Model Predictive Control

8. Robust Model Predictive Control

Jun.-Prof. Dr.-Ing. Daniel Görges
Juniorprofessur für Elektromobilität
Technische Universität Kaiserslautern

Paradigms for Robust Control

Robust Control in Frequency Domain

- Frequency domain models based on additive uncertainty, multiplicative uncertainty, etc.
- Stability analysis and control design based on small gain theorem, $\mathcal{H}_\infty$ and $\mathcal{H}_2$ norm, μ-synthesis and DK-iteration, etc.
- Tools are Riccati equations, LMIs, etc.
- Handling parametric uncertainties is intuitive
- Handling dynamic uncertainties is more intuitive
- Handling time-varying uncertainties is not poss.
- Details can be found in [SP05]
- Addressed in Robust Control

Robust Control in Time Domain

- Time domain models based on polytopic uncertainty, norm-bounded uncertainty, etc.
- Stability analysis and control design based on parameter-dependent Lyapunov functions
- Tools are linear matrix inequalities (LMIs)
- Handling parametric uncertainties is intuitive
- Handling dynamic uncertainties is less intuitive
- Handling time-varying uncertainties is possible
- Details can be found in [BEBF94] and [DB01]
- Addressed in this lecture

Linear Time-Varying Systems

- **Discrete-Time Linear Time-Varying (LTV) System**

$$\boldsymbol{x}(k+1) = \boldsymbol{A}(k)\boldsymbol{x}(k) + \boldsymbol{B}(k)\boldsymbol{u}(k) \qquad \text{state equation} \tag{8.1}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) \qquad \text{output equation} \tag{8.2}$$

- **Symbols**

$\boldsymbol{x}(k) \in \mathbb{X} \subseteq \mathbb{R}^n$ state vector — $\boldsymbol{u}(k) \in \mathbb{U} \subseteq \mathbb{R}^m$ input vector

$\boldsymbol{y}(k) \in \mathbb{Y} \subseteq \mathbb{R}^p$ output vector

$\boldsymbol{A}(k) \in \mathbb{R}^{n \times n}$ system matrix — $\boldsymbol{B}(k) \in \mathbb{R}^{n \times m}$ input matrix

$\boldsymbol{C} \in \mathbb{R}^{p \times n}$ output matrix

- **Remarks**
  - The matrices $\boldsymbol{A}(k)$ and $\boldsymbol{B}(k)$ can be time-varying and uncertain or time-varying but known
  - The system $(8.1)/(8.2)$ is also denoted as discrete-time linear parameter-varying (LPV) system
  - The extension for a time-varying output matrix is straightforward

Systems with Polytopic Uncertainty

- **Polytopic Uncertainty**

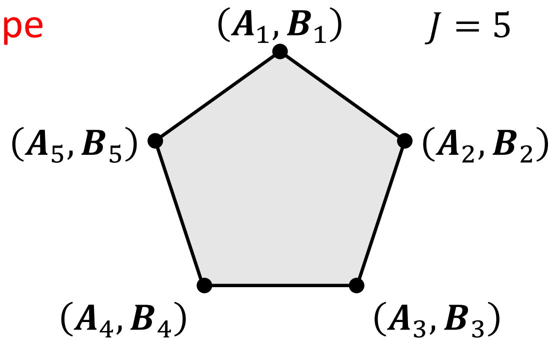
$$\boldsymbol{A}(k) = \sum_{j=1}^{J} \mu_j(k) \boldsymbol{A}_j \tag{8.3}$$

$$\boldsymbol{B}(k) = \sum_{j=1}^{J} \mu_j(k) \boldsymbol{B}_j \tag{8.4}$$

$$\sum_{j=1}^{J} \mu_j(k) = 1 \tag{8.5}$$

$$\mu_j(k) \geq 0 \ \forall j \in \mathbb{J} = \{1, \dots, J\} \tag{8.6}$$

- **Interpretation**
  - The matrices $\boldsymbol{A}_j \in \mathbb{R}^{n \times n}$ and $\boldsymbol{B}_j \in \mathbb{R}^{n \times m}$ are the vertices of a polytope
  - The scalars $\mu_j(k) \in \mathbb{R}$ are uncertain time-varying parameters
  - The condition (8.5) leads to a convex combination
  - The condition (8.5) ensures a “movement” between the vertices
  - The scalars $\mu_j(k)$ can also be time-varying but known parameters



