Semidefinite Relaxation for Detection of 16-QAM Signaling in MIMO Channels

Betegón García, Miguel[1]
mbg51@alumnos.unican.es

Sierra Menéndez, Sergio[1]
ssm86@alumnos.unican.es

[1] *Master's Degree in Telecommunication Engineering*

Outline

MIMO[1] detection is one of the fundamental problems in state-of-the-art communication systems

- The optimal algorithm is the *maximum likelihood (ML) detector*
- The most common suboptimal detectors are the linear receivers:
  - (I) *Matched filter (MF)*
  - (II) *Zero forcing (ZF)*
  - (III) *Minimum mean-squared error (MMSE)*
- One of the most promising suboptimal detection strategies is the ***semidefinite relaxation (SDR)*** detector

[1] **MIMO** – Multiple-input multiple-output

ML detector is a non convex optimization problem

- SDR is an attempt to approximate it using a convex program that can be efficiently solved in polynomial time

Two approaches for deriving the SDR

Formulate the ML problem in a higher dimension and then relax the nonconvex constraints

The Lagrange bidual of the ML optimization problem, i.e., *the dual program of the dual of the ML problem*

The resulting SDR is a semidefinite program (SDP)

- SDR was proposed for detection of binary/quadratic phase shift keying (BPSK/QPSK) constellations
  - Simulation results show that the SDR provides near-ML performance
  - At high signal-to-noise ratios (SNRs), there is a high probability that SDR will yield the true ML decision
- [1] Extended these works to the detection of other constellations used in digital communications

$$\bar{\mathbf{y}} = \bar{\mathbf{H}}\bar{\mathbf{s}} + \bar{\mathbf{w}}$$

Where:

$\bar{\boldsymbol{y}}$ – Received signal of length N

$\bar{\boldsymbol{H}}$ – $N \times N$ channel matrix

$\bar{\boldsymbol{s}}$ – Length K vector of transmitted symbols

$\bar{\boldsymbol{w}}$ – Length N complex normal zero-mean noise vector with covariance $\sigma^2\mathbf{I}$

N – Number of transmit and receive antennas

In order to avoid the need to handle complex-valued variables, it is customary to use the following decoupled model:

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{w}$$

$$\mathbf{y} = \begin{bmatrix} \Re(\bar{\mathbf{y}}) \\ \Im(\bar{\mathbf{y}}) \end{bmatrix}; \quad \mathbf{s} = \begin{bmatrix} \Re(\bar{\mathbf{s}}) \\ \Im(\bar{\mathbf{s}}) \end{bmatrix}; \quad \mathbf{w} = \begin{bmatrix} \Re(\bar{\mathbf{w}}) \\ \Im(\bar{\mathbf{w}}) \end{bmatrix};$$

$$\mathbf{H} = \begin{bmatrix} \Re(\bar{\mathbf{H}}) & -\Im(\bar{\mathbf{H}}) \\ \Im(\bar{\mathbf{H}}) & \Re(\bar{\mathbf{H}}) \end{bmatrix};$$

$$\text{ML:} \begin{cases} \min_{s} ||\mathbf{y} - \mathbf{Hs}||^2 \\ \text{s.t.} \quad s_i \in \{\pm 1, \pm 3\}, \quad i = 1, \ldots, 2K \end{cases}$$

- It is a combinatorial problem
- It can be solved in a brute-force fashion by searching over all of the 16^K possibilities
- as K increases, this option becomes impractical

1. Replace the finite alphabet constraint by a polynomial constraint. So being $s_i \in \{\pm 1 \pm 3\}$:

$$(s_i + 1)(s_i - 1)(s_i + 3)(s_i - 3) = 0, \qquad i = 1, \ldots, 2K$$

2. Introduce slack variables $t_i = s_i^2$ for $i = 1, \ldots, 2K$ to replace the high-order polynomial constraint by multiple quadratic constraints:

$$\begin{cases} \min\limits_{s,t} ||\mathbf{y} - \mathbf{Hs}||^2 \\ \textbf{s.t.} \quad s_i^2 - t_i = 0, \quad i = 1, \ldots, 2K \\ \qquad\quad t_i^2 - 10t_i + 9 = 0, \quad i = 1, \ldots, 2K \end{cases}$$

3. The constraints must be convexified, reformulating them in a higher dimension and relaxing.
    (I) Replace the vectors $\mathbf{s}$ and $\mathbf{t}$ with a rank-one semidefinite matrix:

$$\mathbf{W} = \mathbf{w}\mathbf{w}^T, \quad \text{where} \quad \mathbf{w}^T = [\mathbf{s}^T \quad \mathbf{t}^T \quad 1]$$

