Technical University of Crete

Department of Electronic and Computer Engineering

Polynomial-complexity Maximum-likelihood Block Noncoherent MPSK Detection

A thesis submitted in partial satisfaction of the requirements for the degree

Diploma
in
Electronic and Computer Egnineering

by
Dimitris S. Papailiopoulos
E-Mail: papailiopoulos@telecom.tuc.gr

Advisor
Assistant Prof. George N. Karystinos
E-Mail: karystinos@telecom.tuc.gr

Submitted: October 22, 2007

“*Information is the resolution of uncertainty.*”
- Claude Shannon

“*Nature uses only the longest threads to weave her patterns, so that each small piece of her fabric reveals the organization of the entire tapestry.*”
- Richard P. Feynman

“*I am interested in mathematics only as a creative art.*”
- G. H. Hardy

ABSTRACT

In wireless channels, maximum-likelihood (ML) block noncoherent detection offers significant gains over conventional symbol-by-symbol detection when the fading channel coefficients are not available and cannot be estimated at the receiver. Certainly, in general the complexity of the block detector grows exponentially with the symbol sequence length. However, it has been recently shown that for M-ary phase-shift keying (MPSK) modulation block noncoherent detection can be performed with polynomial complexity. In this work, we develop a new ML block noncoherent detector for MPSK transmission of arbitrary order and multiple-antenna reception. The proposed algorithm introduces auxiliary spherical variables and constructs with polynomial complexity a polynomial-size set which includes the ML data sequence. It is shown that the complexity of the proposed algorithm is polynomial in the sequence length and at least one order of magnitude lower than the complexity of computational-geometry based noncoherent detection algorithms that have been developed recently.

Contents

I. Introduction **6**

II. System Model **7**

III. Efficient ML Block Noncoherent MPSK Detection **9**
A. Theoretic developments 9
B. Algorithmic developments 14

IV. Simulation Results **17**

This thesis is dedicated to my beloved mother and father,

to my grandmother, and my aunt Keti,

to my dear friends Nikos, Stayros, Manolis, Konstantinos, who were always there for me,

to all university students that struggle to find their way in this education system,

to the memory of my dear grandfather.

I. Introduction

Multiple-antenna wireless systems are well known to attain increased orders of diversity resulting in substantially higher system capacity compared to single-antenna systems. When perfect channel state information (CSI) is available or can be retrieved through adequate channel estimation at the receiver, several coherent detection schemes can be followed. However, the very nature of wireless channels suggests rapidly changing channel conditions, thus making channel estimation complex and cost inefficient. Even when channel fades occur slowly, phase distortion is introduced and must be accounted for at the receiver end to avoid performance loss.

Alternatively, noncoherent detection has been studied extensively [1]-[5] and implemented in modern digital communication standards. Since noncoherent detection does not need any channel knowledge or estimation, it is applicable even in most degraded and fast fading channels, making it much more attractive than coherent detection under unfavorable channel conditions. Due to the memory in the received data sequence induced by fading channel memory, noncoherent maximum likelihood sequence detection (MLSD) has recently been a subject of extensive research [1]-[4]. Optimal receivers that suffer from exponential complexity with respect to the data sequence length as well as approximate and sub-optimal detection algorithms were developed in [1], [5]. However, very recent studies [3], [4] proved the existence of efficient noncoherent MLSD receiver schemes that attain optimality with polynomial complexity by utilizing computational-geometry (CG) based optimization algorithms.

The present work shows that noncoherent MLSD of MPSK symbols in SIMO systems can be expressed as a rank-deficient quadratic form maximization problem and computed efficiently in polynomial time. We follow a completely different approach than [3],[4] and, inspired by the work in [6]-[8],[1] construct a polynomial-complexity noncoherent MLSD method that is at least one order of magnitude faster than the method in [4]. The proposed method that is developed in this present work is also applicable to any arbitrary-order MPSK modulation. Further analysis shows that the computational complexity depends only on the data sequence length and receive-diversity order and does not depend on SNR.

[1]The work in [6]-[8] considers the efficient computation of the binary vector that maximizes a rank-deficient quadratic form. The authors prove the existence of the optimal solution and develop a method that computes it in polynomial time. Although rank-deficient quadratic form maximization was also treated in [10] based on CG principles, the method in [6]-[8] requires at least one order of magnitude less complexity compared to the method in [10].

II. System Model

We consider the transmission of a sequence of N uncoded M-ary phase-shift keying (MPSK) data symbols $\mathbf{s} = [s_1, s_2, ..., s_N]^T$, where s_n is selected from an M-ary alphabet $\mathcal{A}_M \stackrel{\triangle}{=} \{e^{j\frac{\pi}{M}(2m+1)} | m = 0, 1, \ldots, M-1\}$, $n = 1, 2, \ldots, N$. The data sequence is shaped and transmitted over D independent and identically distributed (i.i.d.) frequency flat Rayleigh fading wireless channels. The downconverted and pulse-matched equivalent received signal at the dth antenna is

$$\mathbf{y}_d = \sqrt{P} h_d \mathbf{s} + \mathbf{n}_d \tag{1}$$

where P is the constant transmitted power, and h_d denotes the coefficient of the channel between the transmit antenna and the dth receive antenna and is modeled as zero-mean complex Gaussian with variance σ_h^2. Furthermore, $\mathbf{n}_d$ represents additive white complex Gaussian noise (AWGN) and is modeled as a zero-mean complex Gaussian vector with co-variance matrix $\sigma_n^2 \mathbf{I}$. We collect all received data from the D receive antennas and form the $N \times D$ "received matrix"

$$\mathbf{Y} \stackrel{\triangle}{=} [\mathbf{y}_1 \mathbf{y}_2 \ldots \mathbf{y}_D]. \tag{2}$$

The D channel coefficients h_d, $d = 1, 2, \ldots, D$, are assumed unknown to both the transmitter and the receiver, implying that noncoherent detection has to be performed. The MLSD decision for the transmitted sequence $\mathbf{s}$ given the $N \times D$ observation matrix $\mathbf{Y}$ maximizes the conditional probability density function (pdf) of $\mathbf{Y}$ given $\mathbf{s}$. Thus, the maximization problem becomes

$$\begin{aligned} \mathbf{s}_{opt} &\stackrel{\triangle}{=} \arg \max_{\mathbf{s} \in \mathcal{A}_M^N} f(\mathbf{Y} | \mathbf{s}) \\ &= \arg \max_{\mathbf{s} \in \mathcal{A}_M^N} f(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_D | \mathbf{s}). \end{aligned} \tag{3}$$

Due to independence among the D channels, the columns of the received matrix $\mathbf{Y}$ are i.i.d. given the transmitted sequence $\mathbf{s}$. Therefore,

$$\begin{aligned} \mathbf{s}_{opt} &= \arg \max_{\mathbf{s} \in \mathcal{A}_M^N} \prod_{d=1}^{D} f(\mathbf{y}_d | \mathbf{s}) \\ &= \arg \max_{\mathbf{s} \in \mathcal{A}_M^N} \sum_{d=1}^{D} \ln f(\mathbf{y}_d | \mathbf{s}). \end{aligned} \tag{4}$$

The conditional received vector at the dth antenna given the transmitted sequence is $\mathbf{y}_d | \mathbf{s} = h_d \mathbf{s} + \mathbf{n}_d$ where $h_d \mathbf{s}$ is a singular complex Gaussian vector independent from $\mathbf{n}_d$, $d = 1, 2, \ldots, D$. The following proposition identifies the pdf of $\mathbf{y}_d | \mathbf{s}$.

