

Applied Optimization for Wireless, Machine Learning, Big Data
Prof. Aditya K. Jagannatham
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

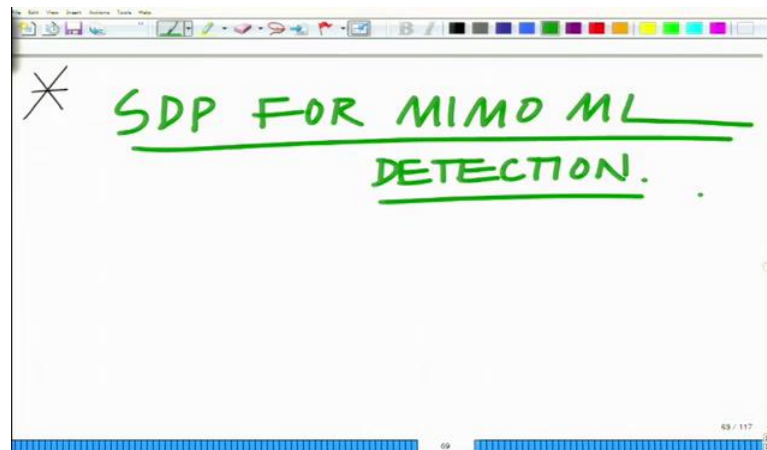
Lecture – 76

Application: SDP for MIMO Maximum Likelihood (ML) Detection

Keywords: *Semi Definite Program, MIMO Maximum Likelihood Detection*

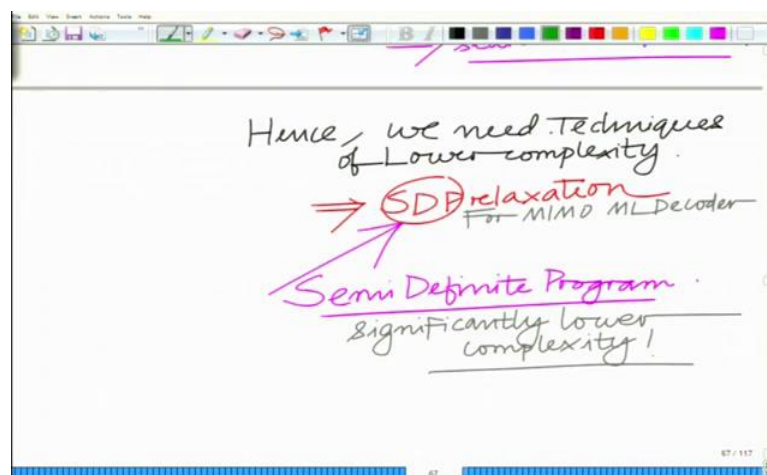
Hello, welcome to another module in this Massive Open Online Course. So we are looking at the SDP that is Semi Definite Programming and its application in the context of MIMO detection that is how to reduce the complexity of the MIMO detector.

(Refer Slide Time: 00:30)



So we are looking at SDP for MIMO ML detection.

(Refer Slide Time: 00:52)



And SDP employs a positive semi definite constraint, that is the linear combination of matrices has to be positive semi definite and this is termed as a linear matrix inequality, LMI.

(Refer Slide Time: 01:01)

Very important & powerful class of problems.

Linear objective generalized inequality

Weighted combination of matrices is PSD.

min $z^T x$

s.t. $Ax = b$

$x_1 F_1 + x_2 F_2 + \dots + x_n F_n + G \succeq 0$

Affine combination of matrices

Linear Matrix Inequality (LMI)

F_1, F_2, \dots, F_n, G matrices.

Linearly weighted combination of matrices

So SDP enforces a linear matrix inequality that is what is novel about SDP.

(Refer Slide Time: 01:38)

