Linear Matrix Inequalities in Control

Carsten Scherer
Delft Center for Systems and Control (DCSC)
Delft University of Technology
The Netherlands

Siep Weiland
Department of Electrical Engineering
Eindhoven University of Technology
The Netherlands
Introduction

• Overview: What can be expected from LMI techniques?
• What are LMI’s and what are they good for?
• Example: Truss topology design
• Software
• Some aspects of linear algebra
Merging Control and Optimization

In contrast to classical control, $H_\infty$-synthesis allows to design controllers in an optimal fashion. However the $H_\infty$-paradigm is restricted:

- Performance spec in terms of complete closed-loop transfer-matrix. Sometimes only particular channels are relevant.

- One measure of performance with clear interpretation in frequency domain. Often particular time-domain specs have to be imposed.

- No incorporation of structured time-varying/nonlinear uncertainties.

- Can only design LTI controllers.

View controller as decision variable of optimization problem. Desired specifications are constraints on controlled closed-loop system.
Major Goals for Optimization and Control

- Distinguish easy from difficult problems: **Convexity** is key.
- What are the consequences of convexity in optimization?
- What is **robust optimization**?
- How can we check **robust stability** by convex optimization?
- Which **performance measures** can be incorporated?
- How can **controller synthesis** be convexified?
- What are the limits for the synthesis of **robust controllers**?
- How can we perform systematic **gain-scheduling**?
Linear Matrix Inequalities (LMIs)

An **LMI** is an inequality of the form

\[ F_0 + x_1 F_1 + \cdots + x_n F_n \prec 0 \]

where \( F_0, F_1, \ldots, F_n \) are real symmetric matrices and \( x_1, \ldots, x_n \) are real scalar **unknowns**.

**LMI feasibility problem**: Test whether there exist \( x_1, \ldots, x_n \) that render the LMI satisfied.

**LMI optimization problem**: Minimize \( c_1 x_1 + \cdots + c_n x_n \) over all \( x_1, \ldots, x_n \) that satisfy the LMI.

Only simple cases can be treated analytically \( \rightarrow \) **Numerical techniques**.
Recap

For a real or complex matrix $A$ the inequality $A \prec 0$ means that $A$ is **Hermitian** and **negative definite**.

- $A$ is defined to be Hermitian if $A = A^* = \bar{A}^T$. If $A$ is real then this amounts to $A = A^T$ and $A$ is called symmetric.

  Set of $n \times n$ Hermitian and symmetric matrices: $\mathbb{H}^n$ and $\mathbb{S}^n$.

  All eigenvalues of Hermitian matrices are real.

- Suppose $A$ is Hermitian. By definition $A$ is negative definite if

  $$ x^* A x < 0 \text{ for all complex vectors } x \neq 0. $$

  $A$ is negative definite iff all its eigenvalues are negative.

- $A \prec B, A \preceq B, A \succeq B, A \succ B$ defined/characterized analogously.
Observation: System of LMI’s is LMI

The system of $m$ individual LMI’s

$$F_0^1 + x_1 F_1^1 + \cdots + x_n F_n^1 \prec 0$$

$$\vdots$$

$$F_0^m + x_1 F_1^m + \cdots + x_n F_n^m \prec 0$$

is equivalent to the single LMI

$$\begin{pmatrix} F_0^1 & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & F_0^m \end{pmatrix} + \sum_{k=1}^{n} x_k \begin{pmatrix} F_k^1 & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & F_k^m \end{pmatrix} \prec 0.$$

Assuming single LMI constraint causes no loss of generality.
Good solvers exploit diagonal structure for computational efficiency!
What are they good for?

- Many engineering optimization problem can be easily translated into LMI problems.
- Various computationally difficult optimization problems can be effectively approximated by LMI problems.
- In practice description of data is affected by uncertainty. Robust optimization problems can be either translated or approximated by standard LMI problems.

Essential topic of this course

How to translate/approximate a given (uncertain) optimization problem into/by an LMI problem?
Example: Truss Topology Design

• Connect nodes with $N$ bars of length $l = \text{col}(l_1, \ldots, l_N)$ (fixed) and cross-sections $s = \text{col}(s_1, \ldots, s_N)$ (to-be-designed).

• Impose bounds $a_k \leq s_k \leq b_k$ on cross-section and $l^T s \leq v$ on total volume (weight). Abbreviate $a = \text{col}(a_1, \ldots, a_N), b = \text{col}(b_1, \ldots, b_N)$.

• If applying external forces $f = \text{col}(f_1, \ldots, f_M)$ (fixed) on nodes the construction reacts with the node displacement $d = \text{col}(d_1, \ldots, d_M)$.

