Optimal Phase Control for Equal-Gain Transmission in MIMO Systems With Scalar Quantization: Complexity and Algorithms

Kin-Kwong Leung, Chi Wan Sung, Majid Khabbazian, and Mohammad Ali Safari

Abstract—**The complexity of the optimal phase control problem in wireless MIMO systems with scalar feedback quantization and equal-gain transmission is studied. The problem is shown to be NP-hard when the number of receive antennas grows linearly with the number of transmit antennas. For the case where the number of receive antennas is constant, the problem can be solved in polynomial time. An optimal algorithm is explicitly constructed. For practical purposes, a low-complexity algorithm based on local search is presented. Simulation results show that its performance is nearly optimal.**

Index Terms—**Beamforming, closed-loop diversity, equal-gain transmission, limited feedback, MIMO systems, phase control.**

I. Introduction

Space diversity is a key technology to improve the performance of a multiple-input multiple-output (MIMO) wireless communication system in a fading environment, which can be utilized at the transmitter, or the receiver, or both. At the receiver side, it can be achieved by suitably combining multiple copies of receive signals to boost the average signal-to-noise ratio (see [19] and the references therein). At the transmitter side, different techniques are available, which are usually classified into open loop or closed loop, depending on whether there is any feedback information sent from the receiver to the transmitter [5]. Open-loop diversity can be realized by the use of space-time codes (e.g., [1], [11], [22], [23]). Closed-loop diversity, first proposed by Gerlach and Paulraj [9], can yield a better performance at the expense of feedback information. Based on the estimated channel gain matrix, the receiver calculates the optimal antenna weight vector and then sends it back to the transmitter. This technique is also called beamforming [6], [7]. When space diversity is employed at both the transmitter and receiver, a diversity order in proportion to the product of the number of transmit and receive antennas can be achieved in the narrow-band Rayleigh fading channel.

Manuscript received March 25, 2008; revised July 31, 2009. Current version published June 16, 2010.

K.-K. Leung is with the Cluster Technology, Ltd., Hong Kong Science Park, Hong Kong (e-mail: kinkwong@clustertech.com).

C. W. Sung is with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong (e-mail: albert.sung@cityu.edu.hk).

M. Khabbazian is with the MIT Computer Science and Artificial Intelligence Laboratory, Cambridge, MA 02139 USA (e-mail: majidk@mit.edu).

M. A. Safari is with the Department of Computer Engineering, Sharif University of Technology, Tehran, Iran (e-mail: msafari@sharif.edu)

Communicated by A. J. Goldsmith, Associate Editor for Communications.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2010.2048457

In this paper, we focus on closed-loop diversity. Since information needs to be fed back to the transmitter from the receiver via a feedback channel typically of limited bandwidth, the optimal antenna weight vector has to be quantized before being sent back. If the quantization level is sufficiently dense, one may simply choose a quantized point that is closest in Euclidean distance to the optimal vector. In practice, however, the allowed feedback information is rather limited. Therefore, two questions need to be answered:

1) How should one design the set of quantized weight vectors? We call this set *quantization set* or *codebook*.
2) Given the codebook, how to find the element with optimal beamforming? This is the codeword assignment problem.

A simple design of the codebook is to quantize the amplitude and phase of each transmit antenna independently. This class of quantization schemes is called *scalar quantization* (SQ). To obtain better beamforming performance, one can consider jointly quantizing the parameters for different antennas. This class of schemes is called *vector quantization* (VQ). VQ codebook design has been widely studied in the literature, e.g., [18]. A design criterion that guarantees full diversity order is obtained in [14] by relating it to Grassmannian line packings [4]. Codebooks that meet Welch lower bound in certain cases are constructed in [25].

In practice, additional constraints on the codebook may be imposed. For example, a technique called equal-gain transmission, which does not require the antenna amplifier to modify the amplitudes of the transmit signal, allows the use of inexpensive amplifiers at the antennas. Indeed, equal-gain transmission at the transmitter and equal-gain combining at the receiver have already been considered low-cost alternatives to maximal ratio transmission and combining, respectively [3], [17], [26]. Codebook design for equal-gain transmission can be found in [13], [16]. A separate gain and phase codebook is designed in [20].

Given a codebook, we have to assign an appropriate codeword in real time, based on the current channel realization. In the ideal case where the codebook is the entire complex space, it is well known that the optimal codeword can be obtained by solving an eigenvector problem. With the additional equal-gain constraint, the problem is shown to be nonconvex, and semi-definite relaxation can be used to obtain approximate solutions [27]. In practice, however, the codebook should consist of only a finite number of points, which can be designed by SQ and VQ techniques as mentioned above. To find the optimal codeword, most of the aforementioned papers assume exhaustive search, which is computationally inefficient.

In this paper, we investigate whether fast algorithm exists. While many VQ codebooks are generated by numerical methods and have no explicit form, we conduct our studies under the assumptions of SQ codebooks and equal-gain transmission. In other words, all the feedback bits are used to adjust the transmission *phases* of the transmit antennas. We call it the *optimal phase control* problem for SQ system, which is combinatorial in nature. Finding the optimal solution by brute force has computational complexity increasing exponentially with the number of transmit antennas, N. Suboptimal solutions for this problem have been proposed in the literature. One example is the Co-phase Algorithm, whose complexity is linear with N [10], [12]. Another suboptimal approach is Local Search [21], which has better performance but higher complexity. It is not clear, however, whether polynomial-time optimal algorithm exists. We provide an answer to this question in this paper. Our results are summarized as follows:

- When the number of receive antennas grows linearly with transmit antennas, the optimal phase control problem is NP-hard. An implication of this result is that the optimal beamforming problem with VQ codebooks is also NP-hard.
- When either the number of receive or transmit antennas is constant and the other number grows, the problem can be solved in polynomial time. The proof is constructive.

The rest of this paper is organized as follows. The next section is the problem definition. The NP-hardness of the first case is proved in Section III, by reducing the maximum cut, which is a well-known NP-complete problem in graph theory. In Section IV, we prove the existence of polynomial time algorithm for the second case by construction. In Section V, we consider sub-optimal algorithms for solving the general phase control problem. Section VI is the simulations and the paper is concluded in Section VII. Most of the proofs are put in the appendices.

II. Problem Definition

We consider a single user multiple-input multiple-output (MIMO) system, with N transmit and M receive antennas. We call it an $(N \times M)$-MIMO system. We assume that the channels are under flat fading and let $\boldsymbol{G}$ be an $M \times N$ complex matrix with entry $g_{m,n}$ representing the link gain between transmit antenna n and receive antenna m, and $\boldsymbol{u} = (u_1, \ldots, u_N)$ be an N-dim complex column vector representing the transmit signal. Assume that the channel is quasi-static so that $\boldsymbol{G}$ remains constant for the duration of a block. For simplicity, we omit the time index, and, therefore, the noise-free receive signal is

$$\boldsymbol{v} = \boldsymbol{G}\boldsymbol{u}. \tag{1}$$

Assume that the receiver end has perfect channel state information (CSI). It controls the signals of the transmit antennas through a band-limited feedback channel.

A. General Beamforming Problem

The general beamforming problem assumes that a predefined codebook $\mathcal{U}_{\mathrm{VQ}}$ of transmit vectors is known to both transmit and receive ends. The receiver calculates the optimal beamforming vector based on the CSI and sends the index of the vector back to the transmitter. The optimal beamforming is to determine $\boldsymbol{u} \in \mathcal{U}_{\mathrm{VQ}}$ such that the power of the receive signal $\|\boldsymbol{v}\|^2 = \sum_{m=1}^{M} |v_m|^2$ is maximized. This is equivalent to maximizing the signal-to-noise ratio (SNR), which is optimal for maximum likelihood decoding, provided that the noise term is additive and white Gaussian.

Therefore, the optimal beamforming problem can be expressed as

$$\begin{array}{ll} \text{maximize} & \|\boldsymbol{G}\boldsymbol{u}\|^2 \\ \text{subject to} & \boldsymbol{u} \in \mathcal{U}_{\mathrm{VQ}}. \end{array} \tag{2}$$

We denote

$$f(\boldsymbol{u}) = \|\boldsymbol{G}\boldsymbol{u}\|^2 = \boldsymbol{u}^H \boldsymbol{G}^H \boldsymbol{G}\boldsymbol{u} \tag{3}$$

where $\boldsymbol{A}^H$ is the Hermitian transpose of the complex matrix $\boldsymbol{A}$. We use $\boldsymbol{A}^T$ to denote its transpose.

B. Phase Control Problem

We also define a special case of the general beamforming problem—the phase control problem. It assumes equal-gain transmission and scalar quantization. In other words, each component u_n has a constant amplitude and the phase of u_n is quantized into L_n discrete values as follows:

$$u_n \in \left\{\exp\left(j\theta_n^1\right), \ldots, \exp\left(j\theta_n^{L_n}\right)\right\}$$

where $0 = \theta_n^1 < \cdots < \theta_n^{L_n} < 2\pi$.

Let $u_n^l = \exp(j\theta_n^l), \mathcal{U}_n = \{u_n^1, \ldots, u_n^{L_n}\}$ and $\mathcal{U} = \mathcal{U}_1 \times \cdots \times \mathcal{U}_N$. To simplify our model and improve readability, we assume in the derivation that $L_n = L$ for all $n \in \{1, \ldots, N\}$. However, all our theoretical results on optimal phase control can be easily extended to cover the general model with different L_n's. For convenience in notation, we extend the sets of $\{\theta_n^l\}$ and $\{u_n^l\}$ for l beyond L with $\theta_n^{l+L} = \theta_n^l$ and $u_n^{l+L} = u_n^l$ for all n, l. Similar to the optimal beamforming problem, the objective of the optimal phase control problem is

$$\begin{array}{ll} \text{maximize} & \|\boldsymbol{G}\boldsymbol{u}\|^2 \\ \text{subject to} & \boldsymbol{u} \in \mathcal{U}. \end{array} \tag{4}$$

III. NP-Hardness of the Optimal Phase Control and Beamforming Problems

In this section, we will prove that when the number of receive antennas grows linearly with the number of transmit antennas, the phase control problem is NP-hard. Since the phase control problem is a special case of the general beamforming problem, the result implies that the optimal beamforming problem is also NP-hard.

