LMI-based Robust MPC Design

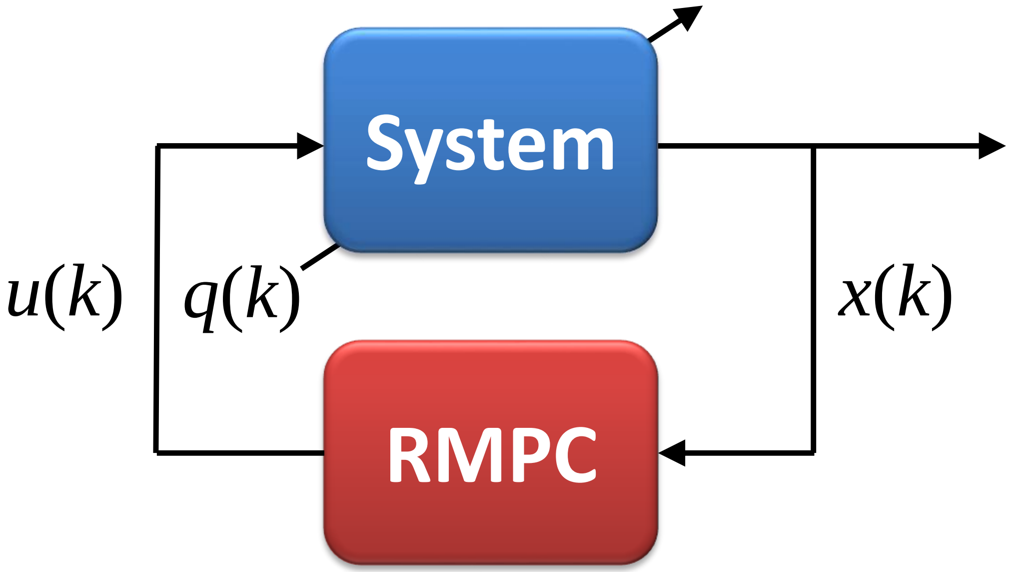
Exercise on Robust MPC Design

Juraj Oravec – Monika Bakošová

STU

Slovak University of Technology in Bratislava

Task



Task

$$x(k+1) = A\,x(k) + B\,u(k), \qquad x(0) = [0.05, 0]^\top,$$
$$y(k) = C\,x(k), \qquad u \in \mathbb{U},$$
$$[A, B] \in \text{convhull}\{[A_v, B_v]\}, \qquad y \in \mathbb{Y},$$

$$A = \begin{bmatrix} 1 & 0.1 \\ 0 & 0.495 \pm 0.495 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ -0.1 \end{bmatrix} \qquad C = \begin{bmatrix} 0 & 4.95 \end{bmatrix}$$

$$\mathbb{U} = \{u : \|u\| \leq \|2\|\} \qquad W_{\mathrm{u}} = 2 \times 10^{-5}$$

$$\mathbb{Y} = \{y : \|y\| \preceq \|[1, 1.5]^\top\|\} \qquad W_{\mathrm{x}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Kothare et al., Automatica, 1996

Task

$$x(k+1) = A\,x(k) + B\,u(k), \qquad x(0) = [0.05, 0]^\top,$$
$$y(k) = C\,x(k), \qquad u \in \mathbb{U},$$
$$[A, B] \in \text{convhull}\{[A_v, B_v]\}, \qquad y \in \mathbb{Y},$$

$$A = \begin{bmatrix} 1 & 0.1 \\ 0 & 0.495 \pm 0.495 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ -0.1 \end{bmatrix} \qquad C = \begin{bmatrix} 0 & 4.95 \end{bmatrix}$$

$$\mathbb{U} = \{u : \|u\| \leq \|2\|\} \qquad W_{\mathrm{u}} = 2 \times 10^{-5}$$

$$\mathbb{Y} = \{y : \|y\| \preceq \|[1, 1.5]^\top\|\} \qquad W_{\mathrm{x}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

```
%% Load benchmark
benchmark_kothare
```

SDP

$$\min \gamma$$

s.t. :

$$\begin{bmatrix} 1 & \star \\ x & X \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} X & \star & \star & \star \\ A_v X + B_v Y & X & \star & \star \\ \sqrt{W_{\mathrm{x}}} X & 0 & \gamma I & \star \\ \sqrt{W_{\mathrm{u}}} Y & 0 & 0 & \gamma I \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} u_{\max}^2 I & \star \\ Y & X \end{bmatrix} \succeq 0 \qquad \begin{bmatrix} U & \star \\ Y & X \end{bmatrix} \succeq 0, \;\; U_{i,i} \leq u_{\max,i}^2$$

$$\begin{bmatrix} x_{\max}^2 I & \star \\ C(A_v X + B_v Y) & X \end{bmatrix} \succeq 0$$