Proposition 1 *The sum of a singular complex Gaussian vector and an independent complex Gaussian vector results in a complex Gaussian vector.*

Proof Consider a singular Gaussian vector of the form

$$\mathbf{q} = [a_1 q, a_2 q, \ldots, a_N q]^T, \quad q \sim \text{CN}(m_q, \sigma_q^2), \quad a_1, a_2, \ldots, a_N \in \mathbb{R} \tag{5}$$

and an independent complex Gaussian vector

$$\mathbf{n} = [n_1, n_2, \ldots, n_N]^T, \quad \mathbf{n} \sim \text{CN}(\mathbf{m}_n, \mathbf{C}_n). \tag{6}$$

Consider the sum $\sum_{i=1}^{N}(q_i + n_i) = \sum_{i=1}^{N} n_i + \sum_{i=1}^{N} a_i q$ and observe that both $\left(q \sum_{i=1}^{N} a_i\right)$ and $\left(\sum_{i=1}^{N} n_i\right)$ are independent complex Gaussian random variables. Then $\sum_{i=1}^{N}(q_i + n_i)$ is also a complex Gaussian random variable. Consequently $\mathbf{z} = \mathbf{n} + \mathbf{q}$ is a complex Gaussian vector [12]. $\square$

According to Proposition 1, since h_d and $\mathbf{n}_d$ are both zero-mean, $\mathbf{y}_d|\mathbf{s}$ is a zero-mean complex Gaussian vector with covariance matrix

$$\mathbf{R}_{\mathbf{y}_d|\mathbf{s}} \stackrel{\triangle}{=} E\{\mathbf{y}_d \mathbf{y}_d^H | \mathbf{s}\} = \sigma_n^2 \mathbf{I} + P\sigma_h^2 \mathbf{s}\mathbf{s}^H. \tag{7}$$

As a result, the MLSD receiver of (4) becomes

$$\begin{aligned} \mathbf{s}_{opt} &= \arg\max_{\mathbf{s} \in \mathcal{A}_M^N} \sum_{d=1}^{D} \ln \frac{1}{\pi^N |\mathbf{R}_{\mathbf{y}_d|\mathbf{s}}|} \exp\left\{-\mathbf{y}_d^H \mathbf{R}_{\mathbf{y}_d|\mathbf{s}}^{-1} \mathbf{y}_d\right\} \\ &= \arg\max_{\mathbf{s} \in \mathcal{A}_M^N} \sum_{d=1}^{D} \left(-\mathbf{y}_d^H \mathbf{R}_{\mathbf{y}_d|\mathbf{s}}^{-1} \mathbf{y}_d + \ln \frac{1}{\pi^N |\mathbf{R}_{\mathbf{y}_d|\mathbf{s}}|}\right). \end{aligned} \tag{8}$$

Using the property $|\mathbf{A} + \mathbf{c}\mathbf{d}^H| = |\mathbf{A}|(1 + \mathbf{d}^H \mathbf{A}^{-1} \mathbf{c})$ found in [11], we compute $|\mathbf{R}_{\mathbf{y}_d|\mathbf{s}}| = \sigma_n^{2N}(1 + NP\frac{\sigma_h^2}{\sigma_n^2})$. Since, $|\mathbf{R}_{\mathbf{y}_d|\mathbf{s}}|$ is not a function of $\mathbf{s}$, it can hence be dropped from the detector in (8). Moreover, using the matrix inversion lemma, the inverse of $\mathbf{R}_{\mathbf{y}_d|\mathbf{s}}$ becomes $\mathbf{R}_{\mathbf{y}_d|\mathbf{s}}^{-1} = \frac{1}{\sigma_n^2}\left(\mathbf{I} - \frac{P\sigma_h^2}{\sigma_n^2 + NP\sigma_h^2}\mathbf{s}\mathbf{s}^H\right)$, implying that the decision rule in (8) is simplified to

$$\begin{aligned} \mathbf{s}_{opt} &= \arg\max_{\mathbf{s} \in \mathcal{A}_M^N} \sum_{d=1}^{D} \frac{1}{\sigma_n^2}\left(-||\mathbf{y}_d||^2 + \frac{P\sigma_h^2}{\sigma_n^2 + NP\sigma_h^2} \mathbf{y}_d^H \mathbf{s}\mathbf{s}^H \mathbf{y}_d\right) \\ &= \arg\max_{\mathbf{s} \in \mathcal{A}_M^N} \sum_{d=1}^{D} |\mathbf{y}_d^H \mathbf{s}|^2 \\ &= \arg\max_{\mathbf{s} \in \mathcal{A}_M^N} ||\mathbf{Y}^H \mathbf{s}||. \end{aligned} \tag{9}$$

If the above optimization is performed through exhaustive search, then it costs $\mathcal{O}(M^N)$ computations which is an intractable complexity even for moderate vales of N. In the next section, we follow an approach similar to the one of [6]-[8] but tailored to our detection problem in (9). Specifically, we introduce $2D-1$ spherical coordinates and develop an efficient algorithm to build a set $\mathcal{S}(\mathbf{Y}_{N\times D}) \subset \mathcal{A}_M^N$ that consists of $|\mathcal{S}(\mathbf{Y}_{N\times D})| = \mathcal{O}((NM)^{2D-1})$ signal vectors, is constructed with $\mathcal{O}((MN)^{2D})$ computations, and contains the optimal vector $\mathbf{s}_{opt}$ in (9).

III. Efficient ML Block Noncoherent MPSK Detection

A. Theoretic developments

In order to develop an efficient technique for solving the maximization problem in (9), we introduce $2D-1$ auxiliary hyperspherical coordinates $\phi_1 \in (-\pi, \pi]$, $\phi_2, \ldots, \phi_{2D-1} \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right]$ and define the $2D \times 1$ hyperspherical vector

$$\tilde{\mathbf{c}}(\phi_1, \ldots, \phi_{2D-1}) \triangleq \begin{bmatrix} \sin\phi_1 \\ \cos\phi_1 \sin\phi_2 \\ \vdots \\ \cos\phi_1 \ldots \cos\phi_{2D-2} \sin\phi_{2D-1} \\ \cos\phi_1 \ldots \cos\phi_{2D-2} \cos\phi_{2D-1} \end{bmatrix} \tag{10}$$

as well as the $D \times 1$ hyperspherical complex vector

$$\mathbf{c}(\phi_1, \ldots, \phi_{2D-1}) \triangleq \tilde{\mathbf{c}}_{1:D,1}(\phi_1, \ldots, \phi_{2D-1}) + j\tilde{\mathbf{c}}_{D+1:2D,1}(\phi_1, \ldots, \phi_{2D-1}). \tag{11}$$

Then, the problem in (9) is rewritten equivalently as

$$\begin{aligned} \mathbf{s}_{opt} &= \arg \max_{\mathbf{s} \in \mathcal{A}_M^N} ||\mathbf{Y}^H \mathbf{s}|| \\ &= \arg \max_{\mathbf{s} \in \mathcal{A}_M^N} \max_{\phi_1 \in (-\pi, \pi]} \max_{\phi_2, \ldots, \phi_{2D-1} \in (-\frac{\pi}{2}, \frac{\pi}{2}]} |\mathbf{s}^H \mathbf{Y} \mathbf{c}(\phi_1, \ldots, \phi_{2D-1})| \end{aligned} \tag{12}$$

due to Cauchy-Schwartz Inequality which states that for any $\mathbf{v} \in \mathbb{C}^D$

$$|\mathbf{v}^H \mathbf{c}(\phi_1, \ldots, \phi_{2D-1})| \leq ||\mathbf{v}|| \cdot \underbrace{||\mathbf{c}(\phi_1, \ldots, \phi_{2D-1})||}_{=1} \tag{13}$$

with equality if and only if $\phi_1, \ldots, \phi_{2D-1}$ are the hyperspherical coordinates of $\mathbf{v}$. Furthermore $\forall \mathbf{v} \in \mathbb{C}^D$,

$$\Re\{\mathbf{v}^H \mathbf{c}(\phi_1, \ldots, \phi_{2D-1})\} \leq |\mathbf{v}^H \mathbf{c}(\phi_1, \ldots, \phi_{2D-1})|, \tag{14}$$

with equality if and only if $\phi_1, \ldots, \phi_{2D-1}$ are the hyperspherical coordinates of $\mathbf{v}$. Hence, the maximization problem in (12) becomes

$$\mathbf{s}_{opt} = \arg \max_{\mathbf{s} \in \mathcal{A}_M^N} \max_{\phi_1 \in (-\pi, \pi]} \max_{\phi_2, \ldots, \phi_{2D-1} \in (-\frac{\pi}{2}, \frac{\pi}{2}]} \Re\{\mathbf{s}^H \mathbf{Y} \mathbf{c}(\phi_1, \ldots, \phi_{2D-1})\}. \tag{15}$$