We show the NP-hardness by reducing the maximum cut (MAXCUT) problem [8] to the phase control problem with two quantized phases, i.e., $\mathcal{U}_n = \{+1, -1\}$ for all $n \in \{1, \ldots, N\}$. Let $\boldsymbol{g}_n$ be the nth column of $\boldsymbol{G}$ and define the function

$$\delta(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Also define $\mathrm{Re}\,\{z\}$ and $\mathrm{Img}\,\{z\}$ as the real and imaginary parts of the complex number z respectively. When z is a matrix, $\mathrm{Re}\,\{z\}$ and $\mathrm{Img}\,\{z\}$ are applied componentwise.

When each u_n is either +1 or -1, we have

$$\|\boldsymbol{G}\boldsymbol{u}\|^2 = \left\| \sum_{n=1}^{N} u_n \boldsymbol{g}_n \right\|^2 \tag{6}$$

$$= \sum_{n=1}^{N} \|u_n \boldsymbol{g}_n\|^2 + \sum_{m<n} u_m u_n \left[\boldsymbol{g}_m^H \boldsymbol{g}_n + \boldsymbol{g}_n^H \boldsymbol{g}_m\right] \tag{7}$$

$$= C - 2 \sum_{m<n} \delta(u_m + u_n) \left[\boldsymbol{g}_m^H \boldsymbol{g}_n + \boldsymbol{g}_n^H \boldsymbol{g}_m\right] \tag{8}$$

$$= C + 4 \sum_{m<n} \delta(u_m + u_n)\, \mathrm{Re}\,\left\{-\boldsymbol{g}_m^H \boldsymbol{g}_n\right\} \tag{9}$$

where $C = \sum_{n=1}^{N} \|\boldsymbol{g}_n\|^2 + \sum_{m<n} [\boldsymbol{g}_m^H \boldsymbol{g}_n + \boldsymbol{g}_n^H \boldsymbol{g}_m]$. Hence, with bipolar phases, the phase control problem (4) can be expressed as

$$\begin{aligned} &\text{maximize} && \sum_{m<n} \delta(u_m + u_n)\, \mathrm{Re}\,\left\{-\boldsymbol{g}_m^H \boldsymbol{g}_n\right\} \\ &\text{subject to} && \boldsymbol{u} \in \mathcal{U}. \end{aligned} \tag{10}$$

Any algorithm $\mathcal{A}$ that solves the phase control problem (4) solves (10).

On the other hand, consider an undirected graph with the vertex set $\mathcal{N} = \{1, \ldots, N\}$. Let $\boldsymbol{W}$ be a real symmetric matrix where the entry W_{mn} represents the weight of the edge between vertices m and n. The MAXCUT problem is to find a subset $\mathcal{S} \subset \mathcal{N}$ such that

$$\sum_{m \in \mathcal{S}, n \in \bar{\mathcal{S}}} W_{mn} \tag{11}$$

is maximized, where $\bar{S} = \mathcal{N} - \mathcal{S}$. We define

$$u_n = \begin{cases} +1, & \text{if } n \in \mathcal{S}; \\ -1, & \text{if } n \in \bar{\mathcal{S}}. \end{cases} \tag{12}$$

The MAXCUT problem can be written as

$$\begin{aligned} &\text{maximize} && \sum_{m<n} \delta(u_m + u_n) w_{mn} \\ &\text{subject to} && \boldsymbol{u} \in \mathcal{U}. \end{aligned} \tag{13}$$

Since $-\boldsymbol{W}$ is a symmetric matrix, we can diagonalize it such that $-\boldsymbol{W} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^T$, where $\boldsymbol{P}$ is a real orthonormal matrix composed of the eigenvectors of $-\boldsymbol{W}$ and $\boldsymbol{D}$ is a real diagonal matrix. Any MAXCUT problem with N vertices can be transformed to the optimal phase control problem in an $(N \times N)$-MIMO system with link gain matrix $\boldsymbol{G} = \sqrt{\boldsymbol{D}}\boldsymbol{P}^T, \mathcal{U}_n = \{-1, +1\}$. Note that some entries of $\boldsymbol{D}$ may be negative and, hence, $\boldsymbol{G}$ is a complex matrix in general. With the above reduction, any algorithm $\mathcal{A}$ that solves such a phase control problem can solve the MAXCUT problem and we have proved the following theorem:

Theorem 1: The phase control problem in an $(N \times N)$-MIMO system is NP-hard.

Corollary 2: The phase control problem in an $(N \times M)$-MIMO system is NP-hard when M grows linearly with N.

Proof: Let $K = \min\{M, N\}$. It suffices to show that an $M \times N$ phase control problem can be reduced to a $K \times K$ problem. To do this, we choose the following instance of the $M \times N$ problem: let all entries other than the upper left $K \times K$ submatrix of $\boldsymbol{G}$ be zeroes. This is equivalent to a $K \times K$ problem, since only the first K components of $\boldsymbol{u}$ needs to be determined. ■

Since the optimal phase control problem is a special case of the optimal beamforming problem, we have the following corollary:

Corollary 3: The optimal beamforming in vector quantization is NP-hard when M grows linearly with N.

IV. COMPLEXITY OF OPTIMAL PHASE CONTROL FOR SOME SPECIAL CASES

Although the phase control problem is NP-hard, efficient algorithms may exist when either M or N is small while the other grows. Firstly, we consider fixing the number of transmit antennas N. For this case, we can simply evaluate all L^N candidates and choose the optimal one. The complexity of this exhaustive search method is $O(\mathrm{MNL}^N)$, which is linear in the growing variable M. We, therefore, have the following result.

Theorem 4: When the number of receive antennas grows while fixing the number of transmit antennas, there is polynomial-time algorithm to solve the optimal phase control problem.

A more complicated problem is the other case, which keeps M constant while N grows. The rest of this section will focus on this case and shows that polynomial-time algorithm also exists. Before describing the algorithm, we first illustrate the idea geometrically.

A. Geometric Illustration

Although the number of possible transmit signals is exponential in N, when we look at the receive-signal space, we can sort out a subset of candidates of cardinality polynomial in N.

We first consider the receive signal due to one transmit antenna. There are L choices of transmit phases and the L signals differ only by a complex scalar at the receiver side. Therefore, the L receive signals are all on the same complex circle in an M-dim complex space. We then partition the circle into L regions using the bisectors of two neighbouring phases as the boundaries, as illustrated in Fig. 1. Here is an important fact: If the projection of the optimal receive signal $\boldsymbol{v}^*$ onto the circle in Fig. 1 falls into Region 1, then the corresponding transmit phase must be θ_n^1. Otherwise, there would be contradiction to the optimality assumption of $\boldsymbol{v}^*$.

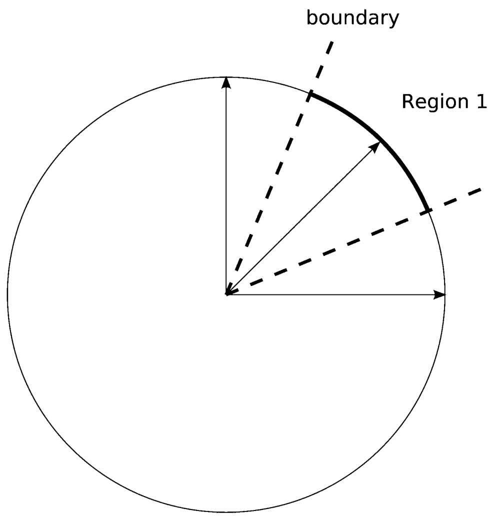

Fig. 1. Projection space showing three quantized phases of the signal from one transmit antenna. The boundaries are the bisectors of the neighbouring signals.

Now we know that by projecting $\boldsymbol{v}^*$ onto the complex circle corresponding to the received signal from transmit antenna n, we can derive u_n^*. Therefore, we can divide the unit sphere in the M-dimensional receive signal space along the phase bisectors of all transmit antennas and form the regions as shown in Fig. 2. Each region is commonly called a *Voronoi cell*. By cutting the sphere in this way, the projection of all vectors within the same cell onto a given complex circle would fall in the same region as shown in Fig. 1. In other words, if we know which cell the optimal receive vector $\boldsymbol{v}^*$ is in, we can conclude which transmit phases had been used in the transmit signal. Each Voronoi cell corresponds to one transmit signal. We call the signal a *candidate transmit signal*. The number of candidate transmit signals is equal to the number of Voronoi cells. Our next step is to show that the number of cells is polynomial in N.

Observe that each cell must have some corner points, each of which is the intersection of $2M-1$ hyperplanes (bisectors). Since each of the N antennas have L bisectors, there are totally NL hyperplanes. If the normal vectors $\boldsymbol{r}_n^l$ of any $2M-1$ hyperplanes are linearly independent, there are exactly $\binom{NL}{2M-1}$ corner points, each of which has 2^{2M-1} neighboring cells, and, thus, the number of Voronoi cells is upper bounded by $\binom{NL}{2M-1}2^{2M-1}$.

In general, however, the linear independence assumption may not hold; some corner points may be the intersections of more than $2M-1$ hyperplanes as shown in Fig. 3. In that case, a naive way to count the number of neighboring cells to such a corner point may lead to a loose upper bound that grows exponentially with N. To figure out the actual number of cells neighboring to such a corner point, we consider a neighborhood of that corner on the S^{2M-1} surface, as shown by the dotted line in Fig. 3. If the corner point is the intersection of K hyperplanes, the problem now becomes counting the number of Voronoi cells formed by K hyperplanes in a $(2M-1)$-dimensional space. This is essentially the same as our original problem, which is to count the number of Voronoi cells formed by NL hyperplanes in a $2M$-dimensional space. As illustrated in the figure, the problem space is reduced from a 3-D sphere to a 2-D circle (in dotted line). Based on this idea, a recursive algorithm is developed in the next section. The number of Voronoi cells turns out to be $O((NL)^{2M})$.

B. Polynomial-Time Algorithm

We first state some notations and lemmas for describing the algorithm. Proofs of lemmas are left to the appendices.

Since $\mathcal{U}$ is finite, optimal transmit vector exists for any given $\boldsymbol{G}$. Note that there may be multiple optimal transmit vectors. An example is where all the phases of the transmit vectors are uniformly and identically quantized, a constant shift of all transmit phases of an optimal transmit signal results in another optimal transmit vector. Although the problem may have multiple optimal solutions, our algorithm is able to find all of them.