argmin $x \in S$

$\|y - Hx\|^2$

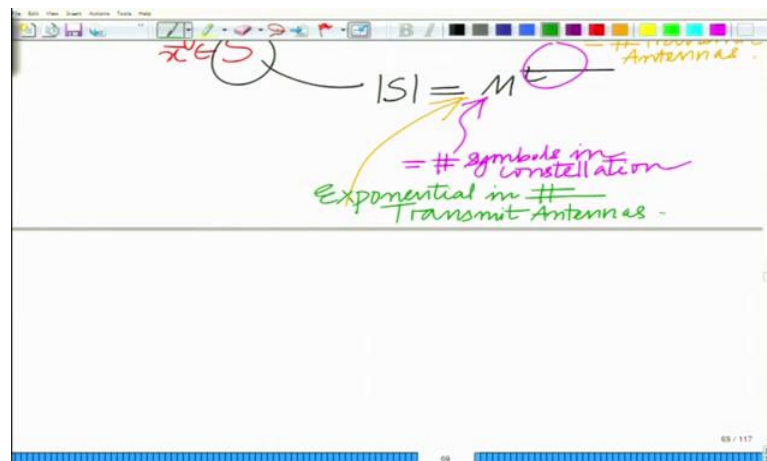
$|S| = M$

$M = \# \text{ Transmit Antennas}$

$M = \# \text{ symbols in constellation}$

We have the MIMO detection problem as $\min_{x \in S} \|y - Hx\|^2$.

(Refer Slide Time: 02:46)



And so basically this is exponential in the number of transmit antennas, which is of very high complexity.

(Refer Slide Time: 03:12)

Handwritten notes on a slide:

- Exponential in $\#$ Transmit Antennas (written in green)
- $\#$ of constellation (written in purple)
- Objective Function (written in red)
- Derivation of the cost function:

$$\begin{aligned}
 & \| \underline{y} - H \underline{x} \|^2 \\
 &= (\underline{y} - H \underline{x})^T (\underline{y} - H \underline{x}) \\
 &= (\underline{y}^T - \underline{x}^T H^T) (\underline{y} - H \underline{x}) \\
 &= \underline{y}^T \underline{y} - \underline{x}^T H^T \underline{y} - \underline{y}^T H \underline{x} + \underline{x}^T H^T H \underline{x}
 \end{aligned}$$

Now the cost function is simplified as shown in slide. So we have the simplified cost function as $\underline{y}^T \underline{y} - \underline{x}^T H^T \underline{y} - \underline{y}^T H \underline{x} + \underline{x}^T H^T H \underline{x}$.

(Refer Slide Time: 04:39)

Handwritten derivation of the normal equations for linear regression. The first line shows the expansion of the cost function derivative:
$$= y^T y - \bar{x}^T H^T \bar{y} - y^T H \bar{x} + \bar{x}^T H^T H \bar{x}$$
 The second line shows the matrix form of the same equation, with the vector $\bar{s} = [\bar{x}^T \ 1]$ and the matrix $L = \begin{bmatrix} H^T H & -H^T \bar{y} \\ -\bar{y}^T H & \bar{y}^T \bar{y} \end{bmatrix}$ and the vector $\begin{bmatrix} \bar{x} \\ 1 \end{bmatrix}$.

This can be written as shown in slide. So I am making a column vector by stacking it along with this number 1. So we have this $t + 1$ dimensional vector, \bar{s} .

(Refer Slide Time: 05:49)

Handwritten definition of the augmented vector \bar{s} and the matrix L . The vector \bar{s} is defined as $\bar{s} = \begin{bmatrix} \bar{x} \\ 1 \end{bmatrix}$, where \bar{x} is a t dimensional vector and the 1 is a scalar, making \bar{s} a $t+1$ dimensional vector. The matrix L is defined as $L = \begin{bmatrix} H^T H & -H^T \bar{y} \\ -\bar{y}^T H & \bar{y}^T \bar{y} \end{bmatrix}$.

Now the matrix L is given as $\begin{bmatrix} H^T H & -H^T \bar{y} \\ -\bar{y}^T H & \bar{y}^T \bar{y} \end{bmatrix}$.

(Refer Slide Time: 07:08)

Handwritten slide content:

$$= \min_{\mathbf{s}} \mathbf{s}^T \mathbf{L} \mathbf{s}$$

$$\mathbf{s} = \begin{bmatrix} x \\ 1 \end{bmatrix}$$

$(s_i)_{1 \leq i \leq t} \in \{\pm 1\}$

So we have written this as $\mathbf{s}^T \mathbf{L} \mathbf{s}$. So this \mathbf{L} can be thought of as a weighting matrix. So let us say this is BPSK constellation.

(Refer Slide Time: 08:59)

Handwritten slide content:

$$|s_i|^2 = 1$$

$$s_i^2 = 1$$

All Diagonal elements = 1

$$\mathbf{s} \mathbf{s}^T = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{t+1} \end{bmatrix} \begin{bmatrix} s_1 & s_2 & \dots & s_{t+1} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}$$

Off Diagonal

Now this proceeds as shown in slide.

