LMI-based Robust MPC Design

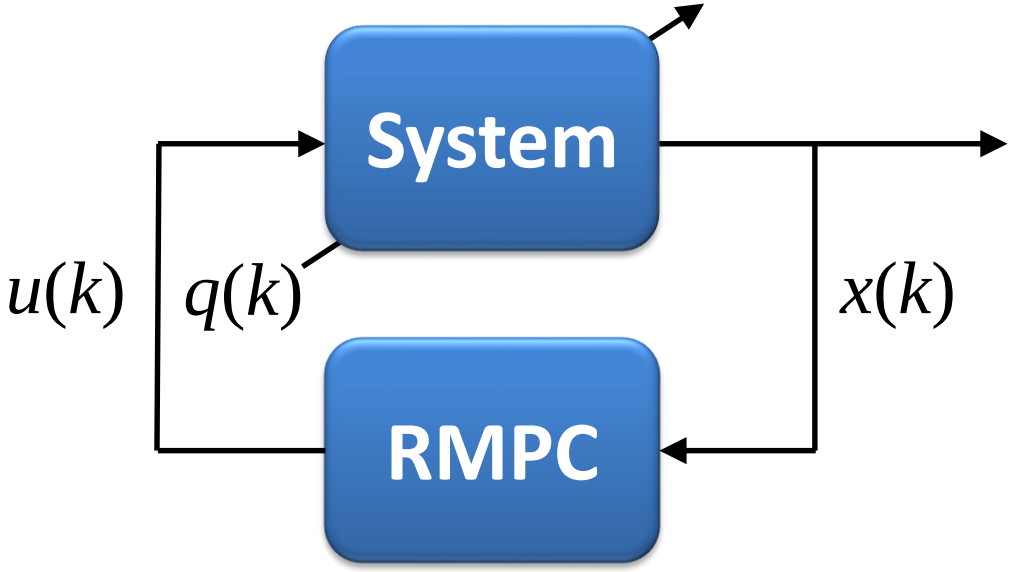
Toolbox MUP

Juraj Oravec – Monika Bakošová

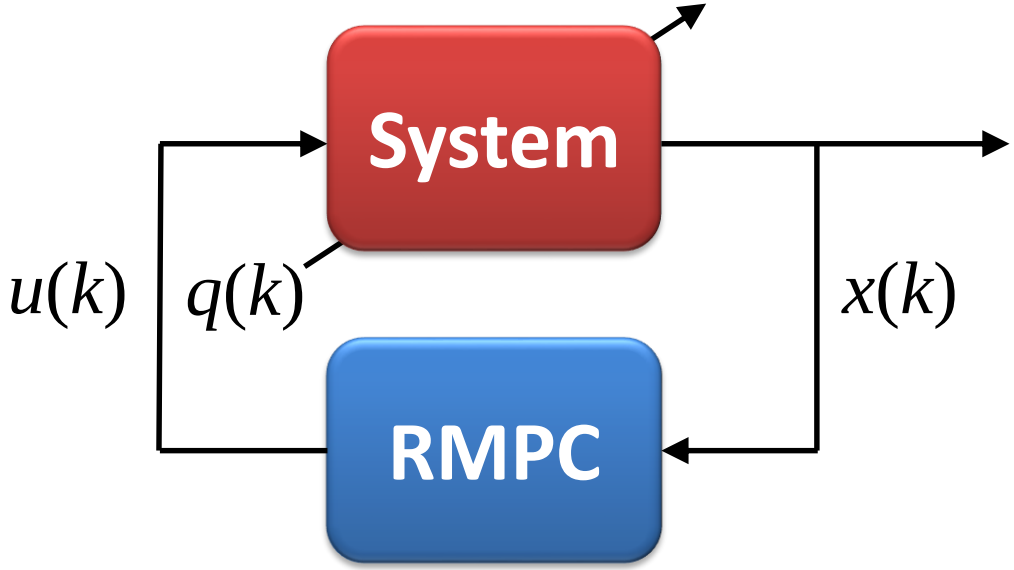
STU

Slovak University of Technology in Bratislava

Problems to solve

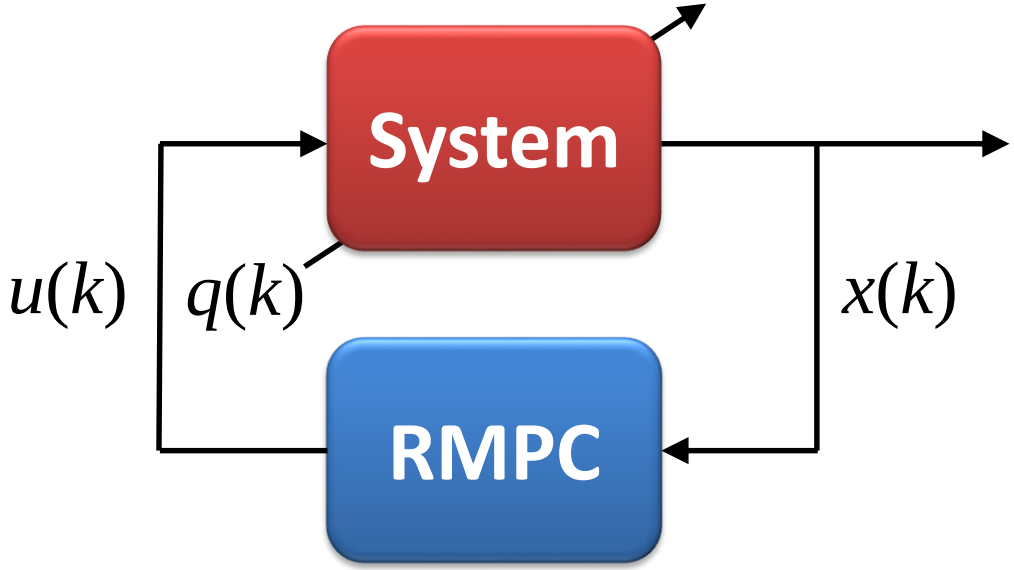


Problems to solve



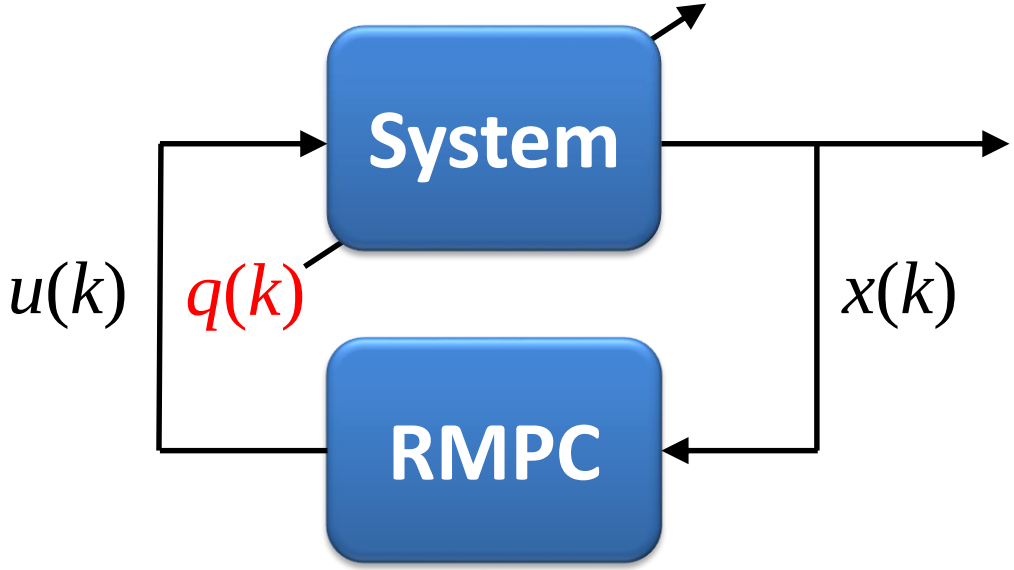
$$x(k+1) = A\,x(k) + B\,u(k), \qquad x(0) = x_0,$$
$$y(k) = C\,x(k),$$

Problems to solve



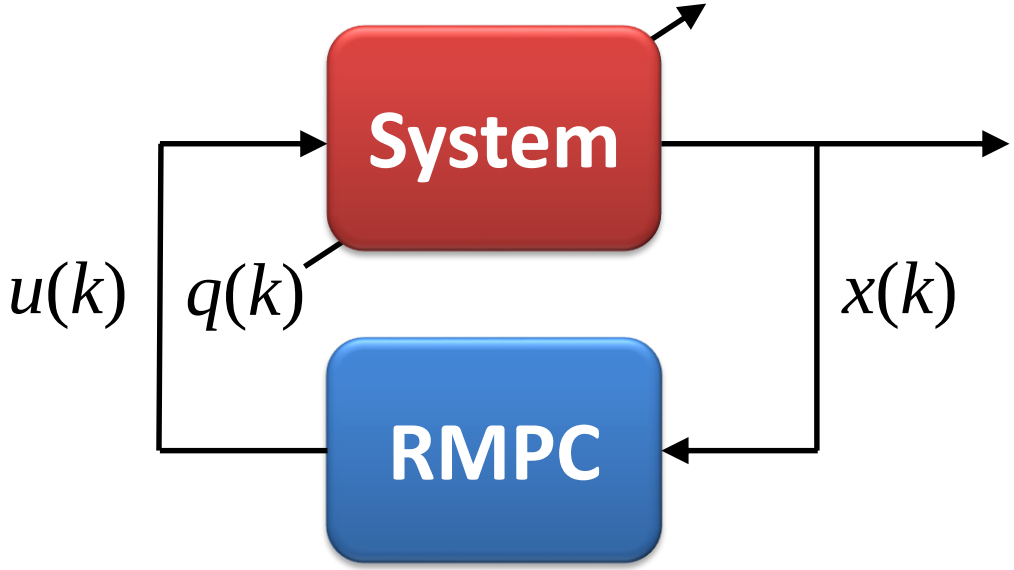
$$x(k+1) = A\,x(k) + B\,u(k), \qquad x(0) = x_0,$$
$$y(k) = C\,x(k), \qquad u \in \mathbb{U},\ y \in \mathbb{Y},$$

Problems to solve



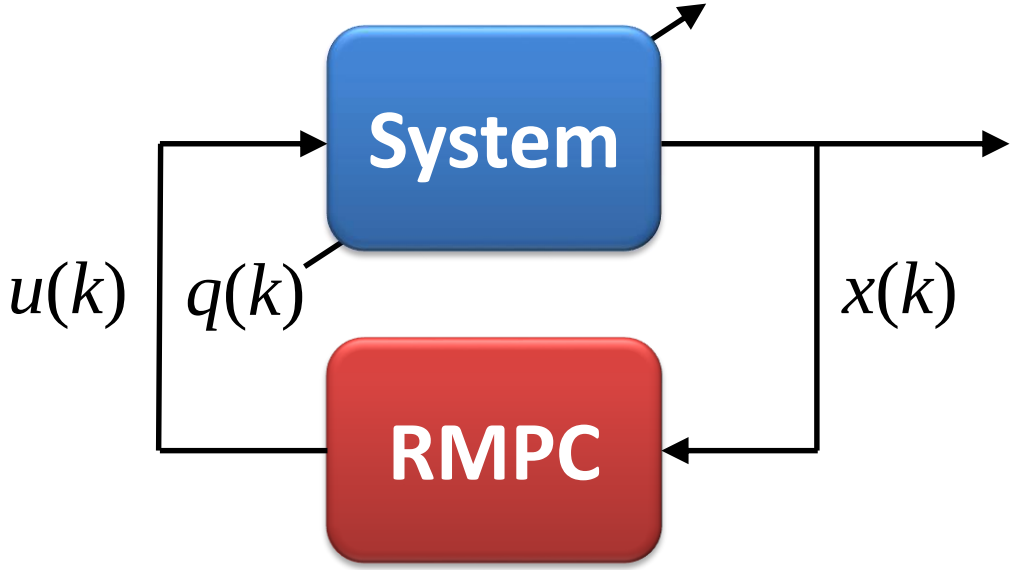
$$x(k+1) = A\,x(k) + B\,u(k), \qquad x(0) = x_0,$$
$$y(k) = C\,x(k), \qquad u \in \mathbb{U},\ y \in \mathbb{Y},$$
$$q \in \mathbb{Q}.$$

Problems to solve



$$x(k+1) = A\,x(k) + B\,u(k), \qquad x(0) = x_0,$$
$$y(k) = C\,x(k), \qquad u \in \mathbb{U},\ y \in \mathbb{Y},$$
$$[A, B] \in \text{convhull}\{[A_v, B_v]\}, \qquad q \in \mathbb{Q}.$$

Analysis

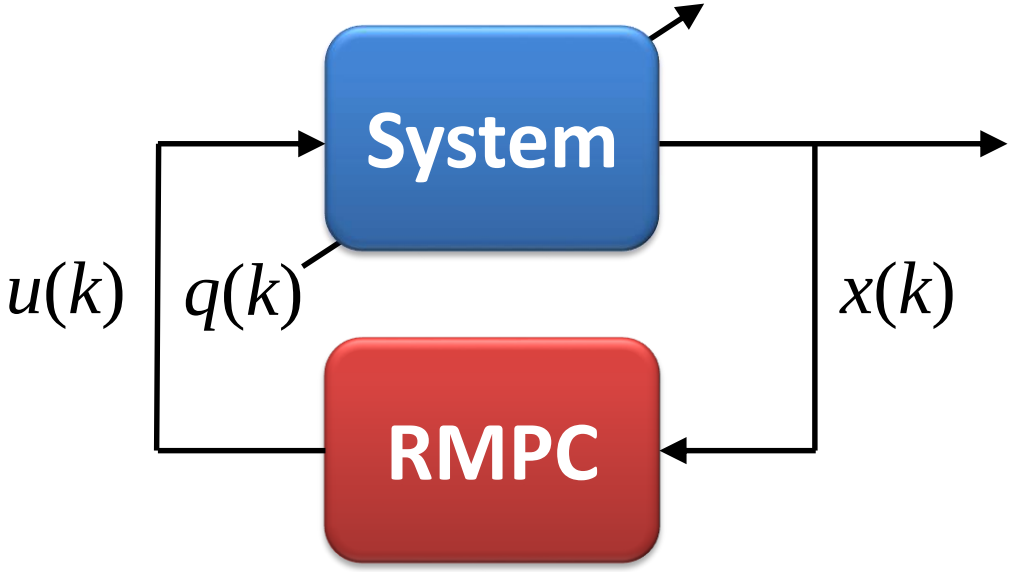


***SDP*:**

$$\min \; c^\top x$$

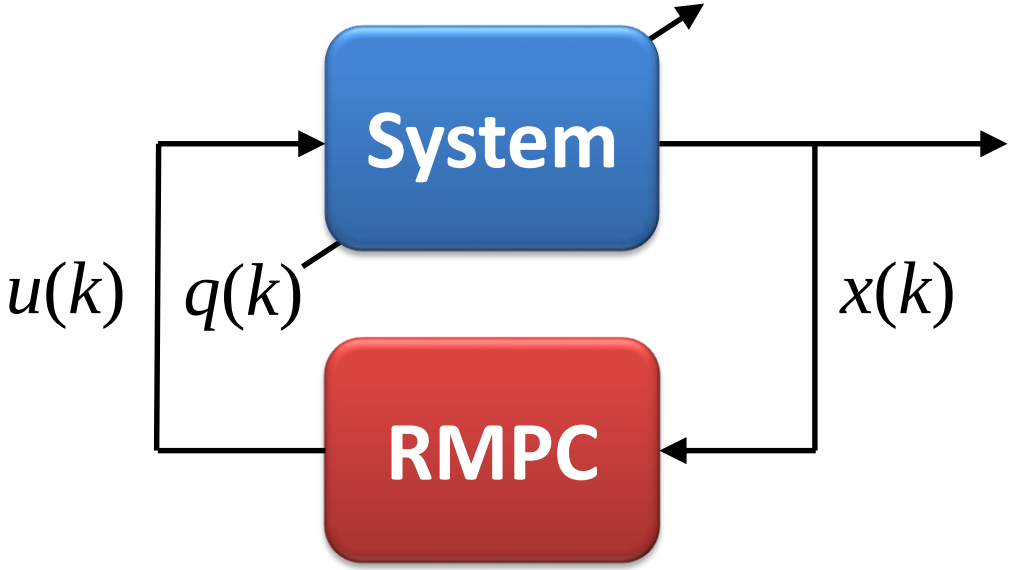
$$\text{s.t.:} \; x \preceq_{\mathcal{K}} 0 \Rightarrow M(x) \prec 0$$

Synthesis



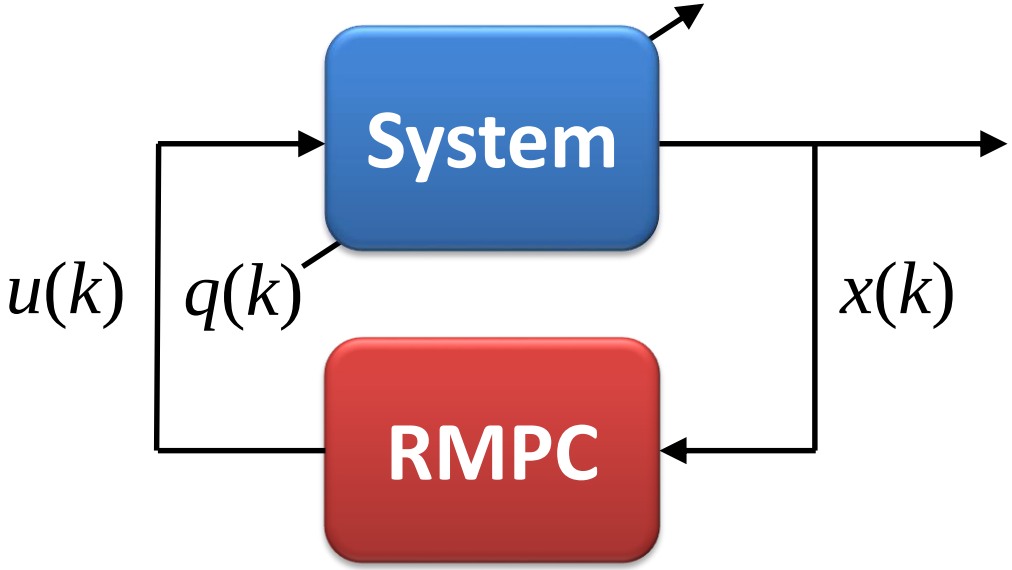
Control law: $u(k) = F(x(k))\,x(k)$

Synthesis



Control law: $u(k) = F(x(k))\, x(k)$

Synthesis

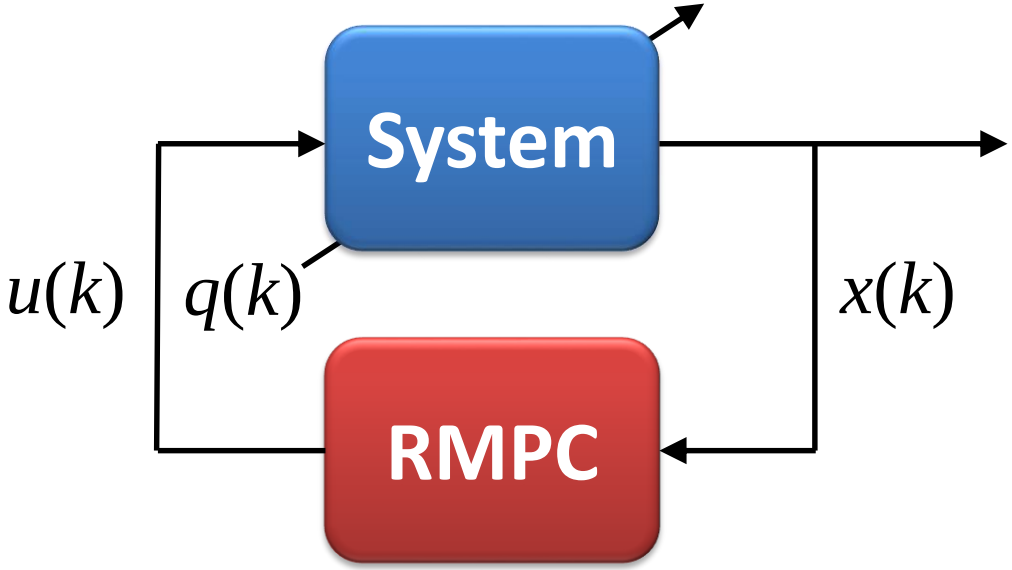


Control law: $u(k) = F(x(k))\,x(k)$

Quality criterion:

$$J(k) = \sum_{i=0}^{\infty} (x(k+i)^\top Q\, x(k+i) + u(k+i)^\top R\, u(k+i))$$

Synthesis

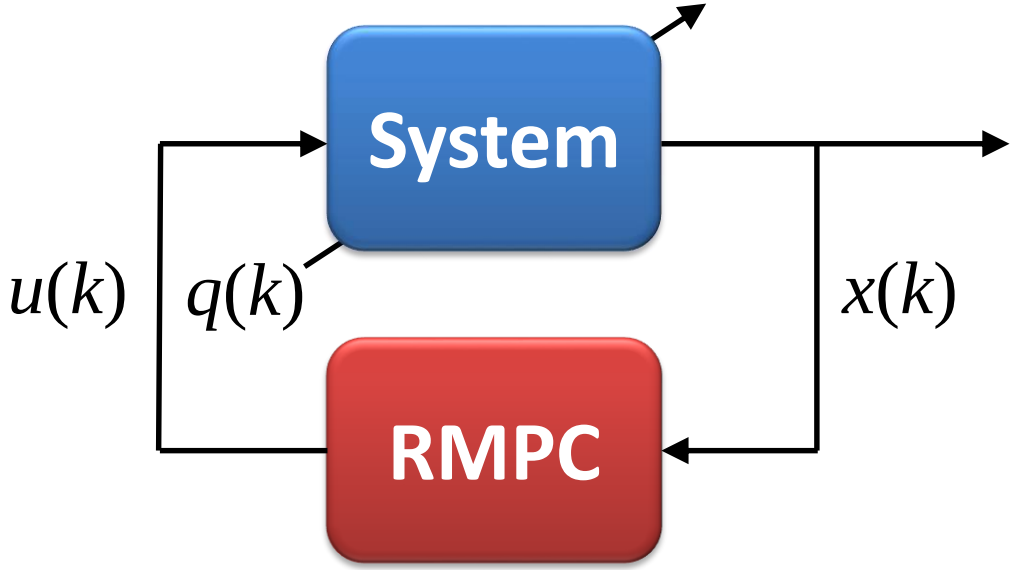


Control law: $u(k) = F(x(k))\,x(k)$

Quality criterion:

$$J(k) = \sum_{i=0}^{\infty} (x(k+i)^\top Q\, x(k+i) + u(k+i)^\top R\, u(k+i))$$

Synthesis

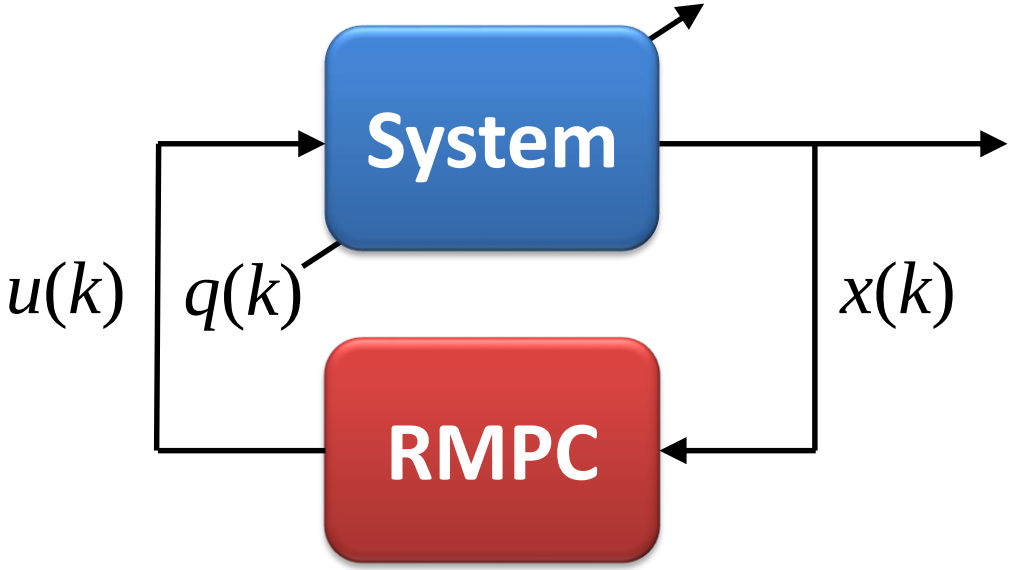


Control law: $u(k) = F(x(k))\,x(k)$

Quality criterion:

$$J(k) = \sum_{i=0}^{\infty} (x(k+i)^\top W_{\mathrm{x}}\, x(k+i) + u(k+i)^\top W_{\mathrm{u}}\, u(k+i))$$

Synthesis

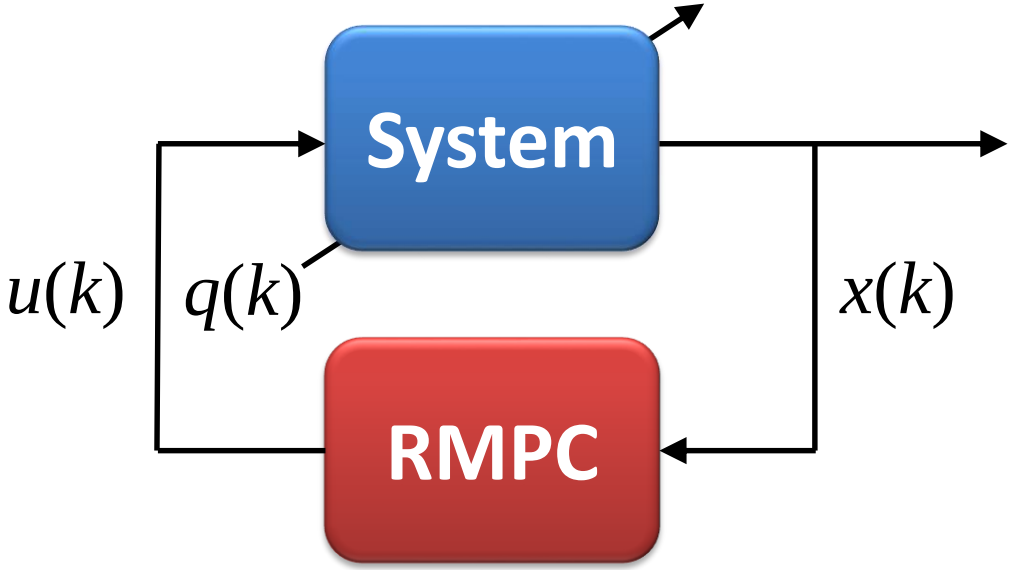


Control law: $u(k) = F(x(k))\,x(k)$

Quality criterion:

$$J(k) = \sum_{i=0}^{\infty} \ell(x(k+i), u(k+i))$$

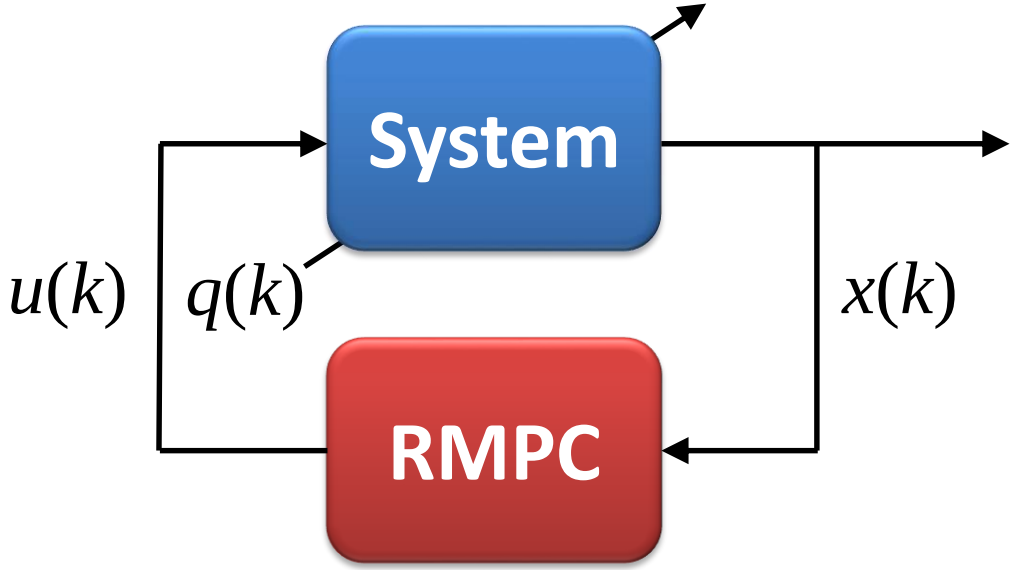
Toolbox



MATLAB toolbox for on-line RMPC design by LMIs

★ ***bitbucket.org/oravec/mup***

Toolbox

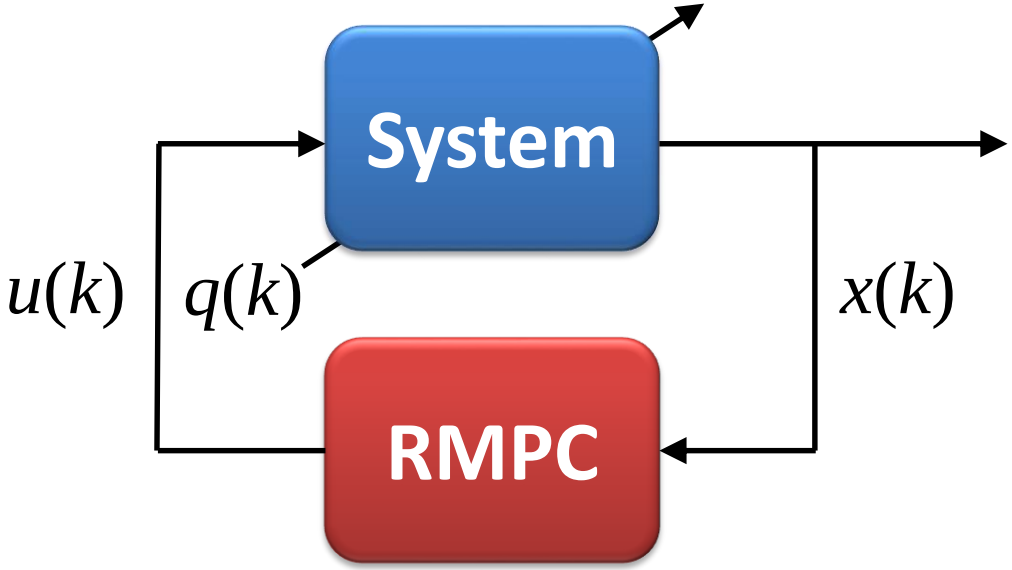


MATLAB toolbox for on-line RMPC design by LMIs

- speed up RMPC design and tuning
- advanced RMPC design analysis
- user-friendly tool

★ ***bitbucket.org/oravec/mup***

Toolbox

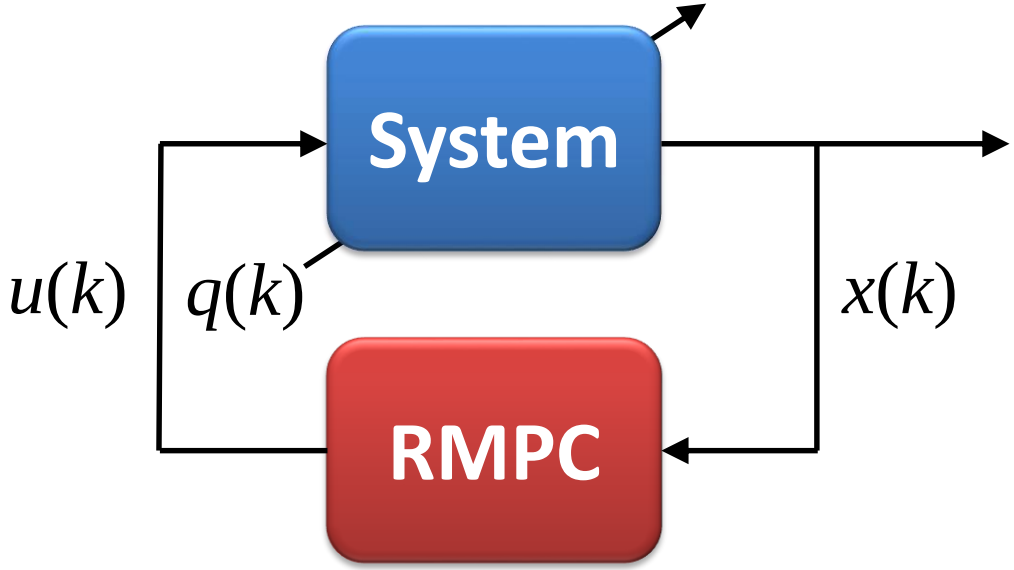


MATLAB toolbox for on-line RMPC design by LMIs

- freely available★
- support★ (demo projects, wiki, issues)
- under development (*new release coming soon*)

★ ***bitbucket.org/oravec/mup***

Toolbox

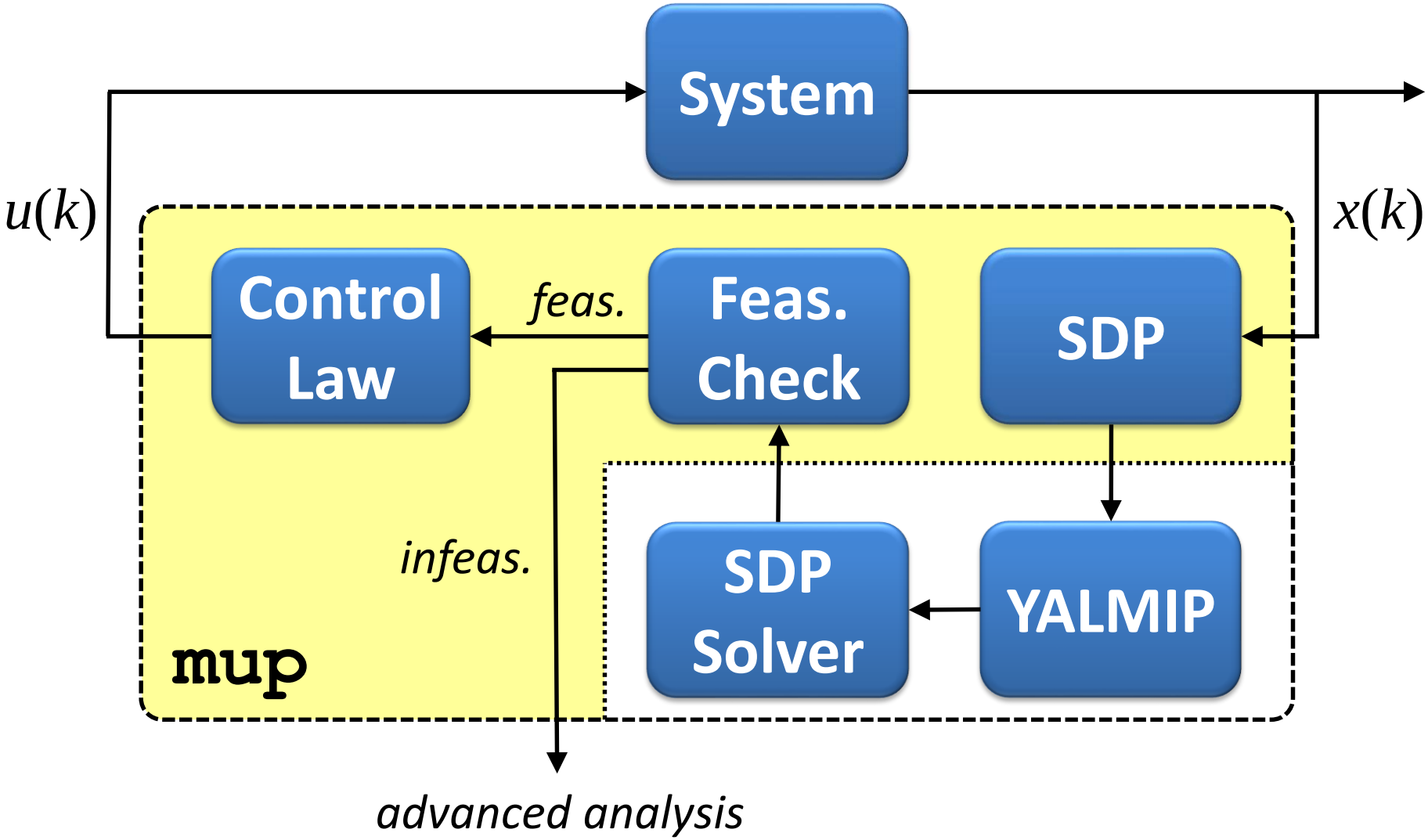


Necessary theoretical backgrounds:

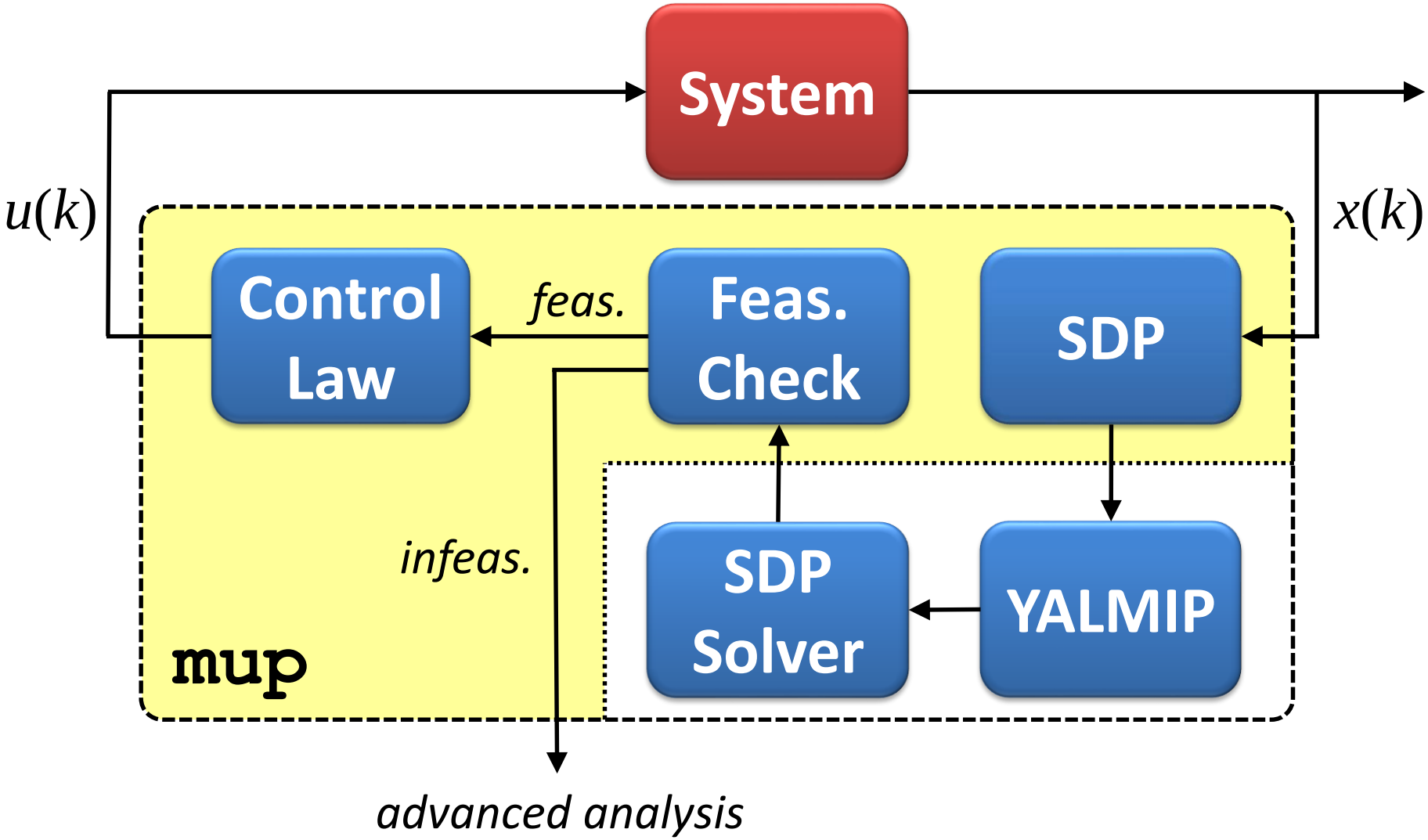
- LQR/MPC basic (*required*)
- RMPC (*optional*)
- SDP (*optional*)

★ ***bitbucket.org/oravec/mup***

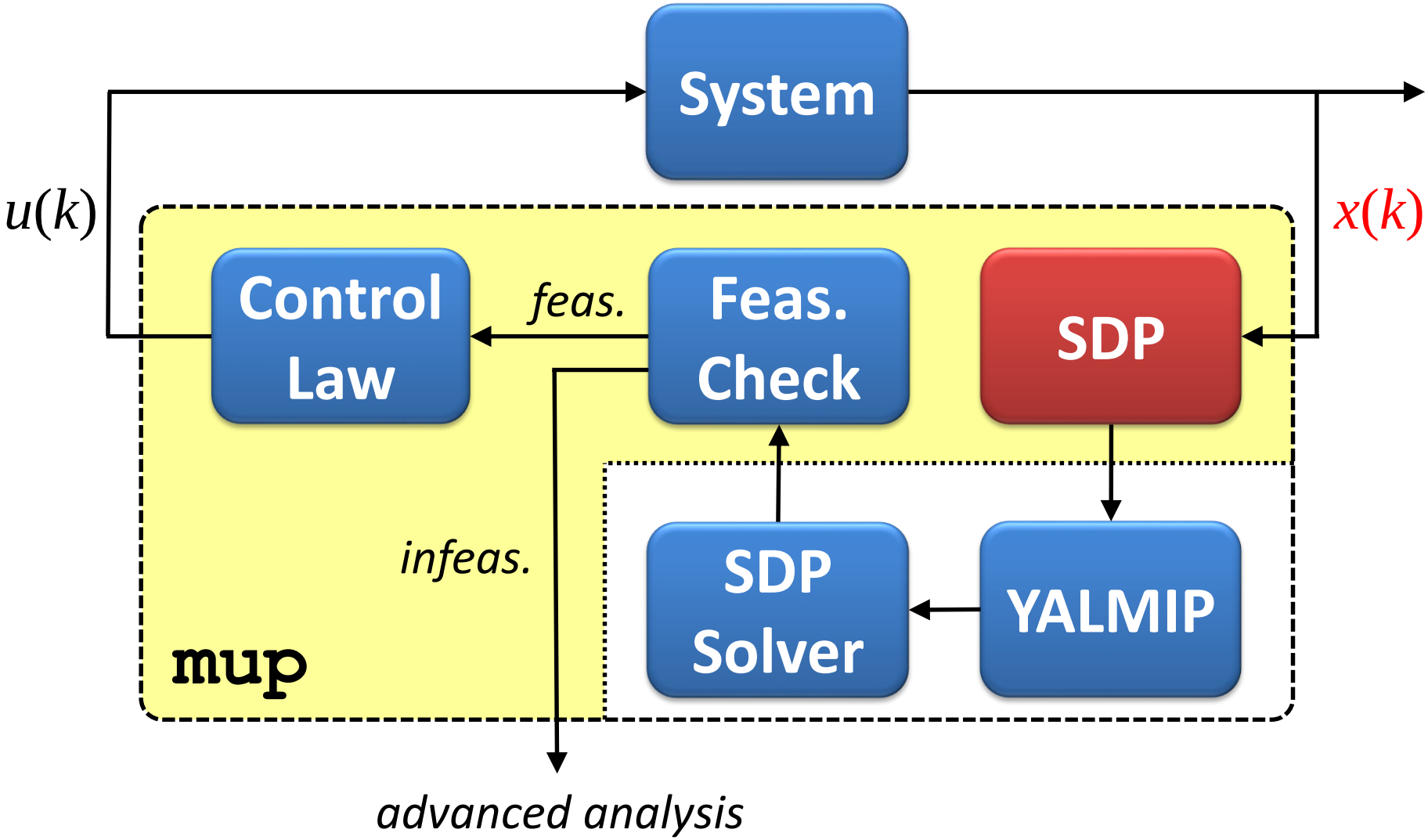
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

