_k = e^{j\phi_k}$, where ϕ_k is the receiver's estimate of θ_k . If $\phi_k \approx \theta_k$, then the product $z_k = U_k^* y_k$ reduces to $z_k \approx x_k$, and decisions based on z_k will be accurate.

As shown in Fig. 2-a, the decision-directed carrier loop generates U_k by first estimating the phase difference ϵ_k between the slicer input z_k and slicer output \hat{x}_k using a phase detector. To deal with the noisy case when z_k and \hat{x}_k may have different magnitudes, it is common for the phase detector to normalize

both to have unity magnitude before estimating the phase difference, yielding an estimate of [15]:

$$\epsilon_k = \angle\{x^* z\} = \sin^{-1}[\text{Im}\{x^* z\}], \quad (5)$$

where $x = \hat{x}_k / |\hat{x}_k|$ and $z = z_k / |z_k|$.

The phase estimate ϵ_k is then passed through a loop filter with constant transfer function $L(z) = K$. (A constant loop filter is sufficient to track a constant phase error, but a higher-order filter is necessary to track a constant frequency error.) The loop filter output drives a VCO whose output is U_k . By definition, the VCO output U_k is related to its input $K\epsilon_k$ by:

$$U_k = \exp\{j \sum_{i=0}^k K\epsilon_i\} = \prod_{i=0}^k \exp\{jK\epsilon_i\}. \quad (6)$$

Therefore, a VCO can be viewed in two ways: as a sum-accumulator followed by a complex exponentiator, or as a complex exponentiator followed by a product-accumulator. We choose the second viewpoint to facilitate its generalization to vector signals.

As shown in the figure, we define the *rotation detector* as the cascade of the phase detector, loop filter, and exponentiator. Its output $T_k = \exp\{jK\epsilon_k\}$ rotates a fraction K of the way from \hat{x}_k to z_k . The positive constant K is typically much less than unity. Note that, using (5), T_k can also be expressed as $T_k = (x^* z)^K$.

IV. A MULTIDIMENSIONAL PHASE-LOCKED LOOP

We now propose a new blind equalization technique for MIMO channels by generalizing the scalar PLL of the previous section to multiple dimensions. Specifically, rather than modifying the vector CMA equalizer to avoid unitary rotations, we propose to append a multidimensional PLL to identify and eliminate any rotation.

Because the vector CMA cost function is invariant to unitary transformations, the equalizer output \mathbf{y}_k after convergence may be related to the transmitted symbol vector \mathbf{x}_k by a unitary matrix. We propose to append a multidimensional generalization

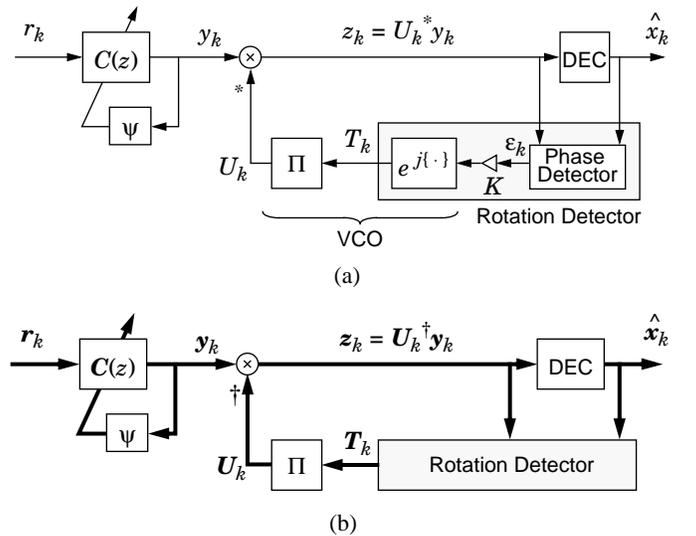


Fig. 2. A CMA equalizer followed by a decision-directed PLL for (a) scalar channels; (b) MIMO channels.

of the scalar decision-directed phase-locked loop to generate an estimate \mathbf{U}_k of this unitary matrix. If this estimate is accurate, then the product $\mathbf{z}_k = \mathbf{U}_k^\dagger \mathbf{y}_k$ [where $(\cdot)^\dagger$ denotes conjugate transpose] reduces to $\mathbf{z}_k \approx \mathbf{x}_k$, so that decisions based on \mathbf{z}_k will be accurate.