(II) Therefore, problem stated in 2. is equivalent to:

$$\begin{cases} \min\limits_{\mathbf{W}} \mathbf{Tr}\left\{ \mathbf{W} \begin{bmatrix} \mathbf{H}^T\mathbf{H} & \mathbf{0} & -\mathbf{H}^T\mathbf{y} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{y}^T\mathbf{H} & \mathbf{0} & \mathbf{y}^T\mathbf{y} \end{bmatrix} \right. \\ \textbf{s.t.} \quad diag\{\mathbf{W_{1,1}}\} - \mathbf{W_{2,3}} = \mathbf{0} \\ \qquad diag\{\mathbf{W_{2,2}}\} - 10\mathbf{W_{2,3}} + 9\mathbf{1} = \mathbf{0} \\ \qquad \mathbf{W} \succeq 0 \\ \qquad \mathbf{W_{3,3}} = 1 \\ \qquad rank(\mathbf{W}) = 1 \end{cases}$$

(III) The program (II) is not convex because of the rank-one constraint. Dropping this constraint **results in the SDR**:

$$\begin{cases} \min\limits_{\mathbf{W}} \mathbf{Tr}\left\{\mathbf{W}\begin{bmatrix} \mathbf{H}^T\mathbf{H} & \mathbf{0} & -\mathbf{H}^T\mathbf{y} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{y}^T\mathbf{H} & \mathbf{0} & \mathbf{y}^T\mathbf{y} \end{bmatrix}\right. \\ \textbf{s.t.} \quad diag\{\mathbf{W_{1,1}}\} - \mathbf{W_{2,3}} = \mathbf{0} \\ \qquad diag\{\mathbf{W_{2,2}}\} - 10\mathbf{W_{2,3}} + 9\mathbf{1} = \mathbf{0} \\ \qquad \mathbf{W} \succeq 0 \\ \qquad \mathbf{W_{3,3}} = 1 \end{cases}$$

- As the SDR algorithm is an approximation of the ML algorithm, there is a strict relationship between $\mathbf{W}$ and the transmitted symbols.

- According to the standard, there are three techniques for approximation of symbols sent from W. All of then are based on the following quantiz function.

Quantiz function

```
def quantiz(entry, symbols):
    result = np.empty((len(entry),1))
    for i in range(len(entry)):
        minimum = float("inf")
        for val in symbols:
            if abs(val - entry[i]) < minimum:
                result[i,0] = val
                minimum = abs(val - entry[i])
    return result
```

- **Simple quantization**: $\hat{s}_i = quantiz(\mathbf{W}_i; 4N+1)$.
- **Eigenvalue decomposition**: Where $\mathbf{u}$ are the eigenvalues of $\widetilde{\mathbf{W}}$, being $\widetilde{\mathbf{W}}$:

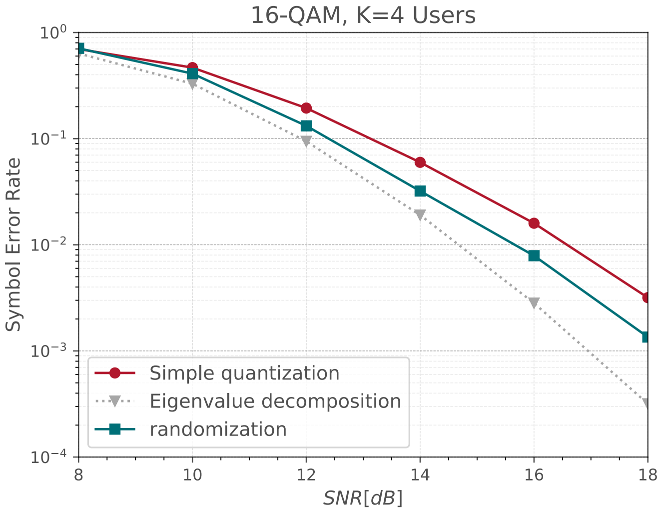
$$\widetilde{\mathbf{W}} = \begin{pmatrix} \mathbf{W_{1,1}} & \mathbf{W_{1,3}} \\ \mathbf{W_{3,1}} & \mathbf{1} \end{pmatrix}$$