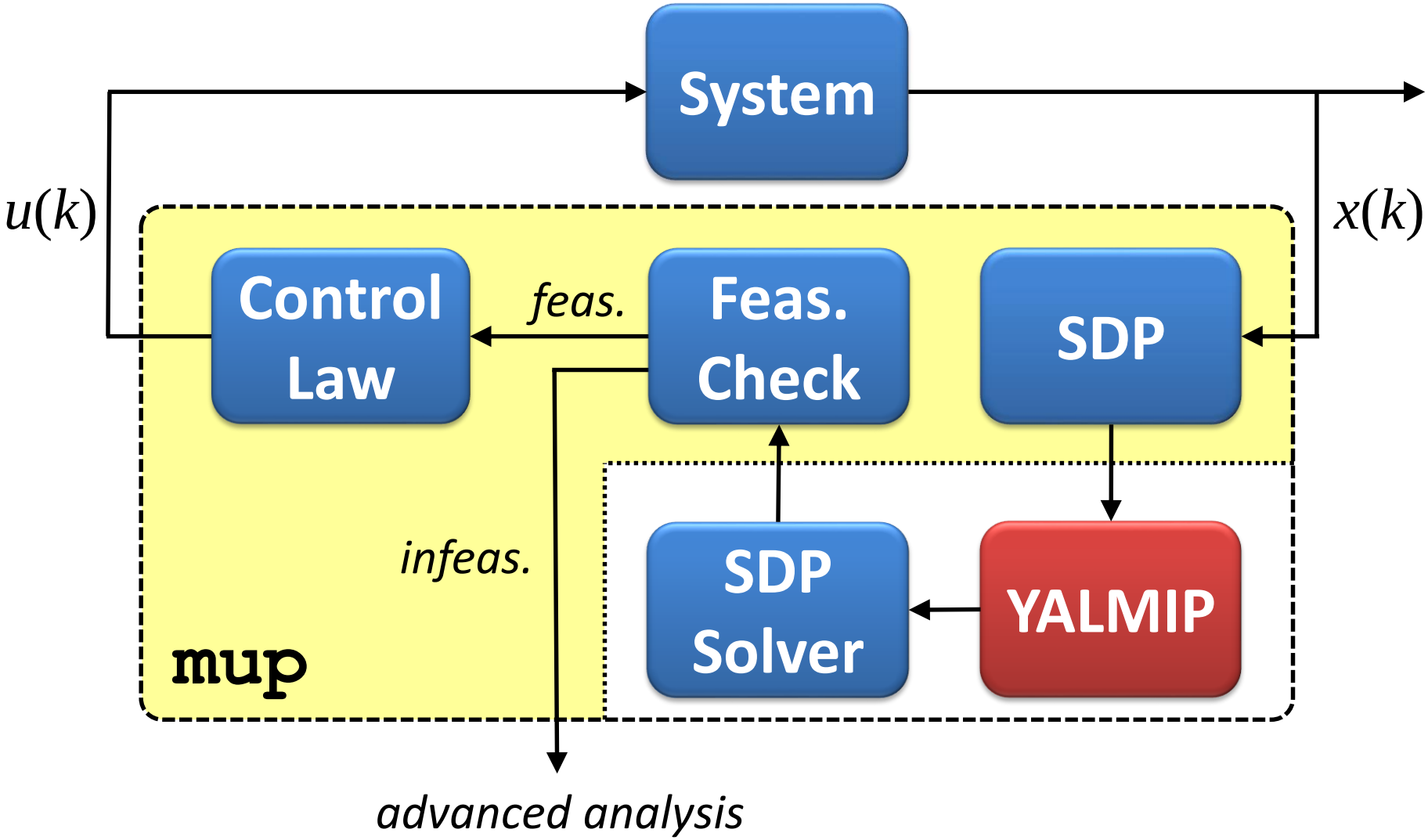
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

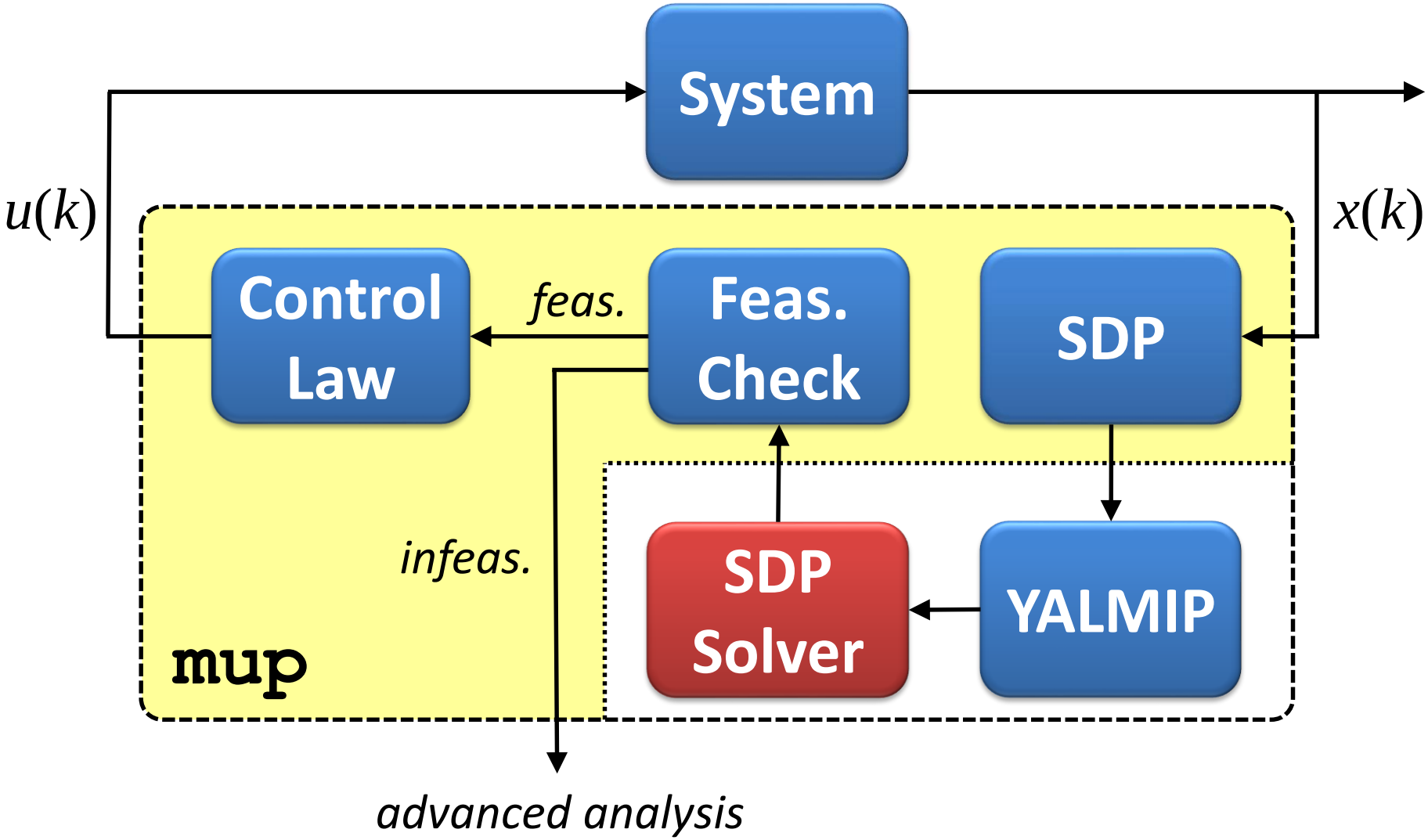
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

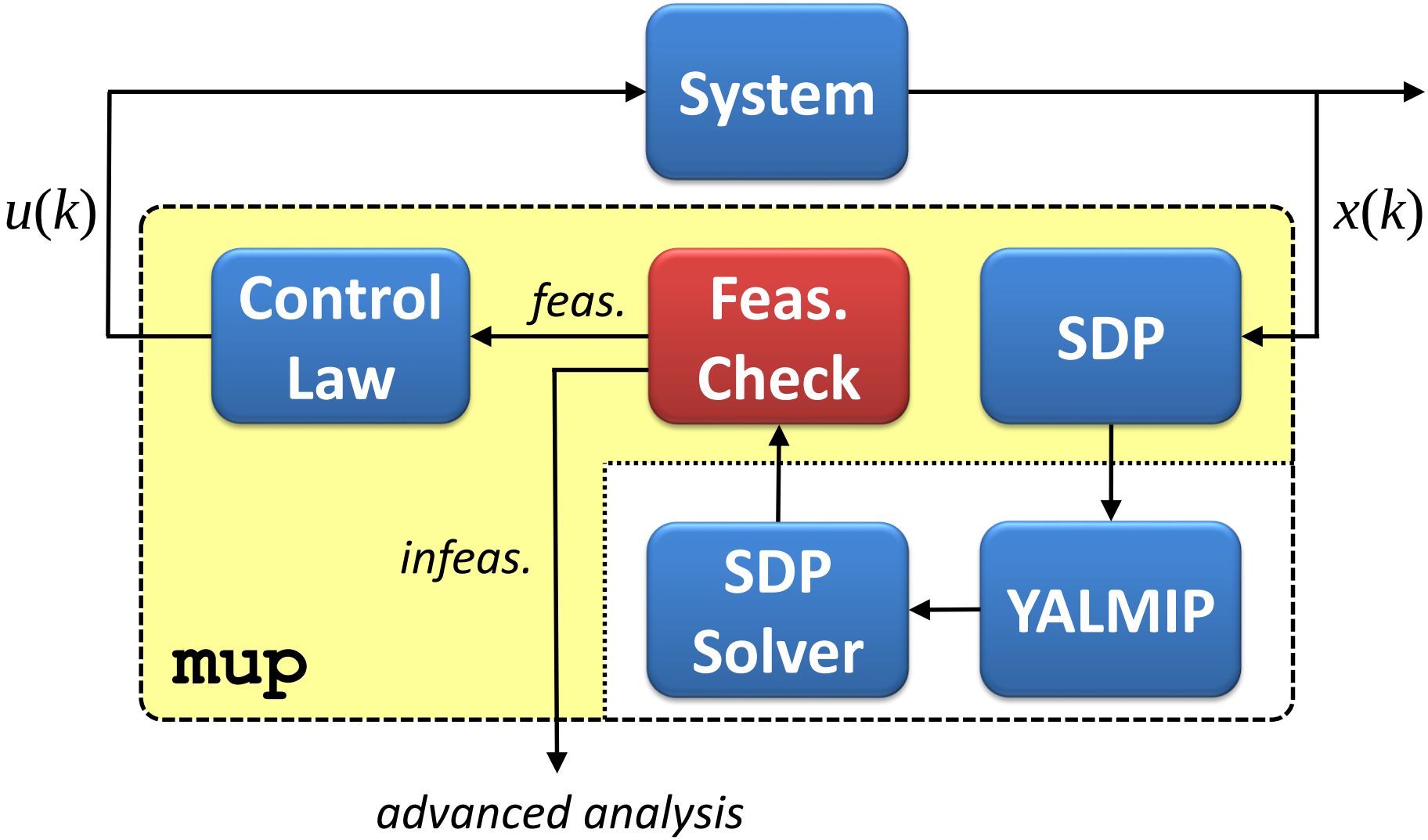
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

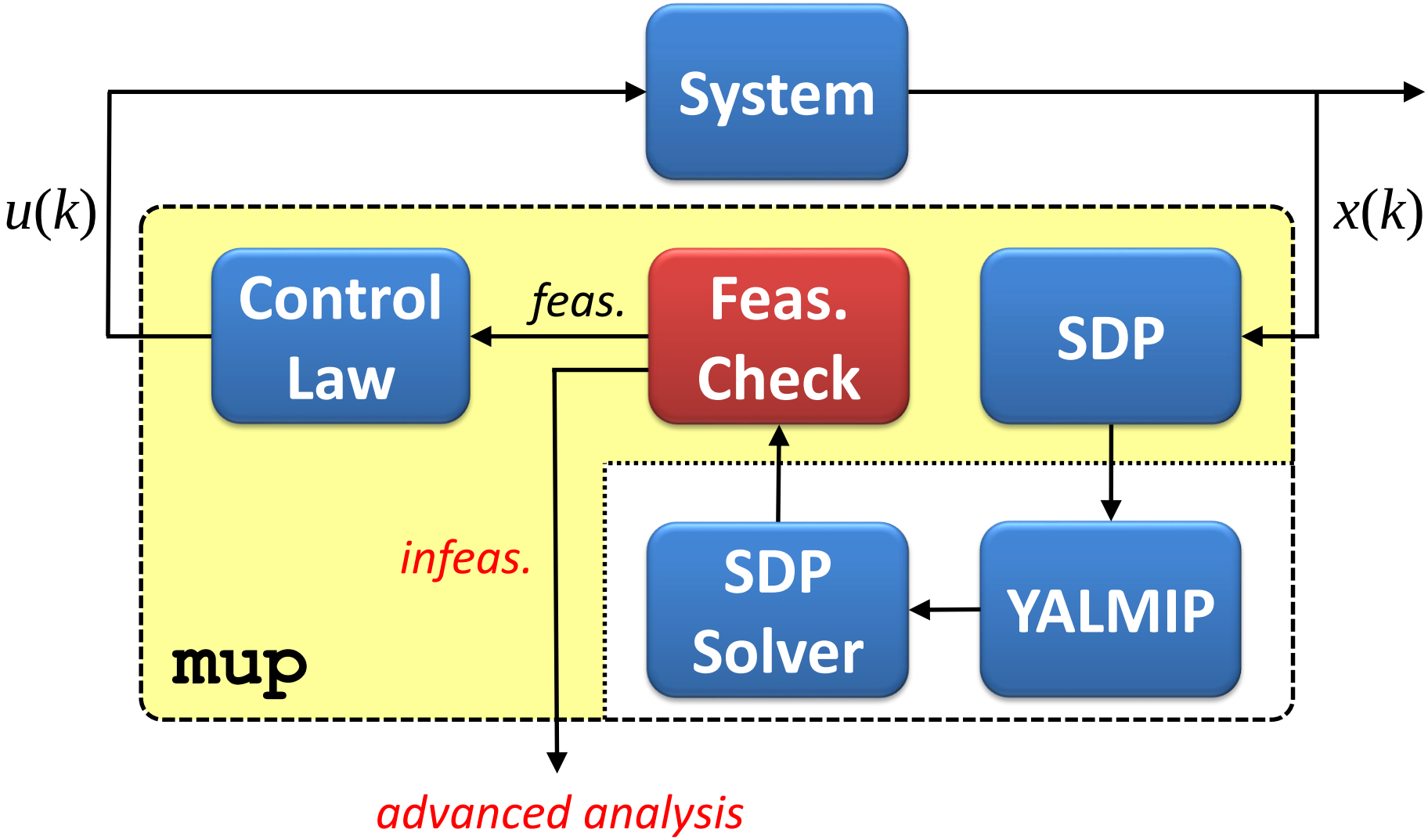
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

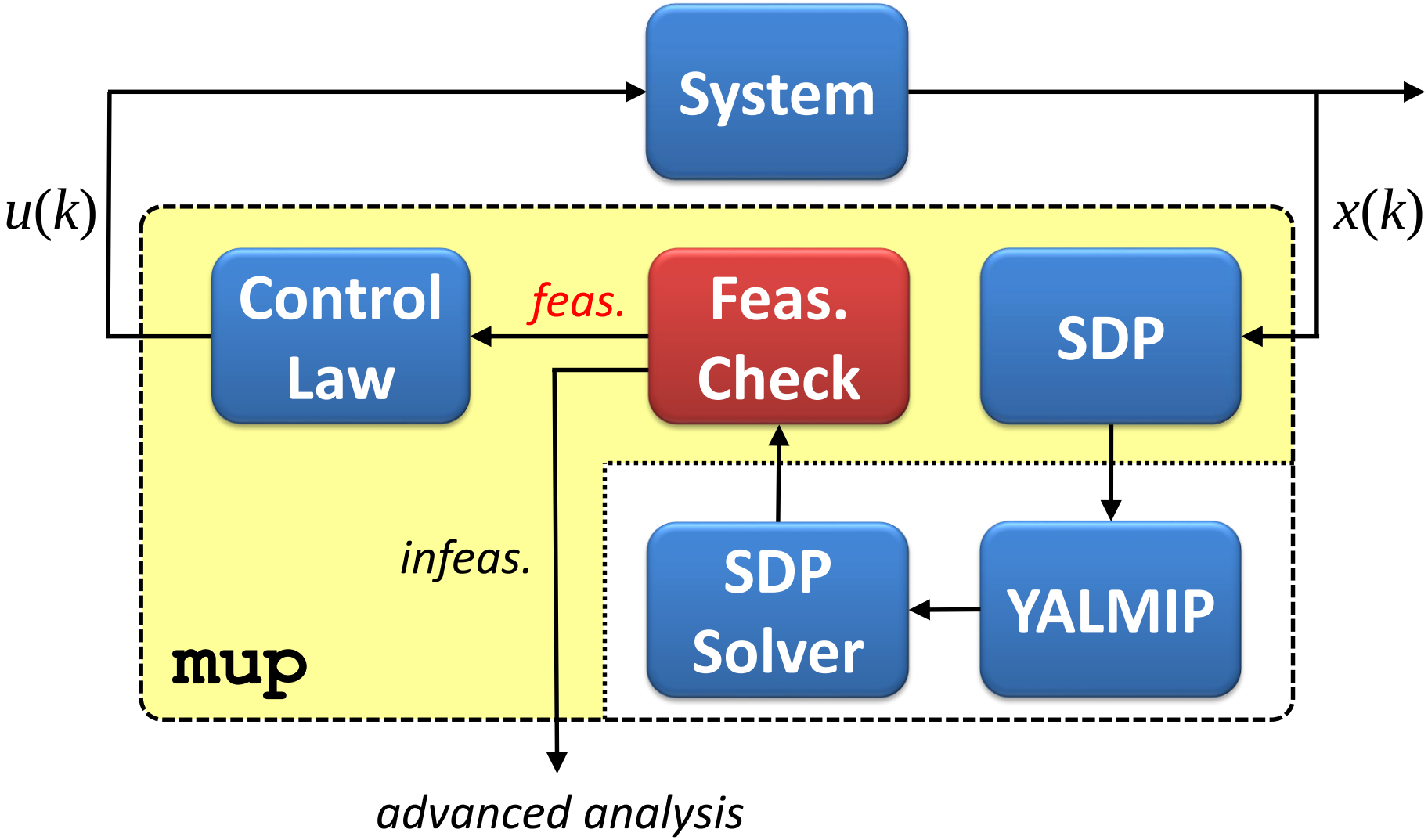
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