We denote one of the optimal transmit vector(s) by $\boldsymbol{u}^* = (u_1^*, \ldots, u_N^*)$, and the corresponding receive vector by $\boldsymbol{v}^* = (v_1^*, \ldots, v_M^*) = \boldsymbol{G}\boldsymbol{u}^*$. Recall that $\boldsymbol{g}_n$ is the nth column of $\boldsymbol{G}$. Without loss of generality, we assume that $\boldsymbol{g}_n \neq \boldsymbol{0}$ for all $n \in \{1, \ldots, N\}$. Our first lemma below states that the optimal transmit phases of the transmit antennas must be the ones that are closest to the projection of $\boldsymbol{v}^*$.

Lemma 5:

$$\mathrm{Re}\{(u_n^*\boldsymbol{g}_n)^H\boldsymbol{v}^*\} > \mathrm{Re}\{(u_n\boldsymbol{g}_n)^H\boldsymbol{v}^*\}$$

for all $u_n \in \mathcal{U}_n \backslash \{u_n^*\}, n = 1, \ldots, N$.

To define the Voronoi cells, let $\mathcal{E}_n^l$ be the set

$$\mathcal{E}_n^l = \big\{\boldsymbol{v} \in \mathbb{C}^M : \mathrm{Re}\ \{(u_n^l\boldsymbol{g}_n)^H\boldsymbol{v}\} > \mathrm{Re}\ \{(u_n\boldsymbol{g}_n)^H\boldsymbol{v}\}, \forall u_n \in \mathcal{U}_n \backslash \{u_n^l\}\big\}. \quad (14)$$

Then each of the Voronoi cells can be written as $\cap_{n=1}^N \mathcal{E}_n^{l_n}$ for some $1 \leq l_1, \ldots, l_N \leq L$. It is obvious that $\mathcal{E}_n^l$ and $\mathcal{E}_n^{l'}$ are disjoint for $l \neq l'$. Hence, the Voronoi cells are disjoint with one another. Furthermore, the closure of all cells tessellates the whole space.

Let $\boldsymbol{r}_n^l$ be a row vector defined as

$$\boldsymbol{r}_n^l = \left[\left(u_n^l - u_n^{l-1}\right)\boldsymbol{g}_n\right]^H \quad (15)$$

for $n = 1, \ldots, N, l = 1, \ldots, L$. We see that $\mathrm{Re}\ \{\boldsymbol{r}_n^l\boldsymbol{v}\} = 0$ defines the bisector of the receive signal points $u_n^{l-1}\boldsymbol{g}_n$ and $u_n^l\boldsymbol{g}_n$. Indeed, $\mathcal{E}_n^l$ can be re-written in the following form:

Lemma 6:

$$\mathcal{E}_n^l = \left\{\boldsymbol{v} \in \mathbb{C}^M : \mathrm{Re}\left\{\boldsymbol{r}_n^l\boldsymbol{v}\right\} > 0 \mathrm{and} \mathrm{Re}\left\{\boldsymbol{r}_n^{l+1}\boldsymbol{v}\right\} < 0\right\}.$$

Let $\boldsymbol{R}_n$ be the $L \times M$ matrix with $\boldsymbol{r}_n^l$ as the lth row. Then each row of $\boldsymbol{R}_n$ represents a bisector of the quantized phases for the nth transmitter antenna. Concatenate $\boldsymbol{R}_n$, for $n = 1, 2, \ldots, N$, to form the block matrix:

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_1 \\ \vdots \\ \boldsymbol{R}_N \end{bmatrix}. \quad (16)$$

Therefore, $\boldsymbol{R}$ is an $NL \times M$ matrix.

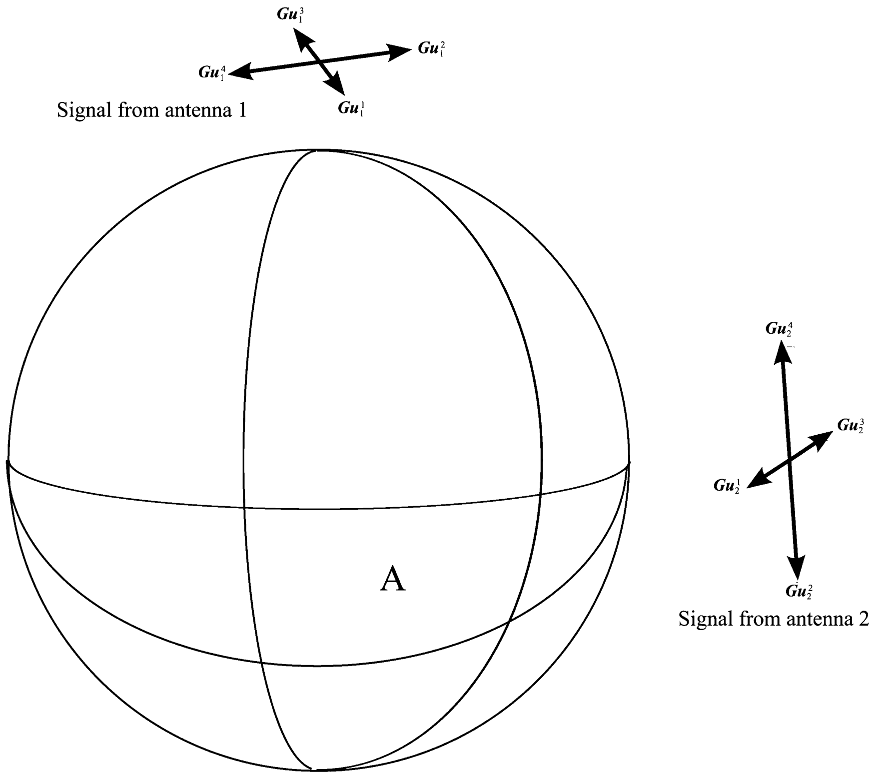

Fig. 2. M-dim sphere of the receive signal space is partitioned into many Voronoi cells. Each cell corresponds to an M-dimensional transmit phase vector.

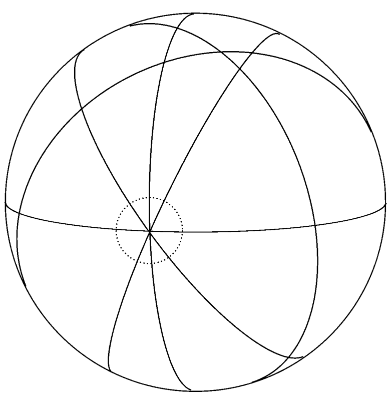

Fig. 3. When the linear independence assumption does not hold, many hyperplanes intersect at one vertex. Identifying the regions in the circle (in dotted line) is similar to the original problem but in a lower dimension space.

The next step is to construct a function that gives us the corresponding candidate transmit signal from receive signal vectors. Define the sign function for matrices as follows: If

$$\boldsymbol{Y} = \mathrm{sign}(\boldsymbol{X}) \tag{17}$$

then $\boldsymbol{Y}$ is a matrix with the same dimension as $\boldsymbol{X}$ and the (i,j)th entry of $\boldsymbol{Y}$ is defined as

$$Y_{ij} = \begin{cases} 1, & \mathrm{Re}\{X_{ij}\} > 0 \\ 0, & \mathrm{Re}\{X_{ij}\} = 0 \\ -1, & \mathrm{Re}\{X_{ij}\} < 0. \end{cases} \tag{18}$$

Let

$$\Sigma = \{\mathrm{sign}(\boldsymbol{Rv}) : \boldsymbol{v} \in \mathbb{C}^M\}. \tag{19}$$

Since $\boldsymbol{R}$ is a block matrix, $\boldsymbol{\sigma} \in \Sigma$ is also a block matrix, with $N \times 1$ blocks and each block an $L \times 1$ matrix. Label the lth entry in the nth block of $\boldsymbol{\sigma}$ as $\sigma^{n,l}$.

Define the following partial function, $\xi : \Sigma \longrightarrow \mathcal{U}$: $\xi(\boldsymbol{\sigma}) = (u_1^{l_1}, \ldots, u_N^{l_N})$ if for each $n = 1, \ldots, N$, there exists a unique l_n such that $\sigma^{nL+l_n} = 1$ and $\sigma^{nL+l'_n} = -1$ where $l'_n = l_n (\mathrm{mod}\ L) + 1$. Otherwise, $\xi(\boldsymbol{\sigma})$ is undefined. Then we immediately have the following result:

Lemma 7: If $\boldsymbol{v}^*$ is an optimal receive signal vector, then the corresponding optimal transmit vector $\boldsymbol{u}^*$ is $\xi(\boldsymbol{Rv}^*)$.

The key idea of our algorithm is to search through all Voronoi cells rather than the set of all transmit signals. Note that each Voronoi cell can be identified by a $\boldsymbol{\sigma} \in \{-1, 1\}^{NL}$. It might seem that there are 2^{NL} possible values for $\boldsymbol{\sigma}$. In that case, the number of possible candidates grows exponentially in N. In reality, however, this is not the case, since $\cap_{n=1}^{N} \mathcal{E}_n^{l_n}$ is empty for some $l_1, l_2, \ldots, l_N$ and thus does not correspond to any (nonempty) Voronoi cells. As we will see later, the number of valid Voronoi cells is only $O((NL)^{2M})$. That explains why we can construct a polynomial-time algorithm.

Definition 1: Let $\mathrm{cspan}(\Phi)$ be the set of vectors that are linear combinations of the columns of matrix Φ with real numbers as coefficients.

Definition 2: Let Φ be a $p \times q$ matrix and J be a subset of $\{1, \ldots, p\}, |J| = p' \le p$. We construct a row submatrix of Φ, written as $\Phi[J]$, by taking the wth row from Φ for all $w \in J$. Therefore, $\Phi[J]$ is a $p' \times q$ matrix.

For any complex matrix Φ, define Φ^E as the block matrix $[\mathrm{Re}\,\{\Phi\}\ \mathrm{Img}\,\{\Phi\}]$, and the *extended rank* of Φ as the rank of Φ^E.

Lemma 8: For any $p \times q$ complex matrix Φ with extended rank k, and q-dim column complex vector $\boldsymbol{x}$, there are a subset $J \subset \{1,\ldots,p\}$ and $\boldsymbol{x}' \in \mathrm{cspan}(\Phi^T)$ such that

$$\mathrm{sign}(\Phi[\bar{J}]\boldsymbol{x}') = \mathrm{sign}\,(\Phi[\bar{J}]\boldsymbol{x}) \tag{20}$$

$$\text{and } \mathrm{Re}\,\{\Phi[J]\boldsymbol{x}'\} = 0 \tag{21}$$

where $\bar{J} = \{1,\ldots,p\}\backslash J$ and $\Phi[J]$ has extended rank $(k-1)$. Moreover, the choice of $\boldsymbol{x}'$ is unique for each J, up to scalar multiples.