We interchange the maximizations in (15) and obtain the equivalent problem

$$\max_{\phi_1 \in (-\pi, \pi]} \max_{\phi_2, \ldots, \phi_{2D-1} \in (-\frac{\pi}{2}, \frac{\pi}{2}]} \sum_{n=1}^{N} \max_{s_n \in \mathcal{A}_M} \Re\{s_n^* \mathbf{Y}_{n,1:D} \mathbf{c}(\phi_1, \phi_2, \ldots, \phi_{2D-1})\}, \tag{16}$$

we observe that the original maximization problem in (9) is decomposed in a set of symbol-by-symbol coherent detection rules for a given set of angles $(\phi_1, \ldots, \phi_{2D-1}) \in (-\pi, \pi] \times \left(-\frac{\pi}{2}, \frac{\pi}{2}\right]^{2D-2}$. For such a set of angles the maximizing argument of a coherent decision metric of the sum in (16), say for symbol s_n, depends only on the corresponding row of "received" matrix $\mathbf{Y}$, since that is the one related with the nth transmitted symbol. As $\phi_1, \phi_2, \ldots, \phi_{2D-1}$ vary, the decision in favor of s_n is maintained as long as a decision boundary is not crossed. Due to the structure of $\mathcal{A}_M$, the $\frac{M}{2}$ decision boundaries that affect the maximization in (16) are given by

$$\mathbf{Y}_{n,1:D}\mathbf{c}(\phi_1, \ldots, \phi_{2D-1}) = Ae^{j2\pi\frac{k}{M}}, k = 0, 1, \ldots, \frac{M}{2} - 1, \quad n = 1, 2, \ldots, N. \tag{17}$$

The decision boundaries in (17) can be rewritten without loss of generality (w.l.o.g.) as

$$\Im\{e^{-j2\pi\frac{k}{M}}\mathbf{Y}_{n,1:D}\mathbf{c}(\phi_1, \ldots, \phi_{2D-1})\} = 0, \quad k = 0, 1, \ldots, \frac{M}{2} - 1, \quad n = 1, 2, \ldots, N \tag{18}$$

which is equivalent to

$$\tilde{\mathbf{Y}}_{l,1:2D}\tilde{\mathbf{c}}(\phi_1, \ldots, \phi_{2D-1}) = 0, \quad l = 1, \ldots, \frac{MN}{2}, \tag{19}$$

where

$$\tilde{\mathbf{Y}} \triangleq \left[\Im(\hat{\mathbf{Y}}) \quad \Re(\hat{\mathbf{Y}})\right], \tag{20}$$

$$\hat{\mathbf{Y}} \triangleq \mathbf{Y} \otimes \left[1 \; e^{-j\frac{2\pi}{M}} \; e^{-j\frac{4\pi}{M}} \ldots \; e^{-j\frac{2\pi}{M}\left(\frac{M}{2}-1\right)}\right]^T, \tag{21}$$

and $\otimes$ denotes Kronecker product. The inner maximization rule in (16) motivates us to define a *decision function* s that maps a set of angles $(\phi_1, \phi_2, \ldots, \phi_{2D-1})$ to a certain value of set $\mathcal{A}_M$ according to

$$s(\mathbf{y}^T; \phi_1, \phi_2, \ldots, \phi_{2D-1}) \triangleq \arg\max_{s \in \mathcal{A}_M} \Re\left\{s^* \mathbf{y}^T \mathbf{c}(\phi_1 \ldots, \phi_{2D-1})\right\} \tag{22}$$

for any $\mathbf{y} \in \mathbb{C}^D$. Then, for the given $N \times D$ matrix $\mathbf{Y}$, each set of angles in $(-\pi, \pi] \times \left(-\frac{\pi}{2}, \frac{\pi}{2}\right]^{2D-2}$ is mapped to a candidate M-ary vector

$$\mathbf{s}(\mathbf{Y}_{N\times D}; \phi_1, \ldots, \phi_{2D-1}) \triangleq \begin{bmatrix} s(\mathbf{Y}_{1,1:D}; \phi_1, \ldots, \phi_{2D-1}) \\ s(\mathbf{Y}_{2,1:D}; \phi_1, \ldots, \phi_{2D-1}) \\ \vdots \\ s(\mathbf{Y}_{N,1:D}; \phi_1, \ldots, \phi_{2D-1}) \end{bmatrix} \tag{23}$$

and the optimal vector $\mathbf{s}_{opt}$ in (15) belongs to $\bigcup_{\phi_1 \in (-\pi,\pi]} \bigcup_{\phi_2,\ldots,\phi_{2D-1} \in (-\frac{\pi}{2},\frac{\pi}{2}]} \mathbf{s}(\mathbf{Y}_{N\times D}; \phi_1, \ldots, \phi_{2D-1})$. Furthermore, since opposite M-ary vectors result in the same metric in (9), we can ignore the values of ϕ_1 in $\left(-\pi, -\frac{\pi}{2}\right] \cup \left(\frac{\pi}{2}, \pi\right]$ and

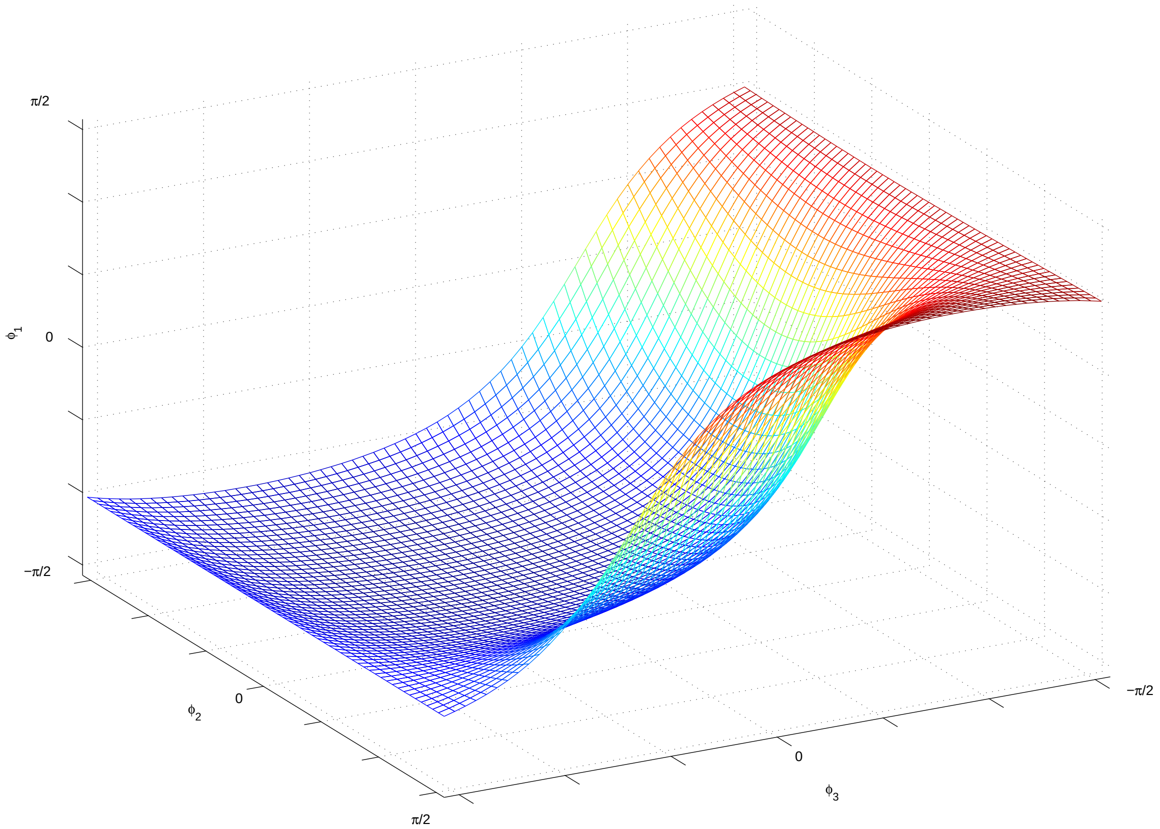

Fig. 1.

consider $\phi_1, \ldots, \phi_{2D-1} \in \varPhi \triangleq \left(-\frac{\pi}{2}, \frac{\pi}{2}\right]$, hence the maximization in (16) can be rewritten as

$$\max_{\phi_1,\phi_2,\ldots,\phi_{2D-1}\in\varPhi} \sum_{n=1}^{N} \max_{s_n\in\mathcal{A}_M} \Re\{s_n^* \mathbf{Y}_{n,1:D}\mathbf{c}(\phi_1,\phi_2,\ldots,\phi_{2D-1})\}. \tag{24}$$

Since we have defined a new problem space, we may now collect all M-ary candidate vectors to set

$$\mathcal{S}(\mathbf{Y}_{N\times D}) \triangleq \bigcup_{\phi_1,\ldots,\phi_{2D-1}\in\varPhi} \{\mathbf{s}(\mathbf{Y}_{N\times D};\phi_1,\ldots,\phi_{2D-1})\} \subseteq \mathcal{A}_M^N. \tag{25}$$

Such set includes the maximizer of our detection problem, hence

$$\mathbf{s}_{opt} = \arg\max_{s\in\mathcal{S}(\mathbf{Y})} ||\mathbf{Y}^H\mathbf{s}||. \tag{26}$$

The set $\mathcal{S}(\mathbf{Y}_{N\times D})$ that includes $\mathbf{s}_{opt}$ is later proved to have cardinality $|\mathcal{S}(\mathbf{Y}_{N\times D})| = \mathcal{O}((NM)^{2D-1})$. Moreover, an efficient algorithm for the construction of the aforementioned set is developed in subsection B with $\mathcal{O}((NM)^{2D})$ complexity.