SDP - given params

$$\min \gamma$$

s.t. :

$$\begin{bmatrix} 1 & \star \\ x & X \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} X & \star & \star & \star \\ A_v X + B_v Y & X & \star & \star \\ \sqrt{W_{\mathrm{x}}} X & 0 & \gamma I & \star \\ \sqrt{W_{\mathrm{u}}} Y & 0 & 0 & \gamma I \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} u_{\max}^2 I & \star \\ Y & X \end{bmatrix} \succeq 0 \qquad \begin{bmatrix} U & \star \\ Y & X \end{bmatrix} \succeq 0, \; U_{i,i} \leq u_{\max,i}^2$$

$$\begin{bmatrix} x_{\max}^2 I & \star \\ C(A_v X + B_v Y) & X \end{bmatrix} \succeq 0$$

SDP - optimizers

$$\min \gamma$$

s.t. :

$$\begin{bmatrix} 1 & \star \\ x & X \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} X & \star & \star & \star \\ A_v X + B_v Y & X & \star & \star \\ \sqrt{W_{\mathrm{x}}} X & 0 & \gamma I & \star \\ \sqrt{W_{\mathrm{u}}} Y & 0 & 0 & \gamma I \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} u_{\max}^2 I & \star \\ Y & X \end{bmatrix} \succeq 0 \qquad \begin{bmatrix} U & \star \\ Y & X \end{bmatrix} \succeq 0, \;\; U_{i,i} \leq u_{\max,i}^2$$

$$\begin{bmatrix} x_{\max}^2 I & \star \\ C(A_v X + B_v Y) & X \end{bmatrix} \succeq 0$$

SDP - optimizers

```
%% Optimizers
gamma = sdpvar(1);
   xk = sdpvar(nx,1);
    X = sdpvar(nx);
    Y = sdpvar(nu,nx);
    U = sdpvar(nu,nu);
```

SDP - aux params

```
%% Auxiliary parameters
ZERO = 1e-6; % ZERO-Tolerance of strict LMIs
ZEROx = zeros(nx,nx);
ZEROux = zeros(nu,nx);
ZEROxu = zeros(nx,nu);
Ix = eye(nx);
Iu = eye(nu);
```

Objective function

$$\min \gamma$$

```
%% Objective function
obj = gamma
```

Objective function

$$\min \gamma + \mathrm{tr}(X)$$

```
%% Objective function
obj = gamma + trace(X)
```

LMIs

$$\begin{bmatrix} 1 & \star \\ x & X \end{bmatrix} \succeq 0, \quad X \succeq 0$$

```
%% Invariant ellipsoid
LMI_ELL = [ [1, x';...
 x, X] >= ZERO ];
%% Lyapunov matrix
LMI_LYAP = [ X >= ZERO ];
```

LMIs

$$\begin{bmatrix} X & \star & \star & \star \\ A_v X + B_v Y & X & \star & \star \\ \sqrt{W_{\mathrm{x}}} X & 0 & \gamma I & \star \\ \sqrt{W_{\mathrm{u}}} Y & 0 & 0 & \gamma I \end{bmatrix} \succeq 0$$

```
%% Convergence
LMI_CON = [[
 X,(A{v}*X+B{v}*Y )',(sqrt(Wx)*X)',(sqrt(Wu)*Y)';...
 A{v}*X + B{v}*Y, X, ZEROx, ZEROxu;...
 sqrt(Wx)*X, ZEROx, gamma*Ix, ZEROxu;...
 sqrt(Wu)*Y, ZEROux, ZEROux, gamma*Iu] >= ZERO ];
```

LMIs

$$\begin{bmatrix} u_{\max}^2 I & \star \\ Y & X \end{bmatrix} \succeq 0 \qquad \begin{bmatrix} U & \star \\ Y & X \end{bmatrix} \succeq 0,\ U_{i,i} \leq u_{\max,i}^2$$

```
%% Constraints on control inputs: L2-norm
LMI_U_L2 = [ [ diag(u_max.^2), Y;...
 Y', X] >= ZERO ];
%% Constraints on control inputs: L1-norm
LMI_U_L1 = [ [ U, Y;...
 Y', X] >= ZERO ];
LMI_U_L1 = LMI_U_L1 + [ U(i,i) <= u_max(i)^2 ];
```

LMIs

$$\begin{bmatrix} x_{\max}^2 I & \star \\ C(A_v X + B_v Y) & X \end{bmatrix} \succeq 0$$

```
%% Constraints on system states
LMI_X = [[diag(x_max.^2), C{v}*(A{v}*X+B{v}*Y);...
          (A{v}*X + B{v}*Y)'*C{v}', X ] >= ZERO ];
```