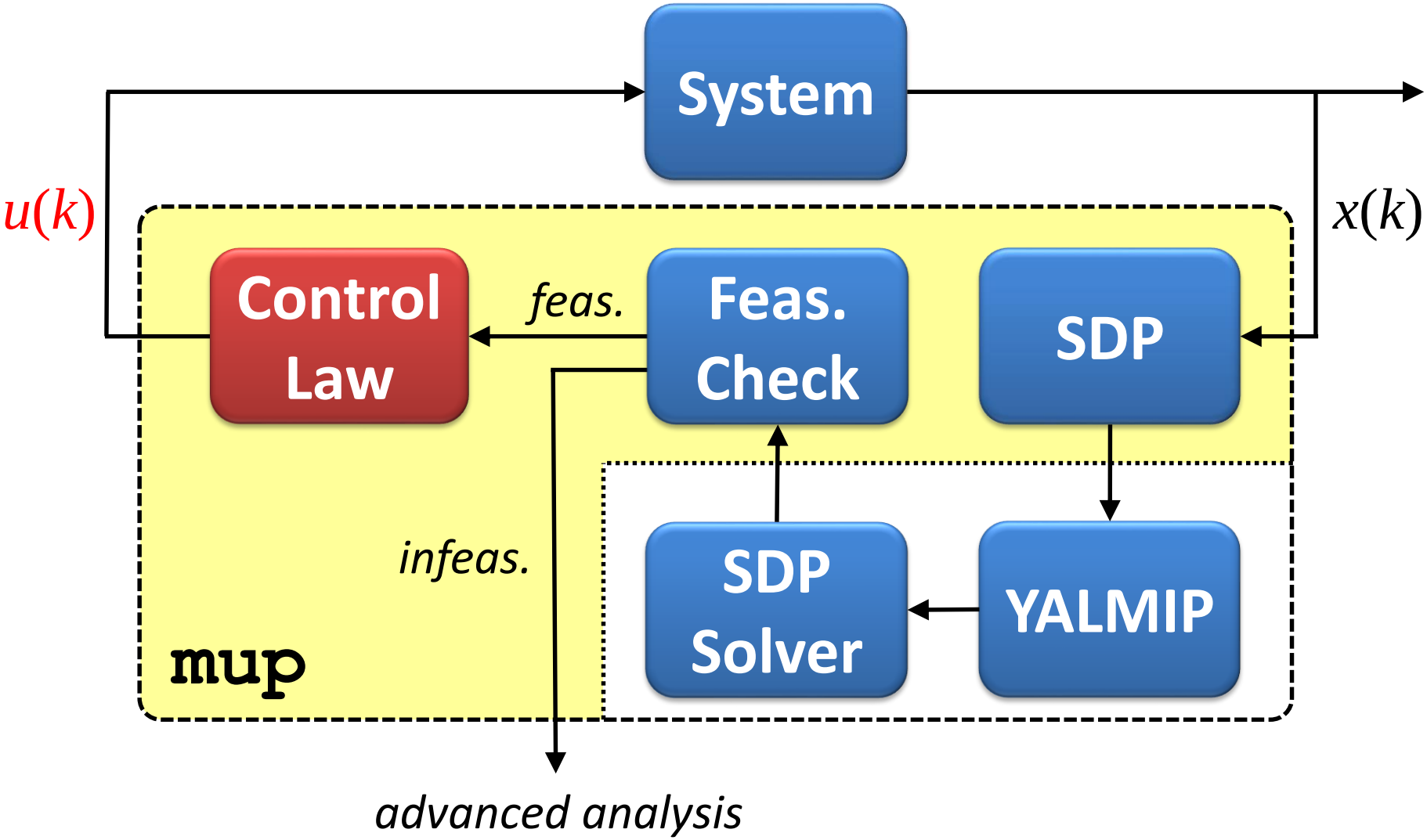
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

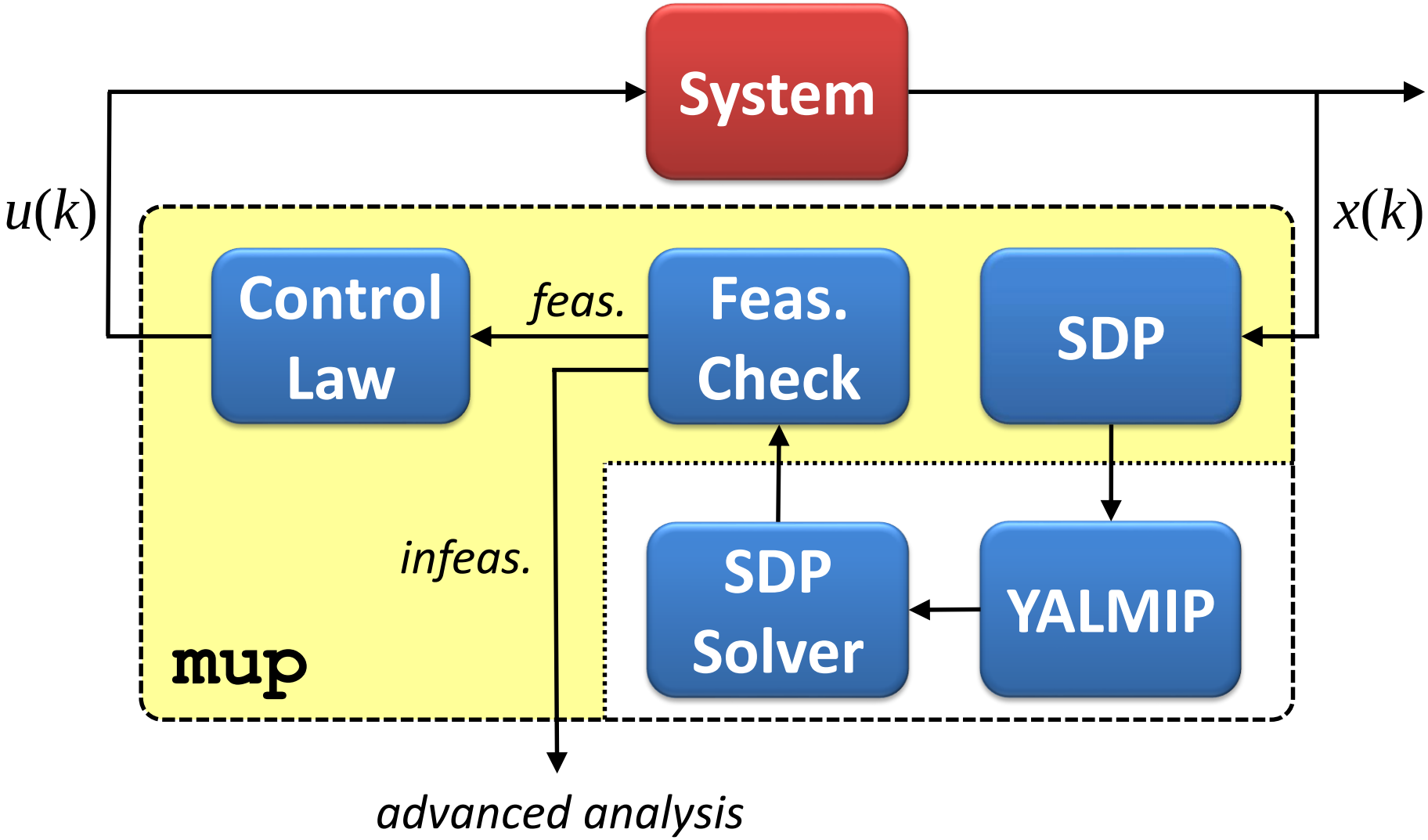
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

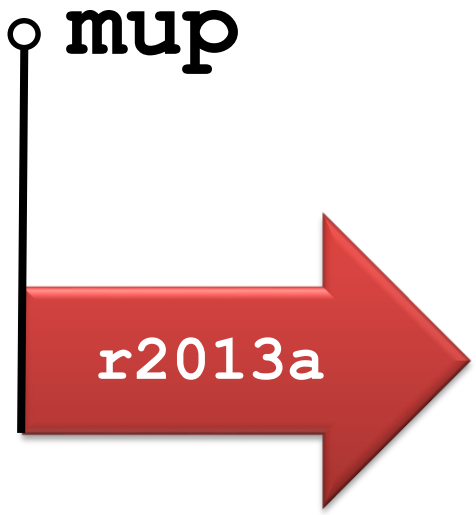
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

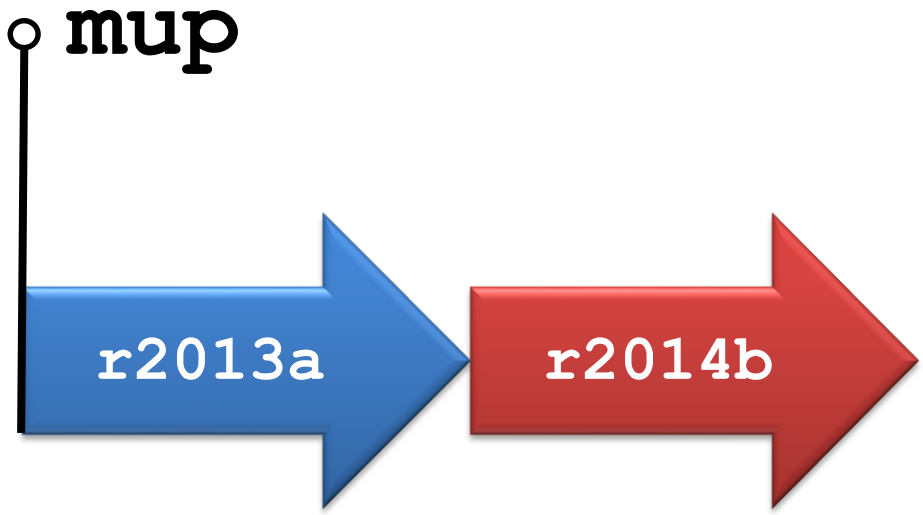
Framework


System
$u(k)$
$x(k)$
Control Law
feas.
Feas. Check
SDP
infeas.
SDP Solver
YALMIP
mup
advanced analysis

MATLAB Compatibility

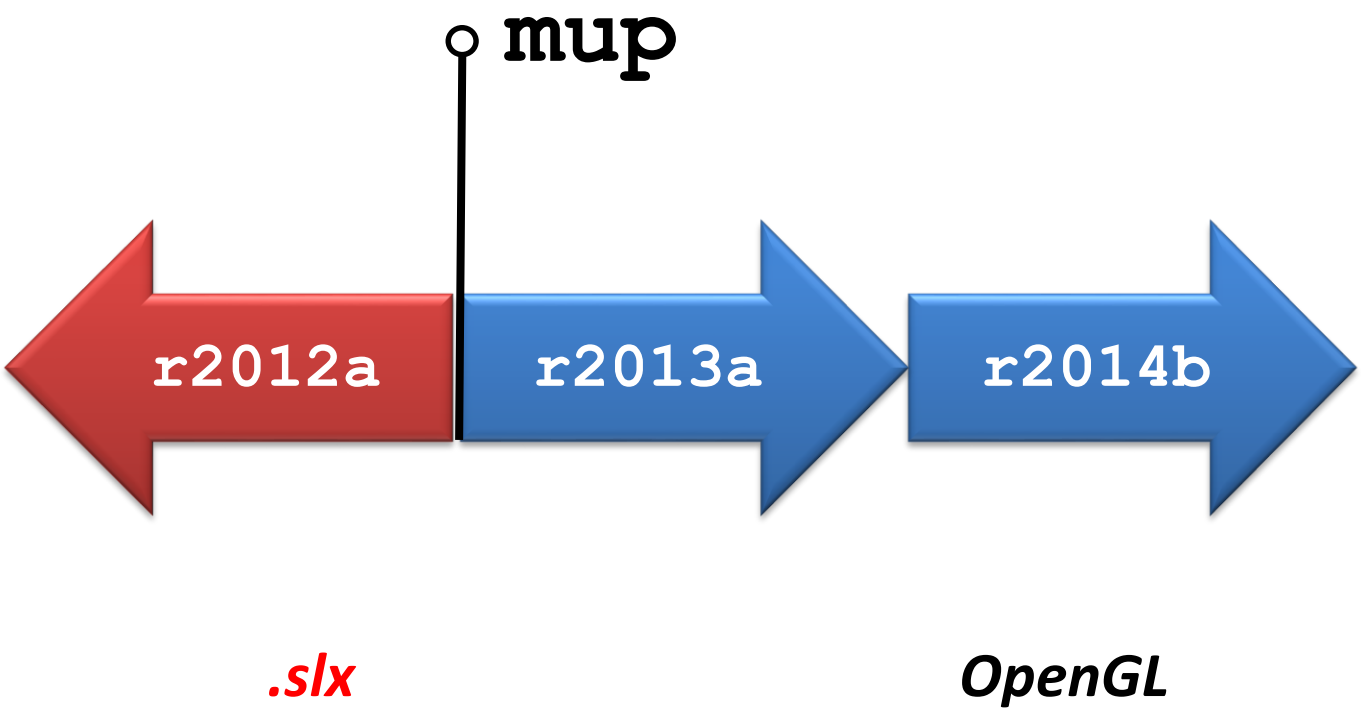


MATLAB Compatibility

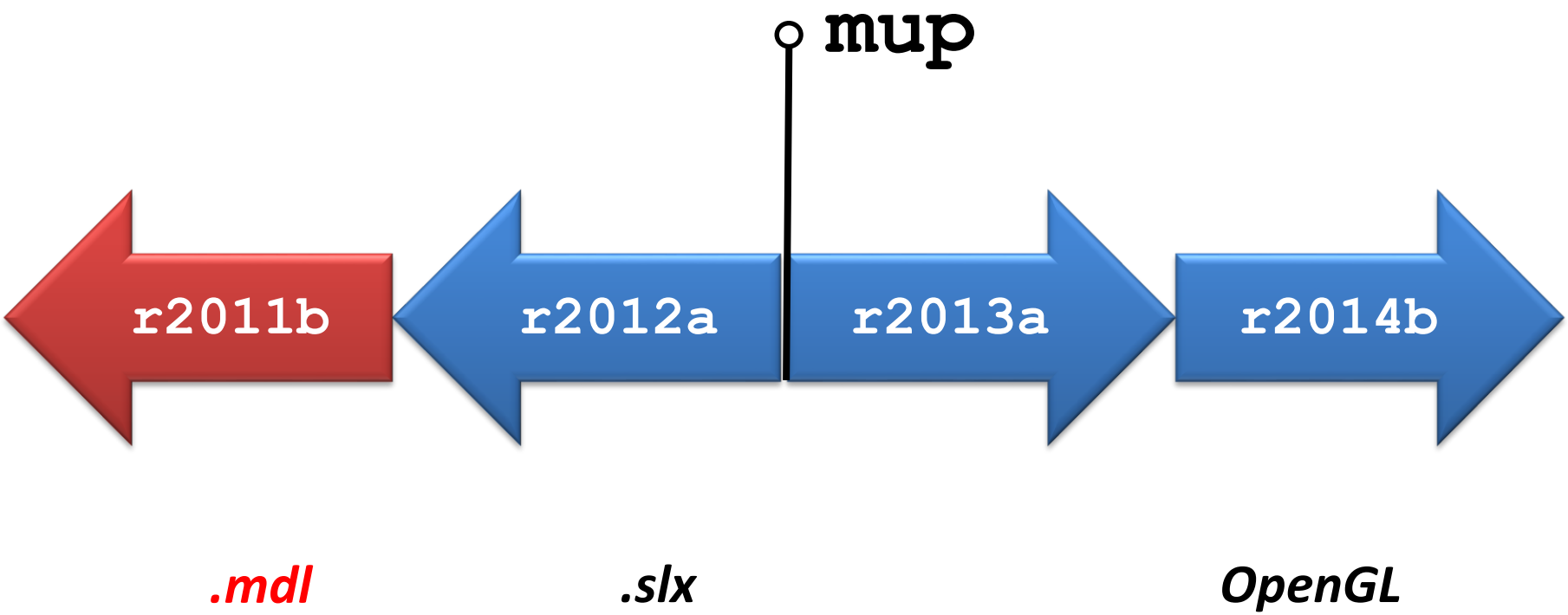


OpenGL

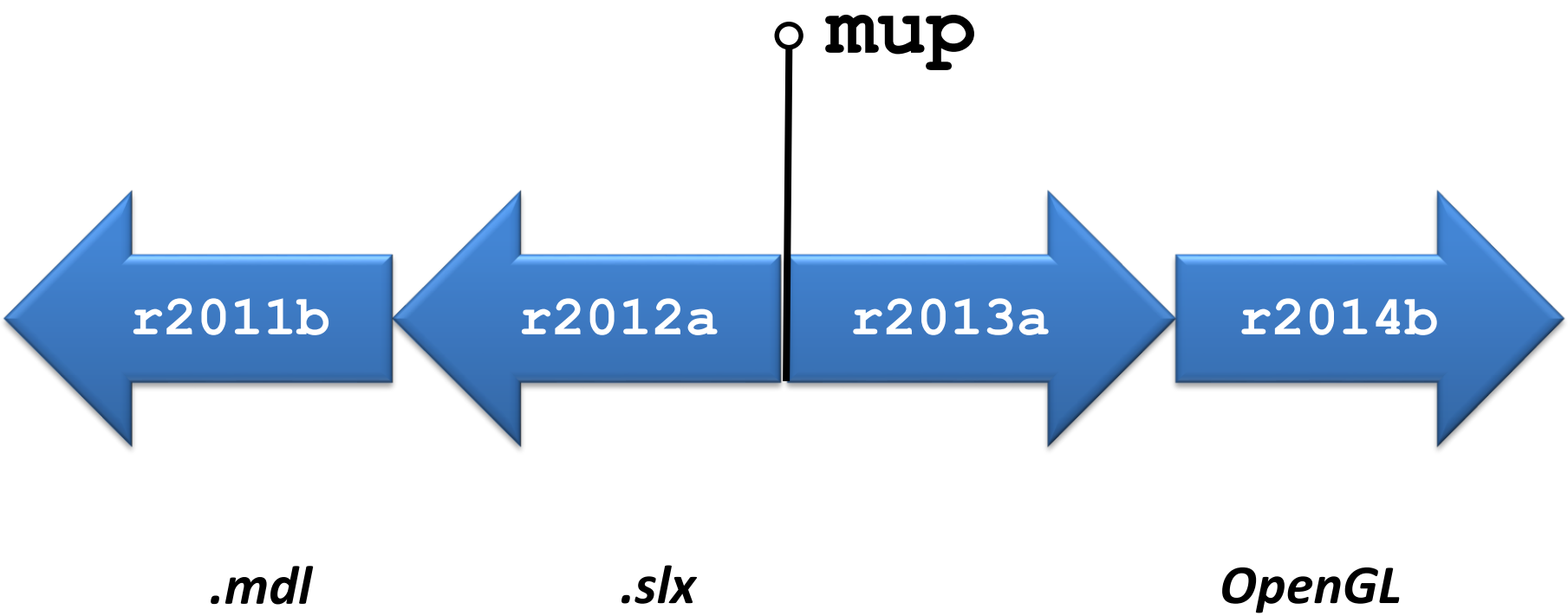
MATLAB Compatibility



MATLAB Compatibility



MATLAB Compatibility



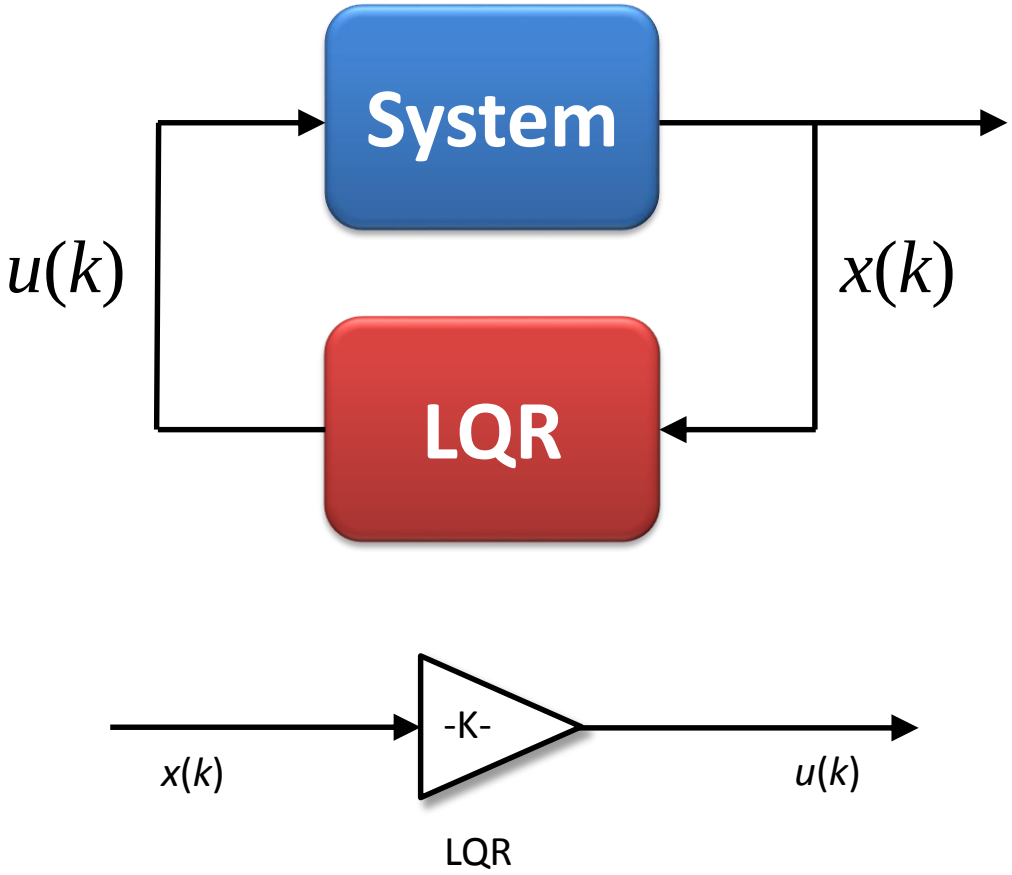
How to use it?