Systems with Polytopic Uncertainty

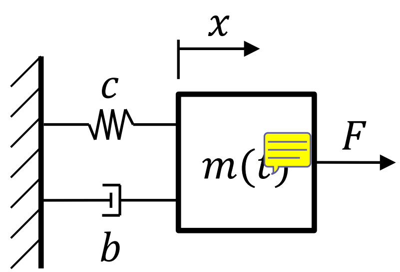
- **Illustrative Example**
  - The equation of motion is given by

    $$m\ddot{x} = F - cx - b\dot{x}$$

  - The state-space model then results as

    $$\underbrace{\begin{pmatrix} \dot{x} \\ \ddot{x} \end{pmatrix}}_{\dot{\boldsymbol{x}}} = \underbrace{\begin{pmatrix} 0 & 1 \\ -\frac{c}{m(t)} & -\frac{b}{m(t)} \end{pmatrix}}_{\boldsymbol{A}_\mathrm{c}(t)} \underbrace{\begin{pmatrix} x \\ \dot{x} \end{pmatrix}}_{\boldsymbol{x}} + \underbrace{\begin{pmatrix} 0 \\ \frac{1}{m(t)} \end{pmatrix}}_{\boldsymbol{B}_\mathrm{c}(t)} \underbrace{F}_{\boldsymbol{u}}$$

    $$\underbrace{y}_{y} = \underbrace{\begin{pmatrix} 1 & 0 \end{pmatrix}}_{\boldsymbol{C}_\mathrm{c}} \underbrace{\begin{pmatrix} x \\ \dot{x} \end{pmatrix}}_{\boldsymbol{x}}$$



$m(t) \in [2\,\mathrm{kg}, 4\,\mathrm{kg}]$

$c = 2\,\frac{\mathrm{N}}{\mathrm{m}}$

$b = 1\,\frac{\mathrm{Ns}}{\mathrm{m}}$

Mass-Spring-Damper System

  - How can we represent this continuous-time LTV system as a discrete-time LTV system $(8.1)/(8.2)$ with polytopic uncertainty $(8.3)/\cdots/(8.6)$?

Systems with Polytopic Uncertainty

- **Illustrative Example**
  - The discretization based on the forward difference for the sampling period $h = 0.5$ s yields

$$\boldsymbol{A}\big(\alpha(k)\big) \approx \boldsymbol{I} + \boldsymbol{A}_{\mathrm{c}}(kh)h = \begin{pmatrix} 1 & h \\ -\frac{ch}{m(kh)} & 1 - \frac{bh}{m(kh)} \end{pmatrix} = \begin{pmatrix} 1 & h \\ -ch\alpha(k) & 1 - bh\alpha(k) \end{pmatrix}$$

$$\boldsymbol{B}\big(\alpha(k), \beta(k)\big) \approx \left(\boldsymbol{I} + \boldsymbol{A}_{\mathrm{c}}(kh)\frac{h}{2}\right) h\boldsymbol{B}_{\mathrm{c}}(kh) = \begin{pmatrix} \frac{h^2}{2m(kh)} \\ \frac{h}{m(kh)} - \frac{bh^2}{2m^2(kh)} \end{pmatrix} = \begin{pmatrix} \frac{h^2}{2}\alpha(k) \\ h\alpha(k) - \frac{bh^2}{2}\beta(k) \end{pmatrix}$$

$$\boldsymbol{C} = \boldsymbol{C}_{\mathrm{c}}$$

with the uncertain time-varying parameters $\alpha(k) = \frac{1}{m(kh)}, \beta(k) = \frac{1}{m^2(kh)}$

  - The uncertain time-varying parameters are characterized by

$$m(kh) \in [2\ \mathrm{kg}, 4\ \mathrm{kg}] \rightarrow \alpha(k) \in \left[\frac{1}{4}\ \mathrm{kg}^{-1}, \frac{1}{2}\ \mathrm{kg}^{-1}\right], \beta(k) \in \left[\frac{1}{16}\ \mathrm{kg}^{-2}, \frac{1}{4}\ \mathrm{kg}^{-2}\right]$$