Optimization

```
%% Optimization
constr = LMI_LYAP + LMI_ELL + LMI_CON + ...
 LMI_U_L1 + LMI_L2 + LMI_X;
sol = optimize(constr,obj)
```

Robust MPC

$$F = YX^{-1}$$

```
%% Robust MPC state controller design
X_opt = value(X)
Y_opt = value(Y)
F_opt = Y*X^(-1)
```

Control law

$$u(k) = F(x(k))\,x(k)$$

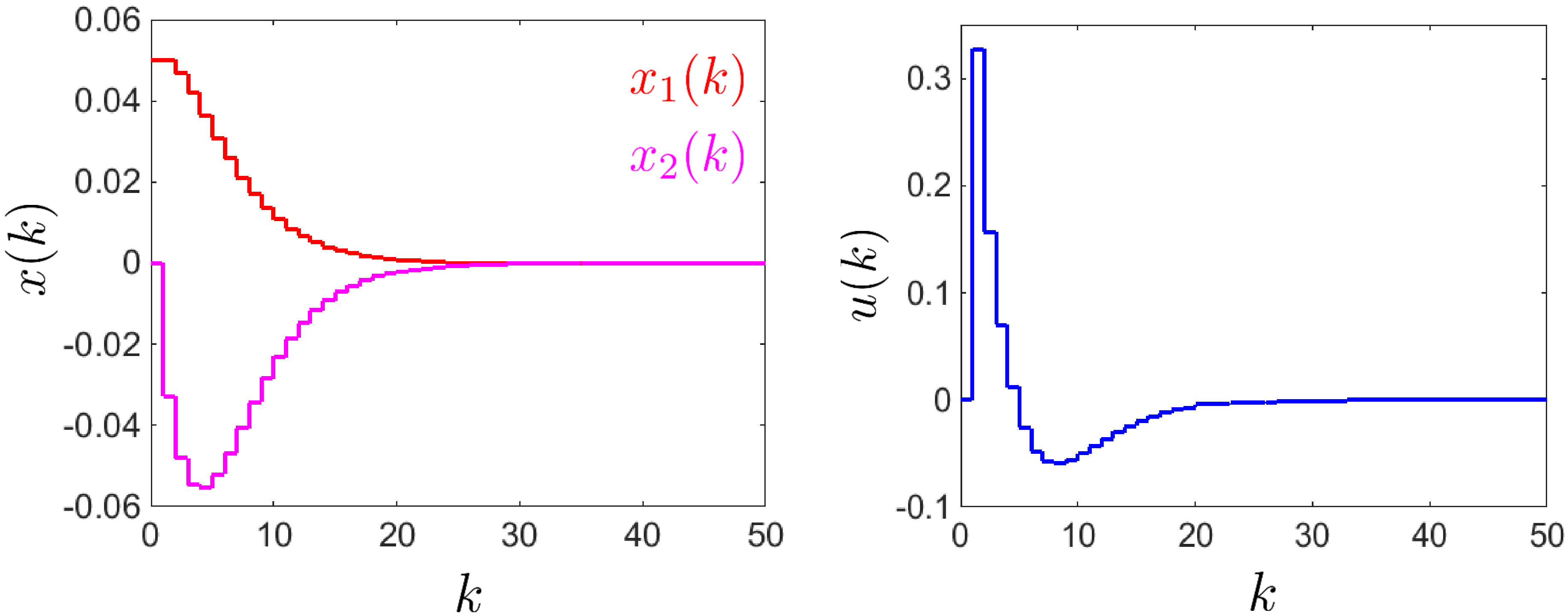
```
%% Control law
u = F_opt*x
```

Closed-loop performance