The above lemma is crucial to the development of our algorithm. It implies that given any $\boldsymbol{R}$, there is a subset J and vector $\boldsymbol{v}' \in \mathrm{cspan}(\boldsymbol{R}^T)$ such that

$$\mathrm{sign}(\boldsymbol{R}[\bar{J}]\boldsymbol{v}') = \boldsymbol{\sigma}[\bar{J}] \tag{22}$$

$$\mathrm{Re}\,\{\boldsymbol{R}[J]\boldsymbol{v}'\} = 0. \tag{23}$$

Note that $\boldsymbol{v}'$ is a corner point of a Voronoi cell, and can be obtained by solving (23). Since $\boldsymbol{R}[J]$ has rank at most $2M-1$, the number of distinct $\boldsymbol{v}'$'s (up to real scalar multiple) that satisfy the above conditions for some J is bounded by $\binom{NL}{2M-1}$. Once $\boldsymbol{v}'$ is determined, the jth component of $\boldsymbol{\sigma}, j \in \bar{J}$, is given by (22). If the number of components not in J is small (in the sense that it does not grow with N), then we are done because the number of Voronoi cells (or equivalently, the possible values for $\boldsymbol{\sigma}$) is polynomial in N. Otherwise, we can apply the same idea recursively.

We are now ready to describe the polynomial-time algorithm.

Description of the Algorithm: From the link gain matrix $\boldsymbol{G}$ and phase quantization set $\mathcal{U}$, construct matrix $\boldsymbol{R}$ of dimension $NL \times M$ from (15) and (16). Let K be the extended rank of $\boldsymbol{R}$, where $K \leq 2M$. The algorithm is an iterative process composed of K levels. J is a subset of $\{1,\ldots,p\}$. At each level, the cardinality of J increases and the extended rank of $\boldsymbol{R}[J]$ is reduced by 1.

1) At level 1, we find all possible $\boldsymbol{v}_J$'s that satisfy

$$\mathrm{Re}\,\{\boldsymbol{R}[J]\boldsymbol{v}_J\} = 0$$

for some $J \subset \{1,\ldots,NL\}$. We will see in the next subsection that there are at most $\binom{NL}{K-1}$ *distinct* $\boldsymbol{v}_J$'s. Label $J(i_1)$, $\boldsymbol{v}(i_1)$ and $\sigma(i_1)$. Here is the detailed procedure:

   a) For each subset S of $\{1,\ldots,NL\}$ that has cardinality $K-1$, if the extended rank of $\boldsymbol{R}[S]$ is $K-1$, then solve the following equations for $\boldsymbol{v}$:

   $$\mathrm{Re}\,\{\boldsymbol{R}[S]\boldsymbol{v}\} = 0, |\boldsymbol{v}| = 1, \boldsymbol{v} \in \mathrm{cspan}\,(\boldsymbol{R}^T).$$

   Put all solutions $\boldsymbol{v}$'s (for any S) into a list.

   b) Let N_1 be the number of distinct elements in the list of $\boldsymbol{v}$. Identify and label each of them as $\boldsymbol{v}(i_1), i_1 = 1,2,\ldots,N_1$. Put the index (i_1) into the set I_1, for $i_1 = 1,2,\ldots,N_1$.

   c) For $i_1 = 1,2,\ldots,N_1$, define $J(i_1)$ as the set that consists of the indices of all rows $\boldsymbol{r}_i$ of $\boldsymbol{R}$ that satisfy $\mathrm{Re}\,\{\boldsymbol{r}_i\boldsymbol{v}(i_1)\} = 0$, and define $\boldsymbol{\sigma}(i_1)$ as $\mathrm{sign}(\boldsymbol{R}\boldsymbol{v}(i_1))$.

2) At level $k, k \geq 2$, we have a set of $\{J(i_1,\ldots,i_{k-1}), \boldsymbol{\sigma}(i_1,\ldots,i_{k-1}), \boldsymbol{v}(i_1,\ldots,i_{k-1}) : (i_1,\ldots,i_{k-1}) \in I_{k-1}\}$, which was generated at level $k-1$. For each $(i_1,\ldots,i_{k-1}) \in I_{k-1}$, we do the following:

   a) For each subset S of $J(i_1,\ldots,i_{k-1})$ that has cardinality $K-k$, if the extended rank of $\boldsymbol{R}[S]$ is $K-k$, then solve the following equations for $\boldsymbol{v}$:

   $$\mathrm{Re}\,\{\boldsymbol{R}[S]\boldsymbol{v}\} = 0, |\boldsymbol{v}| = 1, \boldsymbol{v} \in \mathrm{cspan}(\boldsymbol{R}^T).$$

   Put all solutions $\boldsymbol{v}$'s into a list.

   b) Let N_k be the number of distinct elements in the list. Identify and label each of them as $\boldsymbol{v}(i_1,\ldots,i_k), i_k = 1,2,\ldots,N_k$. Put the index $(i_1,\ldots,i_k)$ into the set I_k, for $i_k = 1,2,\ldots,N_k$.

   c) For $i_k = 1,2,\ldots,N_k$, define $J(i_1,\ldots,i_k)$ as the subset of $J(i_1,\ldots,i_{k-1})$ that consists of the indices of all rows $\boldsymbol{r}_i$ of $\boldsymbol{R}$ that satisfy $\mathrm{Re}\,\{\boldsymbol{r}_i\boldsymbol{v}(i_1,\ldots,i_k)\} = 0$, and define the jth component of $\boldsymbol{\sigma}(i_1,\ldots,i_k)$ as follows:

$$\boldsymbol{\sigma}_j(i_1,\ldots,i_k) = \begin{cases} \mathrm{sign}(\boldsymbol{r}_j\boldsymbol{v}(i_1,\ldots,i_k)) \\ \text{for } j \in J(i_1,\ldots,i_{k-1}) \setminus J(i_1,\ldots,i_k) \\ \boldsymbol{\sigma}_j(i_1,\ldots,i_{k-1}), & \text{otherwise.} \end{cases}$$

3) After level K, we have obtained the set $\{\boldsymbol{\sigma}(i_1,\ldots,i_K) : (i_1,\ldots,i_K) \in I_k\}$. The optimal solution can be obtained by comparing all $f(\xi(\boldsymbol{\sigma}(i_1,\ldots,i_K)))$'s. Let $\boldsymbol{\sigma}^*$ be a maximizer of f. An optimal solution is then $\xi(\boldsymbol{\sigma}^*)$. If it is needed to determine all the optimal solutions, one can first find all the maximizers of f and then apply the function ξ to each of them.

By Lemma 8, we can check that if the ith coordinate of $\xi(\boldsymbol{\sigma}(i_1,\ldots,i_{k-1}))$ and $\boldsymbol{u}^*$ are equal for all $i \notin J(i_1,\ldots,i_{k-1})$, then there is i_k such that the jth coordinate of $\xi(\boldsymbol{\sigma}(i_1,\ldots,i_{k-1},i_k))$and $\boldsymbol{u}^*$ are equal for all $j \notin J(i_1,\ldots,i_{k-1},i_k)$. Since the condition is true when $k = 0$, by mathematical induction, there is $\boldsymbol{\sigma}(i_1,\ldots,i_K)$ after level K such that $\boldsymbol{u}^* = \xi(\boldsymbol{\sigma}^*)$. Therefore, comparing all $f(\xi(\boldsymbol{\sigma}(i_1,\ldots,i_K)))$ gives us all the optimal transmit vectors.

The pseudocode of the above algorithm can be found in Appendix D.

C. Complexity Analysis of Algorithm

Now we show that the complexity of the algorithm is polynomial in N. We need one more lemma from basic linear algebra to bound the number of steps.

Lemma 9: Let Φ be a $p \times q$ complex matrix with extended rank k. There exists a subset $B \subset \{1,\ldots,p\}$ such that B has exactly k elements and $\Phi[B]$ has extended rank k.

- The initialization stage involves the construction of $\boldsymbol{R}$ and the evaluation of its extended rank, which can be done in $O(NLM^2)$ computations.
- To analyze the complexity of level 1, we decompose it into different steps:
  — The first step is to find out all distinct $\boldsymbol{v}(i_1)$. By Lemma 9, there are at most $N_v = \binom{NL}{2M-1}$ choices of subset B. For each of them, evaluation of $\boldsymbol{v}(i_1)$ requires solving a system of $2M-1$ equations, which has complexity of $O(M^3)$. To label distinct $\boldsymbol{v}$'s as $\boldsymbol{v}(i_1)$, we may do a sorting on all $\boldsymbol{v}$'s. This requires $O(N_v \log N_v)$ computations. Hence, the overall complexity in this step is $O(N_v \log N_v)$, assuming that M is a constant.
  — To find $J(i_1)$ for each $\boldsymbol{v}(i_1)$, we need to figure out which row $\boldsymbol{r}_i$ of $\boldsymbol{R}$ satisfies the equation $\mathrm{Re}\,\{\boldsymbol{r}_i \boldsymbol{v}(i_1)\} = 0$. This requires $O(NL)$ computations. Since $\{\boldsymbol{v}(i_1) : i_1 = 1, 2, \ldots, N_1\}$ can have at most $2N_v$ members, this step requires $O(N_v NL)$ computations.
  — To determine $\boldsymbol{\sigma}(i_1)$, we need $O(NL)$ computations for each i_1. Totally, $O(N_v NL)$ computations is required.