Systems with Polytopic Uncertainty

- **Illustrative Example**
  - The vertices of the polytope then result for all possible combinations of the bounds of $\alpha(k)$ and $\beta(k)$

$$\boldsymbol{A}_1 = \boldsymbol{A}(1/4) = \begin{pmatrix} 1 & 0.5 \\ -0.25 & 0.875 \end{pmatrix}, \quad \boldsymbol{B}_1 = \boldsymbol{B}(1/4, 1/16) = \begin{pmatrix} 0.0313 \\ 0.1172 \end{pmatrix}$$

$$\boldsymbol{A}_2 = \boldsymbol{A}(1/4) = \begin{pmatrix} 1 & 0.5 \\ -0.25 & 0.875 \end{pmatrix}, \quad \boldsymbol{B}_2 = \boldsymbol{B}(1/4, 1/4) = \begin{pmatrix} 0.0313 \\ 0.0938 \end{pmatrix}$$

$$\boldsymbol{A}_3 = \boldsymbol{A}(1/2) = \begin{pmatrix} 1 & 0.5 \\ -0.5 & 0.75 \end{pmatrix}, \quad \boldsymbol{B}_3 = \boldsymbol{B}(1/2, 1/16) = \begin{pmatrix} 0.0625 \\ 0.2422 \end{pmatrix}$$

$$\boldsymbol{A}_4 = \boldsymbol{A}(1/2) = \begin{pmatrix} 1 & 0.5 \\ -0.5 & 0.75 \end{pmatrix}, \quad \boldsymbol{B}_4 = \boldsymbol{B}(1/2, 1/4) = \begin{pmatrix} 0.0625 \\ 0.2188 \end{pmatrix}$$

Systems with Norm-Bounded Uncertainty

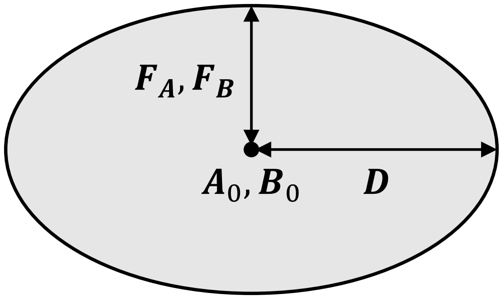
- **Norm-Bounded Uncertainty**

$$\boldsymbol{A}(k) = \boldsymbol{A}_0 + \boldsymbol{D}\boldsymbol{\Delta}(k)\boldsymbol{F}_{\boldsymbol{A}} \quad (8.7)$$

$$\boldsymbol{B}(k) = \boldsymbol{B}_0 + \boldsymbol{D}\boldsymbol{\Delta}(k)\boldsymbol{F}_{\boldsymbol{B}} \quad (8.8)$$

$$\|\boldsymbol{\Delta}(k)\|_2 \leq 1 \quad (8.9)$$

- **Interpretation**
  - The matrices $\boldsymbol{A}_0 \in \mathbb{R}^{n\times n}$ and $\boldsymbol{B}_0 \in \mathbb{R}^{n\times m}$ are constant “nominal” matrices
  - The matrices $\boldsymbol{D} \in \mathbb{R}^{n\times n}$, $\boldsymbol{F}_{\boldsymbol{A}} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{F}_{\boldsymbol{B}} \in \mathbb{R}^{n\times m}$ are constant “structuring” matrices
  - The matrix $\boldsymbol{\Delta}(k) \in \mathbb{R}^{n\times n}$ is an uncertain time-varying parameter
  - $\|\boldsymbol{\Delta}(k)\|_2 = \rho(\boldsymbol{\Delta}^T(k)\boldsymbol{\Delta}(k))$ is the induced 2-norm of the matrix $\boldsymbol{\Delta}(k)$
  - The norm-bound uncertainty can be interpreted as a hyperellipsoid with center $\boldsymbol{A}_0, \boldsymbol{B}_0$ and semi-axes $\boldsymbol{D}$ and $\boldsymbol{F}_{\boldsymbol{A}}, \boldsymbol{F}_{\boldsymbol{B}}$
  - The condition (8.9) ensures a “movement” within the hyperellipsoid