As shown in Fig. 2-b, the MIMO receiver generates \mathbf{U}_k in a way similar to the scalar receiver. The key component of the loop is the rotation detector, whose output \mathbf{T}_k is a unitary rotation matrix that approximately rotates a fraction of the way from $\hat{\mathbf{x}}_k$ to \mathbf{z}_k . Specifically, let \mathbf{z}' be a vector between $\hat{\mathbf{x}}_k$ and \mathbf{z}_k :

$$\mathbf{z}' = \lambda \mathbf{z}_k + (1 - \lambda) \hat{\mathbf{x}}_k, \quad (7)$$

where $\lambda \in (0, 1)$, and define the unit-length vectors \mathbf{x} and \mathbf{z} by:

$$\mathbf{x} = \hat{\mathbf{x}}_k / \|\hat{\mathbf{x}}_k\| \quad \text{and} \quad \mathbf{z} = \mathbf{z}' / \|\mathbf{z}'\|. \quad (8)$$

Then we define \mathbf{T}_k as a unitary matrix mapping \mathbf{x} to \mathbf{z} .

Unlike the scalar case, \mathbf{T}_k is not unique. We now derive the family of unitary matrices \mathbf{T}_k mapping the unit-length vector \mathbf{x} to the unit-length vector \mathbf{z} . Let p denote the inner product:

$$p = \mathbf{x}^\dagger \mathbf{z}. \quad (9)$$

Because \mathbf{x} and \mathbf{z} are unit length, p satisfies $|p| \leq 1$. We consider the cases $|p| = 1$ and $|p| < 1$ separately, starting with $|p| = 1$.

If $|p| = 1$ then $\mathbf{z} = p\mathbf{x}$. Let $\mathbf{V} = [\mathbf{x}, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4, \dots, \mathbf{v}_n]$ be a unitary matrix whose columns form a basis for the entire space. In this case, the set of all unitary matrices \mathbf{T}_k mapping \mathbf{x} to \mathbf{z} is given by:

$$\mathbf{T}_k = \mathbf{V} \begin{bmatrix} p & \mathbf{0} \\ \mathbf{0} & \mathbf{J} \end{bmatrix} \mathbf{V}^\dagger, \quad (10)$$

where \mathbf{J} is an arbitrary $(n-1) \times (n-1)$ unitary matrix. Because \mathbf{x} and \mathbf{z} convey information about the one-dimensional subspace spanned by \mathbf{x} only, we should choose \mathbf{J} so as not to affect the vectors orthogonal to this subspace; in other words, we should choose $\mathbf{J} = \mathbf{I}$, in which case (10) reduces to:

$$\mathbf{T}_k = \mathbf{I} + (p-1)\mathbf{x}\mathbf{x}^\dagger \quad (\text{when } |p| = 1). \quad (11)$$

On the other hand, if $|p| < 1$, then the span of \mathbf{x} and \mathbf{z} is a two-dimensional subspace. Let us introduce a basis $\{\mathbf{x}, \mathbf{y}\}$ for this subspace, where from the Gram-Schmidt procedure we have:

$$\mathbf{y} = (\mathbf{z} - p\mathbf{x}) / \sqrt{1 - |p|^2}. \quad (12)$$

(The basis vector \mathbf{y} should not be confused with the equalizer output \mathbf{y}_k .) Let $\mathbf{V} = [\mathbf{x}, \mathbf{y}, \mathbf{v}_3, \mathbf{v}_4, \dots, \mathbf{v}_n]$ be a unitary matrix whose columns form an orthonormal basis of the entire space. In terms of this basis, \mathbf{x} and \mathbf{z} are given by the vectors $[1, 0, \dots, 0]^T$ and $[p, \sqrt{1 - |p|^2}, 0, 0, \dots, 0]^T$, respectively. It can be shown that any unitary matrix mapping $[1, 0]^T$ to $[p, \sqrt{1 - |p|^2}]^T$ must have the form:

$$\tilde{\mathbf{R}} = \begin{bmatrix} p & -\sqrt{1 - |p|^2} \\ \sqrt{1 - |p|^2} & p^* \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & e^{j\beta} \end{bmatrix}, \quad (13)$$

for some $\beta \in (-\pi, \pi]$. Therefore, the set of all unitary matrices \mathbf{T}_k mapping \mathbf{x} to \mathbf{z} is given by:

$$\mathbf{T}_k = \mathbf{V} \begin{bmatrix} \tilde{\mathbf{R}} & \mathbf{0} \\ \mathbf{0} & \mathbf{J} \end{bmatrix} \mathbf{V}^\dagger, \quad (14)$$

where $\beta \in (-\pi, \pi]$ and \mathbf{J} is an arbitrary $(n-2) \times (n-2)$ unitary matrix.

Again, because \mathbf{x} and \mathbf{z} convey information about the subspace spanned by \mathbf{x} and \mathbf{z} only, we choose \mathbf{J} so as not to affect the vectors orthogonal to this subspace; in other words, we choose $\mathbf{J} = \mathbf{I}$. The choice of β is not as obvious. Observe from (13) and (14) that $\det \mathbf{T}_k = e^{j\beta}$ when $\mathbf{J} = \mathbf{I}$ and $|p| < 1$. On the other hand, from (11) we see that $\det \mathbf{T}_k = p$ when $|p| = 1$. This suggests that, to be consistent in our choice of β as $|p|$ approaches unity, we should choose $\beta = \angle p = \sin^{-1}[\text{Im}\{p/|p|\}]$. The choice $\beta = \angle p$ also minimizes the Frobenius norm of $\tilde{\mathbf{R}} - \mathbf{I}$, which is intuitively pleasing because we expect \mathbf{T}_k to approximate \mathbf{I} near convergence. For these reasons we choose $\beta = \angle p$ in our algorithm. With $\mathbf{J} = \mathbf{I}$ and $\beta = \angle p$, (14) reduces to:

$$\mathbf{T}_k = \mathbf{I} + \begin{bmatrix} \mathbf{x}, \mathbf{y} \end{bmatrix} \begin{bmatrix} p-1 & \frac{-p}{|p|\sqrt{1-|p|^2}} \\ \sqrt{1-|p|^2} & |p|-1 \end{bmatrix} \begin{bmatrix} \mathbf{x}^\dagger \\ \mathbf{y}^\dagger \end{bmatrix} \quad (|p| < 1). \quad (15)$$

In summary, the rotation detector of Fig. 2-b is defined by (15) and (11), together with (7) – (9) and (12), and is parameterized by the constant λ .

As in the scalar case of (6), the MIMO VCO accumulates the rotation vectors, so that the receiver's estimate of the overall rotation matrix at time k is given by:

$$\mathbf{U}_k = \prod_{i=0}^k \mathbf{T}_i, \quad (16)$$

which can be implemented recursively:

$$\begin{aligned} \mathbf{U}_{-1} &= \mathbf{I}, \\ \mathbf{U}_k &= \mathbf{T}_k \mathbf{U}_{k-1}. \end{aligned} \quad (17)$$

We remark that a second-order loop is not necessary, and that the first-order MPLL described above is sufficient for our purposes, because the unitary matrix being estimated after convergence will be a constant.