  Mechanical model: $A(s)d = f$ where $A(s)$ is the stiffness matrix which depends linearly on $s$ and has to be positive definite.

• Goal is to maximize stiffness what amounts to minimizing the elastic stored energy $f^T d$. 
Example: Truss Topology Design

Find $s \in \mathbb{R}^N$ which maximizes $f^T d$ subject to the constraints

$$A(s) \succeq 0, \quad A(s)d = f, \quad l^T s \leq v, \quad a \leq s \leq b.$$  

Features

- **Data:** Scalar $v$, vectors $f$, $a$, $b$, $l$, and symmetric matrices $A_1, \ldots, A_N$ which define the linear mapping $A(s) = A_1 s_1 + \cdots + A_N s_N$.
- **Decision variables:** Vectors $s$ and $d$.
- **Objective function:** $d \rightarrow f^T d$ which happens to be linear.
- **Constraints:** Semi-definite constraint $A(s) \succeq 0$, nonlinear equality constraint $A(s)d = f$, and linear inequality constraints $l^T s \leq v$, $a \leq s \leq b$. Latter interpreted **elementwise**!
From Truss Topology Design to LMI’s

Render LMI inequality strict. Equality constraint $A(s)d = f$ allows to **eliminate** $d$ which results in

$$\begin{align*}
\text{minimize} & \quad f^T A(s)^{-1} f \\
\text{subject to} & \quad A(s) \succ 0, \ l^T s \leq v, \ a \leq s \leq b.
\end{align*}$$

**Push objective to constraints** with auxiliary variable:

$$\begin{align*}
\text{minimize} & \quad \gamma \\
\text{subject to} & \quad \gamma > f^T A(s)^{-1} f, \ A(s) \succ 0, \ l^T s \leq v, \ a \leq s \leq b.
\end{align*}$$

**Linearize with Schur lemma** to equivalent LMI problem

$$\begin{align*}
\text{minimize} & \quad \gamma \\
\text{subject to} & \quad \begin{pmatrix} \gamma & f^T \\ f & A(s) \end{pmatrix} \succ 0, \ l^T s \leq v, \ a \leq s \leq b.
\end{align*}$$
Congruence Transformations

Given a Hermitian matrix $A$ and a square non-singular matrix $T$, 

$$A \rightarrow T^*AT$$

is called a **congruence transformation** of $A$.

Congruence transformations preserve negative/positive definiteness of a matrix. The following general statement is easy to remember.

If $A$ is Hermitian and $T$ is nonsingular, the matrices $A$ and $T^*AT$ have the **same number** of negative, zero, positive eigenvalues.

What is true if $T$ is not square? ... if $T$ has full column rank?
Schur-Lemma

The Hermitian block matrix \( \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} \) is negative definite if and only if

\[ Q \prec 0 \quad \text{and} \quad R - S^T Q^{-1} S \prec 0 \]

if and only if

\[ R \prec 0 \quad \text{and} \quad Q - S R^{-1} S^T \prec 0. \]

Proof. First equivalence follows from

\[
\begin{pmatrix}
I & 0 \\
-S^T Q^{-1} & I
\end{pmatrix}
\begin{pmatrix}
Q & S \\
S^T & R
\end{pmatrix}
\begin{pmatrix}
I & -Q^{-1} S \\
0 & I
\end{pmatrix}
= \begin{pmatrix}
Q & 0 \\
0 & R - S^T Q^{-1} S
\end{pmatrix}.
\]

Often allows to turn rational matrix inequalities into LMI’s!
Yalmip-Coding: Truss Topology Design

\[
\begin{align*}
\text{minimize} & \quad \gamma \\
\text{subject to} & \quad \begin{pmatrix}
\gamma & f^T \\
f & A(s)
\end{pmatrix} \succeq 0, \quad l^T s \leq v, \quad a \leq s \leq b.
\end{align*}
\]

Suppose \( A(s) = \sum_{k=1}^{N} s_k \mathbf{m}_k \mathbf{m}_k^T \) with vectors \( \mathbf{m}_k \) collected in matrix \( M \).

The following code with Yalmip commands solves LMI problem:

```matlab
gamma = sdpvar(1,1); x = sdpvar(N,1,'full');
lmi = set([gamma f'; f M*diag(x)*M']);
lmi = lmi + set(l'*x<=v);
lmi = lmi + set(a<=x<=b);
options = sdpsettings('solver','csdp');
solvesdp(lmi,gamma,options); s = double(x);
```
Result: Truss Topology Design
Quickly Accessible Software

General purpose Matlab interface Yalmip:

- Free code developed by J. Löfberg and accessible at
  
  http://control.ee.ethz.ch/~joloef/yalmip.msql

- Can use usual Matlab-syntax to define optimization problem.
  Is extremely easy to use and very versatile. Highly recommended!