- MATLAB/Simulink: `rmpc_block` (*beginner*)
- MATLAB: CLI (*advanced*)

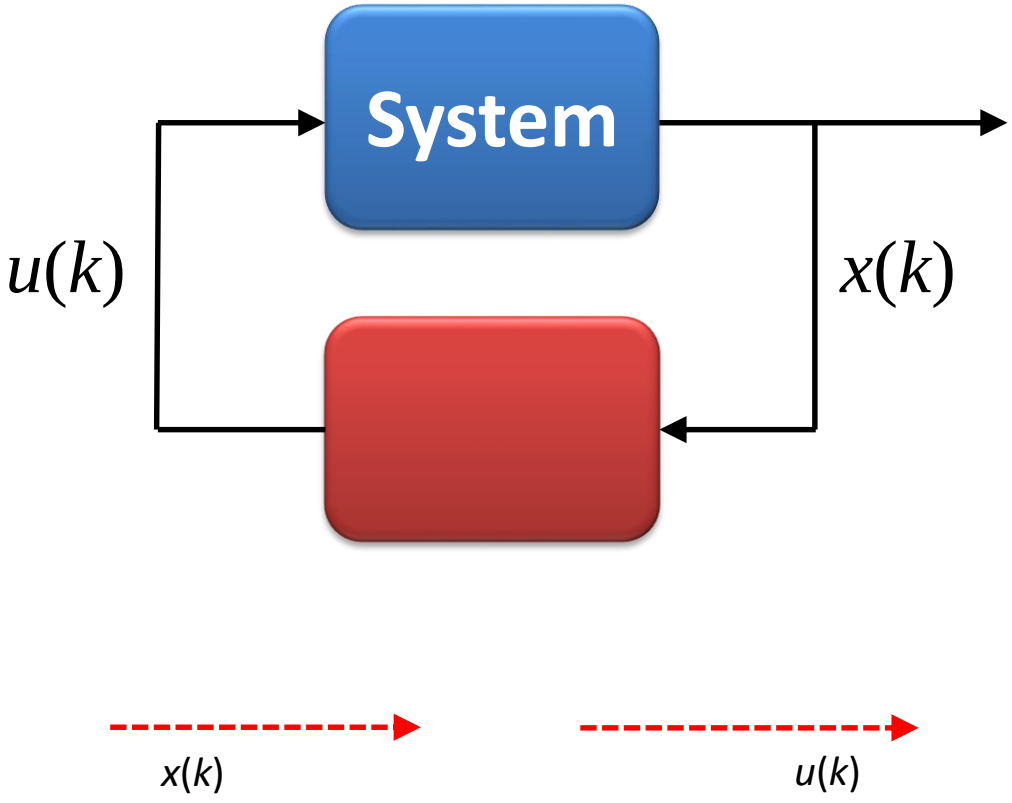
How to use it?

- MATLAB/Simulink: `rmpc_block` (*beginner*)
- MATLAB: CLI (*advanced*)

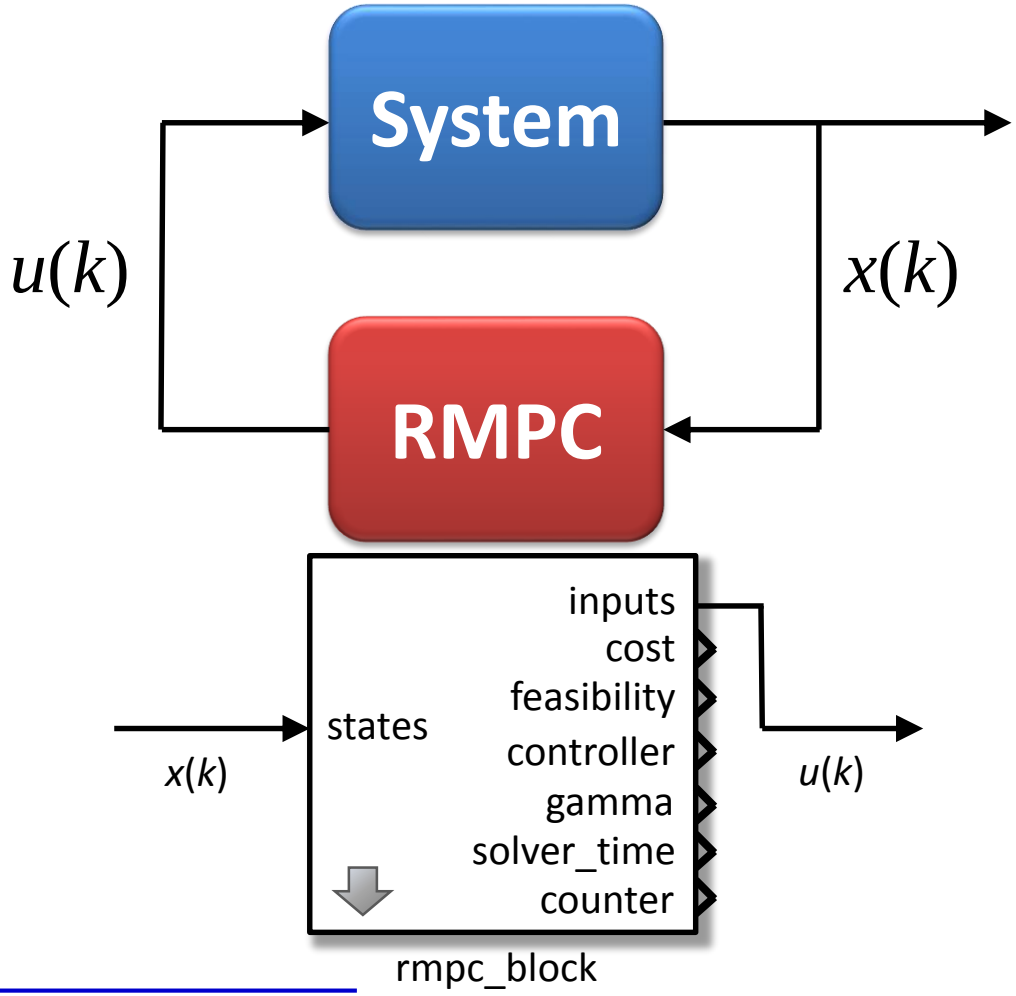
Simulink Block



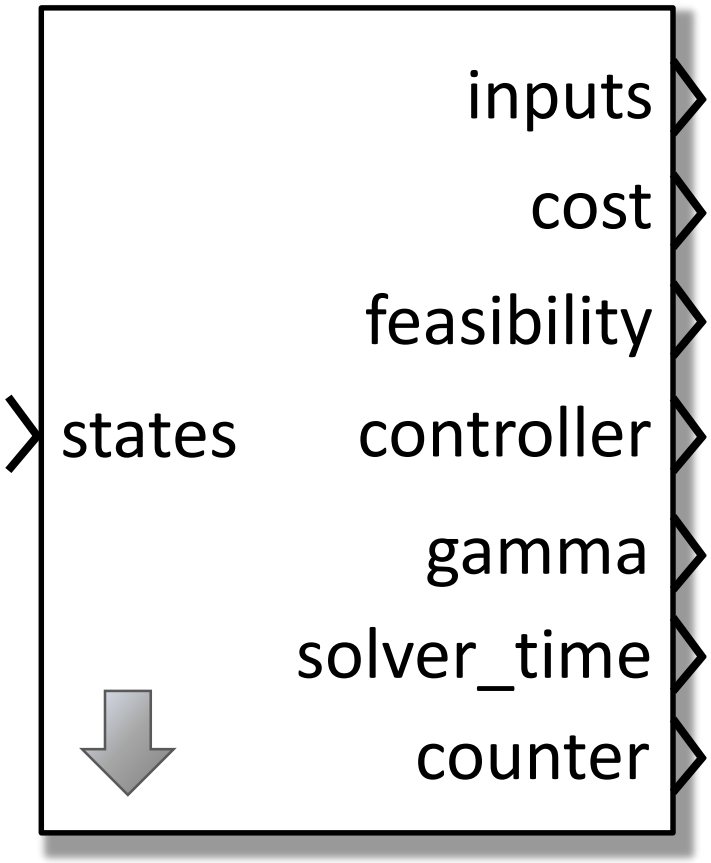
Simulink Block



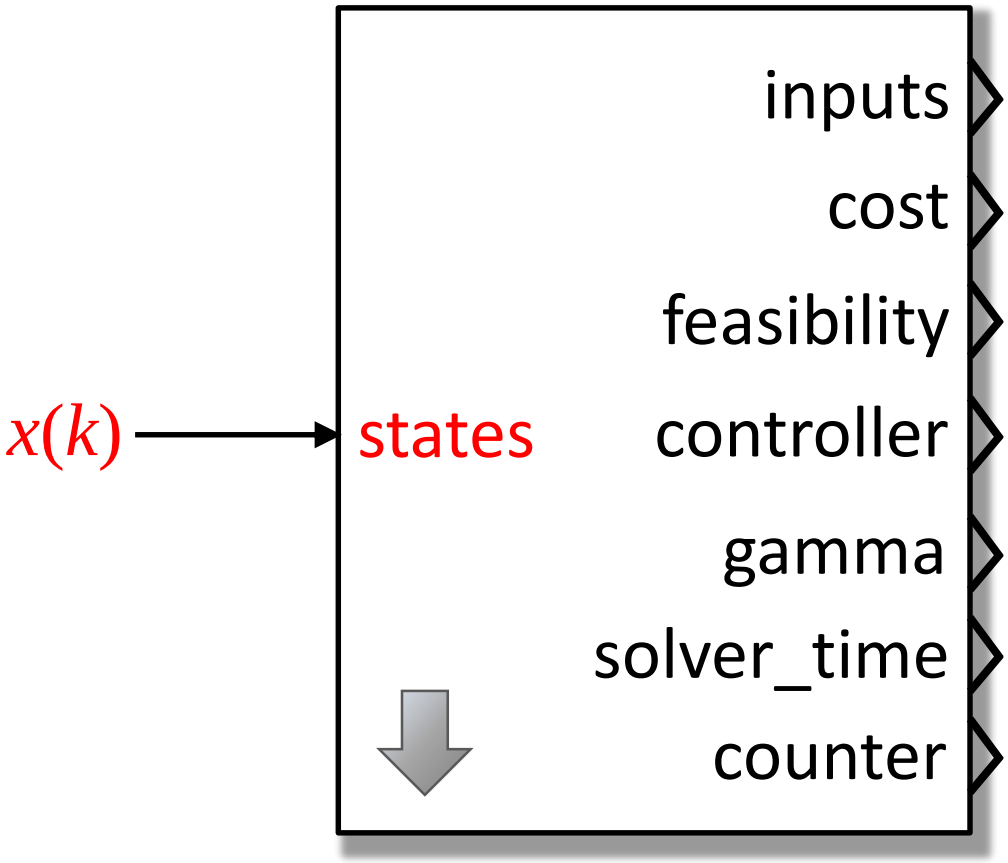
Simulink Block



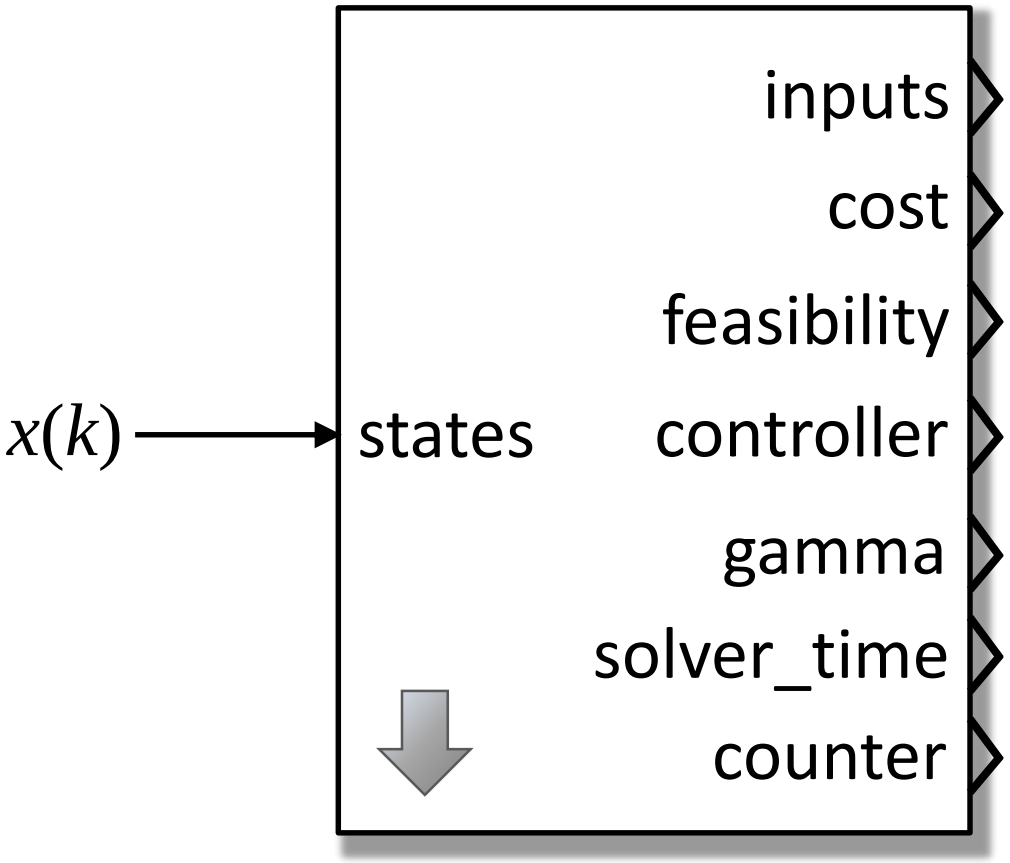
★ *bitbucket.org/oravec/mup*

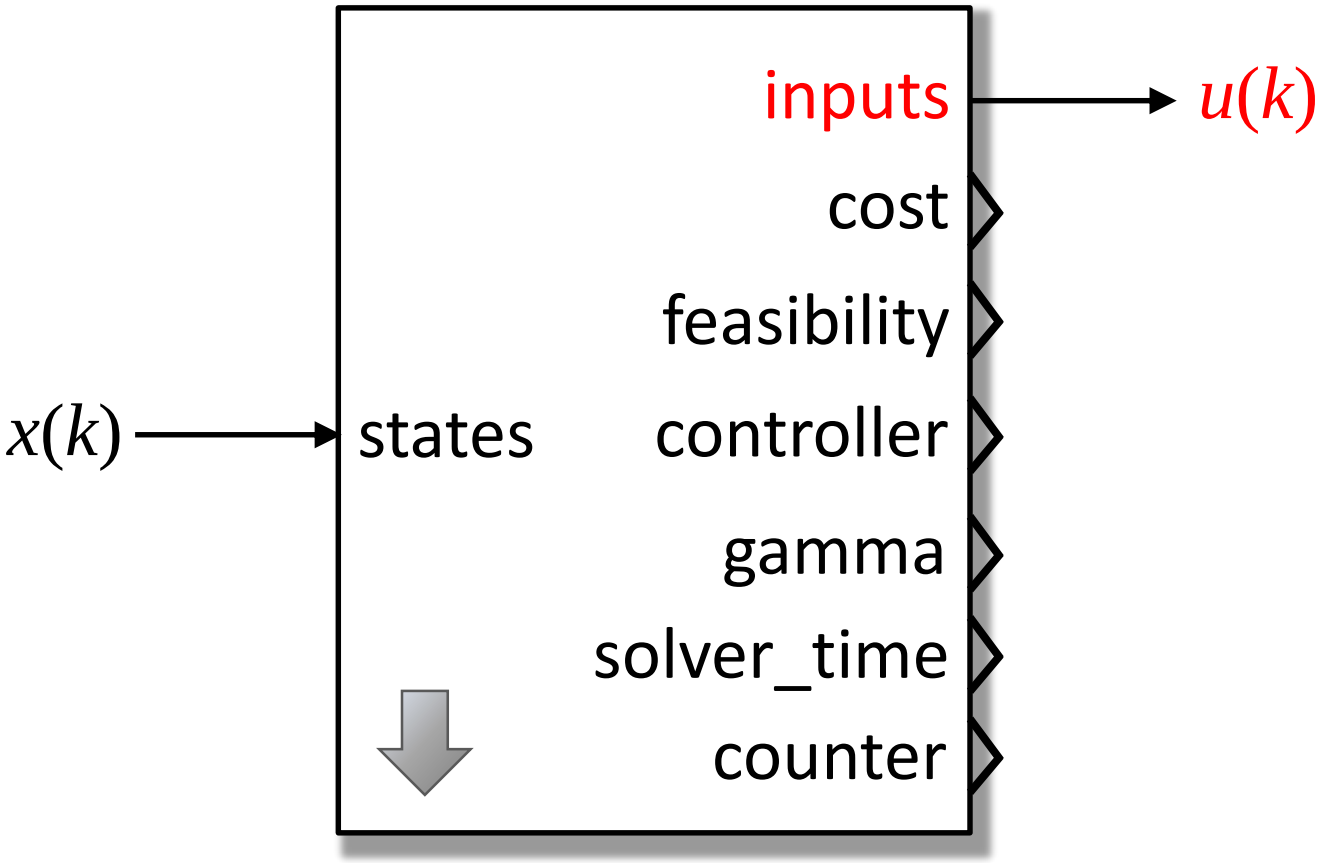

inputs
cost
feasibility
states
controller
gamma
solver_time
counter

rmpc_block

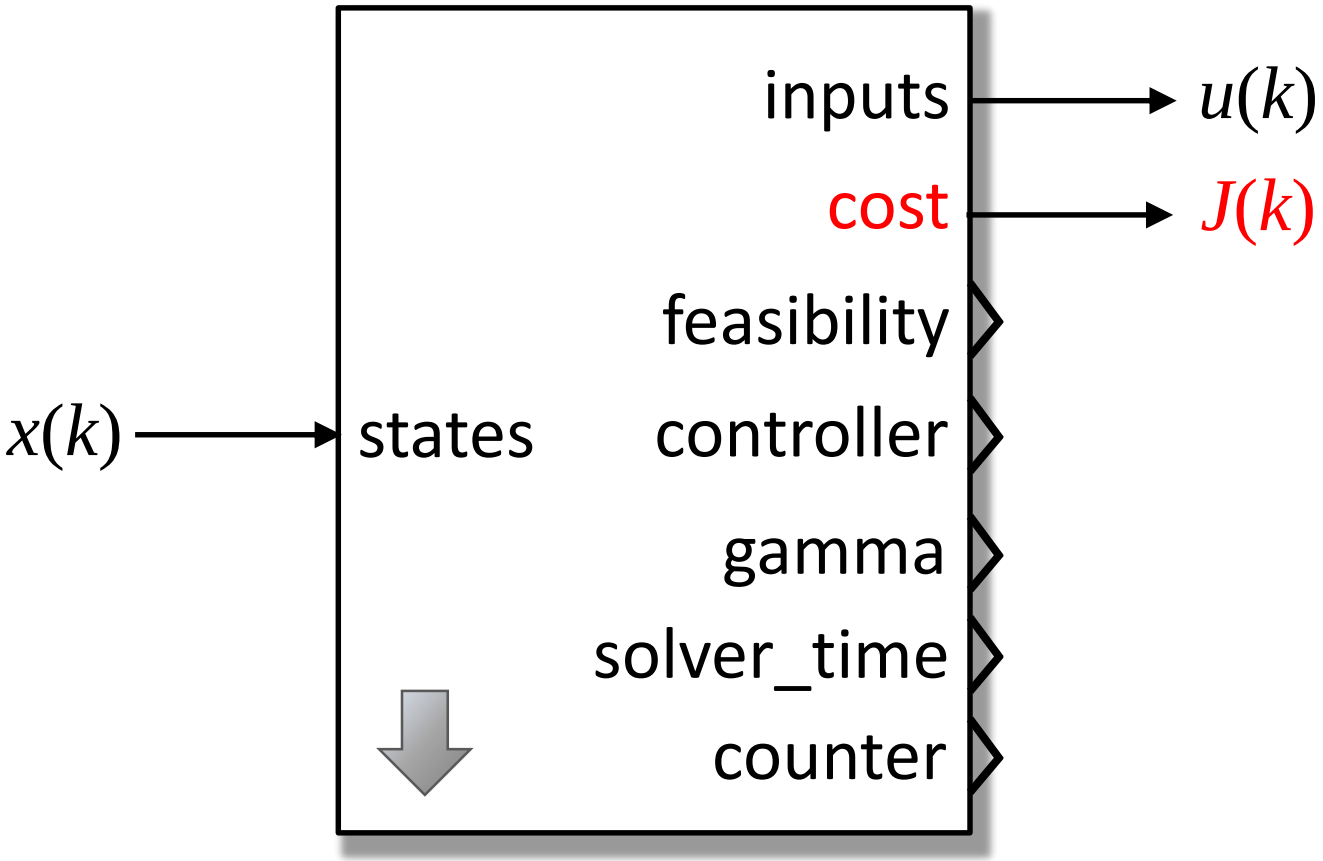

$x(k)$
states
inputs
cost
feasibility
controller
gamma
solver_time
counter

rmpc_block

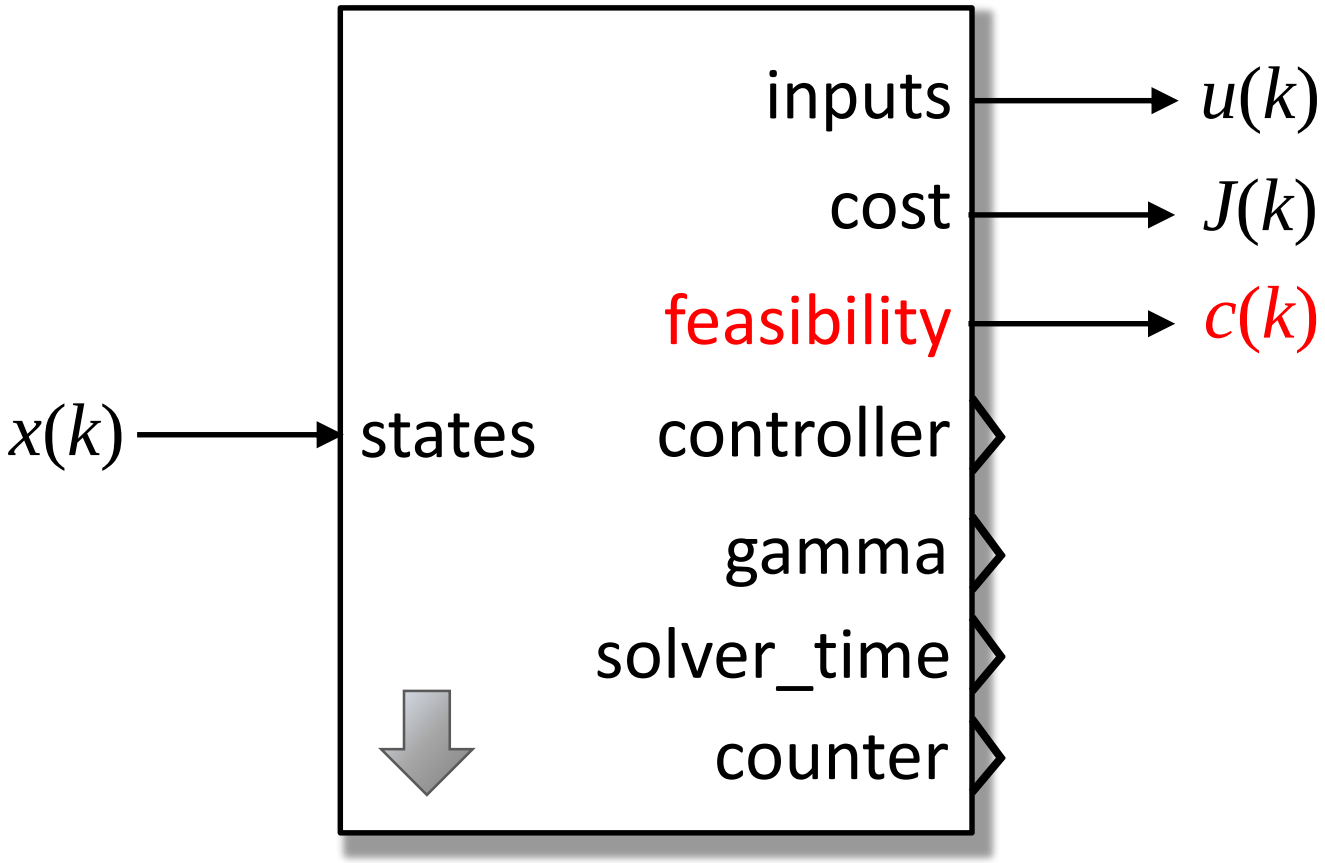

$x(k)$
states
inputs
cost
feasibility
controller
gamma
solver_time
counter
rmpc_block