V. SIMULATION EXAMPLES

In this section we present simulation results for three experiments, and compare the performance of the multidimensional PLL with that of the combination and pointwise CMA equalizers. In all cases we assume a noiseless 2×2 system, with the two transmitters being independent and uniformly distributed over a 16-QAM alphabet. We assume that the combination CMA parameters are $A = 4$ and $B = 1$, and that each equalizer is initialized to the identity matrix. The modulus parameters for a two-user 16-QAM system are $M = 23.2$ and $M_i = 13.2$.

A. Experiment One: Memoryless Unitary Channel

In the first experiment we consider a channel matrix that is unitary, so as to demonstrate the speed with which the multidimensional PLL can identify and eliminate unitary transformations. Specifically, consider a memoryless two-user system with a unitary channel matrix $\mathbf{H} = \mathbf{R}$, where \mathbf{R} is defined in (13) with parameters $p = 0.5 + j\pi^{-1}$ and $\beta = \pi/\ln 2$. Assume that the equalizer has one tap, and assume a zero step size for vector CMA and a step size of $\mu = 10^{-4}$ for both pointwise and combination CMA. (A nonzero step size for vector CMA will not appreciably affect the results, because the output of a unitary channel already minimizes the vector CMA cost function; hence, a nonzero step size will merely cause the equalizer to hover about the identity matrix.) The MPLL parameter was chosen to be $\lambda = 0.1$.

In Fig. 3 we plot the eight MPLL coefficients $\{\text{Re}[U_{ij}], \text{Im}[U_{ij}], i, j = 1, 2\}$ versus time. Observe that the MPLL converges after only about 100 iterations. In Fig. 4 we plot the real part of

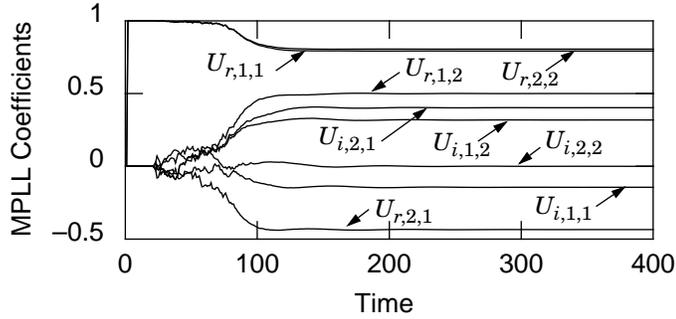


Fig. 3. PLL coefficients versus time for experiment one.

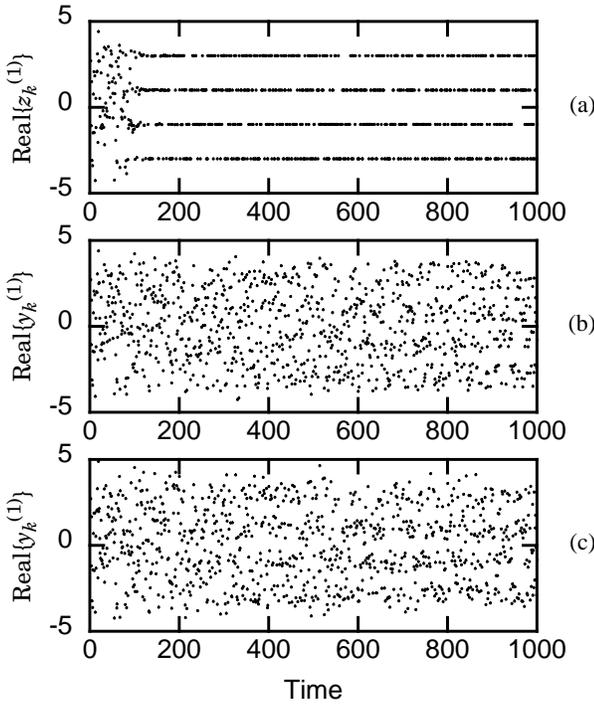


Fig. 4. Real part of output 1 versus time for experiment one: (a) vector CMA after MPPL; (b) pointwise CMA; (c) combination CMA.

output one versus time for the three algorithms. (The results for the imaginary part and those for output two are not shown but are qualitatively similar.) In Fig. 5 we show the constellation diagrams (for time 200 to 400) before the PLL, after the PLL, for the pointwise CMA, and for the combination CMA. These figures show that the MPLL converges quicker and more abruptly than the other techniques.