  Since $N_v = \binom{NL}{2M-1} \le (NL)^{2M-1}$ and M is assumed to be constant, the overall complexity of level 1 is $O((NL)^{2M})$.
- Basically, level 2 has the same procedure as level 1. Therefore, they have the same complexity, except that the cardinality of $\{\boldsymbol{v}(i_1, i_2)\}$ is higher than $\{\boldsymbol{v}(i_1)\}$. To find a suitable upper bound, we focus on a particular $\boldsymbol{v}(i_1^*)$. Let $|J(i_1^*)| = P \ge K-1$. Note that any subset $B \subset J(i_1^*)$ with extended rank of B equal $K-1$ would have yielded the same $\boldsymbol{v}(i_1^*)$, up to a scalar, by solving the equation
$$\mathrm{Re}\,\{\boldsymbol{R}[B]\boldsymbol{v}\} = 0 \tag{24}$$
  for $\boldsymbol{v}$. At level 1, we have evaluated at most $\binom{NL}{K-1}$ subsets of $\{1, \ldots, NL\}$ and $\binom{P}{K-1}$ of them have produced $\boldsymbol{v}(i_1^*)$. To evaluate vector $\boldsymbol{v}(i_1^*, i_2)$ at level 2, there are at most $\binom{P}{K-2}$ subsets of $J(i_1^*)$. Since this relation holds for any other $\boldsymbol{v}(i_1)$, the ratio between the number of distinct $J(i_1, i_2)$ to the number of $J(i_1)$ must be bounded by
$$\max_{2M-1 \le P' \le N_v} \binom{P'}{K-2} \Big/ \binom{P'}{K-1} \tag{25}$$
$$\le \frac{K-1}{P' - (K-1) + 1} \tag{26}$$
$$< K \tag{27}$$
$$\le 2M. \tag{28}$$
  Therefore, the number of distinct $\boldsymbol{v}(i_1, i_2)$ is less than $4MN_v$, and the overall complexity of level 2 can be proved to be $O(M(NL)^{2M})$.
- Similarly the complexity of level k is $O(M^{k-1}(NL)^{2M})$ for $k = 1, \ldots, K$. Since the number of levels, K, is less than $2M$, the overall complexity of the algorithm is $O((1 + \cdots + M^{2M-1})(NL)^{2M}) \le O((MNL)^{2M})$. Keeping M constant, the complexity is $O((NL)^{2M})$, which is polynomial in N and L.

The following result concludes this section:

Theorem 10: When the number of transmit antennas grows while fixing the number of receive antennas, there is polynomial-time algorithm to solve the optimal phase control problem.

V. Suboptimal Algorithms

In Section III, it is shown that the optimal phase control problem is NP-hard in general. However, if we relax our solution to a sub-optimal one, efficient algorithms can be developed. In this section, we present two such sub-optimal algorithms.

A. Co-Phase Algorithm

The Co-phase Algorithm was proposed in [10], [12]. The idea is to determine the phase of each antenna independently against the phase of the first antenna. Precisely, for $n = 2, \ldots, N$, we find $\boldsymbol{u}_n$, which maximizes the expression
$$\begin{bmatrix} \bar{u}_1 & \bar{u}_n \end{bmatrix} \begin{bmatrix} a_{11} & a_{1n} \\ a_{n1} & a_{nn} \end{bmatrix} \begin{bmatrix} u_1 \\ u_n \end{bmatrix} \tag{29}$$
where a_{ij} is the (i,j)th entry of the matrix $\boldsymbol{G}^H\boldsymbol{G}$. A straightforward implementation of this algorithm has complexity proportional to $(N-1)L$.

B. Local Search Algorithm

Local Search is an iterative algorithm. At each iteration, it generates an updated transmit vector $\boldsymbol{u}(\tau+1)$ from the neighborhood of $\boldsymbol{u}(\tau)$. We define the neighborhood of a vector $\boldsymbol{u}$, denoted by $\mathcal{N}(\boldsymbol{u})$, as the set of vectors whose Hamming distance from $\boldsymbol{u}$ is smaller than or equal to 1. The weight vector is iterated as follows:
$$\boldsymbol{u}(\tau+1) = \arg \max_{\boldsymbol{u} \in \mathcal{N}(\boldsymbol{u}(\tau))} f(\boldsymbol{u}). \tag{30}$$
The algorithm outputs the vector $\boldsymbol{u}(\tau)$ when no more improvement can be made, i.e., $f(\boldsymbol{u}(\tau+1)) = f(\boldsymbol{u}(\tau))$.

The algorithm always stops, since $f(\boldsymbol{u}(0)), f(\boldsymbol{u}(1)), f(\boldsymbol{u}(2)), \ldots$ are strictly increasing and there are only a finite number of points. However, there is no guarantee on how close is the SNR of the output to the optimal one. According to [2], heuristics based on Local Search are well adapted for problems with flat landscape. To obtain some insight on the suitability of Local Search, we perform a ruggedness analysis of the problem landscape. Our result shows that Local Search is well adapted for the phase control problem with large L. Details can be found in Appendix E.

The complexity of Local Search depends on the cardinality of $\mathcal{N}(\boldsymbol{u})$ and the number of iterations. Recall that L is the number of quantized phases of the signal from each transmit antenna. Let b be the number of bits to represent each phase. Then we have $b = \log_2 L$. It is easy to see that the cardinality of $\mathcal{N}(\boldsymbol{u})$ is $N(L-1)+1$. Regarding the average number of iterations, our experimental results show that it increases both linearly with N for fixed b, and with b for fixed N (see Figs. 4 and 5). However,

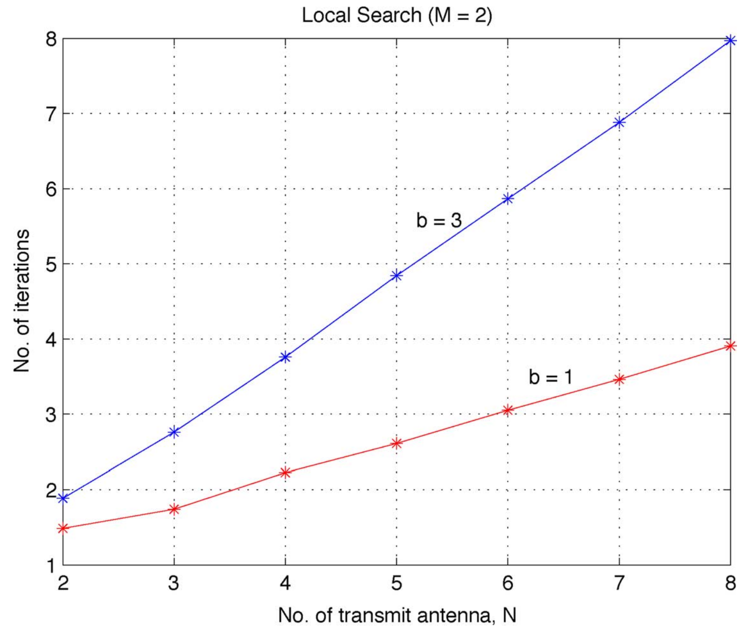

Fig. 4. Average number of iterations with different values of N.

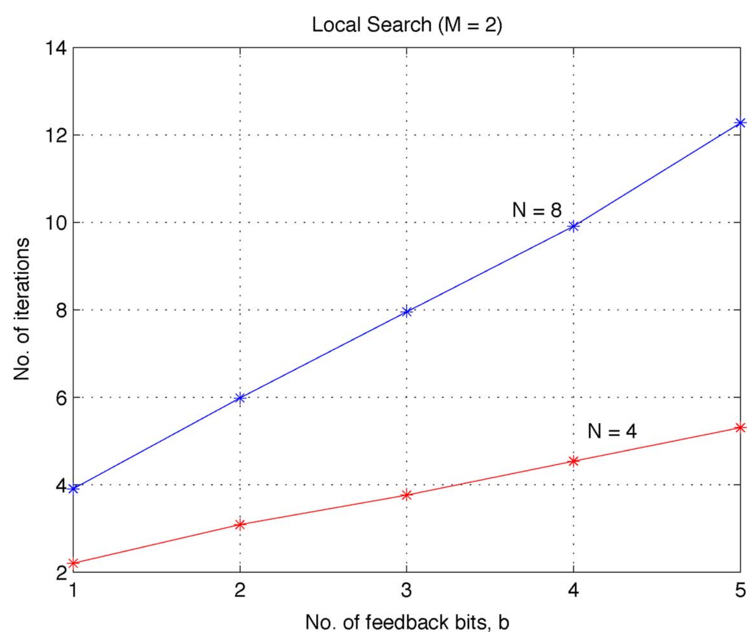

Fig. 5. Average number of iterations with different values of b.

it is independent of the number of receive antenna, M (figure omitted). Hence, the overall complexity is $O(N^2 L \log L)$.

VI. Performance Comparisons

We summarize the computational complexities of different algorithms as follows, assuming that M is fixed:

Algorithm	Complexity
Exhaustive Search	$O(NL^N)$
Optimal Algorithm	$O((NL)^{2M})$
Local Search	$O(N^2 L \log L)$
Co$-$phase Algorithm	$O(NL)$

It can be seen that Co-phase Algorithm is the most efficient one. Local Search also has a very low complexity. The complexity of the optimal algorithm, though polynomial in N, is quite high, especially when M is not small. For practical applications, Local Search and Co-phase Algorithm may be more suitable candidates.

Next we compare the algorithms in terms of the average SNR. In our simulations, we assume that the channel between each transmitter-receiver pair is subject to independent quasi-static Rayleigh fading. The link gain is modeled by a complex Gaussian random variable with independent I and Q components, each of them having mean zero and standard deviation one. The transmit power is assumed to be one. The receiver

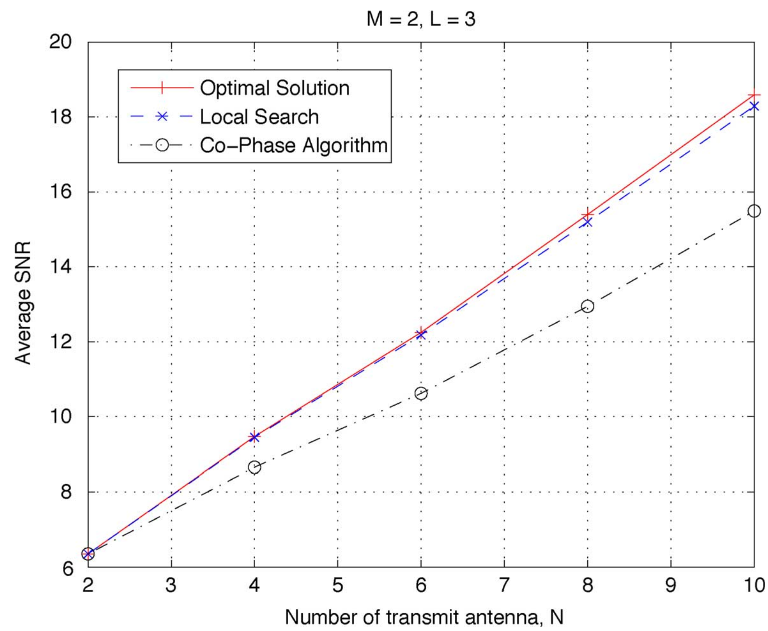

Fig. 6. Average SNR against the number of transmit antennas, N.