- Provides access to a whole suite of public and commercial optimization solvers, including fastest available dedicated LMI-solvers.

Matlab LMI-Toolbox for dedicated control applications. Has been recently integrated with updated version of $\mu$-toolbox.
Example: Stability of LTI Systems

The linear time-invariant dynamical system

\[ \dot{x}(t) = Ax(t) \]

is exponentially stable if and only if there exists \( K \) with

\[ K \succ 0 \quad \text{and} \quad A^T K + KA \prec 0. \]

Two inequalities can be combined as

\[
\begin{pmatrix}
-K & 0 \\
0 & A^T K + KA
\end{pmatrix} \prec 0.
\]

Since the left-hand side depends affinely on the matrix variable \( K \), this is indeed a standard strict feasibility test!

**Matrix variables** are fully supported by Yalmip and LMI-toolbox!
General Formulation of LMI Problems

Let $\mathcal{X}$ be a finite-dimensional real vector space. Suppose the mappings

$$c : \mathcal{X} \to \mathbb{R} \quad \text{and} \quad F : \mathcal{X} \to \mathbb{H}^m$$

are affine (constant plus linear).

**LMI feasibility problem**: Test existence of $X \in \mathcal{X}$ with $F(X) \prec 0$.

**LMI optimization problem**: Minimize $c(X)$ over all $X \in \mathcal{X}$ that satisfy the LMI $F(X) \prec 0$.

**Translation to standard form**: Choose basis $X_1, \ldots, X_n$ of $\mathcal{X}$ and parameterize $X = x_1X_1 + \cdots + x_nX_n$. For any affine $f$ infer

$$f \left( \sum_{k=1}^n x_kX_k \right) = f(0) + \sum_{k=1}^n x_k[f(X_k) - f(0)].$$
Diverse Remarks

- The **standard basis** of $\mathbb{R}^{p \times q}$ is $X_{(k,l)}$, $k = 1, \ldots, p$, $l = 1, \ldots, q$, where the only nonzero element of $X_{(k,l)}$ is one at position $(k, l)$.

- General **affine equation** constraint can be routinely eliminated - just recall how we can parameterize the solution set of general affine equations. This might be cumbersome and is **not required in Yalmip**.

- If $F(X)$ is **linear** in $X$, then

  $$ F(X) \prec 0 \text{ implies } F(\alpha X) \prec 0 \text{ for all } \alpha > 0. $$

  With some solvers this might cause numerical trouble. Avoided by normalization or extra constraints.

**Example.** In stability test verify feasibility of following equivalent LMI system: $\text{trace}(K) = 1$, $K \succ 0$ and $A^T K + KA \prec 0$. 

Recap

Let $A$ be a real or complex matrix. The set of its eigenvalues is denoted as $\lambda(A)$. If $A$ has only real eigenvalues then $\lambda_{\text{max}}(A)$ and $\lambda_{\text{min}}(A)$ denote the largest and the smallest eigenvalue of $A$.

**Remember:** If $A = A^*$ then $\lambda_{\text{min}}(A)I \lesssim A \lesssim \lambda_{\text{max}}(A)I$.

The elements of $\sqrt{\lambda(A^*A)}$ are called the singular values of $A$, and $\sigma_{\text{max}}(A)$, $\sigma_{\text{min}}(A)$ denote the largest, smallest singular values. The **spectral norm** of $A$ is defined by

$$\|A\| = \sigma_{\text{max}}(A) = \sqrt{\lambda_{\text{max}}(A^*A)}.$$ 

**Remember:** $\sigma_{\text{min}}(A)^2I \lesssim A^*A \lesssim \sigma_{\text{max}}(A)^2I$. 
Recap

The proofs of these results are very simple. They allow to derive a whole variety of other facts that are required at various points during the course, as those that we already alluded to!

**Example.** If $A \prec 0$, $M = M^*$, $\|M\| < -\lambda_{\text{max}}(A)$ then $A + M \prec 0$.

Two excellent sources for linear algebra:


Lessons to be Learnt

• Many interesting engineering problems are LMI problems.

• Variables can live in arbitrary vector space.

In control: Variables are typically matrices.

Can involve equation and inequality constraints. Just check whether cost function and constraints are affine.

• Translation to standard form is done by parser (Yalmip).

Can choose among many efficient LMI solvers.

• Main trick in removing nonlinearities so far: Schur Lemma.