B. Experiment Two: Memoryless Near-Singular Channel

In this experiment we consider a memoryless two-user system with the following channel matrix:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 \\ 1 & 0.5 \end{bmatrix}. \quad (18)$$

Although there is no crosstalk for the first user, the crosstalk for the second user is particularly severe, with the interference power exceeding the signal power by 6 dB. With this channel there is a tendency for some blind equalization algorithms to converge to an undesirable overall transfer function of:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}. \quad (19)$$

Assume step sizes of 8×10^{-4} , 5×10^{-4} , and 10^{-4} for vector CMA, pointwise CMA, and combination CMA, respectively. Assume a MPLL parameter of $\lambda = 0.05$.

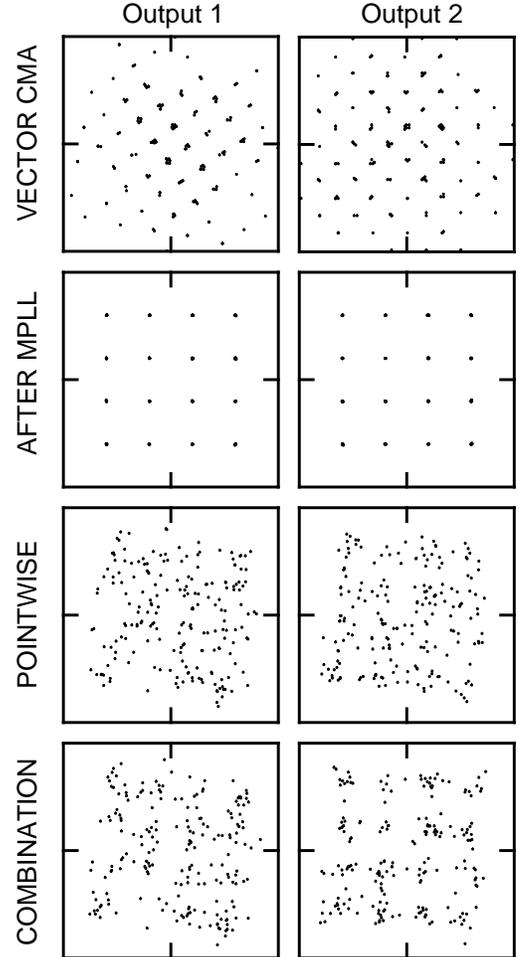


Fig. 5. Constellations for experiment one (time 200 to 400).

In Fig. 6-a we plot the eight vector CMA equalizer coefficients $\{\text{Re}[C_{ij}], \text{Im}[C_{ij}], i, j = 1, 2\}$ versus time, and in Fig. 6-b we plot the MPLL coefficients $\{\text{Re}[U_{ij}], \text{Im}[U_{ij}], i, j = 1, 2\}$ versus time. Observe that the PLL achieves lock near time 500.

In Fig. 7 we show the constellation diagrams (from time 1000 to 1200) for the different algorithms. The constellations after the MPLL are clean, and the overall transfer function $\mathbf{U}^\dagger \mathbf{C}_v \mathbf{H}$ converges approximately to the identity matrix. On the other hand, despite the fact that the constellations for the pointwise algorithm appear to be clean, the overall transfer function $\mathbf{C}_p \mathbf{H}$ converges approximately to the undesirable matrix of (19); thus, the pointwise equalizer is a failure, because both of its outputs lock onto the signal of user 1, ignoring the signal of user 2.