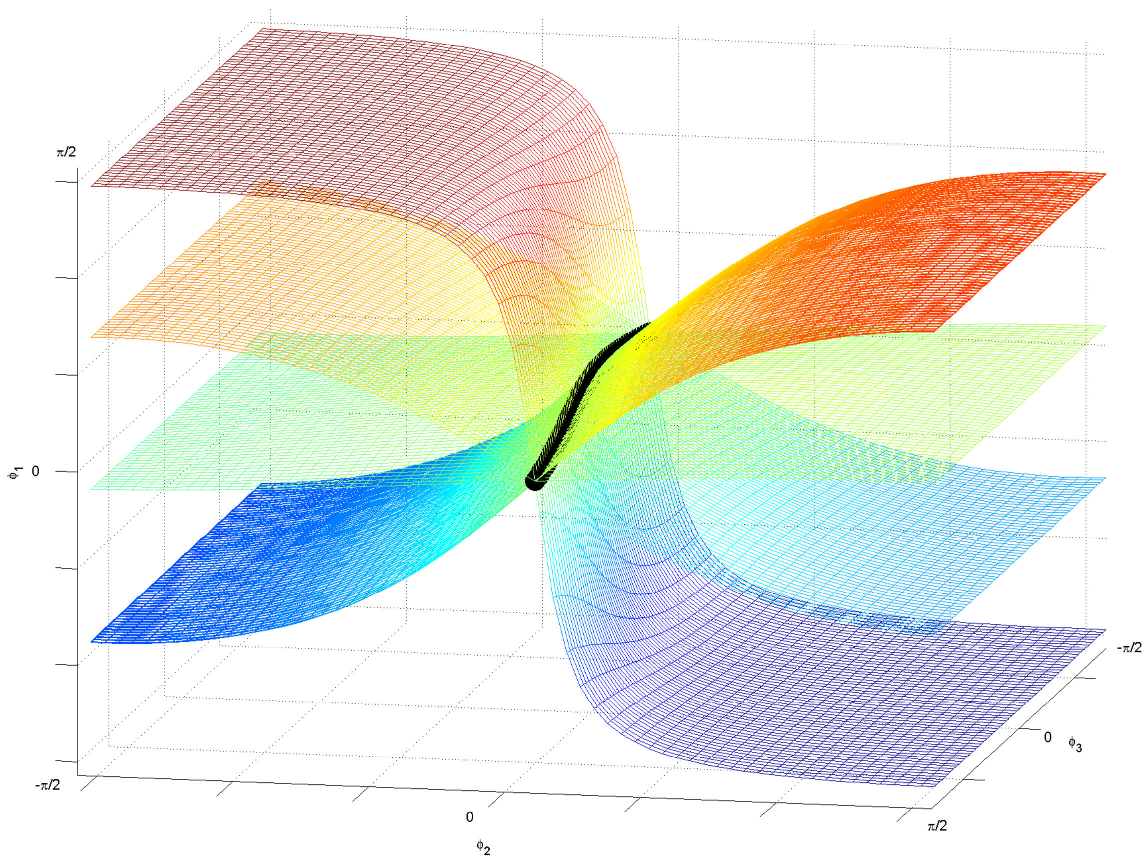

Fig. 2.

From (21), we observe that the rows of the $\frac{MN}{2} \times 2D$ matrix $\tilde{\mathbf{Y}}$ determine $\frac{MN}{2}$ hypersurfaces $\mathcal{F}(\tilde{\mathbf{Y}}_{1,1:2D}), \mathcal{F}(\tilde{\mathbf{Y}}_{2,1:2D}), \ldots, \mathcal{F}(\tilde{\mathbf{Y}}_{\frac{MN}{2},2:D})$. An example of such hypersurface is shown in Fig.1 for $D = 2$. Each hypersurface partitions the hypercube Φ^{2D-1} into two regions, and all hypersurfaces $\mathcal{F}(\tilde{\mathbf{Y}}_{1,1:2D}), \mathcal{F}(\tilde{\mathbf{Y}}_{2,1:2D}), \ldots, \mathcal{F}(\tilde{\mathbf{Y}}_{\frac{MN}{2},2:D})$ partition the hypercube Φ^{2D-1} into K cells $C_1, C_2, \ldots, C_K$ such that $\bigcup_{k=1}^{K} C_k = \Phi^{2D-1}$, $C_k \cap C_j \neq 0 \; \forall \; k \neq j$, with each cell C_k corresponding to a unique $\mathbf{s}_k \in \mathcal{A}_M^N$. Let $\{i_1, i_2, \ldots, i_{2D-1}\} \subset \{1, 2, \ldots, \frac{MN}{2}\}$ be a subset of $2D-1$ indices (that correspond to $2D-1$ hypersurfaces) and $\phi(\tilde{\mathbf{Y}}_{\frac{MN}{2} \times D}; i_1, \ldots, i_{2D-1}) \in \Phi^{2D-1}$ equal the vector of coordinates of the intersection of hypersurfaces $\mathcal{F}(\tilde{\mathbf{Y}}_{i_1,1:2D}), \ldots, \mathcal{F}(\tilde{\mathbf{Y}}_{i_{2D-1},1:2D})$. The basic property of such an intersection is presented in the following proposition. The proof is given in Appendix A.

Proposition 2 *Any combination of $2D-1$ hypersurfaces, say $\mathcal{F}(\tilde{\mathbf{Y}}_{i_1,1:2D}), \mathcal{F}(\tilde{\mathbf{Y}}_{i_2,1:2D}), \ldots, \mathcal{F}(\tilde{\mathbf{Y}}_{i_{2D-1},1:2D})$, has a unique intersection (which is a vertex of a cell) if and only if no more than two hypersurfaces originate from the same row of the* *observation matrix* $\mathbf{Y}$. $\square$

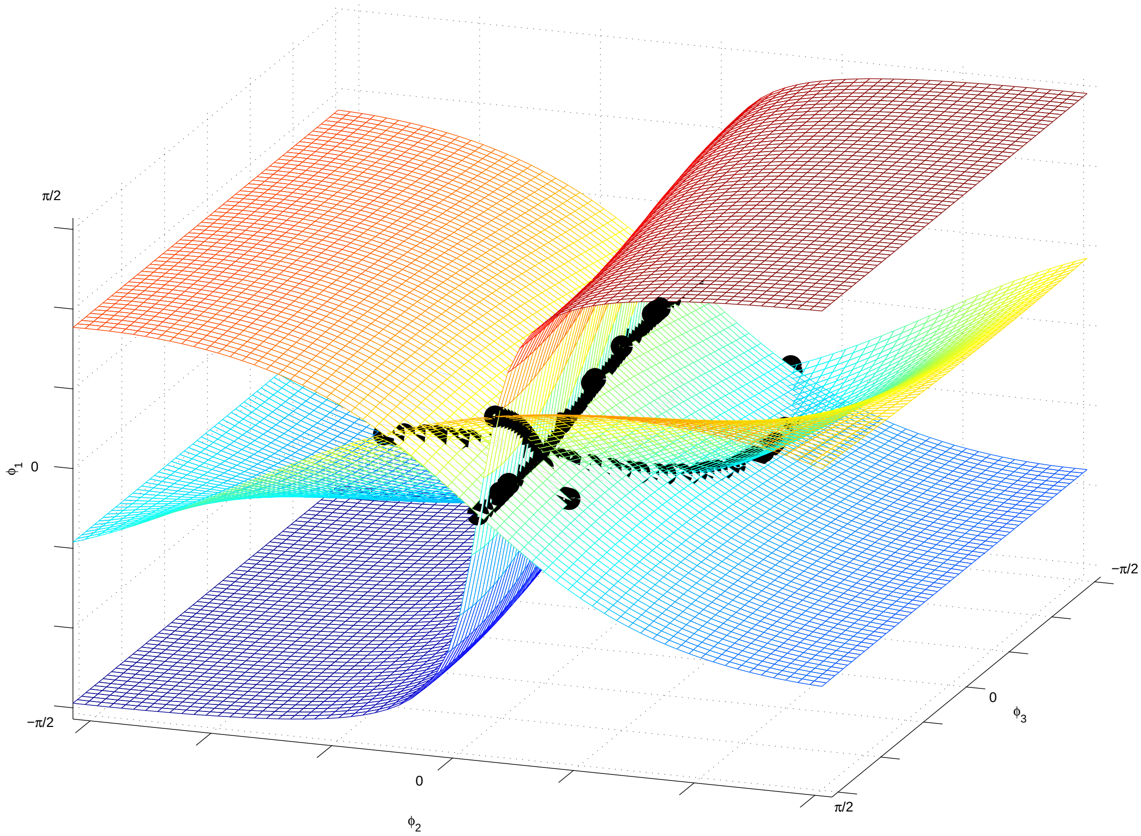

Fig. 3.