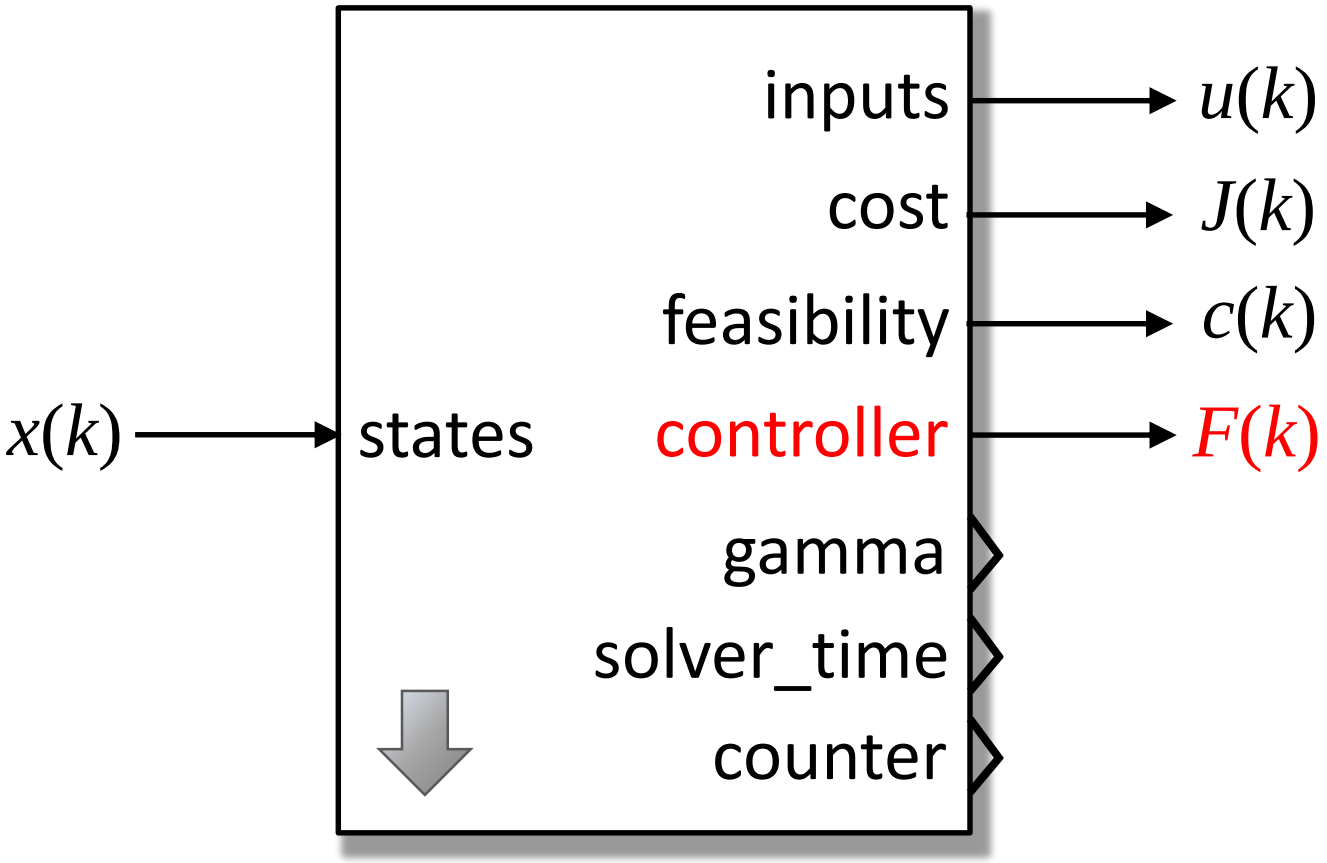


Control law: $u(k) = F(x(k))\, x(k)$

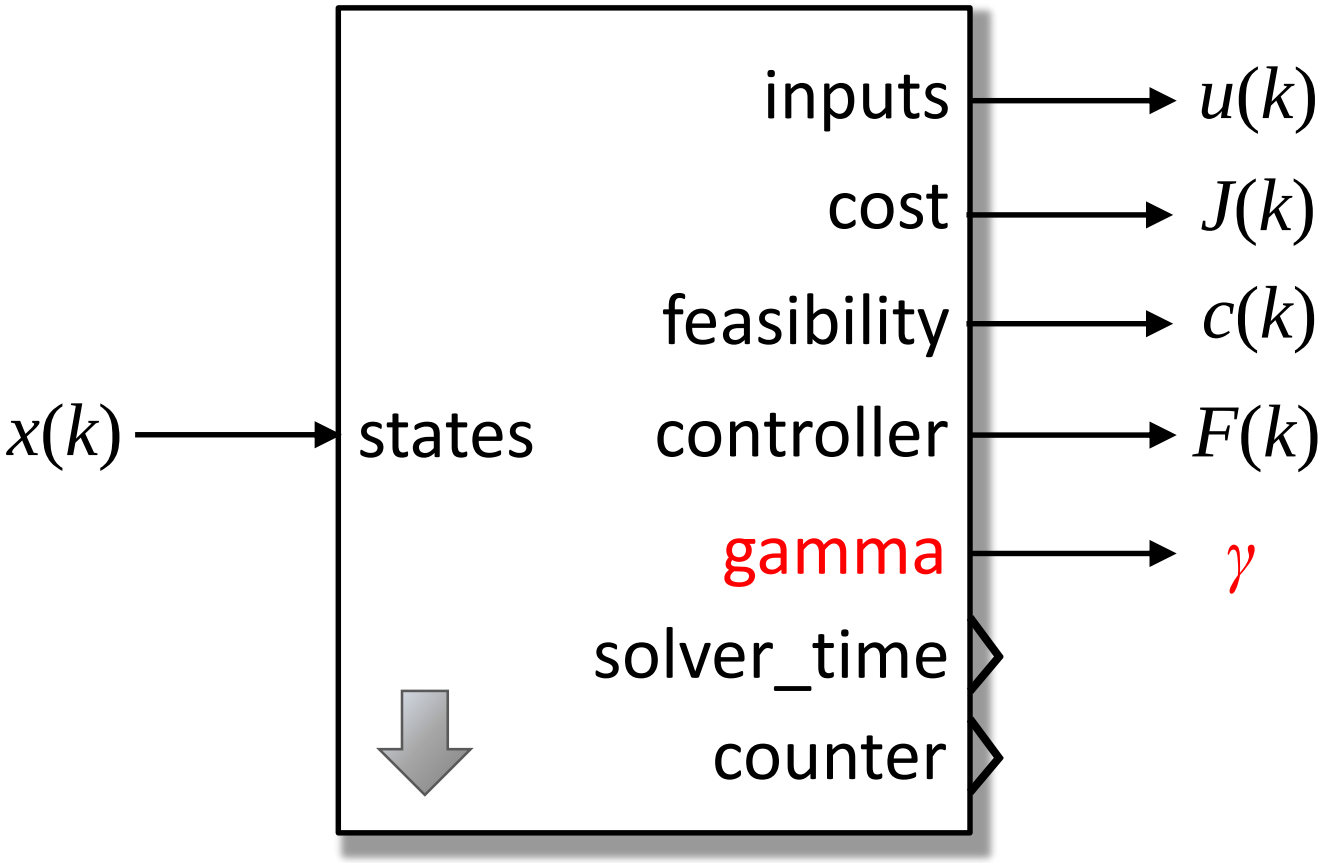


$$J(k) = \sum_{i=0}^{\infty} (x(k+i)^\top W_{\mathrm{x}}\, x(k+i) + u(k+i)^\top W_{\mathrm{u}}\, u(k+i))$$


inputs
cost
feasibility
controller
gamma
solver_time
counter
states
$x(k)$
$u(k)$
$J(k)$
$c(k)$
rmpc_block

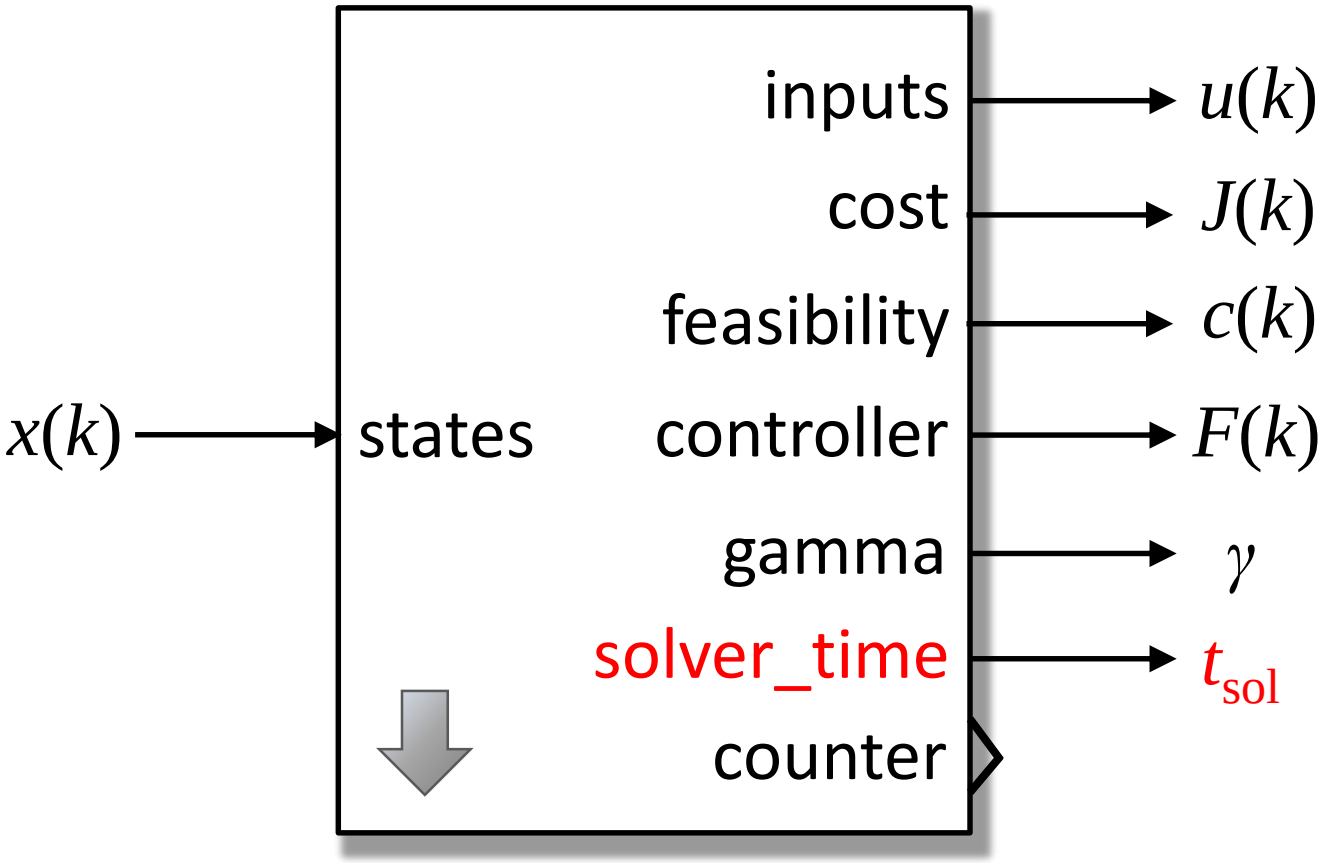


Control law: $u(k) = F(x(k))\, x(k)$

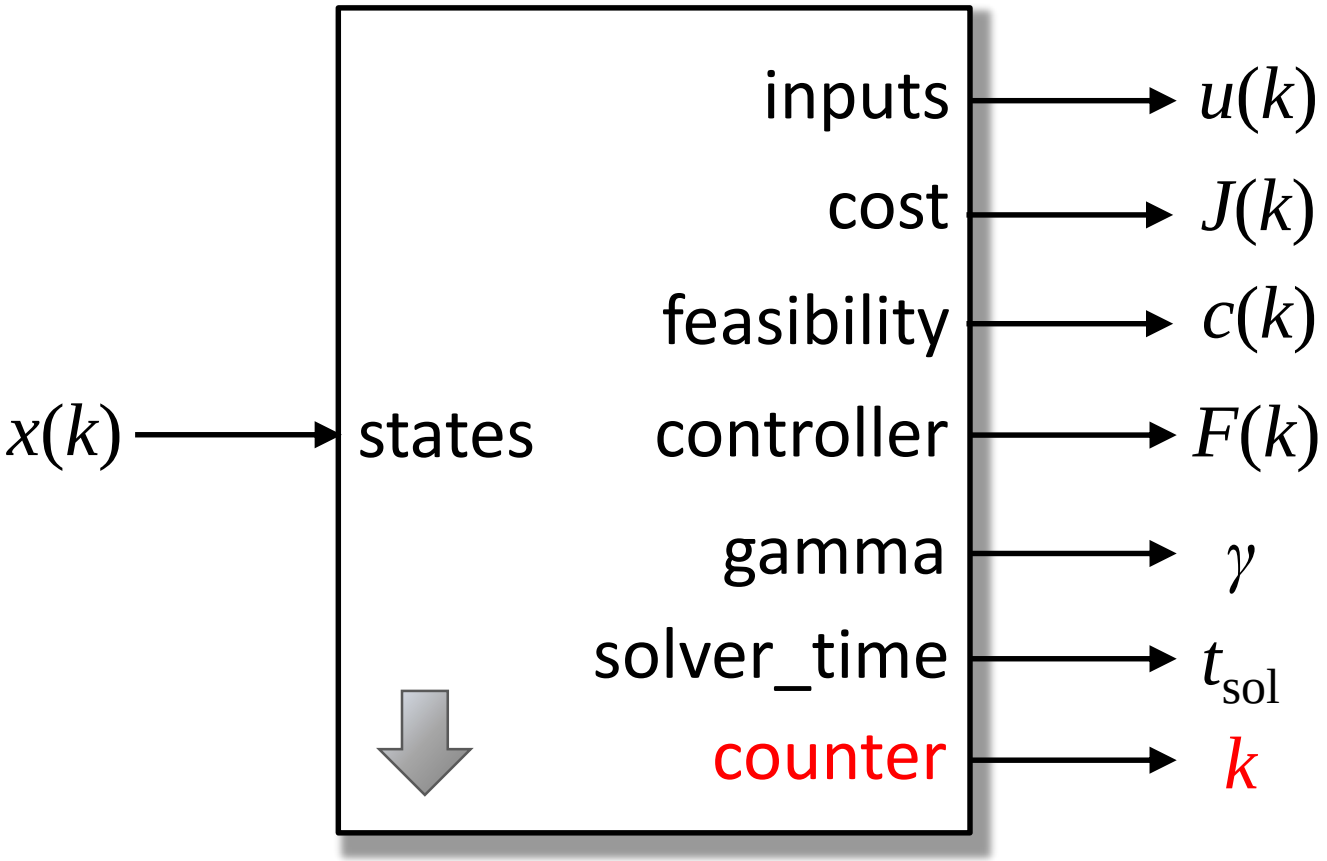


rmpc_block

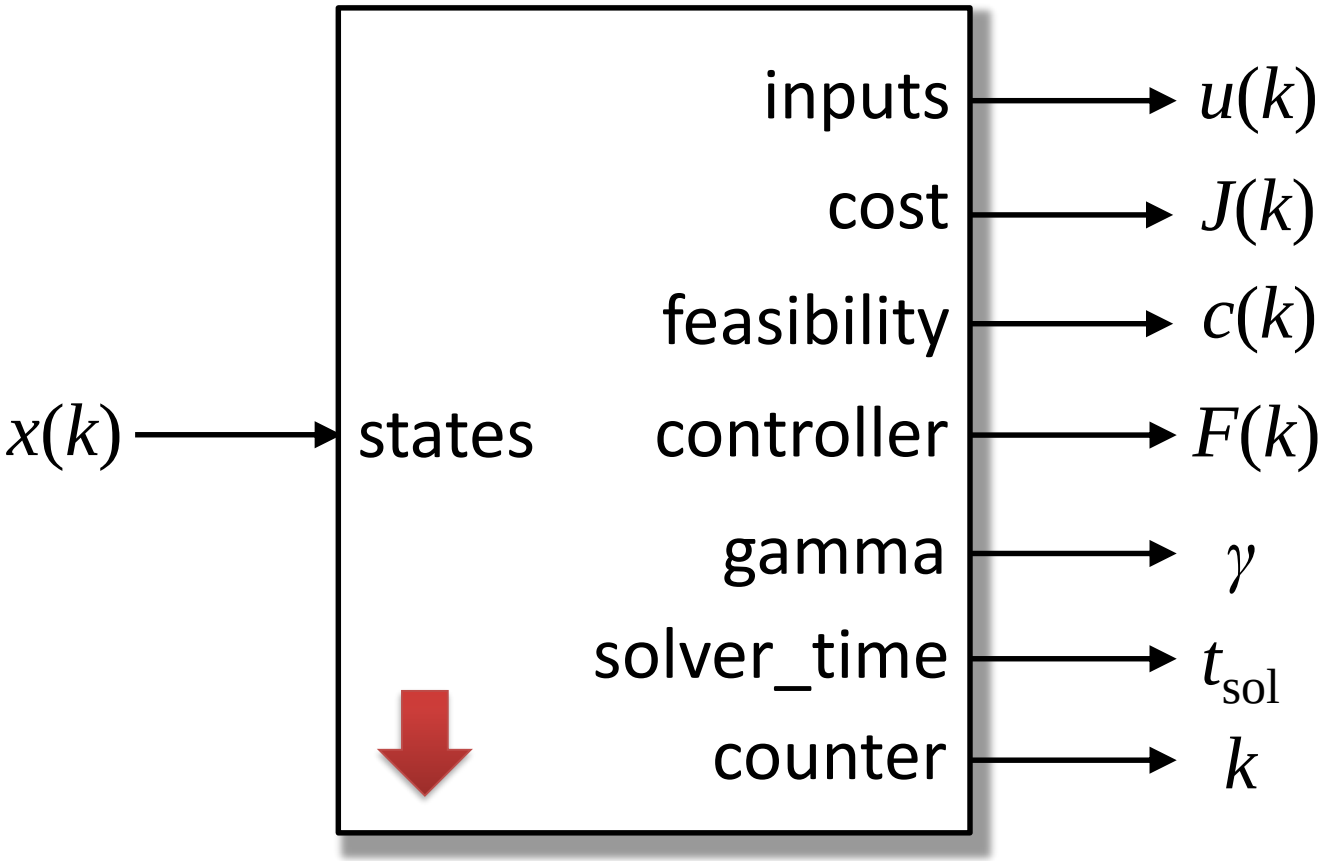
SDP: $\min\ \gamma, \quad \text{s.t.:}\ x \preceq_{\mathcal{K}} 0 \Rightarrow M(x) \prec 0$