C. Experiment Three: Channel with Memory

Consider a system with the following channel matrix:

$$\mathbf{H}(z) = \begin{bmatrix} e^{j0.4\pi} & 0.7e^{j0.3\pi} \\ 0.3e^{j1.2\pi} & 0.9e^{j0.9\pi} \end{bmatrix} + \begin{bmatrix} 0.3e^{j0.7\pi} & 0 \\ 0.4e^{j0.1\pi} & 0.1e^{j0.4\pi} \end{bmatrix} z^{-1}. \quad (20)$$

Assume a five-tap equalizer, and assume step sizes of 10^{-4} , 10^{-5} , and 2×10^{-5} for vector CMA, pointwise CMA, and combination CMA, respectively. Assume a MPLL parameter of $\lambda = 0.03$.

In Fig. 8 we plot the real part of output 1 versus time for the vector CMA with MPLL and combination CMA algorithms. In Fig. 9 we show the constellations for the various algorithms from time 4600 to 5000. As before, the MPLL converges faster than the other equalizers.

In each of the above experiments, the vector CMA with a multidimensional PLL converged faster than the combination CMA or the pointwise CMA. Although the simulations considered only a few sample channels, we expect faster convergence on a broad class of channels. Intuitively, this is because the channel matrix will almost always be closer to a unitary matrix

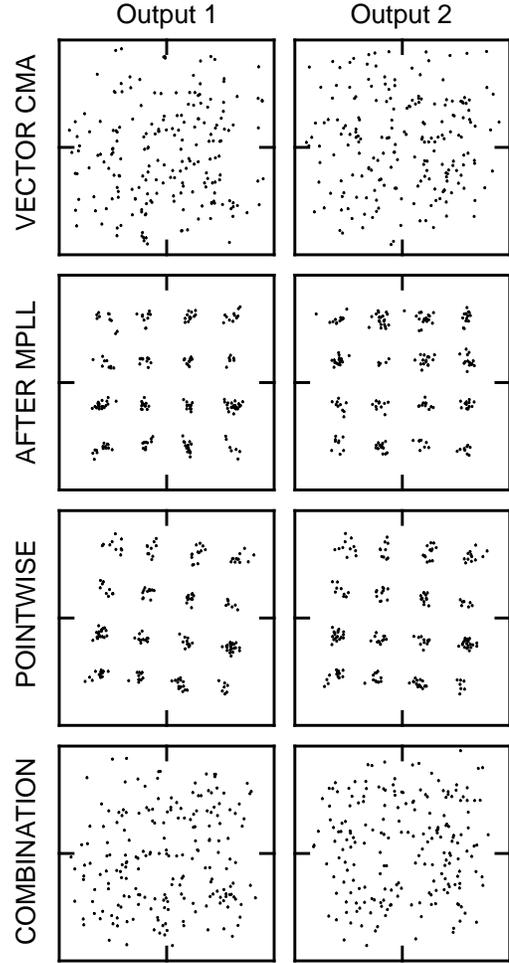


Fig. 7. Constellations for experiment two (time 1000 to 1200).

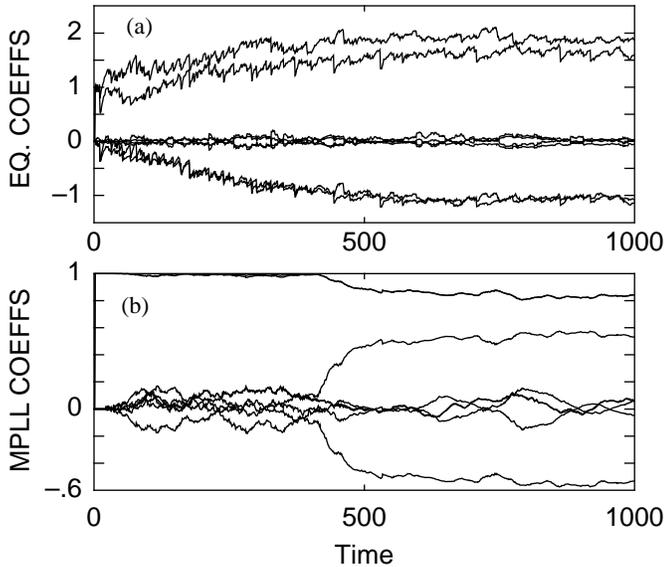


Fig. 6. Experiment two: (a) Equalizer coefficients for vector CMA versus time; (b) MPLL coefficients versus time.