In Fig.2 we present the intersection of $\frac{M}{2}$ hypersurfaces that originate from the same row of $\mathbf{Y}$, with $M = 8$, $D = 2$. We observe that such an ensemble of hypersurfaces partitions the hypercube Φ^{2D-1} into M partitions, with each partition corresponding to a unique element of the $\mathcal{A}_M$ alphabet. Moreover as an illustrative example for proposition 2, we present, in Fig.3, the intersection of $2D-1$ hypersurfaces, where no more than two hypersurfaces originate from the same row of $\mathbf{Y}$, and observe that such unique intersection is a single point and a vertex of a cell.

Any cell, say $C\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D}; i_1, \ldots, i_{2D-1}\right)$ for example, is exclusively associated with a unique M-ary vector $\mathbf{s}(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D};$ $i_1, \ldots, i_{2D-1})$, in the sense that within that cell all single points of the form $(\phi_1, \phi_2, \ldots, \phi_{2D-1}) \in C\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D}; i_1, \ldots, i_{2D-1}\right)$

yield the same candidate vector, $\mathbf{s}\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D}; i_1,\ldots,i_{2D-1}\right)$ that is; we collect all such M-ary vectors to set

$$J(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D}) \triangleq \bigcup_{\{i_1,\ldots,i_{2D-1}\}\subset\{1,\ldots,\frac{MN}{2}\}} \left\{\mathbf{s}\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D}; i_1,\ldots,i_{2D-1}\right)\right\} \subseteq \mathcal{A}_M^N. \tag{27}$$

The cardinality of $J(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D})$ is given by

$$|J(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D})| = \sum_{d=0}^{D-1}\binom{N}{d}\binom{N-d}{2D-(1+2d)}\left(\frac{M}{2}\right)^{2D-1-d} = \mathcal{O}((NM)^{2D-1}). \tag{28}$$

Thus, $J(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D})$ contains $\mathcal{O}((NM)^{2D-1})$ M-ary vectors. Then, it can be shown [8] that all candidate vectors form the set

$$\begin{aligned}\mathcal{S}(\mathbf{Y}_{N\times D}) &= J\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D}\right)\cup J\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times (2D-2)}\right)\cup\ldots\cup J\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2}\right)\\ &= \bigcup_{d=0}^{D-1} J\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2(D-d)}\right),\end{aligned} \tag{29}$$

with $\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2(D-d)}$ denoting the matrix that contains all first $2(D-d)$ columns of $\tilde{\mathbf{Y}}$. As a result the cardinality of the above set is

$$\begin{aligned}|S(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D})| &= \left|J\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D}\right)\right| + \left|J\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2(D-2)}\right)\right| + \cdots + \left|J\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2}\right)\right|\\ &= \sum_{Q=1}^{D}\sum_{d=0}^{Q-1}\binom{N}{d}\binom{N-d}{2Q-(1+2d)}\left(\frac{M}{2}\right)^{2Q-1-d} =\\ &= \sum_{Q=1}^{D}\mathcal{O}((NM)^{2Q-1}) = \mathcal{O}((NM)^{2D-1}),\end{aligned} \tag{30}$$

which is straightforward since $J\left(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D}\right)$, with cardinality $\mathcal{O}((NM)^{2D-1})$, is the largest set among the subsets of $S(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2D})$.

To summarize the developments of this subsection, we have utilized $2D-1$ auxiliary hyperspherical coordinates, and partitioned the hypercube Φ^{2D-1} into $\mathcal{O}((NM)^{2D-1})$ cells associated with unique M-ary candidate vectors that constitute the set $\mathcal{S}(\mathbf{Y}_{N\times D}) \subseteq \mathcal{A}_M^N$ which includes $\mathbf{s}_{opt}$ in (9). Therefore, the initial detection problem in (9) has been converted into a maximization among $\mathcal{O}((NM)^{2D-1})$ candidate vectors.

B. Algorithmic developments

The construction of $\mathcal{S}(\mathbf{Y}_{N\times D})$ is of special interest since it determines the overall performance of the proposed method. According to (29), it reduces to the parallel construction of $J(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2d})$, for $d = 2D, 2D-2, \ldots, 2$, which can be also fully parallelized since cells in the hypersurface arrangement are examined independently from each other. It can be shown that the decision function in (22) determines definitely the corresponding symbol s_n if and only if no hypersurface originates from $\mathbf{Y}_{n,1:d}$. For the hypersurfaces that pass through the cell intersection, the rule in (22) becomes ambiguous. In such a case, definite determination of s_n is attained if ϕ_{2D-1} is set to $\frac{\pi}{2}$ and (22) is examined at the intersection of the same hypersurfaces except from the hypersurface of interest.

The algorithm visits independently the $|\mathcal{S}(\mathbf{Y}_{N\times D})| = \mathcal{O}((NM)^{2D-1})$ intersections and computes the candidate vector in $\mathcal{A}_M^N$ for each intersection. The cost of the algorithm for each candidate vector is $\mathcal{O}(MN)$ since it needs to check at most MN inequalities. Therefore, the overall complexity for the construction of $\mathcal{S}(\mathbf{Y}_{N\times D})$ becomes $\mathcal{O}((NM)^{2D-1})\ \mathcal{O}(MN) = \mathcal{O}((NM)^{2D})$. The MATLAB code for the construction of $\mathcal{S}(\mathbf{Y}_{N\times D})$ is given in Appendix B..

It remains to describe how the vector of coordinates $\boldsymbol{\phi}(\tilde{\mathbf{Y}}_{\frac{MN}{2}\times d}; i_1, \ldots, i_{2d-1})$ is obtained efficiently. Since such coordinates represent the intersection of $\mathcal{F}(\tilde{\mathbf{Y}}_{i_1,1:2d}), \mathcal{F}(\tilde{\mathbf{Y}}_{i_2,1:2d}), \ldots, \mathcal{F}(\tilde{\mathbf{Y}}_{i_{2d-1},1:2d})$ a unique solution is given by the following system of equations

$$\left.\begin{array}{c} \tilde{\mathbf{Y}}_{[i_1,i_2,\ldots,i_{2d-1}],1:2d}\mathbf{c}(\phi_1,\phi_2,\ldots,\phi_{2d-1}) = \mathbf{0}_{(d-1)\times 1} \\ ||\mathbf{c}(\phi_1,\phi_2,\ldots,\phi_{2d-1})|| = 1 \end{array}\right\}. \tag{31}$$

We observe that the above equations *(i)* explicitly imply orthogonality between the $(2d-1)$-dimensional hyperplane defined by $\tilde{\mathbf{Y}}_{[i_1,i_2,\ldots,i_{2d-1}],1:2d}$ and the hyperspherical vector $\mathbf{c}(\phi_1,\phi_2,\ldots,\phi_{2d-1})$, and *(ii)* constrain the solutions to unitary vectors. To further assist the present work we introduce the following proposition.