Linear Matrix Inequalities

Definition

Definition 8.1 A linear matrix inequality (LMI) is a matrix inequality of the form

$$\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{F}_0 + \sum_{l=1}^{L} x_l \boldsymbol{F}_l \succ \boldsymbol{0}$$

where the vector $\boldsymbol{x} = (x_1 \quad x_2 \quad \cdots \quad x_n)^T \in \mathbb{R}^n$ is the decision variable and the matrices $\boldsymbol{F}_l = \boldsymbol{F}_l^T \in \mathbb{R}^{n \times n}$ with $l \in \{0, \dots, L\}$ are given coefficients.

- **Remarks**
  - Multiple LMIs $\boldsymbol{F}_1(\boldsymbol{x}) \succ \boldsymbol{0}, \dots, \boldsymbol{F}_M(\boldsymbol{x}) \succ \boldsymbol{0}$ can be written as a single LMI $\operatorname{diag}(\boldsymbol{F}_1(\boldsymbol{x}), \dots, \boldsymbol{F}_M(\boldsymbol{x})) \succ \boldsymbol{0}$
  - LMIs in control are often formulated with matrices as decision variables
  - An example is the Lyapunov inequality $\boldsymbol{F}(\boldsymbol{X}) = \boldsymbol{A}^T \boldsymbol{X} \boldsymbol{A} - \boldsymbol{X} + \boldsymbol{Q} \prec \boldsymbol{0}$ with decision variable $\boldsymbol{X} \in \mathbb{R}^{n \times n}$ and given coefficients $\boldsymbol{A}, \boldsymbol{Q} \in \mathbb{R}^{n \times n}$ (cf. Corollary 2.1)
  - An LMI $\boldsymbol{F}(\boldsymbol{X}) \succ \boldsymbol{0}$ can be transformed into an LMI $\boldsymbol{F}(\boldsymbol{x}) \succ \boldsymbol{0}$ by constructing the vector $\boldsymbol{x}$ through "stacking" the columns of the matrix $\boldsymbol{X}$ (cf. [SW04, Remark 1.24] for details)

LMI Problems

Problem 8.1 Find a vector $\boldsymbol{x} \in \mathbb{R}^n$ such that the LMI

$$\boldsymbol{F}(\boldsymbol{x}) \succ \boldsymbol{0}$$

is feasible. This problem is denoted as LMI feasibility problem.

Problem 8.2 Solve the optimization problem

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) \ \text{ subject to } \ \boldsymbol{F}(\boldsymbol{x}) \succ \boldsymbol{0}$$

with the convex cost function $f: \mathbb{R}^n \to \mathbb{R}$. This problem is denoted as LMI optimization problem.

- **Remarks**
  - An LMI feasibility problem can be written as an LMI optimization problem with an arbitrary cost fcn.
  - An LMI optimization problem is a convex optimization problem since $\boldsymbol{F}(\boldsymbol{x}) \succ \boldsymbol{0}$ defines a convex set
  - LMI optimization problems can be solved with polynomial complexity using interior point methods
  - More details on LMIs can be found in [BEBF94], [SW04], and [SP05, Chapter 12]

Tricks in LMI Problems

Lemma 8.1 The following statements are equivalent:

(1) $\begin{pmatrix} \boldsymbol{Q} & \boldsymbol{S} \\ \boldsymbol{S}^T & \boldsymbol{R} \end{pmatrix} \succ \boldsymbol{0}$

(2) $\boldsymbol{R} \succ \boldsymbol{0},\ \boldsymbol{Q} - \boldsymbol{S}\boldsymbol{R}^{-1}\boldsymbol{S}^T \succ \boldsymbol{0}$

This equivalence is denoted as Schur complement.

Lemma 8.2 If $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$ is a positive definite matrix, then $\boldsymbol{W}^T\boldsymbol{Q}\boldsymbol{W}$ with $\boldsymbol{W} \in \mathbb{R}^{n\times n}$ full rank is also a positive definite matrix. This transformation is denoted as congruence transformation. A congruence transformation does in particular not change the number of positive and negative eigenvalues.