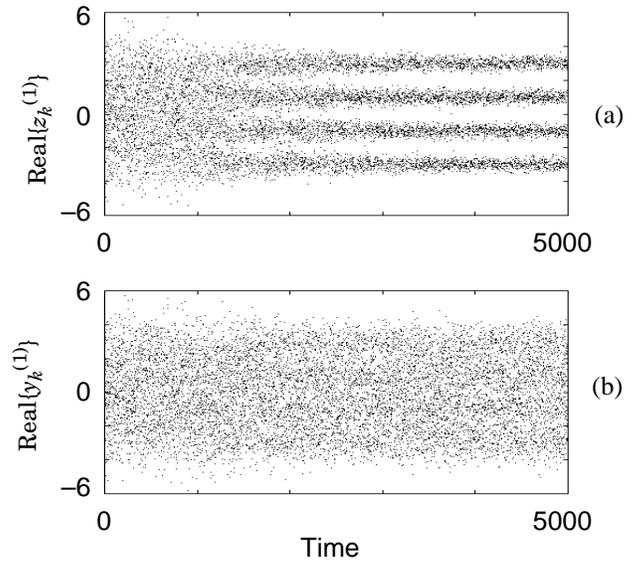


Fig. 8. Experiment three: Output 1 versus time for (a) MPLL; (b) combination CMA.

(of which there are infinitely many) than to the identity matrix. Therefore, if the equalizer is initialized to the identity matrix, the overall transfer function starts out much closer to a unitary matrix than to the identity matrix; thus, the equalizer can converge much faster if it need only force the overall transfer function to be unitary. As illustrated dramatically in Sect. V.A, the MPLL is able to eliminate unitary transformations much faster than the pointwise and combination CMA equalizers.

VI. SUMMARY

After convergence of the vector CMA, the overall transfer function from the channel input to the equalizer output may be an arbitrary unitary matrix. To identify and eliminate this rotation matrix, we generalized the decision-directed phase-locked loop to vector-valued signals. The cascade of a blind MIMO equalizer adapted according to the vector CMA and a multidimensional PLL is a natural extension of a commonly used strategy for blind equalization of scalar channels. Simulation results show that the vector CMA with a multidimensional PLL can converge much faster than other known blind MIMO equalization algorithms.

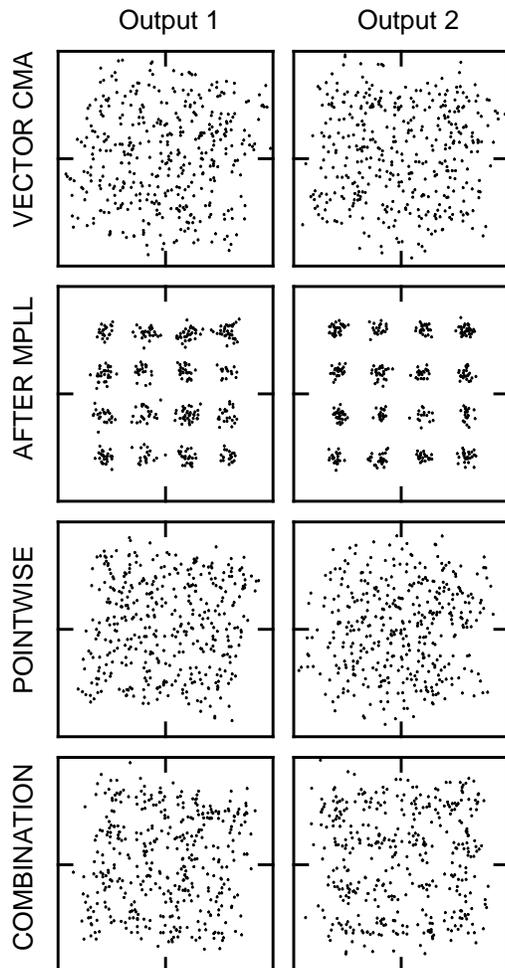


Fig. 9. Constellations for experiment three (time 4600 to 5000).