$$x(k+1) = A_v\, x(k) + B_v\, u(k),$$
$$y(k) = C\, x(k),$$

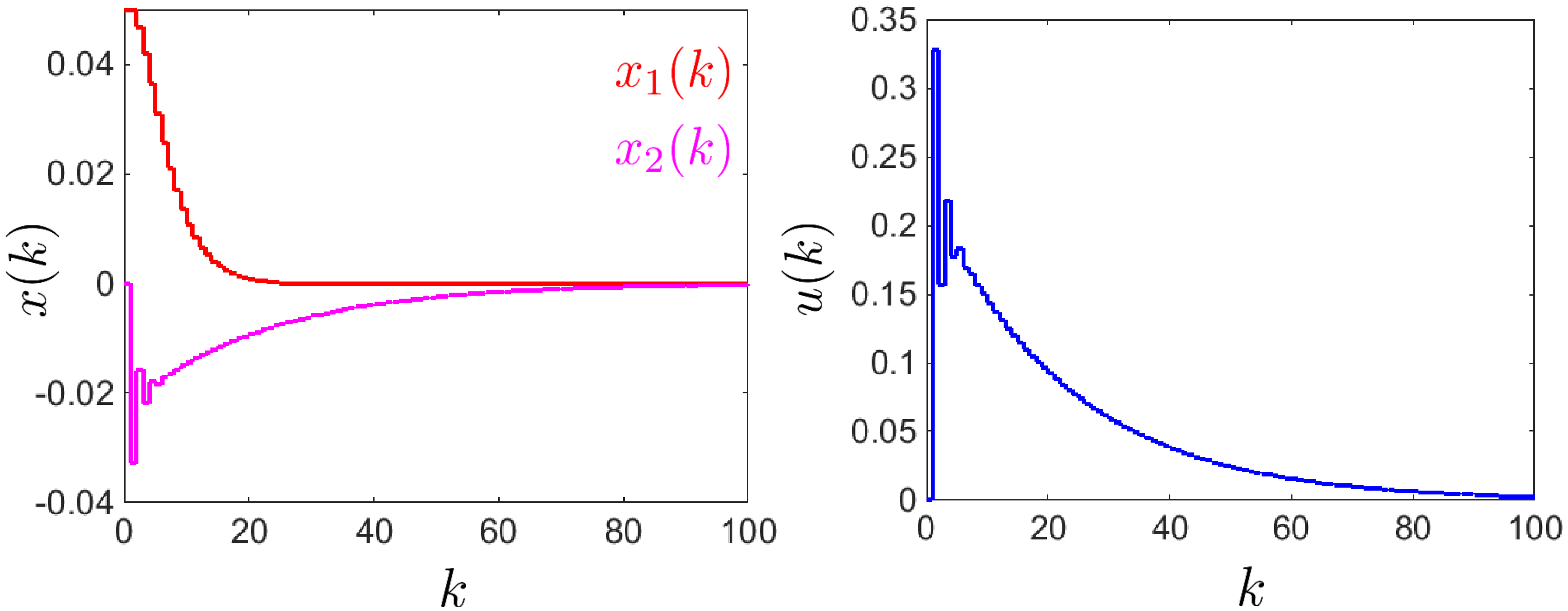
```
%% Closed loop performance
x_plus = A{v}*x + B{v}*u
     y = C{v}*x
```

Closed-loop performance



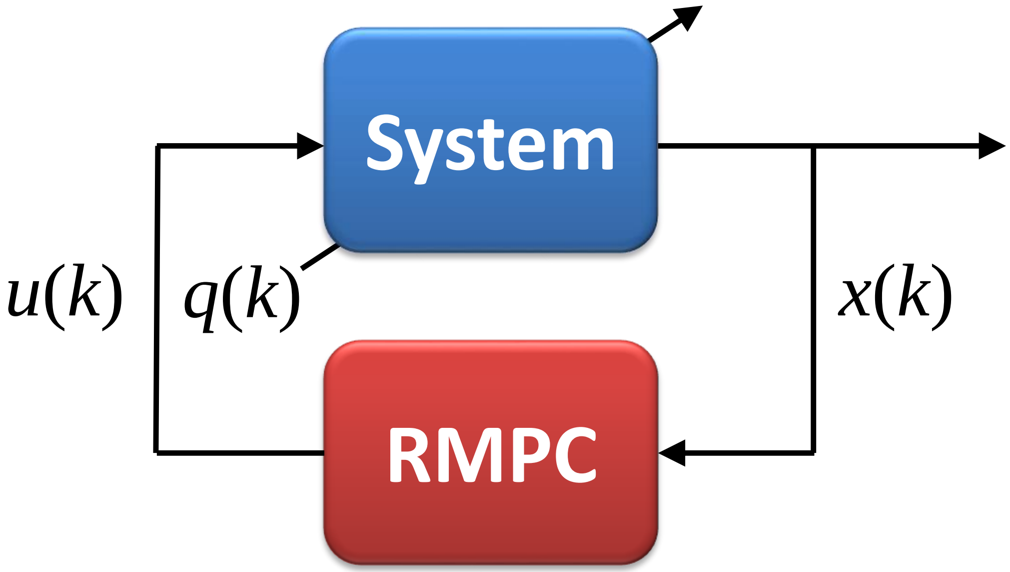
Vertex 1

Closed-loop performance



Vertex 2

MUP



MATLAB toolbox for on-line RMPC design by LMIs

⋆ ***bitbucket.org/oravec/mup***

MUP

```
%% DEMO
mup_rmpc_demo
```