(Refer Slide Time: 10:47)

$$\begin{aligned}
 \min \|y - Hz\|^2 &= \min s^T L s \\
 s_i &\in \{\pm 1\} \\
 \text{scalar } s^T L s &= \text{Tr}(s^T L s) \\
 &= \text{Tr}(L s s^T) \\
 &= \text{Tr}(L S)
 \end{aligned}$$

Now the above problem can be equivalently written as $\min_{\substack{s \in \{\pm 1\} \\ 1 \leq i \leq t}} s^T L s$. Now this $s^T L s$ is a

scalar quantity. So I can write this as $s^T L s = \text{Tr}(L S)$ since trace is the sum of the diagonal elements for a square matrix. So a single number is a special case of square matrix. So the trace will yield the number itself.

(Refer Slide Time: 12:30)

$$\begin{aligned}
 S &= s s^T \quad \text{PSD} \\
 \text{diag}(S) &= \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \\
 S &\geq 0 \\
 \text{Equivalent Problem} \\
 \min \quad &\text{Tr}(L S) \\
 \text{s.t.} \quad &\text{diag}(S) = \mathbf{1}
 \end{aligned}$$

Here $S = s s^T$. So the equivalent problem will become

$$\begin{aligned}
 \min \quad &\text{Tr}(L S) \\
 \text{s.t.} \quad &\text{diag}(S) = \mathbf{1} \\
 &S \geq 0 \\
 &S = s s^T
 \end{aligned}$$

(Refer Slide Time: 14:00)

Equivalence Problem

$$\begin{aligned} &S \text{ is PSD} \quad \min. \text{Tr}(LS) \\ &\text{s.t.} \quad \text{diag}(S) = \mathbf{1} \\ &\quad \quad S \geq 0 \\ &\quad \quad S = \mathbf{s}\mathbf{s}^T \end{aligned}$$

Most Difficult: Non-convex

Rank-1 constraint!
Because $S = \mathbf{s}\mathbf{s}^T$

S is a positive semi definite matrix and of all the constraints this is the most difficult non-convex constraint. This is known as a rank-1 constraint because $S = \mathbf{s}\mathbf{s}^T$. So since this is very difficult to impose we simplify this and in this case we simply ignore this. This is known as an SDP relaxation, so we relax it. So this rank-1 constraint makes it non-convex, so it makes it non SDP. So we relax it as an SDP that is we ignore this rank-1 constraint.

(Refer Slide Time: 15:31)

Rank-1 constraint!
Because $S = \mathbf{s}\mathbf{s}^T$

ignore rank 1 constraint
to obtain SDP.

\Rightarrow SDP relaxation!

ML MIMO Decoder

$$\equiv \min. \text{Tr}(LS) \\ \text{s.t.} \quad \text{Diag}(S) = \mathbf{1} \\ \quad \quad S \geq 0$$

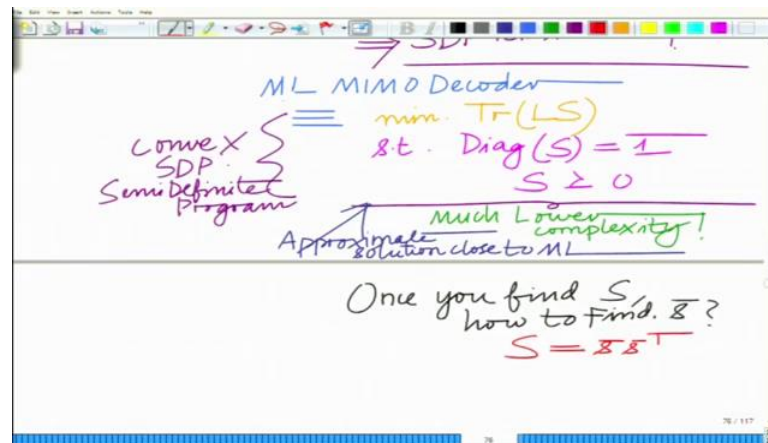
So this is termed as SDP relaxation. So our ML decoder can be equivalently written as,

$$\min \text{Tr}(LS)$$

s.t $\text{diag}(S) = \mathbf{1}$ and this yields an approximate solution.