VII. REFERENCES

- [1] W. van Etten, "An Optimum Linear Receiver for Multiple Channel Digital Transmission Systems," *IEEE Transactions on Communications*, pp. 828-834, Aug. 1975.
- [2] J. Salz, "Digital Transmission Over Cross-Coupled Linear Channels," *AT&T Technical Journal*, 64:6, pp. 1147, July-Aug. 1985.
- [3] S. Verdu, "Recent progress in Multi-User Detection," in *Advances in Communications and Signal Processing*, W. A. Porter and S. C. Kak, Eds. Springer-Verlag, New York (1989).
- [4] A. Duel-Hallen, "Equalizers for Multiple-Input/Multiple-Output Channels and PAM Systems with Cyclostationary Input Sequences," *IEEE J. Select. Areas in Comm.*, 10:3, pp. 630-639, Apr. 1992.
- [5] M. L. Honig, P. Crespo, and K. Steiglitz, "Suppression of Near- and Far-End Crosstalk by Linear Pre- and Post-Filter," *IEEE J. Select. Areas in Comm.*, 10:3, pp. 614-629, Apr. 1992.
- [6] D. D. Falconer, M. Abdulrahman, N. W. K. Lo, B. R. Peterson, and A. U. H. Sheikh, "Advances in Equalization and Diversity for Portable Wireless Systems," *Digital Signal Processing*, vol. 3, no. 3, pp. 148-162, July 1993.
- [7] J. H. Winters, J. Salz, R. D. Gitlin, "The Impact of Antenna Diversity on the Capacity of Wireless Communication Systems," *IEEE Transactions on Communications*, vol. 42, no. 2-4, pp. 1740-51, Feb.-April 1994.
- [8] J. Yang and S. Roy, "Joint Transmitter-Receiver Optimization for Multi-Input Multi-Output Systems with Decision Feedback," *IEEE Trans. Info. Theory*, 40:5, pp. 1334-1347, Sep. 1994.
- [9] P. A. Voois and J. M. Cioffi, "Multichannel Signal Processing for Multiple-Head Digital Magnetic Recording," *IEEE Trans. Magnetics*, 30:6, pp. 5100-5114, Nov. 1994.
- [10] H. Oda, Y. Sato, "A Method of Multi-Dimensional Blind Equalization," *Intern. Symp. on Info. Theory*, San Antonio, p. 327, 1993.
- [11] A. Batra, J. R. Barry, "Blind Cancellation of Co-Channel Interference," *IEEE Global Telecommunications Conference*, Singapore, vol. 1, pp. 157-162, November 1995.
- [12] K. N. Oh and Y. O. Chin, "Modified Constant Modulus Algorithm: Blind Equalization and Carrier Phase Recovery Algorithm," *ICC '95*, Seattle, pp. 498-502, May 1995.
- [13] N. K. Jablon, "Joint Blind Equalization, Carrier Recovery, and Timing Recovery for High-order QAM Signal Constellations," *IEEE Trans. on Sign. Proc.*, 40:6, pp. 1383-1398.
- [14] J. R. Treichler, B. G. Agee, "A New Approach to Multipath Correction of Constant Modulus Signals," *IEEE Trans. Acous., Speech, and Sign. Proc.*, 31:2, pp. 459-472, Apr. 1983.
- [15] E. A. Lee and D. G. Messerschmitt, *Digital Communication*, Second Edition, Kluwer (1994), p. 728.