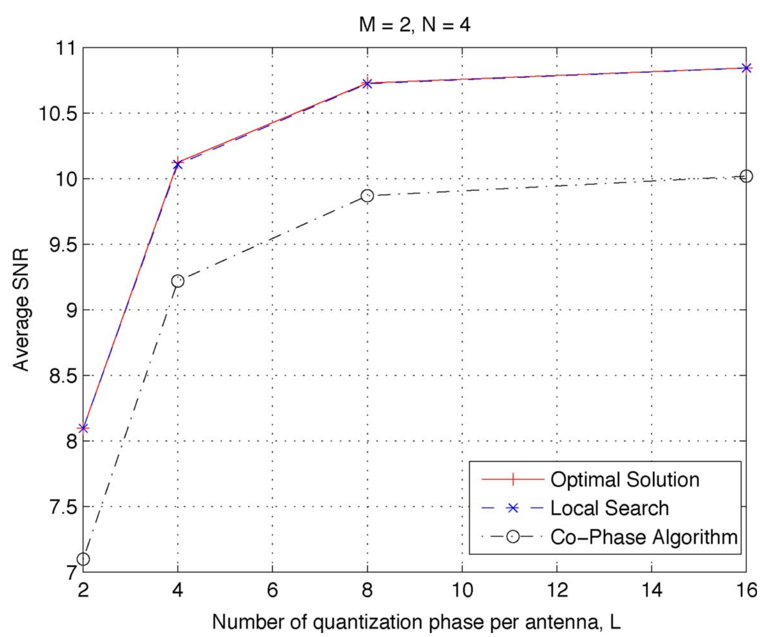

Fig. 7. Average SNR against the number of quantized phases per antenna, L.

noise is assumed to be addictive white Gaussian and its power is equal to one. We consider an $(N \times M)$-MIMO system with M equal to two. We first fix the number of quantized phases per antenna, L, as three. The average SNR of the algorithms are plotted in Fig. 6. Each point is obtained by averaging over 300 simulation runs. From the figure, we can see that Local Search has significant improvement over Co-phase Algorithm. When the number of antennas is small, Local Search performs very close to the optimal.

Fig. 7 shows a similar test where the varying parameter is the number of quantized phases, L, and the numbers of transmit antennas is fixed to be four. We consider four cases: $L = 2, 4, 8$ and 16, which correspond to the cases where the number of feedback bits is equal to 1, 2, 3, and 4 respectively. Similar to the previous test, Local Search Algorithm outperforms Co-phase Algorithm. In this case, it essentially yields optimal performance. Note that the SNR does not improve much when the number of quantization phases is exceedingly large.

VII. Conclusion

We investigated the transmit phase control problem in MIMO systems. We showed that for a general $(N \times M)$-MIMO system, the problem is NP-hard. However, when either M or N is kept constant, the problem can be solved in polynomial time. This is proven by explicitly constructing a polynomial-time algorithm.

Although a polynomial-time algorithm has been constructed, its complexity may still be too high for practical purpose. In view of this, we propose a sub-optimal algorithm based on local search. Ruggedness analysis shows that it is a suitable tool for tackling the phase control problem. Moreover, simulation results show that it outperforms an existing algorithm in terms of SNR, at the expense of slight increase in complexity.

Appendix A
Proof of Lemma 5

Let $\boldsymbol{u}' = (u_1^*, \ldots, u_{n-1}^*, u_n', u_{n+1}^*, \ldots, u_N^*)$, where $u_n' \neq u_n^*$. Assume that $\mathrm{Re}\,\{(u_n^*\boldsymbol{g}_n)^H\boldsymbol{v}^*\} \leq \mathrm{Re}\,\{(u_n'\boldsymbol{g}_n)^H\boldsymbol{v}^*\}$. Then we have

$$\begin{aligned} \boldsymbol{v}' &= \boldsymbol{G}\boldsymbol{u}' && (31)\\ &= \boldsymbol{G}\boldsymbol{u}^* - u_n^*\boldsymbol{g}_n + u_n'\boldsymbol{g}_n && (32)\\ &= \boldsymbol{v}^* - u_n^*\boldsymbol{g}_n + u_n'\boldsymbol{g}_n && (33)\end{aligned}$$

and

$$\begin{aligned} \|\boldsymbol{v}'\|^2 &= \|\boldsymbol{v}^*\|^2 + \|(u_n' - u_n^*)\boldsymbol{g}_n\|^2 \\ &\quad + 2[\mathrm{Re}\,\{(u_n'\boldsymbol{g}_n)^H\boldsymbol{v}^*\} \\ &\quad - \mathrm{Re}\,\{(u_n^*\boldsymbol{g}_n)^H\boldsymbol{v}^*\}] && (34)\\ &> \|\boldsymbol{v}^*\|^2 && (35)\end{aligned}$$

which contradicts to the assumption that $\boldsymbol{v}^*$ is optimal.

Appendix B
Proof of Lemma 6

Lemma 6 can be re-phrased as follows: $\boldsymbol{v} \in \mathcal{E}_n^l$ if and only if $\mathrm{sign}(\boldsymbol{r}_n^l\boldsymbol{v}) = 1$ and $\mathrm{sign}\,(\boldsymbol{r}_n^{l+1}\boldsymbol{v}) = -1$, where the sign function is defined in (17).

By definition, $\boldsymbol{v} \in \mathcal{E}_n^l$ implies that $\mathrm{Re}\,\{[(u_n^l - u_n^{l'})\boldsymbol{g}_n]^H\boldsymbol{v}\} > 0$ for all $l' \neq l$. Then we have $\mathrm{sign}(\boldsymbol{r}_n^l\boldsymbol{v}) = 1$ and $\mathrm{sign}(\boldsymbol{r}_n^{l+1}\boldsymbol{v}) = -1$, which proves the forward part.

To prove the reverse part, define the modulo-π function $\mathrm{mod}\,(x)$ as

$$\mathrm{mod}\,(x) = 2k\pi + x$$

where k is an integer and $-\pi \leq \mathrm{mod}\,(x) < \pi$. Given any $\boldsymbol{v} \in \mathbb{C}^M$, it can be decomposed into two components, $\boldsymbol{v} = e^{j\phi}\boldsymbol{g}_n + \boldsymbol{v}_g^\perp$, for some $-\pi \leq \phi < \pi$ and $\mathrm{Re}\,\{\boldsymbol{g}_n^H\boldsymbol{v}_g^\perp\} = 0$. It can be shown that

$$\mathrm{Re}\,\{[(e^{j\theta} - e^{j\theta'})\boldsymbol{g}_n]^H\boldsymbol{v}\} > 0 \text{ if and only if } |\,\mathrm{mod}\,(\phi - \theta)| < |\,\mathrm{mod}\,(\phi - \theta')|. \quad (36)$$

Therefore

$$\mathrm{sign}\,(\boldsymbol{r}_n^l\boldsymbol{v}) = 1 \quad (37)$$

implies

$$|\,\mathrm{mod}\,(\phi - \theta_n^l)| < |\,\mathrm{mod}\,(\phi - \theta_n^{l-1})| \quad (38)$$

and

$$\mathrm{sign}\,(\boldsymbol{r}_n^l\boldsymbol{v}) = -1 \quad (39)$$

implies

$$|\,\mathrm{mod}\,(\phi - \theta_n^l)| < |\,\mathrm{mod}\,\left(\phi - \theta_n^{l+1}\right)|. \quad (40)$$

Since $0 \leq \theta_n^1 \leq \cdots \leq \theta_n^L < 2\pi$, (38) and (40) imply that

$$|\,\mathrm{mod}\,(\phi - \theta_n^l)| < |\,\mathrm{mod}\,(\phi - \theta_n^{l'})| \quad (41)$$

for all $l' \neq l$, and therefore $\boldsymbol{v} \in \mathcal{E}_n^l$.

Appendix C
Proof of Lemma 8

Without loss of generality, we assume that $\boldsymbol{x}$ is in $\mathrm{cspan}(\Phi^T)$, for otherwise, we can replace $\boldsymbol{x}$ by its projection on $\mathrm{cspan}(\Phi^T)$. Denote Φ_i as the ith row of Φ. Subset J and $\boldsymbol{x}'$ that satisfy the lemma can be obtained by the following iterative process. We will show that the process will terminate after N steps.

Initially, let $\boldsymbol{x}(0) = \boldsymbol{x}$ and $J(0) = \{i : \Phi_i\boldsymbol{x}(0) = 0\}$. At each iteration n, if the extended rank of $\Phi[J(n-1)] < k-1$, we can construct a column vector $\psi_n \in \mathrm{cspan}(\Phi[\bar{J}(n-1)]^T)$ such that

$$\begin{aligned} \boldsymbol{x}^T\psi_n &= 0 && (42)\\ \Phi[J(n-1)]\psi_n &= \mathbf{0} && (43)\\ \text{and} \quad \Phi_j\psi_n &= 1 && (44)\end{aligned}$$

for some $j \in \bar{J}(n-1)$. For each $i \in \bar{J}(n-1)$, if $\Phi_i\psi_n \neq 0$, define

$$\alpha^i = \frac{\Phi_i\boldsymbol{x}(n-1)}{\Phi_i\psi_n}$$

and let

$$\alpha_n = \min_{i:\alpha^i>0}\{\alpha^i\}.$$

Note that α_n is well defined since there is at least one $\alpha^j > 0$, where j is given in (44). Let

$$\boldsymbol{x}(n) = \boldsymbol{x}(n-1) - \alpha_n\psi_n. \quad (45)$$

and $J(n) = \{i : \Phi_i\boldsymbol{x}(n) = 0\}$.

We would like to prove that for all $m \geq 0$

$$\mathrm{sign}\,(\Phi_i\boldsymbol{x}(m)) = \begin{cases} 0, & i \in J(m)\\ \mathrm{sign}\,(\Phi_i\boldsymbol{x}), & \text{otherwise.}\end{cases} \quad (46)$$

The above assertion is obviously true for $m = 0$. Assume that it is true for $m = n-1$. According to (45), we have

$$\mathrm{sign}(\Phi_i\boldsymbol{x}(n)) = \begin{cases} 0, & i \in J(n-1);\\ 0 \text{ or } \mathrm{sign}(\Phi_i\boldsymbol{x}), & \text{otherwise.}\end{cases} \quad (47)$$

Hence, the assertion is true for all m. Furthermore, since α_n is well defined, $J(n-1)$ is a proper subset of $J(n)$. Since the process can be applied as long as $\Phi[J(n-1)]$ has extended rank less than $k-1$, it will eventually terminate, say, after iteration N_0. And $\Phi[J(N_0)]$ has extended rank greater than or equal to $k-1$.