Proposition 3 *Consider a full-rank $(d-1)\times d$ complex matrix $\mathbf{Q}$. Then, the equation*

$$\mathbf{Q}\mathbf{c}(\phi_1,\ldots,\phi_{d-1}) = \mathbf{0}_{(d-1)\times 1} \tag{32}$$

has a unique solution $\boldsymbol{\phi}(\mathbf{Q}) \in \Phi^{2d-1}$ which consists of the hyperspherical coordinates of the zero left singular vector of $\mathbf{Q}$. $\square$

We recall that one way to obtain the zero left singular vector of $\mathbf{Q}$ is by singular value decomposition (SVD). Such a singular vector is known to be unitary, thus norm-constrained to 1. Consequently, the solution of (31) can be efficiently obtained simply by computing the zero left singular vector of matrix $\tilde{\mathbf{Y}}_{[i_1,i_2,\ldots,i_{2d-1}],1:2d}$, which is the hyperspherical vector of interest $\mathbf{c}(\phi_1,\phi_2,\ldots,\phi_{2D-1})$. Now, since $\mathbf{c}(\phi_1,\phi_2,\ldots,\phi_{2d-1})$ is obtained we need only compute $s_n = \arg\max_{s\in\mathcal{A}_M} \Re\{s^*\mathbf{Y}_{n,1:2d}$ $\mathbf{c}(\phi_1\ldots,\phi_{2d-1})\}\ \forall n \in \{1,2,\ldots,N\}$, that correspond to all elements of the candidate vector explicitly defined by the intersection of $\mathcal{F}(\tilde{\mathbf{Y}}_{i_1,1:2d}), \mathcal{F}(\tilde{\mathbf{Y}}_{i_2,1:2d}), \ldots, \mathcal{F}(\tilde{\mathbf{Y}}_{i_{2d-1},1:2d})$. However, as mentioned earlier, the determination of s_n where

$n \in \{i_1, i_2, \ldots, i_{2d-1}\}$ becomes ambiguous, hence, we set $\phi_{2d-1} = \frac{\pi}{2}$ and solve the following system

$$\left.\begin{array}{c} \tilde{\mathbf{Y}}_{[i_1,i_2,\ldots,i_{n-1},i_{n+1},\ldots,i_{2d-1}]}\mathbf{c}(\phi_1, \phi_2, \ldots, \phi_{2d-2}, \frac{\pi}{2}) = \mathbf{0}_{(d-2)\times 1} \\ \left|\left|\mathbf{c}(\phi_1, \phi_2, \ldots, \phi_{2d-2}, \frac{\pi}{2})\right|\right| = 1 \end{array}\right\}. \tag{33}$$

Solution is obtained similarly to (31), by computing the zero left singular vector of $\tilde{\mathbf{Y}}_{[i_1,i_2,\ldots,i_{n-1},i_{n+1},\ldots,i_{2d-1}],1:2d-1}$. Eventually, we compute $s_n = \arg\max_{s\in\mathcal{A}_M} \Re\left\{s^*\mathbf{Y}_{n,1:2d}\mathbf{c}(\phi_1, \ldots, i_{n-1}, i_{n+1}, \ldots, i_{2d-1})\right\}$ for $n \in \{i_1, i_2, \ldots, i_{2d-1}\}$.

We recall that the corresponding complexity of [4] is $\mathcal{O}((NM)^{2D}\mathrm{LP}(NM, 2D))$ where $\mathrm{LP}(NM, 2D)$ is the complexity of a linear programming (LP) optimization problem with MN inequalities and 2D variables. Provided that the worst-case complexity of $\mathrm{LP}(NM, 2D)$ in linear in NM [13], it turns out that the method in [4] costs $\mathcal{O}((NM)^{2D+1})$ calculations, i.e., one order of magnitude more calculations than the proposed algorithm. In addition, [4] treats only the case $M = 2$ (BPSK) and $M = 4$ (QPSK).

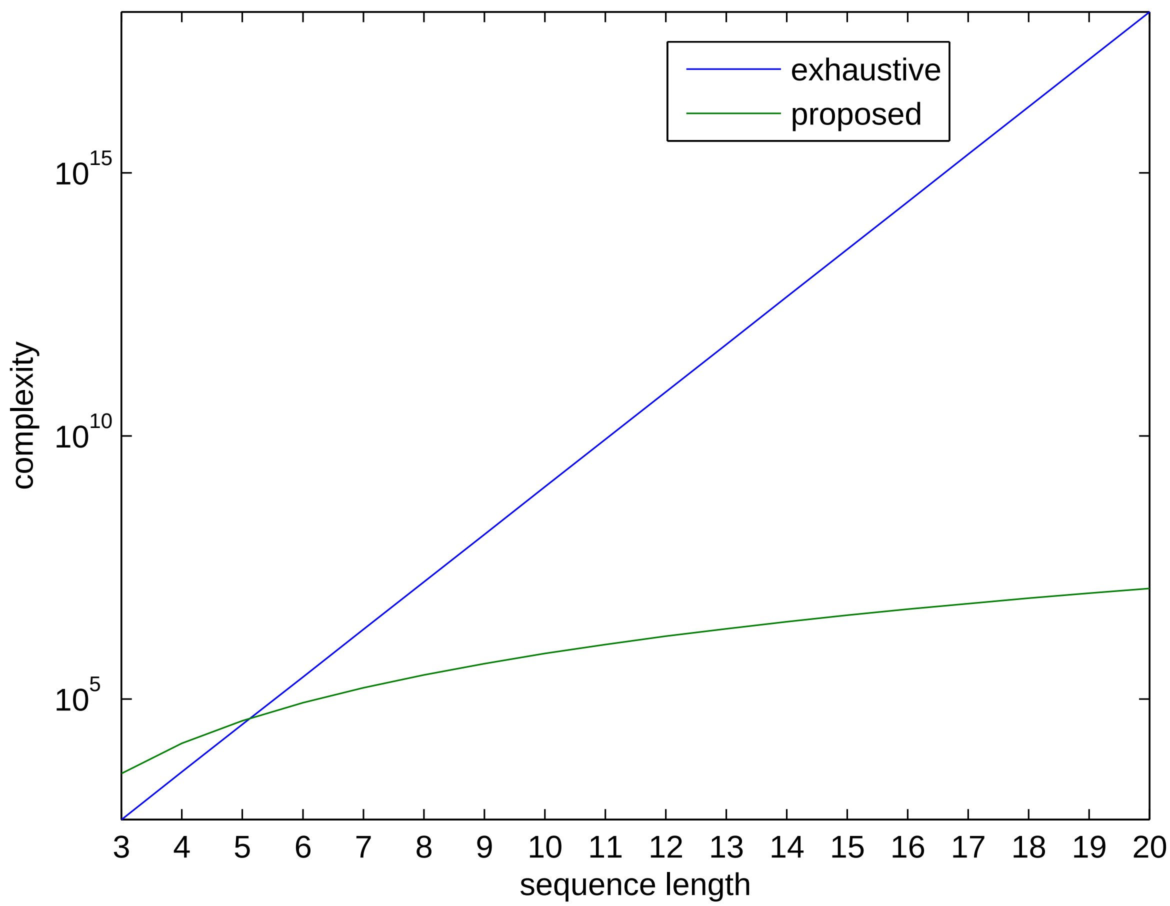

Fig. 4.

Finally, in Fig.4 we draw the complexity of the proposed algorithm and that of the exhaustive search, for $D = 2$ and 8-PSK constellations. We observe that for sequence lengths less than $N = 5$ exhaustive search seems most efficient. However, as sequence length grows, the complexity of the exhaustive search grows exponentially and becomes impractically large even for moderate sequence lengths. Whereas, the complexity of the proposed algorithm grows polynomially with respect to N, is surprisingly faster than exhaustive search, and remains practical even for large sequence lengths, where the candidate vector set of the exhaustive search is incomputably large.

