Algorithm 1

Given $k$ samples in the form:

$$y = Ax + e$$

- $y \in \mathbb{R}^n$
- $A \sim \mathcal{N}(0, 1)$, $x \sim \text{unif.}$ ($n = (2-i \cdot M \cdot i : i \in 1, 2, \ldots, M)^t$), $e \sim \mathcal{N}(0, \sigma^2)$
- $x$ is drawn from either an MPAM or BPSK constellation (the source must be hypercubic).
- Receiver only knows statistics of $A$, $x$, and $e$

Problem: Estimate $A$, and recover an estimate of $x$.

1. We cannot estimate $A$ from a single sample of $y$.
2. One-shot transmissions schemes exist but impose performance loss.
3. In a blocking channel, we can estimate $A$ from multiple samples. Let $Y \in \mathbb{R}^{rk}$ and $X \in \mathbb{R}^{nk}$ represent $k$-user uses over a single fading period.

Admissible Transform Matrices (ATMs)

The distribution of the constellation is invariant under sign flips and $k$-user use. This means that we can only hope to recover $A$ up to some factor of $T$, where $T$ is an $n \times n$ matrix that is the product of a permutation matrix and a diagonal matrix with entries $\pm 1$. Such a matrix is termed an Admissible Transform Matrix (ATM), the set of such matrices is denoted as $T$.

Input: An $n \times k$ matrix of received samples.

Output: An estimate of the inverse of the channel gain matrix $U$, and of the transmitted symbols $\hat{X}$.

1. Pick a random starting point which meets the constraints given by (2).
2. Run gradient descent over (1)–(2) to find an optimal value of $U$.
3. Use $U$ to estimate the transmitted symbols as $\hat{X} = (U{Y})$.

Fitting a Parallelepiped

Since the transmitted constellation forms a hypercube, the received values will occupy an $n$-dimensional parallelepiped, the shape of which will be minimally distorted by AWGN. Thus, we formulate the problem of fitting a parallelepiped to our observed samples as an optimization problem. Given a set of $k$ samples of $y$, consider the program:

\[
\begin{align*}
\text{maximize} & \quad \log |U| \\
\text{subject to} & \quad |Uy_i| \leq M + \varepsilon, \quad i = 1, \ldots, k
\end{align*}
\]

\[
(1)
\]

\[
(2)
\]

Algorithm 1

Empirical Success Probability vs. Number of Samples

Theoretical Analysis

As stated, (1)–(2) is a non-convex optimization problem. Empirical results show that gradient descent solves this problem with high probability when the number of samples, $k$, is sufficiently large. We now explain why and when gradient descent works.

Maximal Subset Property

The following property is sufficient for the set of optima of (1)–(2) to contain solutions to our problem:

Definition: A set $S \subseteq [-1, 1]^n$ of size at least $\frac{n}{2}$ has the maximal subset property if there is a subset $\mathcal{S} \subseteq S$ of size at least $\frac{n}{2}$ whose maximum subset is $\mathcal{S}$.

Equivalence Classes of Hadamard Matrices

For $n = 4$, we can express the set of all Hadamard matrices as two equivalence classes, defined as follows:

\[
\begin{align*}
N^H_4 & = \left\{ T \left[ \begin{array}{cccc}
1 & 1 & 1 & 1 \\
1 & 1 & -1 & -1 \\
1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1
\end{array} \right] \right\} \\
N^H_4 & = \left\{ T \left[ \begin{array}{cccc}
1 & 1 & 1 & 1 \\
1 & 1 & -1 & -1 \\
1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1
\end{array} \right] \right\}
\end{align*}
\]

where $T \in T$. An orthogonal matrix, which is not an ATM, maps between these classes. For $n = 4, k = 4$, if $X$ is maximal, then, empirically, the solver has a 50% chance of success. This observation is fully explained by the existence of two equivalence classes of Hadamard matrices. If $X$ is maximal, and has at least one vector from each equivalence class, the only set of optima will be solutions.

Theoretical Success Probability

Let $r(k)$ denote the probability that a collection of $k$ binary vectors, drawn uniformly at random, will be full rank in $\mathbb{F}_2^k$. This expression is given by:

\[
r(k) = 2^{-k} + \sum_{j=0}^{k-1} \binom{k}{j}^2 2^{-j} (1 - 2^{-j-1})^j
\]

\[
(4)
\]

In [1], we derive the success probability for $n = 2, 3, 4$ in terms of $r(k)$. We show:

\[
\mathbb{P}[\text{Success}(n = 2)] = r(4), \quad \mathbb{P}[\text{Success}(n = 3)] = r(9), \quad \mathbb{P}[\text{Success}(n = 4)] = r(16)
\]

\[
(1 - 2^{-k-1})
\]

Sample Size Requirements

The table on the left shows the number of samples required for various values of $n$ and $k$ to recover $U$ in the correct form with 90% success rate using Algorithm 1. The table on the right represents the number of samples needed to ensure a 90% success rate using the technique in [2].

All Optima are Global

The gradient of the objective function is given by:

\[
\nabla \log |U| = (U^{-1})^T
\]

As long as $U$ is finite and non-singular, there are no critical points. Furthermore, we can always find a direction $D$ such that

\[
(U^{-1})^T \cdot D \neq 0.
\]

This implies that the only optima will apply at the points within the constrained region that have the largest $\ell_2$ norm. Since our problem boundary is defined by half planes, gradient descent will converge on these points!

References


Conclusions

- We pose the problem of decoding from an unknown MIMO channel as a non-convex optimization problem that, under proper conditions, is solved by gradient descent. Our approach simultaneously recovers the channel gain matrix.
- Our algorithm requires far fewer samples than previously known blind decoding methods.
- Our algorithm has approximately 3dB loss compared to ML decoding with perfect CSI.
- We outperform the ML with even a small amount of estimation error.
- We initiate the theoretical analysis of the correctness and performance of this algorithm and introduce the maximal subset property as a promising direction towards a formal proof of correctness.

Acknowledgements

T. Dean is supported by the Fannie and John Hertz Foundation. This work was supported in part by the NSF Center for the Science of Information under grant NSF-CCF-0939370.