- **Remarks**
  - The tricks are very helpful for transforming non-LMI problems into LMI problems
  - E.g. the congruence transformation is very useful for “removing” bilinear terms
  - More tricks are given in [SP05, Section 12.3]

Tools for LMI Problems

- **Open-Source Tools**
  - YALMIP can be utilized for formulating LMIs in MATLAB
    yalmip.github.io
  - SeDuMi can be utilized with YALMIP for solving LMIs in MATLAB
    sedumi.ie.lehigh.edu
  - SDPT3 can be utilized with YALMIP for solving LMIs in MATLAB
    www.math.nus.edu.sg/~mattohkc/sdpt3.html
- **Commercial Tools**
  - LMI Lab in the Robust Control Toolbox can be utilized for formulating and solving LMIs in MATLAB
- **Remark**
  - Sometimes numerical problems occur when solving LMI problems
  - Trying different solvers should then be considered

Robust Stability Condition

Theorem 8.1 The discrete-time linear time-varying system (8.1) with polytopic uncertainty (8.3)/ ⋯/(8.6) is globally asymptotically stable if there exist matrices $\boldsymbol{P}_j = \boldsymbol{P}_j^T \succ \boldsymbol{0}$ with $j \in \mathbb{J}$ such that

$$\boldsymbol{A}_j^T \boldsymbol{P}_i \boldsymbol{A}_j - \boldsymbol{P}_j \prec \boldsymbol{0} \;\; \forall (j, i) \in \mathbb{J} \times \mathbb{J}. \tag{8.10}$$

The quadratic function

$$V(\boldsymbol{x}(k), k) = \boldsymbol{x}^T(k)\boldsymbol{P}(k)\boldsymbol{x}(k) \quad \text{with} \quad \boldsymbol{P}(k) = \textstyle\sum_{j=1}^{J} \mu_j(k)\boldsymbol{P}_j, \sum_{j=1}^{J} \mu_j(k) = 1, \mu_j(k) \geq 0 \; \forall j \in \mathbb{J}$$

is then a parameter-dependent Lyapunov function for the discrete-time linear time-varying system (8.1).

- **Proof**
  - The function $V(\boldsymbol{x}(k), k)$ is positive definite, descrecent and radially unbounded since

    $\alpha_1 \|\boldsymbol{x}(k)\|_2^2 \leq V(\boldsymbol{x}(k), k) \;\; \forall \boldsymbol{x}(k) \in \mathbb{R}^n \;\; \forall k \in \mathbb{N}_0$ with $\alpha_1 = \varepsilon > 0$, cf. Lemma 2.1

    $V(\boldsymbol{x}(k), k) \leq \alpha_2 \|\boldsymbol{x}(k)\|_2^2 \;\; \forall \boldsymbol{x}(k) \in \mathbb{R}^n \;\; \forall k \in \mathbb{N}_0$ with $\alpha_2 = \sum_{j=1}^{J} \lambda_{\max}(\boldsymbol{P}_j) > 0$, cf. Lemma 2.1

    $\alpha_1 \|\boldsymbol{x}(k)\|_2^2 \rightarrow \infty$ as $\|\boldsymbol{x}(k)\|_2 \rightarrow \infty$

Robust Stability Condition

- **Proof**
  - We must still prove when $\Delta V(\boldsymbol{x}(k), k)$ along trajectories of the discrete-time LTV system (8.1), i.e.

    $$\Delta V(\boldsymbol{x}(k), k) = V(\boldsymbol{x}(k+1), k+1) - V(\boldsymbol{x}(k), k) = \boldsymbol{x}^T(k+1)\boldsymbol{P}(k+1)\boldsymbol{x}(k+1) - \boldsymbol{x}^T(k)\boldsymbol{P}(k)\boldsymbol{x}(k)$$
    $$= \boldsymbol{x}^T(k)\boldsymbol{A}^T(k)\boldsymbol{P}(k+1)\boldsymbol{A}(k)\boldsymbol{x}(k) - \boldsymbol{x}^T(k)\boldsymbol{P}(k)\boldsymbol{x}(k) = \boldsymbol{x}^T(k)(\boldsymbol{A}^T(k)\boldsymbol{P}(k+1)\boldsymbol{A}(k) - \boldsymbol{P}(k))\boldsymbol{x}(k),$$