$x(k)$
states
inputs
cost
feasibility
controller
gamma
solver_time
counter
$u(k)$
$J(k)$
$c(k)$
$F(k)$
γ
t_{sol}

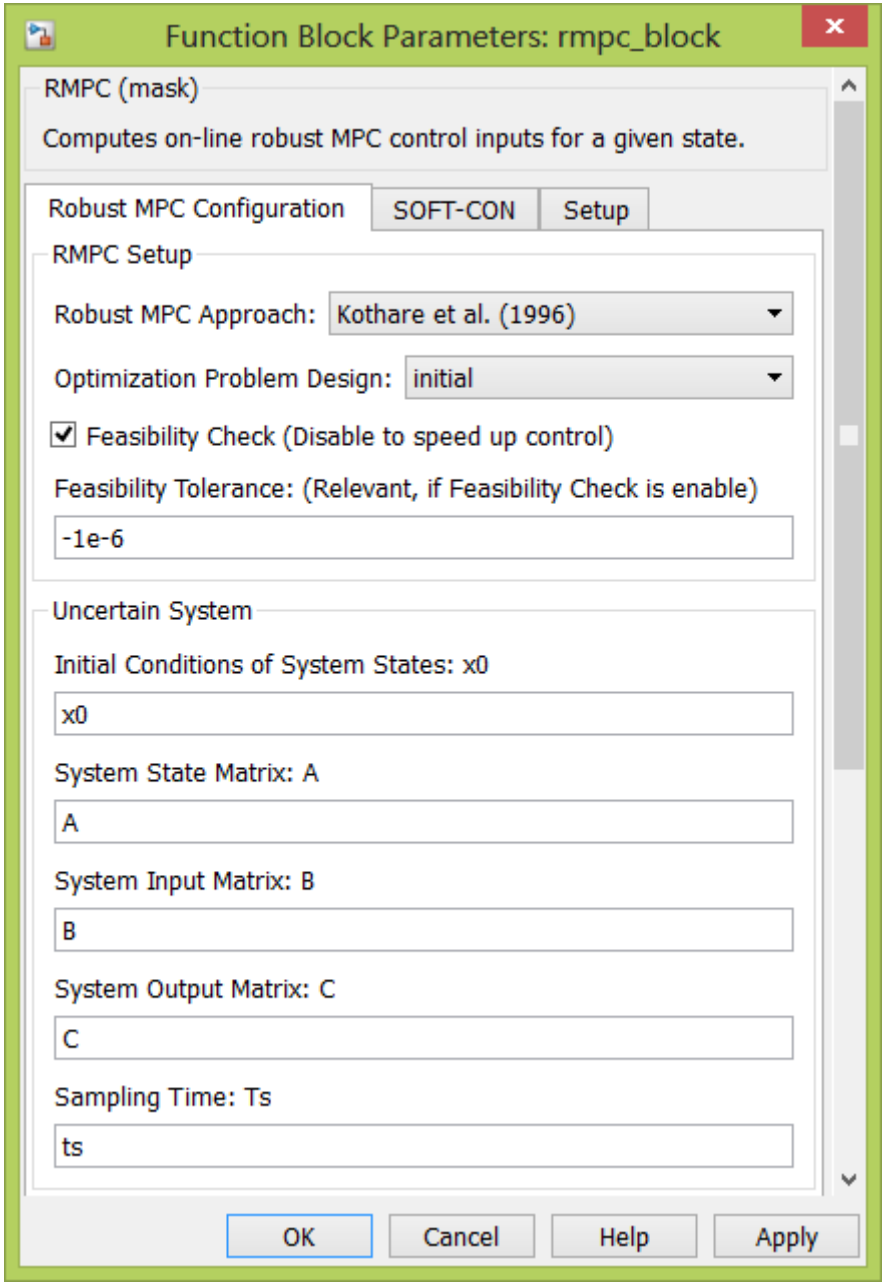
rmpc_block

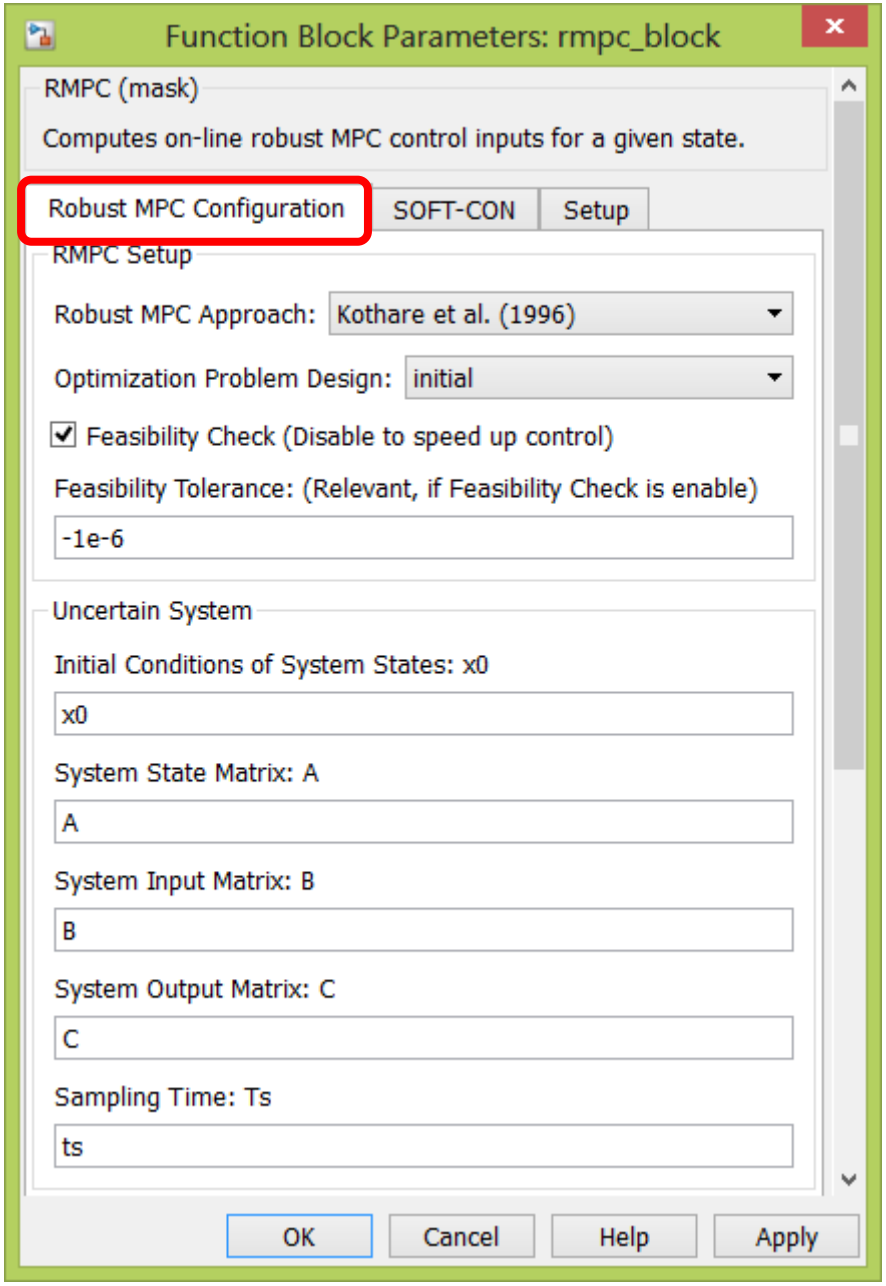

$x(k)$
states
inputs
cost
feasibility
controller
gamma
solver_time
counter
$u(k)$
$J(k)$
$c(k)$
$F(k)$
γ
t_{sol}
k

rmpc_block

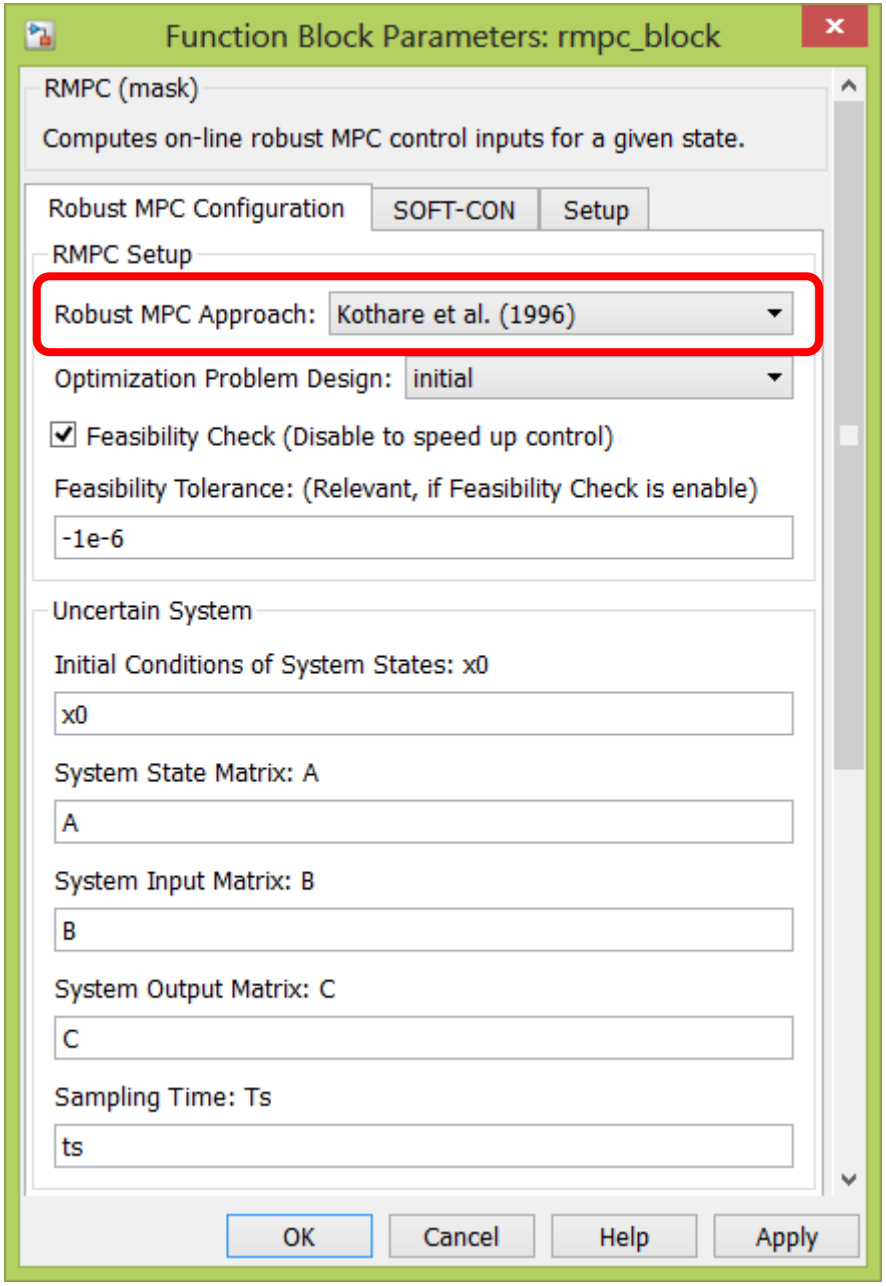

$x(k)$
states
inputs
cost
feasibility
controller
gamma
solver_time
counter
$u(k)$
$J(k)$
$c(k)$
$F(k)$
γ
t_{sol}
k

rmpc_block

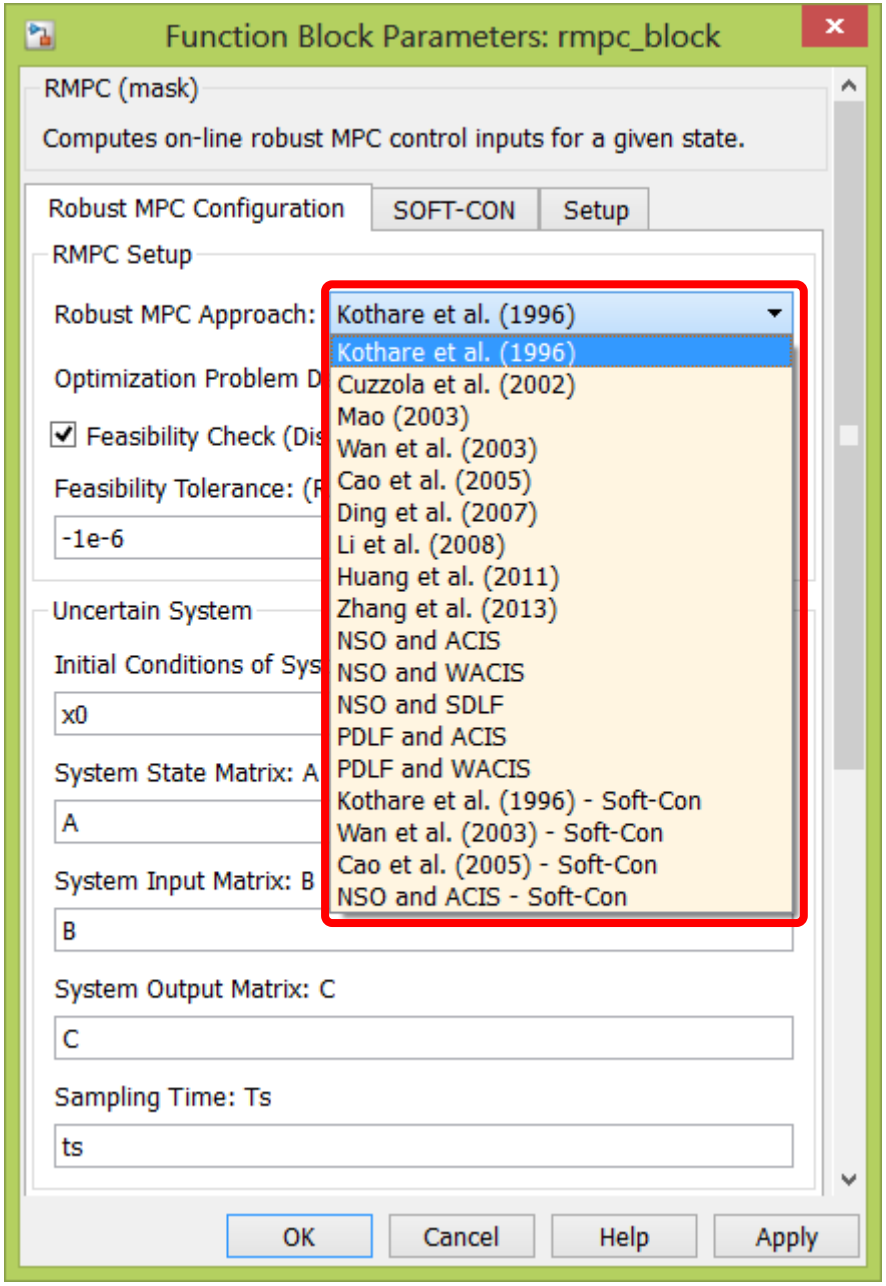

Function Block Parameters: rmpc_block
RMPC (mask)
Computes on-line robust MPC control inputs for a given state.
Robust MPC Configuration
SOFT-CON
Setup
RMPC Setup
Robust MPC Approach:
Kothare et al. (1996)
Optimization Problem Design:
initial
Feasibility Check (Disable to speed up control)
Feasibility Tolerance: (Relevant, if Feasibility Check is enable)
-1e-6
Uncertain System
Initial Conditions of System States: x0
x0
System State Matrix: A
A
System Input Matrix: B
B
System Output Matrix: C
C
Sampling Time: Ts
ts
OK
Cancel
Help
Apply



RMPC Configuration

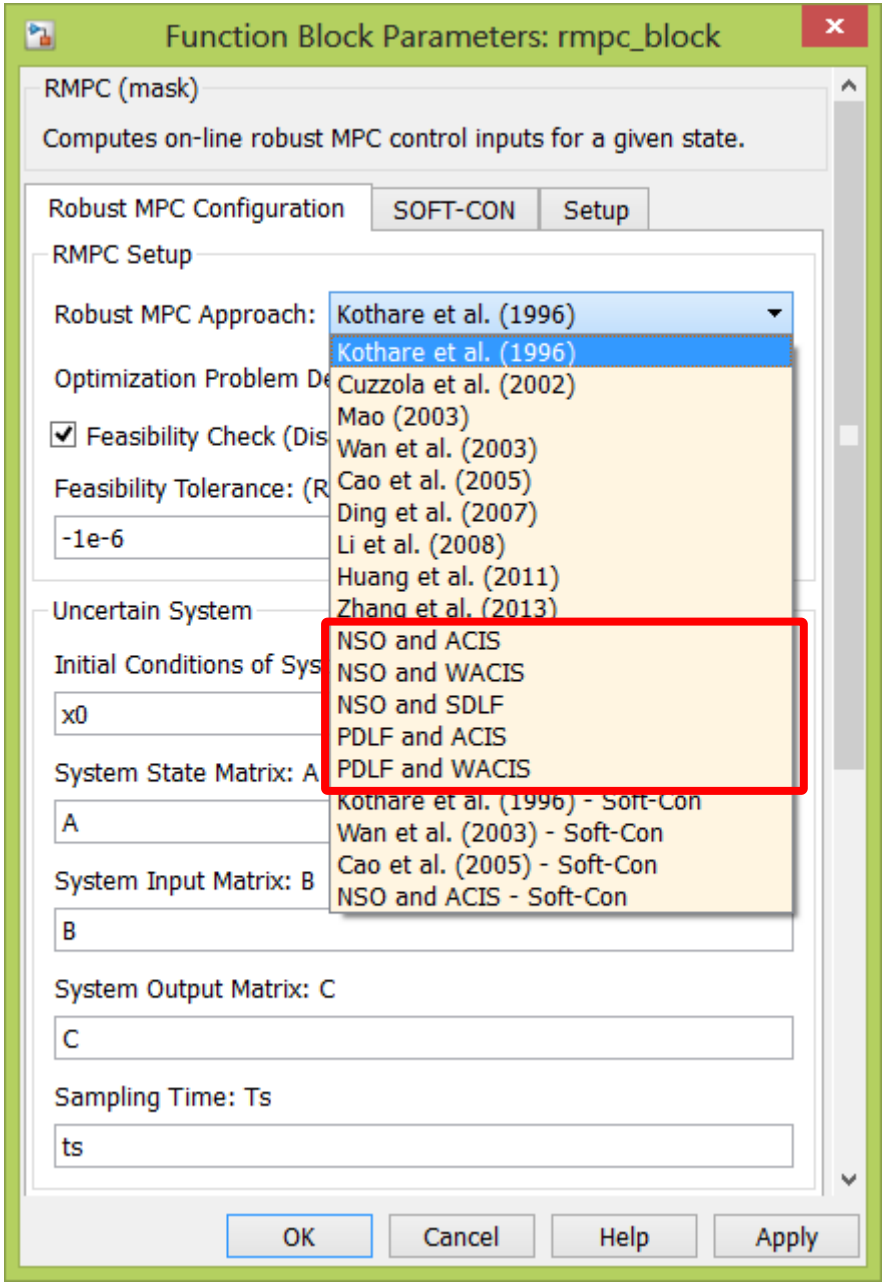


RMPC Approaches



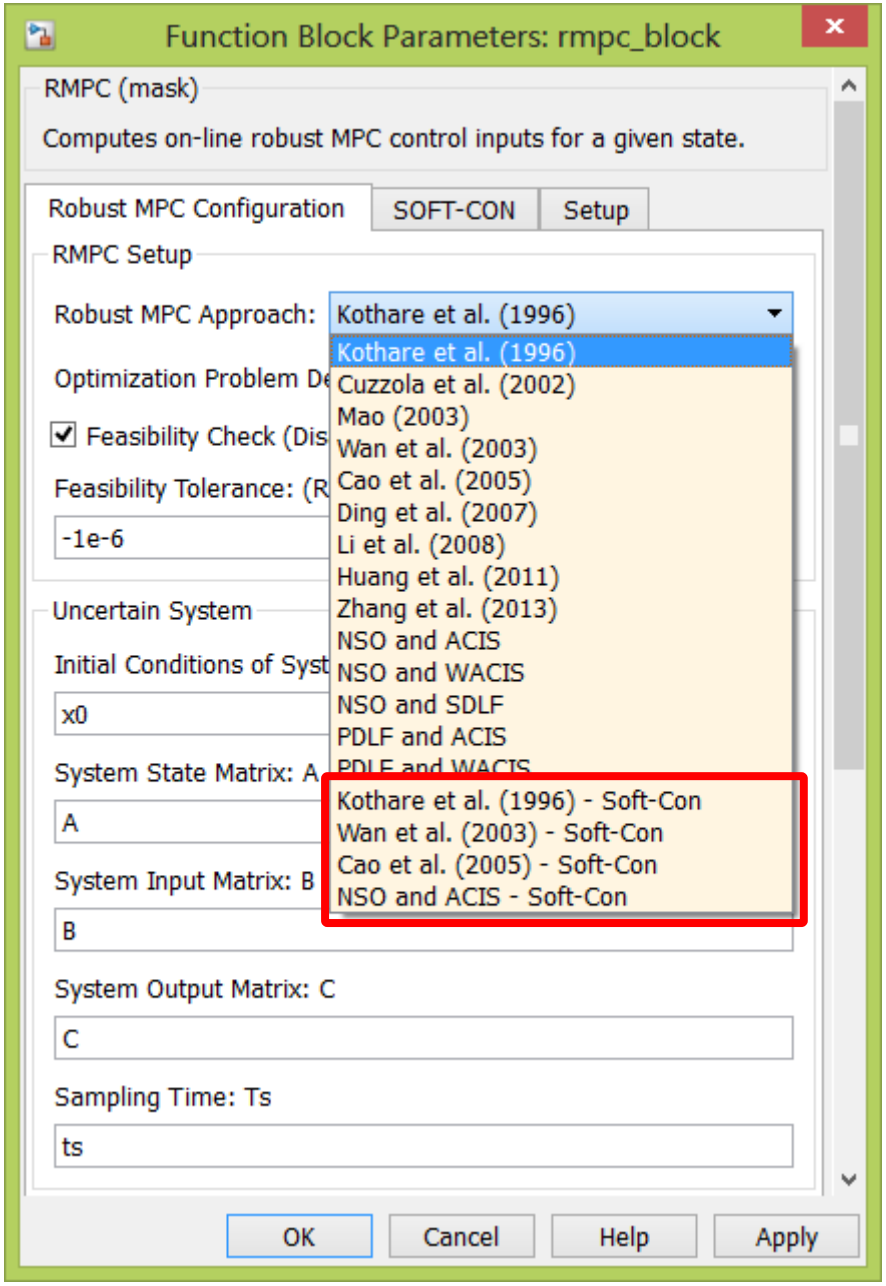
RMPC Approaches

Kothare et al. (1996)
Cuzzola et al. (2002)
Mao (2003)
Wan et al. (2003)
Ding et al. (2007)
Li et al. (2008)
Cao et Li (2005)
Huang et al. (2011)
Zhang et al. (2013)