IV. Simulation Results

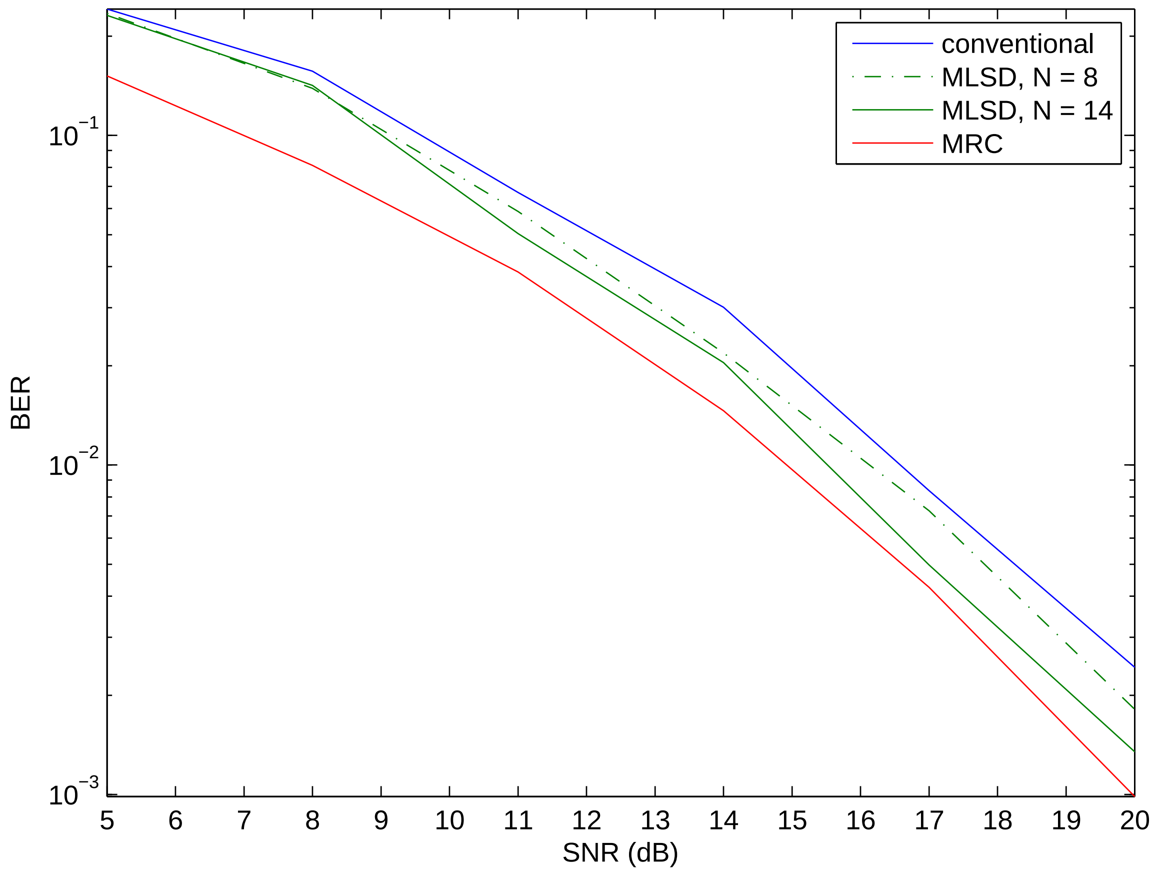

Fig. 5. BER versus SNR for conventional, (proposed) MLSD noncoherent receivers, and MRC.

As an illustration, we consider differentially encoded 8-PSK ($M = 8$) transmission by 1 antenna of a sequence of length

$N = 8, 14$ and reception by $D = 2$ antennas; for the case where coherence detection is made the maximal ratio combining (MRC) method is considered. The signal-to-noise ratio (SNR) ranges from 5dB to 20dB and for each SNR value 10^3 Monte-Carlo simulations are run. In Fig. 5, we present bit error rate (BER) of the conventional (1-lag) differential detector, of the maximum-likelihood sequence detector (MLSD), and of the MRC detector. The noncoherent MLSD is implemented using our proposed algorithm of complexity $\mathcal{O}((NM)^{2D})$. Even for $N = 8$ the candidate set of our proposed algorithm is much smaller than that of the exhaustive search. The reduced set of candidates of the proposed algorithm has size $4480 \approx 2^{12}$ for $N = 8$, compared to the much larger set that needs to be visited by exhaustive search and has size $2^{24} = 16777216$. For $N = 14$ that difference grows with our method reducing the exponential set of 2^{42} to approximately 2^{15} candidate vectors, in which the MLSD vector is included. Also, observe that as block length grows the BER curve of the MLSD algorithm is getting closer to the BER curve of coherent detection. We can conclude that our algorithm truly appears as an efficient noncoherent MLSD method that is applicable to any order of MPSK constellation.

Appendix A

In this appendix, we provide the proof of proposition 2 for arbitrary D, M, and N. For the given matrices $\tilde{\mathbf{Y}}$, $\hat{\mathbf{Y}}$ let $a_d \triangleq \Im\{\hat{y}_{n,d}\}$ and $b_d \triangleq \Re\{\hat{y}_{n,d}\}$, $d = 1, 2, \ldots, D$. Since $\tilde{\mathbf{Y}}_{n,d'} = \Im\{\hat{y}_{n,d}\}$ for $d' = 1, 2, \ldots, D$, and $\tilde{\mathbf{Y}}_{n,d'} = \Re\{\hat{y}_{n,d}\}$ for $d' = D+1, D+2, \ldots, 2D$, $n = 1, 2, \ldots, \frac{MN}{2}$. Moreover, $\Im\{e^{-j\omega}\hat{y}_{n,d}\} = \Im\{\hat{y}_{n,d}\}\cos\omega - \Re\{\hat{y}_{n,d}\}\sin\omega$ and $\Re\{e^{-j\omega}\hat{y}_{n,d}\} = \Re\{\hat{y}_{n,d}\}\cos\omega + \Im\{\hat{y}_{n,d}\}\sin\omega$. Motivated by Section III we define two rotated hypersurfaces in the following manner

$$\begin{bmatrix} \Im\{\hat{y}_{n,1}\} & \ldots & \Im\{\hat{y}_{n,D}\} & \Re\{\hat{y}_{n,1}\} & \ldots & \Re\{\hat{y}_{n,d}\} \\ \Im\{e^{-j\omega}\hat{y}_{n,1}\} & \ldots & \Im\{e^{-j\omega}\hat{y}_{n,D}\} & \Re\{e^{-j\omega}\hat{y}_{n,1}\} & \ldots & \Re\{e^{-j\omega}\hat{y}_{n,D}\} \end{bmatrix} \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 \ldots \cos\phi_{2D-1} \end{bmatrix} = \mathbf{0}_{(2D-1)\times 1} \Leftrightarrow$$

$$\begin{bmatrix} a_1 & \ldots & a_D & b_1 & \ldots & b_D \\ a_1\cos\omega - b_1\sin\omega & \ldots & a_D\cos\omega - b_D\sin\omega & b_1\cos\omega + a_1\sin\omega & \ldots & b_D\cos\omega + a_D\sin\omega \end{bmatrix} \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 \ldots \cos\phi_{2D-1} \end{bmatrix} = \mathbf{0}_{(2D-1)\times 1}. \tag{34}$$

The first row yields the following equation

$$a_1\sin\phi_1 + \ldots + a_D\cos\phi_1 \ldots \cos\phi_{D-2}\sin\phi_{D-1} + b_1\cos\phi_1 \ldots \cos\phi_D\sin\phi_{D+1} + \ldots + b_D\cos\phi_1 \ldots \cos\phi_{2D-1}\cos\phi_{2D-1} = 0 \Leftrightarrow$$

$$a_1\tan\phi_1 + \ldots + a_D\cos\phi_2 \ldots \cos\phi_{D-2}\sin\phi_{D-1} + b_1\cos\phi_2 \ldots \cos\phi_D\sin\phi_{D+1} + \ldots + b_D\cos\phi_2 \ldots \cos\phi_{2D-1}\cos\phi_{2D-1} = 0 \Leftrightarrow$$

$$\tan\phi_1 = -\frac{\begin{bmatrix} a_2 \\ \vdots \\ b_D \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 ... \cos\phi_{2D-1} \end{bmatrix}}{a_1} \tag{35}$$

equivalently from the second row we obtain[2]

$$\tan\phi_1 = -\frac{\begin{bmatrix} a_2\cos\omega - b_2\sin\omega \\ \vdots \\ b_D\cos\omega + a_D\sin\omega \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 ... \cos\phi_{2D-1} \end{bmatrix}}{a_1\cos\omega - b_1\sin\omega}. \tag{36}$$

The two solutions for $\tan\phi_1$ define two different hypersurfaces, if we show that the intersection of those two hypersurfaces is independent of the arbitrary rotation, then proposition 2 will have been proved. Continuingly, the intersection of the two hypersurfaces is given by

$$\frac{\begin{bmatrix} a_2 \\ \vdots \\ b_D \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 ... \cos\phi_{2D-1} \end{bmatrix}}{a_1} = \frac{\begin{bmatrix} a_2\cos\omega - b_2\sin\omega \\ \vdots \\ b_D\cos\omega + a_D\sin\omega \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 ... \cos\phi_{2D-1} \end{bmatrix}}{a_1\cos\omega - b_1\sin\omega} \Leftrightarrow \tag{37}$$

$$\begin{bmatrix} a_2(a_1\cos\omega - b_1\sin\omega) \\ \vdots \\ b_D(a_1\cos\omega - b_1\sin\omega) \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 ... \cos\phi_{2D-1} \end{bmatrix} = \begin{bmatrix} a_1(a_2\cos\omega - b_2\sin\omega) \\ \vdots \\ a_1(b_D\cos\omega + a_D\sin\omega) \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 ... \cos\phi_{2D-1} \end{bmatrix} \Leftrightarrow$$

$$\begin{bmatrix} a_2(a_1\cos\omega - b_1\sin\omega) - a_1(a_2\cos\omega - b_2\sin\omega) \\ \vdots \\ b_D(a_1\cos\omega - b_1\sin\omega) - a_1(b_D\cos\omega + a_D\sin\omega) \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 ... \cos\phi_{2D-1} \end{bmatrix} = 0 \Leftrightarrow$$

$$\begin{bmatrix} a_2a_1\cos\omega - a_1b_1\sin\omega - a_1a_2\cos\omega + a_1b_2\sin\omega \\ \vdots \\ b_Da_1\cos\omega - b_Db_1\sin\omega - a_1b_D\cos\omega - b_Da_D\sin\omega \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 ... \cos\phi_{2D-1} \end{bmatrix} = 0 \Leftrightarrow$$