    is negative definite
  - Assume that (8.10) is fulfilled
  - Rearranging (8.10) yields

    $$\boldsymbol{P}_j - \boldsymbol{A}_j^T\boldsymbol{P}_i\boldsymbol{P}_i^{-1}\boldsymbol{P}_i\boldsymbol{A}_j \succ \boldsymbol{0}$$
  - Applying the Schur complement leads to

    $$\begin{pmatrix} \boldsymbol{P}_j & \boldsymbol{A}_j^T\boldsymbol{P}_i \\ \boldsymbol{P}_i\boldsymbol{A}_j & \boldsymbol{P}_i \end{pmatrix} \succ \boldsymbol{0}$$

Robust Stability Condition

- **Proof**
  - Multiplying by $\mu_i(k+1)$ and summing over $i = 1,2,\dots,J$ results in

$$\begin{pmatrix} \sum_{i=1}^{J}\mu_i(k+1)\boldsymbol{P}_j & \sum_{i=1}^{J}\mu_i(k+1)\boldsymbol{A}_j^T\boldsymbol{P}_i \\ \sum_{i=1}^{J}\mu_i(k+1)\boldsymbol{P}_i\,\boldsymbol{A}_j & \sum_{i=1}^{J}\mu_i(k+1)\,\boldsymbol{P}_i \end{pmatrix} = \begin{pmatrix} \boldsymbol{P}_j\sum_{i=1}^{J}\mu_i(k+1) & \boldsymbol{A}_j^T\sum_{i=1}^{J}\mu_i(k+1)\,\boldsymbol{P}_i \\ \sum_{i=1}^{J}\mu_i(k+1)\boldsymbol{P}_i\boldsymbol{A}_j & \sum_{i=1}^{J}\mu_i(k+1)\,\boldsymbol{P}_i \end{pmatrix} =$$

$$\begin{pmatrix} \boldsymbol{P}_j & \boldsymbol{A}_j^T\boldsymbol{P}(k+1) \\ \boldsymbol{P}(k+1)\boldsymbol{A}_j & \boldsymbol{P}(k+1) \end{pmatrix} \succ \boldsymbol{0}$$

  - Multiplying by $\mu_j(k)$ and summing over $j = 1,2,\dots,J$ results in

$$\begin{pmatrix} \sum_{j=1}^{J}\mu_j(k)\boldsymbol{P}_j & \sum_{j=1}^{J}\mu_j(k)\boldsymbol{A}_j^T\boldsymbol{P}(k+1) \\ \sum_{j=1}^{J}\mu_j(k)\boldsymbol{P}(k+1)\,\boldsymbol{A}_j & \sum_{j=1}^{J}\mu_j(k)\boldsymbol{P}(k+1) \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^{J}\mu_j(k)\boldsymbol{P}_j & \sum_{j=1}^{J}\mu_j(k)\boldsymbol{A}_j^T\boldsymbol{P}(k+1) \\ \boldsymbol{P}(k+1)\sum_{j=1}^{J}\mu_j(k)\boldsymbol{A}_j & \boldsymbol{P}(k+1)\sum_{j=1}^{J}\mu_j(k) \end{pmatrix} =$$

$$\begin{pmatrix} \boldsymbol{P}(k) & \boldsymbol{A}^T(k)\boldsymbol{P}(k+1) \\ \boldsymbol{P}(k+1)\boldsymbol{A}(k) & \boldsymbol{P}(k+1) \end{pmatrix} \succ \boldsymbol{0}$$

Robust Stability

Robust Stability Condition

- **Proof**
  - Applying the Schur complement leads to

    $$\boldsymbol{P}(k) - \boldsymbol{A}^T(k)\boldsymbol{P}(k+1)\boldsymbol{P}^{-1}(k+1)\boldsymbol{P}(k+1)\boldsymbol{A}(k) = \boldsymbol{P}(k) - \boldsymbol{A}^T(k)\boldsymbol{P}(k+1)\boldsymbol{A}(k) \succ \boldsymbol{0}$$

  - Rearranging yields

    $$\boldsymbol{A}^T(k)\boldsymbol{P}(k+1)\boldsymbol{A}(k) - \boldsymbol{P}(k) \prec \boldsymbol{0}$$