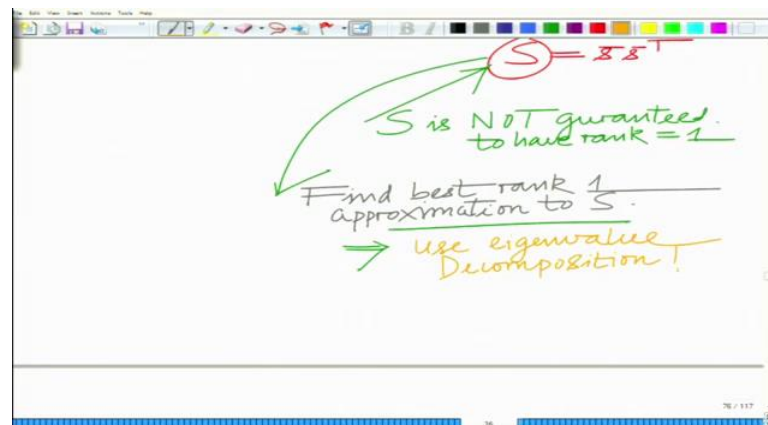
$$S \geq 0$$

(Refer Slide Time: 17:02)



So this is significantly of lower complexity and therefore, it is very amenable to implement this in practice. The only thing is it yields an approximate solution close to the ML solution. Now, once you find S how to find s . So the point is because we have ignored the rank 1 constraint, S is not guaranteed to be $S = ss^T$.

(Refer Slide Time: 19:09)



So in this context we need to find s and the key here is to find best rank 1 approximation to S , for that we use the Eigenvalue decomposition of S .

(Refer Slide Time: 20:23)

$$S = Q \Lambda Q^T$$

matrix of eigenvectors. Diagonal matrix of Eigenvalues.

$$= \sum_{i=1}^{n+1} \lambda_i q_i q_i^T$$

The Eigenvalue decomposition is as follows, we can write S as $S = Q \Lambda Q^T$ where Q is the matrix of eigenvectors and Λ is the diagonal matrix of Eigenvalues and this is then proceeded as shown in slide.

(Refer Slide Time: 21:17)

$$= \sum_{i=1}^{n+1} \lambda_i q_i q_i^T$$

λ_i i^{th} Eigenvalue
 q_i i^{th} eigenvector

$$S = \lambda_1 q_1 q_1^T + \lambda_2 q_2 q_2^T + \dots$$

since S is PSD. Let
 $\Rightarrow \lambda_i \geq 0, \lambda_1 \geq \lambda_2 \geq \lambda_3 \dots$

$$\Rightarrow \lambda_1 = \max \text{Eigenvalue}$$

Now, since S is positive semi definite, note that $\lambda_i \geq 0$. So I can always arrange them in decreasing order.

(Refer Slide Time: 22:42)

$$S = \lambda_1 \bar{q}_1 \bar{q}_1^T + \lambda_2 \bar{q}_2 \bar{q}_2^T + \lambda_3 \bar{q}_3 \bar{q}_3^T + \dots$$

$$= \text{1st Eigenvalue}$$

since S is PSD. Let $\lambda_i \geq 0$. $\lambda_1 \geq \lambda_2 \geq \lambda_3 \dots$

$$\Rightarrow \lambda_1 = \max \text{Eigenvalue}$$

Best rank 1 approximation

Then, the best rank – 1 approximation is simply choose S equal to the largest Eigenvalue.

(Refer Slide Time: 23:11)

$$S \approx \lambda_1 \bar{q}_1 \bar{q}_1^T$$

$$= (\sqrt{\lambda_1} \bar{q}_1)(\sqrt{\lambda_1} \bar{q}_1)^T$$

$$S = \hat{s} \hat{s}^T$$

$$s = \begin{bmatrix} x \\ 1 \end{bmatrix} \Rightarrow \hat{s} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

So this is as shown in slide and the final step is $\hat{s} = \begin{bmatrix} x \\ 1 \end{bmatrix}$, so by choosing the first t symbols of \hat{s} you get transmitted symbols.

(Refer Slide Time: 24:04)

$$= (\sqrt{\lambda_1} q_1) (\sqrt{\lambda_1} q_1)^T$$
$$S = z z^T$$
$$z = \begin{bmatrix} x \\ 1 \end{bmatrix} \Rightarrow \hat{z} = \begin{bmatrix} x \\ 1 \end{bmatrix}$$

By choosing first t symbols in \hat{S} , one can estimate transmit symbols x_i .

So you take the original ML decoder, recast it in a different form and then you relax the rank 1 constraint that makes it a semi definite program, this process is known as SDP relaxation. From the SDP relaxation you get S which is a positive semi definite matrix, from that you perform the Eigenvalue decomposition thereby getting the best rank 1 approximation, so that will be nothing but the principle Eigenvector of S and from that you take the top t symbols. So let us stop here and continue in the subsequent modules. Thank you very much.