$$\hat{s}_i = quantiz\left(\frac{\mathbf{u}_i}{\mathbf{u}_{2K+1}}\right)$$

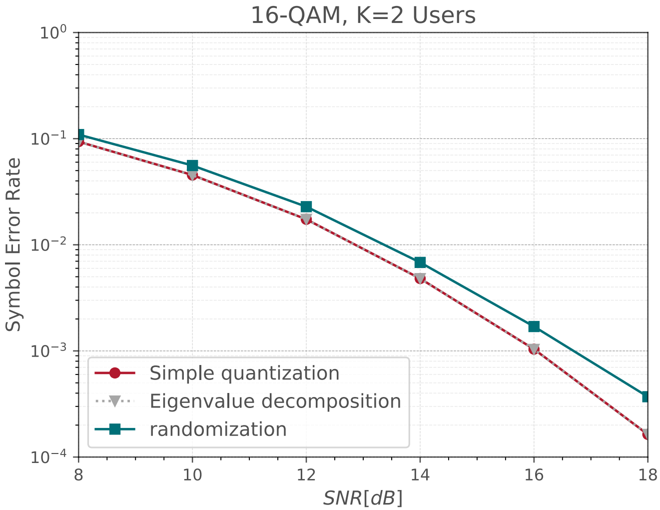
- **Randomization**: It is based on the Cholesky factorization of $\widetilde{\mathbf{W}} = \mathbf{V}^T\mathbf{V}$, being $\mathbf{r}$ a random evenly distributed vector of length 2N+1. We denote the columns of $\mathbf{V}$ by $\mathbf{v}_i$

$$\hat{s}_i = quantiz\left(\frac{\mathbf{v}_i^T\mathbf{r}}{\mathbf{v}_{2K+1}^T\mathbf{r}}\right)$$

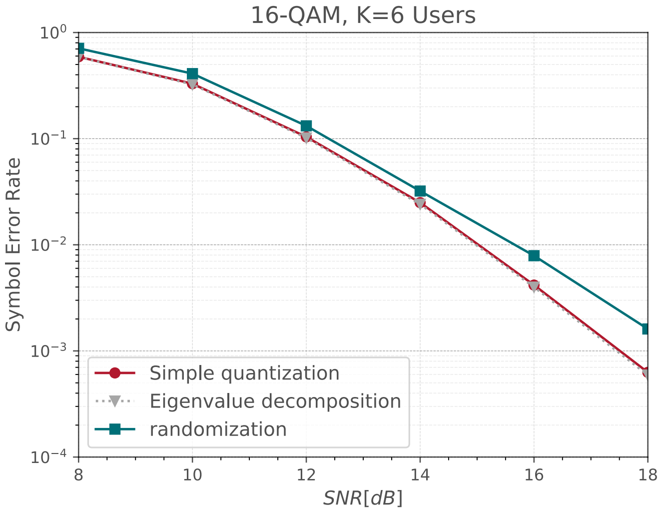
```
Valor de los símbolos de entrada
 [[ 1]
 [-1]
 [-3]
 [ 3]]

Valor simple quantization
 [[ 1.]
 [-1.]
 [-3.]
 [ 3.]]

Valor eigenvalue descomposition
 [[ 1.]
 [-1.]
 [-3.]
 [ 3.]]
Valor randomization
 [[ 1.]
 [-1.]
 [-3.]
 [ 3.]]
```


16-QAM, K=4 Users
Symbol Error Rate
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
8
10
12
14
16
18
SNR[dB]
Simple quantization
Eigenvalue decomposition
randomization

Simulation results III



Simulation results IV



- Our approach shows that other digital constellations, can also be addressed using SDR by formulating the constraints set.
- As we can see in the graphics in some of our results simple quantization and eigenvalue decomposition has similar results in the plots, even being the same with K=2 and K=6.
- We find the computationally efficient SDR detector as a competitive detector in comparison to other suboptimal methods.

A. Wiesel, Y. C. Eldar, S. Shamai
Semidefinite Relaxation for Detection of 16-QAM Signaling in MIMO Channels.
IEEE, Signal Processing Letters, vol. 12, no. 9, pp. 653-656, Sept. 2005.

S. Diamond, S. Boyd
CVXPY: A Python-embedded Modeling Language for Convex Optimization
Journal of Machine Learning Research, vol. 17, no. 83, pp. 1-5, 2016.

References II

M. Betegón, S. Sierra.
16-QAM Detection MIMO, 2019.
https://github.com/betegon/16-QAM-Detection-MIMO

Mosek
Linear, Quadratic, Semidefinite and Mixed Integer problems solver, 2019.
https://www.mosek.com/

Google
Google colaboratory, 2019.
https://colab.research.google.com/