  - This implies that $\Delta V(\boldsymbol{x}(k), k)$ is negative definite
  - This completes the proof
- **Remarks**
  - The robust stability condition (8.10) is only sufficient
  - This means that the discrete-time LTV system (8.1) may be globally asymptotically stable although the robust stability condition (8.10) is not fulfilled, i.e. the robust stability condition may “fail”
  - The “fail rate” of a stability condition is denoted as conservatism

Robust Stability Condition

- **Remarks**
  - Optionally a common Lyapunov function $V(\boldsymbol{x}(k), k) = \boldsymbol{x}^T(k)\boldsymbol{P}\boldsymbol{x}(k), \boldsymbol{P} = \boldsymbol{P}^T \succ \boldsymbol{0}$ can be considered
  - The robust stability condition (8.10) then becomes

$$\boldsymbol{A}_j^T \boldsymbol{P} \boldsymbol{A}_j - \boldsymbol{P} \prec \boldsymbol{0} \quad \forall j \in \mathbb{J} \tag{8.11}$$

  - The robust stability condition (8.11) has a smaller number of LMIs but also a higher conservatism than the robust stability condition (8.10)

Corollary 8.1 The discrete-time linear time-varying system (8.1) with polytopic uncertainty $(8.3)/\cdots/(8.6)$ is globally asymptotically stable if there exist matrices $\boldsymbol{P}_j = \boldsymbol{P}_j^T \succ \boldsymbol{0}$ with $j \in \mathbb{J}$ such that the LMIs

$$\begin{pmatrix} \boldsymbol{P}_j & \boldsymbol{A}_j^T \boldsymbol{P}_i \\ \boldsymbol{P}_i \boldsymbol{A}_j & \boldsymbol{P}_i \end{pmatrix} \succ \boldsymbol{0} \tag{8.12}$$

are feasible for all $(j, i) \in \mathbb{J} \times \mathbb{J}$.

Robust Stability Condition

- **Illustrative Example**
  - Reconsider the Illustrative Example (Mass-Spring-Damper System) from Slide 8-5ff
  - From Corollary 8.1 we obtain an LMI feasibility problem with four matrix variables $\boldsymbol{P}_j = \boldsymbol{P}_j^T \in \mathbb{R}^{2\times 2}$ with $j \in \mathbb{J} = \{1, \dots, 4\}$, two LMIs resulting from $\boldsymbol{P}_j \succ \mathbf{0}$, and four LMIs resulting from (8.12)
  - A feasible solution can be found under MATLAB using YALMIP and SeDuMi in 0.14 s

Robust State Feedback Control

- **Assumptions**
  - No constraints ($\mathbb{X} = \mathbb{R}^n, \mathbb{U} = \mathbb{R}^m, \mathbb{Y} = \mathbb{R}^p$)
  - State feedback ($\boldsymbol{C} = \boldsymbol{I}_{n\times n}$)
  - Regulation of the state to the origin ($\boldsymbol{x}(k) \to \mathbf{0}$ as $k \to \infty$)

Theorem 8.2 The discrete-time linear time-varying system (8.1) with polytopic uncertainty $(8.3)/\cdots/(8.6)$ under the state feedback control law $\boldsymbol{u}(k) = \boldsymbol{K}\boldsymbol{x}(k)$ is globally asymptotically stable if there exist matrices $\boldsymbol{Q}_j = \boldsymbol{Q}_j^T \succ \mathbf{0}$ with $j \in \mathbb{J}$ and matrices $\boldsymbol{G}, \boldsymbol{Y}$ such that the LMIs

$$\begin{pmatrix} \boldsymbol{G} + \boldsymbol{G}^T - \boldsymbol{Q}_j & \boldsymbol{G}^T\boldsymbol{A}_j^T + \boldsymbol{Y}^T\boldsymbol{B}_j^T \\ \boldsymbol{A}_j\boldsymbol{G} + \boldsymbol{B}_j\boldsymbol{Y} & \boldsymbol{Q}_i \end{pmatrix} \succ \mathbf{0} \tag{8.13}$$

are feasible for all $(j, i) \in \mathbb{J} \times \mathbb{J}$. The feedback matrix is then given by $\boldsymbol{K} = \boldsymbol{Y}\boldsymbol{G}^{-1}$.