[2]For the sake of simplicity and space, from now on we shall only include the first and last element of a vector in the equations

$$\begin{bmatrix} \sin\omega(a_1b_2 - a_2b_1) \\ \vdots \\ \sin\omega(-b_1b_D - a_Db_D) \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 \ldots \cos\phi_{2D-1} \end{bmatrix} = 0 \Leftrightarrow$$

$$\sin\omega \begin{bmatrix} a_1b_2 - a_2b_1 \\ \vdots \\ -b_1b_D - a_Db_D \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 \ldots \cos\phi_{2D-1} \end{bmatrix} = 0 \Leftrightarrow$$

$$\begin{bmatrix} a_1b_2 - a_2b_1 \\ \vdots \\ -b_1b_D - a_Db_D \end{bmatrix}^T \begin{bmatrix} \sin\phi_2 \\ \vdots \\ \cos\phi_2 \ldots \cos\phi_{2D-1} \end{bmatrix} = 0. \tag{38}$$

Since we reached the final form in (38) w.l.o.g., we ought to assume that (38) is fully equivalent to (37), and thus the intersection of the two hypersurfaces is independent of the arbitrary rotation $\omega \neq 0$. Thus, proposition 2 holds true.

Appendix B

In this appendix the Matlab code for finding the polynomial-sized reduced set of candidates that includes $\mathbf{s}_{opt}$ is provided. The function complex_rank_M_cand_fast(Y_tilde, Y, M, N) has as input arguments the matrices $\tilde{\mathbf{Y}}_{\frac{MN}{2}\times 2d}$, $\mathbf{Y}_{\frac{N}{\times}d}$, the order M of M-PSK and the sequence lenght N.

```
function S = complex_rank_M_cand_fast(Y_tilde, Y, M, N)

s = exp(j*pi/M*(2*[0:M-1]'+1)); % MPSK constellation

[NM 2D] = size(Y_tilde);

if 2D > 2

    s_reduced = [exp(-j*(pi/M)) ; exp(j*(pi/M)) ; exp(-j*(pi/M+pi)) ; exp(j*(pi/M+pi))];

    s_2_slices = exp(j*pi/M)*[1 ; exp(j*pi)];

    reduced_pairs = reshape(s_reduced, 2, []);

    combs = find_combs1(N, 2D, M);
```

```
S = zeros(N, size(combs, 1));
for i = 1:length(combs)
    J = combs(i,:);
    group = ceil(2*J/M); % generator row
    Y_tilde1 = Y_tilde(J,:); % intersecting HSs
    phi = cartesian_to_spherical(find_intersection(Y_tilde1).');
    c_complex = [sin(phi(1))+j*cos(phi(1))*cos(phi(2))*sin(phi(3)) ;...
                 cos(phi(1))*sin(phi(2))+j*cos(phi(1))*cos(phi(2))*cos(phi(3))];
    %%%% find cadidate %%%
    [dummy, index_s] = max(real([Y(:,1) Y(:,2)]*c_complex*s'), [], 2);
    S(:,i) = s(index_s);
    %%%%%%%%%%%%%%%%%%%%%%
    %%% disambiguate %%%
    group = [group -1];
    for m = 1:2D-1
        k = mod(J(m)-1, M/2);
        rotation = exp(j*(2*k*pi/M));
        Y12_m = [Y(group(m),1) Y(group(m),2)];
        if group(m)- group(m+1) ~= 0 % different row of origin
            s_reduced_m = s_reduced*rotation;
            reduced_pairs_m = reduced_pairs*rotation;
            half_s_reduced_m = s_reduced_m(1:2:end,:);
            [dummy, index_s] = max(real(Y12_m*c_complex*half_s_reduced_m').');
            s_reduced_m = reduced_pairs_m(:,index_s);
            phi_reduced = cartesian_to_spherical(find_intersection(...
             [Y_tilde1([1:m-1 m+1:2D-1],1:2D-1)]).');
             [dummy, index_s] = max(real(Y12_m*[sin(phi_reduced(1))+...
             j*cos(phi_reduced(1))*sin(phi_reduced(2));...
```

```
                cos(phi_reduced(1))*cos(phi_reduced(2))*sin(pi/2)+...
                j*cos(phi_reduced(1))*cos(phi_reduced(2))*cos(pi/2)]*s_reduced_m'));
               S(group(m),i) = s_reduced_m(index_s);
           else
               phi = cartesian_to_spherical(find_intersection(...
                 [Y_tilde1([1:m-1 m+1:2D-1],1:2D-1)]).');
               s_reduced_m = s_2_slices*rotation;
               [dummy, index_s] = max(real(Y12_m*[sin(phi(1))+...
                j*cos(phi(1))*sin(phi(2));...
               cos(phi(1))*cos(phi(2))*sin(pi/2)+...
               j*cos(phi(1))*cos(phi(2))*cos(pi/2)]*s_reduced_m'));
               S(group(m), i) = s_reduced_m(index_s);
               m = m + 1;
           end
       end
   end
   S = [S complex_rank_M_cand_fast(Y_tilde(:, 1:2D-2), Y(:, 1:(2D-2)/2), M)];
else
   phi_crosses = [-pi/2 ; atan(-Y_tilde(:, 2)./Y_tilde(:, 1))];
   [phi_sort, phi_ind] = sort(phi_crosses);
   phi_mid = (phi_sort(1:end-1)+phi_sort(2:end))/2;
   for i = 1:length(phi_mid)
       [dummy, index_s] = max(real(Y*(sin(phi_mid(i))+j*cos(phi_mid(i)))*s').');
       S(:,i) = s(index_s);
   end
end
```

References

[1] D. Divsalar and M. K. Simon, “Multiple-symbol differential detection of MPSK,” *IEEE Trans. Commun.*, vol. 38, pp. 300-308, Mar. 1990.

[2] K. M. Mackenthun, Jr., “A fast algorithm for multiple-symbol differential detection of MPSK,” *IEEE Trans. Commun.*, vol. 42, pp. 1471-1474, Feb./Mar./Apr. 1994.

[3] I. Motedayen, A. Krishnamoorthy, and A. Anastasopoulos, “Optimal joint detection/estimation in fading channels with polynomial complexity,” *IEEE Trans. Inform. Theory*, vol. 53, no. 1, pp. 209 - 223, Jan. 2007.

[4] V. Pauli, L. Lampe, R. Schober, and K. Fukuda, “Multiple-symbol differential detection based on combinatorial geometry,” *IEEE International Conference on Communications (ICC 2007)*, Glasgow, Scotland, Jun. 2007, pp. 827 - 832.

[5] H. Leib, “Data-aided noncoherent demodulation of DPSK,” *IEEE Trans. Commun.*, vol. 43, pp. 722-725 Feb/Mar/Apr 1995.

[6] G. N. Karystinos and A. P. Liavas, “Efficient computation of the binary vector that maximizes a rank-3 quadratic form,” in *Proc. 2006 Allerton Conf. Commun., Control, and Computing*, Allerton House, Monticello, IL, Sept. 2006, pp. 1286-1291.

[7] G. N. Karystinos and D. A. Pados, “Rank-2-optimal adaptive design of binary spreading codes,” *IEEE Trans. Inform. Theory*, vol. 53, pp. 3075-3080, Sept. 2007.

[8] G. N. Karystinos and A. P. Liavas, “Efficient computation of the binary vector that maximizes a rank-deficient quadratic form,” in preparation.

[9] H. Edelsbrunner, J. O’Rouke, and R. Seiel, “Constructing arrangements of lines and hyperplanes with applications,” *SIAM J. Computing*, vol. 15, pp. 341-363, 1986.

[10] D. Avis and K. Fukuda, “Reverse search for enumeration,” *Discrete Applied Mathematics*, vol. 65, pp. 21-46, 1996.

[11] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, siam, 2004.

[12] A. Papoulis and S. U. Pillai *Probability, Random Variables and Stochastic Processes* McGraw Hill, 2001.

[13] Y. Ye, *Interior Point Algorithms: Theory & Analysis*, Wiley, 1997.