It remains to prove that the extended rank of $\Phi[J(N_0)]$ is exactly equal to $k-1$. From (42), we have

$$\boldsymbol{x}^T \psi_n = 0, \text{ for all } n \leq N_0.$$

Together with (45)

$$\boldsymbol{x}^T \boldsymbol{x}(N_0) = \boldsymbol{x}^T \boldsymbol{x}(N_0 - 1) = \cdots = \boldsymbol{x}^T \boldsymbol{x} \neq 0$$

which implies

$$\boldsymbol{x}(N_0) \neq 0.$$

Since $\boldsymbol{x}(N_0) \in \operatorname{cspan}(\Phi^T)$ and $\Phi[J(N_0)]\boldsymbol{x}(N_0) = 0, \Phi[J(N_0)]$ must have extended rank less than k. Therefore, $\Phi[J(N_0)]$ has extended rank exactly equal to $k-1$.

At this point, we have J and $\boldsymbol{x}'$ such that

$$\begin{aligned} &\operatorname{sign}(\Phi[\bar{J}]\boldsymbol{x}') = \operatorname{sign}(\Phi[\bar{J}]\boldsymbol{x}) \\ &\text{and } \Phi[J]\boldsymbol{x}' = 0. \end{aligned}$$

Since $\boldsymbol{x}'$ is chosen from a k-dim space and $\Phi[J]$ has extended rank $k-1$, the choice of $\boldsymbol{x}'$ is unique up to scalar multiples.

APPENDIX D
PSEUDOCODE OF THE OPTIMAL PHASE CONTROL ALGORITHM

Since the pseudocode is used to describe the logical flow, we assume some complex but straight forward steps can be done in one step. Denote $\vec{i}_k$ as the vector $(i_1, \ldots, i_k)$, i.e., $\boldsymbol{v}(i_1, \ldots, i_k)$ can be written as $\boldsymbol{v}(\vec{\boldsymbol{i}}_{\boldsymbol{k}})$ or $\boldsymbol{v}(\vec{i}_{k-1}, i_k)$.

Algorithm 1 Optimal Phase Control($\boldsymbol{G}$)

Construct matrix $\boldsymbol{R}$ from $\boldsymbol{G}$ by (15) and (16).
Let $K =$ extended rank of $\boldsymbol{R}$.
This is a K-level recursive algorithm. Initialize a set of arrays, $\{I_1, \ldots, I_K\}$, one for each level.
Let $I_0 = \{1\}, J(\vec{i}_0) = \{1, \ldots, NL\}$ and $\boldsymbol{\sigma}_{\vec{i}_0} = 0$.
for level $k = 1$ to K **do**
  for each index $\vec{i}_{k-1}$ in I_{k-1} **do**
Input $J(\vec{i}_{k-1}), \boldsymbol{\sigma}_{\vec{i}_{k-1}}$ and k into Algorithm 2: **ReductionStep**$(J(\vec{i}_{k-1}), \boldsymbol{\sigma}_{\vec{i}_{k-1}}, k)$, and label the outputs J_i as $J(\vec{i}_{k-1}, i), \boldsymbol{\sigma}_i$ as $\boldsymbol{\sigma}_{\vec{i}_{k-1}, i}$, for $i = 1, \ldots, n$, where n is given at the output of Algorithm 2.
    Put the index $(\vec{i}_{k-1}, i)$, for $i = 1, \ldots, n$, into array I_k.
  end for
end for
Find the maximum of $f(\xi(\boldsymbol{\sigma}_{\vec{i}_K}))$ over all $\vec{i}_K$ in I_K. The corresponding $\xi(\boldsymbol{\sigma}_{\vec{i}_K})$ is the optimal transmit phase $\boldsymbol{u}^*$.

Algorithm 2 ReductionStep $(\boldsymbol{J}, \boldsymbol{\sigma}_{\text{input}}, \boldsymbol{k})$

Initialize an empty array V.
for each subset $J' \subseteq J$ which has $K-k$ elements **do**
  if the extended rank of $\boldsymbol{R}[J'] = K - k$ **then**
    find a $(K-k)$-dimensional vector $\boldsymbol{x}$ which satisfies the equation

$$\{\boldsymbol{R}[J']\boldsymbol{R}^T\boldsymbol{x}\} = 0.$$

    Let $\boldsymbol{v} = \dfrac{\boldsymbol{R}^T\boldsymbol{x}}{|\boldsymbol{R}^T\boldsymbol{x}|}$ and put $\boldsymbol{v}$ and $-\boldsymbol{v}$ into array V.
  end if
end for
Identify all distinct v in V. Label them as $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$.
For $i = 1, 2, \ldots, n$, construct J_i and $\boldsymbol{\sigma}_i$.
return $n, J_i, \boldsymbol{\sigma}_i$, for $i = 1, \ldots, n$.

APPENDIX E
RUGGEDNESS ANALYSIS OF THE PHASE CONTROL PROBLEM

We first introduce the concept of *ruggedness*, which is a measurement on how flat the problem space is. Let h be the objective function and $\mathcal{N}$ the neighborhood structure. The distance between any two distinct solutions x and y is denoted by $d(x,y)$, which is defined as the smallest integer $k \geq 1$ such that there exists a sequence of solution $x_0, \ldots, x_k$ with $x_0 = x, x_{i+1} \in \mathcal{N}(x_i)$ for $0 \leq i \leq k-1$, and $x_k = y$.

To measure the hardness of a combinatorial optimization problem, the notion of landscape autocorrelation function is introduced in [24]

$$\rho(d) = 1 - \frac{\langle (h(x) - h(y))^2 \rangle_{d(x,y)=d}}{\langle (h(x) - h(y))^2 \rangle} \tag{48}$$

where $\langle g(x,y) \rangle$ denotes the average value of $g(x,y)$ over all solution pairs (x,y), and $\langle g(x,y) \rangle_{d(x,y)=d}$ the average value of $g(x,y)$ over all solution pairs (x,y) with distance d. The above expression can be rewritten as

$$\rho(d) = 1 - \frac{\langle (h(x) - h(y))^2 \rangle_{d(x,y)=d}}{2(\langle h^2 \rangle - \langle h \rangle^2)} \tag{49}$$

where $\langle g \rangle$ denotes the average value of g over the solution space.

In particular, we are interested in $\rho(1)$, which is important for Local Search and its variants. For this purpose, the autocorrelation coefficient λ is defined in [2]

$$\lambda = \frac{1}{1 - \rho(1)} \tag{50}$$

which measures the ruggedness of a landscape. The larger it is, the flatter is the landscape, and the more suitable is the landscape for Local Search.

Recall that in our context $h = \|\boldsymbol{G}\boldsymbol{u}\|^2$, which can be expressed as $\boldsymbol{u}^H\boldsymbol{A}\boldsymbol{u}$, where $\boldsymbol{A} = \boldsymbol{G}^H\boldsymbol{G}$ is an $N \times N$ Hermitian matrix. To gain some insight on the ruggedness of the landscape of our problem, we make the assumption that the signal phases of all transmit antennas are quantized in the same way, each of them being quantized uniformly into L discrete values, that is

$$\mathcal{U}_k = \{1, \omega, \omega^2, \ldots, \omega^{L-1}\} \tag{51}$$

for $k = 1, 2, \ldots, N$, and ω is the Lth root of unity. This simplifies the analysis significantly and the assumption made is a typical setting in practical implementation and is reasonable for i.i.d. channels. With this assumption, we have

$$\langle h \rangle = \left\langle \sum_{i,j} a_{ij} u_j \bar{u}_i \rangle = \sum_{i,j} a_{ij} \right\rangle u_j \bar{u}_i \rangle \tag{52}$$

where $\bar{z}$ denotes the complex conjugate of the complex number z. If $i = j$, then $\langle u_j \bar{u}_i \rangle$ equals 1. Otherwise, $\langle u_j \bar{u}_i \rangle$ equals $\langle u_j \rangle \langle \bar{u}_i \rangle$. Since

$$\langle u_j \rangle = (1 + \omega + \omega^2 + \cdots + \omega^{L-1})/L = 0 \tag{53}$$

we have $\langle u_j \bar{u}_i \rangle = 0$ when $i \neq j$. As a result

$$\langle h \rangle = \sum_i a_{ii}. \tag{54}$$

Similarly, we have

$$\langle h^2 \rangle = \left\langle \sum_{i,j,k,l} a_{ij} a_{kl} u_j \bar{u}_i u_l \bar{u}_k \right\rangle \tag{55}$$

$$= \left(\sum_i a_{ii} \right)^2 + \sum_{i,j(i \neq j)} a_{ij} a_{ji}. \tag{56}$$

Next, we compute $\langle (h(\boldsymbol{u}) - h(\boldsymbol{u}'))^2 \rangle_{d(\boldsymbol{u},\boldsymbol{u}')=1}$. Consider a solution pair $(\boldsymbol{u}, \boldsymbol{u}')$, where $u_i = u'_i$ for all i except when $i = k$

$$h(\boldsymbol{u}) - h(\boldsymbol{u}') = \sum_{i,j} a_{ij}(u_j \bar{u}_i - u'_j \bar{u'}_i) \tag{57}$$

$$= \sum_{i \neq k} a_{ik}(u_k \bar{u}_i - u'_k \bar{u'}_i) + \sum_{j \neq k} a_{kj}(u_j \bar{u}_k - u'_j \bar{u'}_k) \tag{58}$$

$$= \sum_{i \neq k} a_{ik} \epsilon_k \bar{u}_i + \sum_{j \neq k} a_{kj} \bar{\epsilon}_k u_j \tag{59}$$

where $\epsilon_k = u_k - u'_k$. Then

$$(h(\boldsymbol{u}) - h(\boldsymbol{u}'))^2 \tag{60}$$

$$= \left[\sum_{i \neq k} (a_{ik} \epsilon_k \bar{u}_i + a_{ki} \bar{\epsilon}_k u_i) \right]^2 \tag{61}$$

$$= \sum_{i \neq k, j \neq k} \big(a_{ij} a_{jk} \epsilon_k^2 \bar{u}_i \bar{u}_j + a_{ik} a_{kj} \epsilon_k \bar{\epsilon}_k \bar{u}_i u_j + a_{ki} a_{jk} \bar{\epsilon}_k \epsilon_k u_i \bar{u}_j + a_{ki} a_{kj} \bar{\epsilon}_k^2 u_i u_j \big). \tag{62}$$