- **Proof**
  - The proof is similar to the proof of Theorem 8.1. Details are given in [Mao03, Proof of Theorem 1]

Robust State Feedback Control

- **Exercise**
  - Consider the uncertain mass-spring damper system introduced on Slide 8-5ff
  - Design a robust state feedback controller based on Theorem 8.2 under MATLAB using YALMIP
  - Simulate the closed-loop system under MATLAB for
    - the vertices of the polytope $\boldsymbol{A}_j$ and $\boldsymbol{B}_j$ with $j \in \mathbb{J} = \{1, \dots, 4\}$
    - hundred random parameters $\alpha(k) \in \left[\frac{1}{4}\ \mathrm{kg}^{-1}, \frac{1}{2}\ \mathrm{kg}^{-1}\right]$ and $\beta(k) \in \left[\frac{1}{16}\ \mathrm{kg}^{-2}, \frac{1}{4}\ \mathrm{kg}^{-2}\right]$

    over the discrete times $k \in \{0, \dots, 20\}$ and for the initial state $\boldsymbol{x}_0 = \begin{pmatrix} 1\ \mathrm{m} & 0\,\frac{\mathrm{m}}{\mathrm{s}} \end{pmatrix}^T$
  - Visualize the closed-loop state sequences under MATLAB in a single diagram
- **Hints**
  - Simulations of discrete-time systems can be realized in MALTAB using a for-loop
  - Uniformly distributed random numbers between 0 and 1 can be generated in MATLAB with `rand`

Robust Model Predictive Control

- **Robust Model Predictive Control based on LMIs**
  - Relies on the LMI concepts introduced on the previous slides
  - [KBM96] state an LMI optimization problem based on a common Lyapunov function
  - [CGM02], [Mao03] state an LMI opt. problem based on a parameter-dependent Lyapunov function
  - [WK03] extend the concept from [KBM96] to explicit model predictive control
  - [Mac02, Section 8.4] and [CB04, Section 8.4] provide very good introductions
- **Robust Model Predictive Control based on Min-Max Optimization**
  - [BBM15, Chapter 16] provide a very good introduction
- **Robust Model Predictive Control based on Tubes**
  - [RM09, Sections 3.4 and 3.5] provide a very good introduction

Literature

Robust Control in Frequency Domain

[SP05] Sigurd Skogestad and Ian Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. John Wiley & Sons, Chichester, 2nd edition, 2005. – EIT 910/088, L EIT 18

Robust Control in Time Domain

[BEFB94] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1994. – MAT Line, web.stanford.edu/~boyd/lmibook/

[DB01] Jamal Daafouz and Jacques Bernussou. Parameter dependent Lyapunov functions for discrete time systems with time varying parametric uncertainties. *Systems & Control Letters*, 43(5):355–359, 2001.

[DRI01] Jamal Daafouz, Pierre Riedinger, and Claude Iung. Stability analysis and control synthesis for switched systems: A switched Lyapunov function approach. *IEEE Transactions on Automatic Control*, 47(11):1883–1887,2002.

Literature

Robust Control in Time Domain

[SW04] Carsten Scherer and Siep Weiland. *Linear Matrix Inequalities in Control*. DISC Lecture Notes, Delft, 2004. – www.dcsc.tudelft.nl/~cscherer/lmi/notes05.pdf

Robust Model Predictive Control

[CGM02] Francesco A. Cuzzola, Jose C. Geromel, and Manfred Morari. An improved approach for constrained robust model predictive control. *Automatica*, 38(7):1183–1189, 2002.

[KBM96] Mayuresh V. Kothare, Venkataramanan Balakrishnan, and Manfred Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.

[Mao03] Wei-Jie Mao. Robust stabilization of uncertain time-varying discrete systems and comments on “An improved approach for constrained robust model predictive control”. *Automatica*, 39(6):1109–1112, 2003.

[WK03] Zhaoyang Wan and Mayuresh V. Kothare. An efficient off-line formulation of robust model predictive control using linear matrix inequalities. *Automatica*, 39(5):837–846, 2003.