Notice that $w \in \mathcal{U}$ implies $\bar{w} \in \mathcal{U}$, and given any $w_1, w_2 \in \mathcal{U}$, we have $\langle w_1 w_2 \rangle = 0$. Hence

$$\langle (h(\boldsymbol{u}) - h(\boldsymbol{u}'))^2 \rangle_{d(\boldsymbol{u},\boldsymbol{u}')=1, u_k \neq u'_k}$$

$$= 2 \left\langle \sum_{i \neq k} a_{ik} a_{ki} \epsilon_k \bar{\epsilon}_k \bar{u}_i u_i \right\rangle_{d(\boldsymbol{u},\boldsymbol{u}')=1, u_k \neq u'_k} \tag{63}$$

$$= 2 \left\langle \sum_{i \neq k} a_{ik} a_{ki} (2 - u_k \bar{u'}_k - u'_k \bar{u}_k) \right\rangle_{d(\boldsymbol{u},\boldsymbol{u}')=1, u_k \neq u'_k} \tag{64}$$

Consider $w_1, w_2 \in \mathcal{U}$.

$$\langle w_1 \bar{w_2} \rangle_{w_1 \neq w_2}$$

$$= \frac{1}{L(L-1)} [1(\omega + \omega^2 + \ldots + \omega^{L-1}) + \bar{\omega}(1 + \omega^2 + \omega^3 + \ldots + \omega^{L-1}) + \ldots + \bar{\omega}^{L-1}(1 + \omega + \omega^2 + \ldots + \omega^{L-2})] \tag{65}$$

$$= -\frac{1}{L-1}. \tag{66}$$

Hence, we have

$$\langle (h(\boldsymbol{u}) - h(\boldsymbol{u}'))^2 \rangle_{d(\boldsymbol{u},\boldsymbol{u}')=1}$$

$$= \frac{2}{N} \sum_{i,j(i \neq j)} a_{ij} a_{ji} \left(2 + \frac{2}{L-1} \right) \tag{67}$$

$$= \frac{4L}{N(L-1)} \sum_{i,j(i \neq j)} a_{ij} a_{ji}. \tag{68}$$

In consequence

$$\rho(1) = 1 - \frac{2L}{N(L-1)} \tag{69}$$

$$\lambda = \frac{N(L-1)}{2L}. \tag{70}$$

The autocorrelation coefficient, λ, depends not only on the number of transmit antennas, N, but also the number of quantization values, L. It can be seen that λ increases from $N/4$ to $N/2$ when L grows from 2 to infinity.

As mentioned earlier, heuristics based on Local Search are well adapted for problems with a high value of λ. Experimental studies indicate that $N/2$ is a high value for λ whereas $N/8$ is low. Conclusion is difficult to make for intermediate values like $N/4$. Based on this, we can conclude that Local Search is well adapted for the phase control problem with large L.

References

[1] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *J. Sel. Commun.*, vol. 16, no. 8, pp. 1451–1458, Oct. 1998.

[2] E. Angel and V. Zissimopoulos, "On the classification of NP-complete problems in terms of their correlation coefficient," *Discrete Appl. Math.*, vol. 90, pp. 261–277, 2000.

[3] A. Annamalai, C. Tellambura, and V. K. Bhargava, "Equal-gain diversity receiver performance in wireless channels," *IEEE Trans. Commun.*, vol. 48, no. 10, pp. 1732–1745, Oct. 2000.

[4] J. H. Conway, R. H. Hardin, and N. J. A. Sloane, "Packing lines, planes, etc.: Packings in Grassmannian spaces," *Exp. Math.*, vol. 5, no. 2, pp. 139–159, 1996.
[5] R. T. Derryberry, S. D. Gray, D. M. Ionescu, G. Mandyam, and B. Raghothaman, "Transmit diversity in 3G CDMA systems," *IEEE Commun. Mag.*, vol. 40, no. 4, pp. 68–75, Apr. 2002.
[6] P. A. Dighe, R. K. Mallik, and S. S. Jamuar, "Analysis of transmit-receive diversity in Rayleigh fading," *IEEE Trans. Commun.*, vol. 51, no. 4, pp. 694–703, Apr. 2003.
[7] T. K. Y. Lo, "Maximum ratio transmission," *IEEE Trans. Commun.*, vol. 47, no. 10, pp. 1458–1461, Oct. 1999.
[8] M. R. Garey and D. S. Johnson*, Computers and Intractability. A Guide to the Theory of NP-Completeness*. San Francisco, CA: W. H. Freeman, 1979.
[9] D. Gerlach and A. Paulraj, "Adaptive transmitting antenna arrays with feedback," *IEEE Signal Process. Lett.*, vol. 1, no. 10, pp. 150–52, Oct. 1994.
[10] J. Hämäläinen and R. Wichman, "Performance analysis of closed-loop transmit diversity in the presence of feedback errors," *IEEE Proc. PIMRC*, pp. 2297–2301, 2002.
[11] B. M. Hochwald, T. L. Marzetta, T. J. Richardson, W. Sweldens, and R. Urbanke, "Systematic design of unitary space-time constellations," *IEEE Trans. Inf. Theory*, vol. 46, no. 6, pp. 1962–1973, Sep. 2000.
[12] A. Hottinen, O. Tirkkonen, and R. Wichman*, Multi-Antenna Transceiver Techniques for 3G and Beyond*. Hoboken, NJ: Wiley, 2003.
[13] D. J. Love and R. W. Heath Jr., "Equal gain transmission in multiple-input multiple-output wireless systems," *IEEE Trans. Commun.*, vol. 51, no. 7, pp. 1102–1110, Jul. 2003.
[14] D. J. Love, R. W. Heath Jr., and T. Strohmer, "Grassmannian beamforming for multiple-input multiple-output wireless systems," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2735–2747, Oct. 2003.
[15] K. K. Mukkavilli, A. Sabharwal, E. Erkip, and B. Aazhang, "On beamforming with finite rate feedback in multiple-antenna systems," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2562–2579, Oct. 2003.
[16] C. R. Murthy and B. D. Rao, "Quantization methods for equal gain transmission with finite rate feedback," *IEEE Trans. Signal Process.*, vol. 55, no. 1, pp. 233–245, Jan. 2007.
[17] M. A. Najib and V. K. Prabhu, "Analysis of equal-gain diversity with partially coherent fading signals," *IEEE Trans. Veh. Technol.*, vol. 49, no. 5, pp. 783–791, May 2000.
[18] A. Narula, M. J. Lopez, M. D. Trott, and G. W. Wornell, "Efficient use of side information in multiple-antenna data transmission over fading channels," *IEEE J. Sel. Commun.*, vol. 16, no. 8, pp. 1423–1436, Oct. 1998.
[19] J. G. Proakis*, Digital Communications*, 4th ed. New York: McGraw Hill, 2000.
[20] S. H. Shin, B. K. Cho, H. C. Hwang, and K. S. Kwak, "An efficient scheme for transmit antenna diversity with limited feedback channel rate," *IEICE Trans. Commun.*, vol. E88-B, no. 8, pp. 3471–3474, Aug. 2005.
[21] C. W. Sung, S. C. Ip, and K. K. Leung, "A simple algorithm for closed-loop transmit diversity with a low-data-rate feedback channel," presented at the IEEE Pacific Rim Conf. Commun., Comput., and Signal Proc., Victoria, BC, Canada, Aug. 2005.
[22] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 744–765, Mar. 1998.
[23] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Inf. Theory*, vol. 45, no. 7, pp. 1456–1467, Jul. 1999.
[24] E. D. Weinberger, "Correlated and uncorrelated fitness landscapes and how to tell the difference," *Biol. Cybern.*, vol. 63, pp. 325–336, 1990.
[25] P. Xia, S. Zhou, and G. B. Giannakis, "Achieving the Welch bound with difference sets," *IEEE Trans. Inf. Theory*, vol. 51, no. 5, pp. 1900–1907, May 2005.
[26] Q. T. Zhang, "Probability of error for equal-gain combiners over Rayleigh channels: Some closed-form solutions," *IEEE Trans. Commun.*, vol. 45, no. 3, pp. 270–273, Mar. 1997.
[27] X. Zheng, Y. Xie, J. Li, and P. Stoica, "MIMO transmit beamforming under uniform elemental power constraint," *IEEE Trans. Signal Process.*, vol. 55, no. 11, pp. 5395–5406, Nov. 2007.

Kin-Kwong Leung received the B.Eng., M.Phil., and Ph.D. degrees in information engineering from the Chinese University of Hong Kong in 1997, 2000, and 2003, respectively.

He is currently a computational scientist in Cluster Technology Limited. His research interests include data mining, financial modeling, wireless networks, and computational complexity.

Chi Wan Sung received the B.Eng., M.Phil., and Ph.D. degrees in information engineering from the Chinese University of Hong Kong in 1993, 1995, and 1998, respectively.

He was an Assistant Professor at the Chinese University of Hong Kong. He joined the faculty at City University of Hong Kong in 2000, and is now an Associate Professor with the Department of Electronic Engineering. His research interests include multiuser information theory, design and analysis of algorithms, wireless communications, and radio resource management, and he is an editor for the *ETRI Journal*.

Majid Khabbazian received the B.Sc. degree in computer engineering from the Sharif University of Technology, Tehran, Iran, in 2002, and the Ph.D. degree in electrical and computer engineering from the University of British Columbia, Vancouver, BC, Canada, in 2008.

He is currently a postdoctoral fellow at the Massachusetts Institute of Technology Computer Science and Artificial Intelligence Lab. His research interest include wireless networks, network security, and distributed algorithms.

Mohammad Ali Safari received the B.S. degree from the Sharif University, Tehran, Iran, in 2001, the MMath degree from the university of Waterloo, Waterloo, ON, Canada, in 2003, and the Ph.D. degree from the Computer Science Department, University of British Columbia, Vancouver, BC, Canada, in 2007.

He spent one year as a postdoctoral Fellow in the Department of Computing Science, University of Alberta, Canada. He is currently an Assistant Professor in the Department of Computer Engineering, Sharif University of Technology.