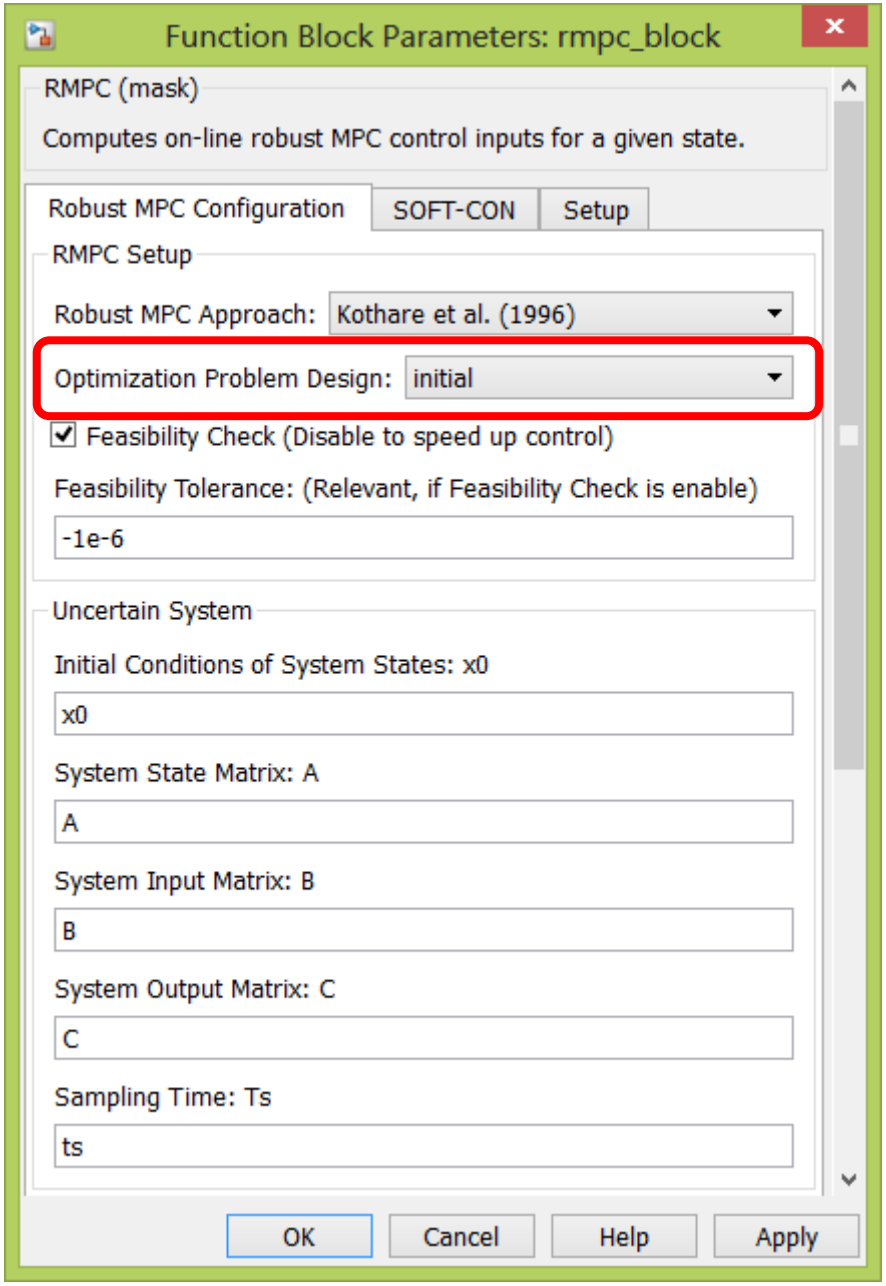
RMPC Approaches

Kothare et al. (1996)
Cuzzola et al. (2002)
Mao (2003)
Wan et al. (2003)
Ding et al. (2007)
Li et al. (2008)
Cao et Li (2005)
Huang et al. (2011)
Zhang et al. (2013)
NSO and ACIS
NSO and WACIS
NSO and SDLF
PDLF and ACIS
PDLF and WACIS

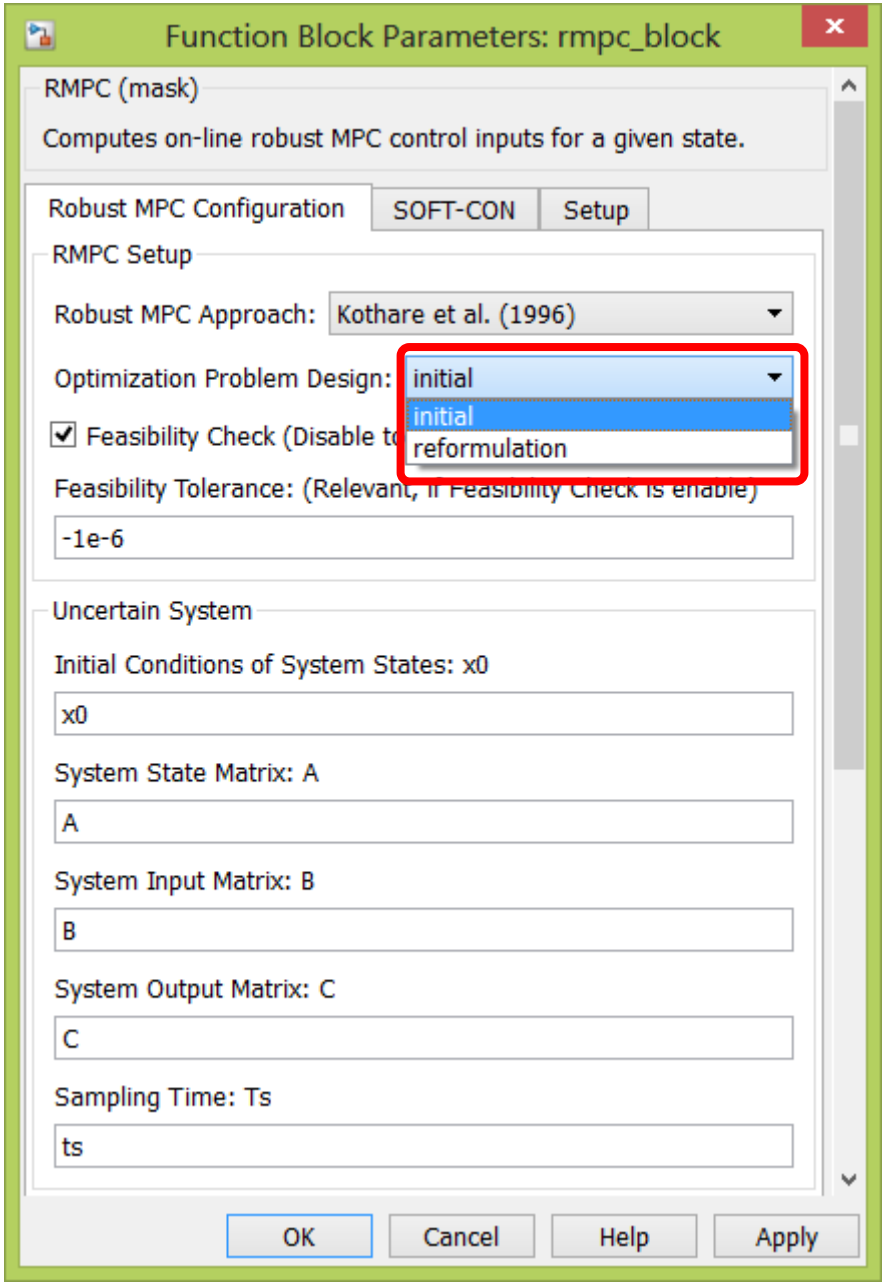


RMPC Approaches

Kothare et al. (1996)
Cuzzola et al. (2002)
Mao (2003)
Wan et al. (2003)
Ding et al. (2007)
Li et al. (2008)
Cao et Li (2005)
Huang et al. (2011)
Zhang et al. (2013)
NSO and ACIS
NSO and WACIS
NSO and SDLF
PDLF and ACIS
PDLF and WACIS
Kothare et al. (1996) – *Soft-Con*
Wan et al. (2003) – *Soft-Con*
Cao et Li (2005) – *Soft-Con*
NSO and ACIS – Soft-Con

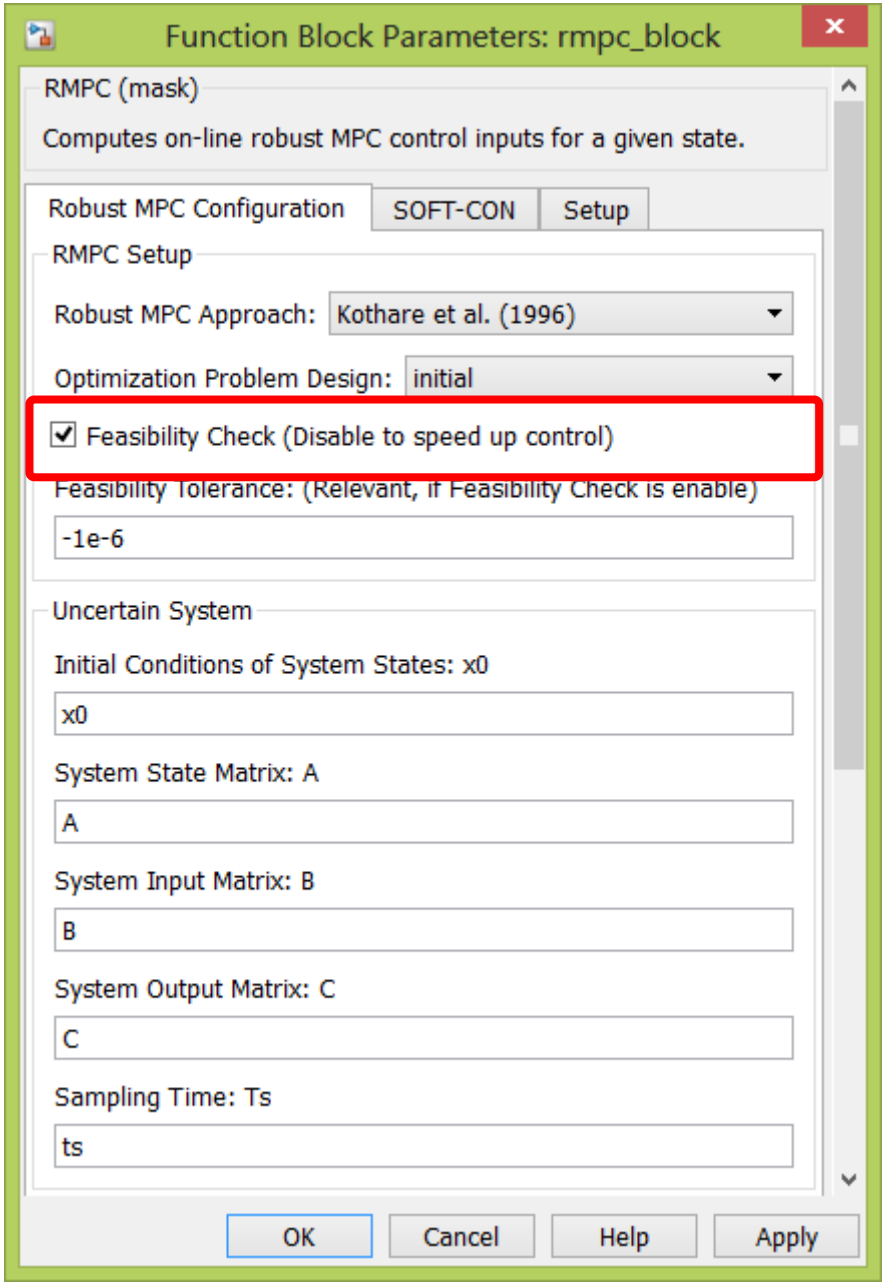


Opt. Problem Design



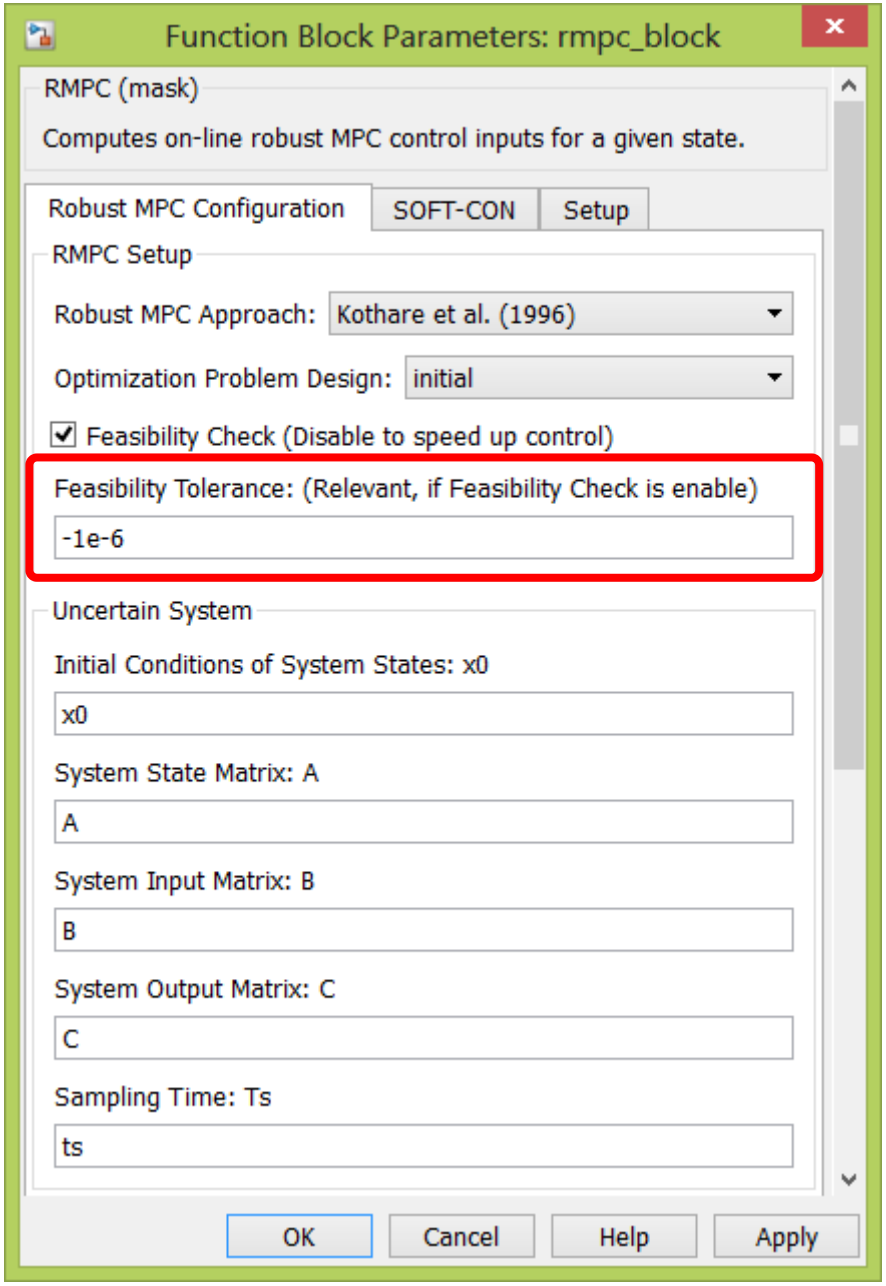
Opt. Problem Design

- initial: *speed up* (YALMIP/optimizer)
- reformulation: *analysis* (YALMIP/optimize)



Feasibility Check

enable

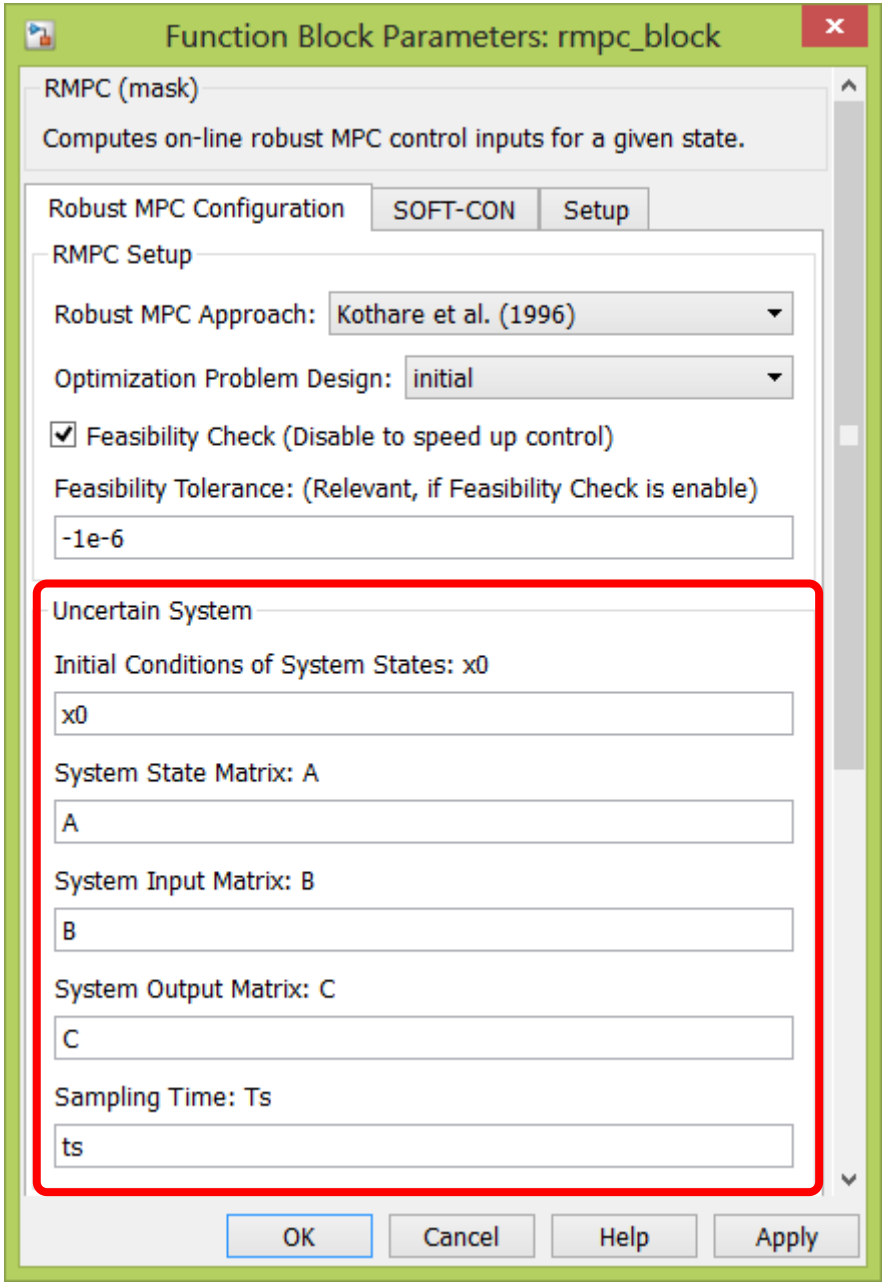


Feasibility Tolerance

- *default:* $\alpha = -1 \times 10^{-6}$
- SDP:

$$\min \ \gamma$$

$$\text{s.t.:} \ \ x \preceq_{\mathcal{K}} 0 \Rightarrow M(x) \preceq \alpha$$

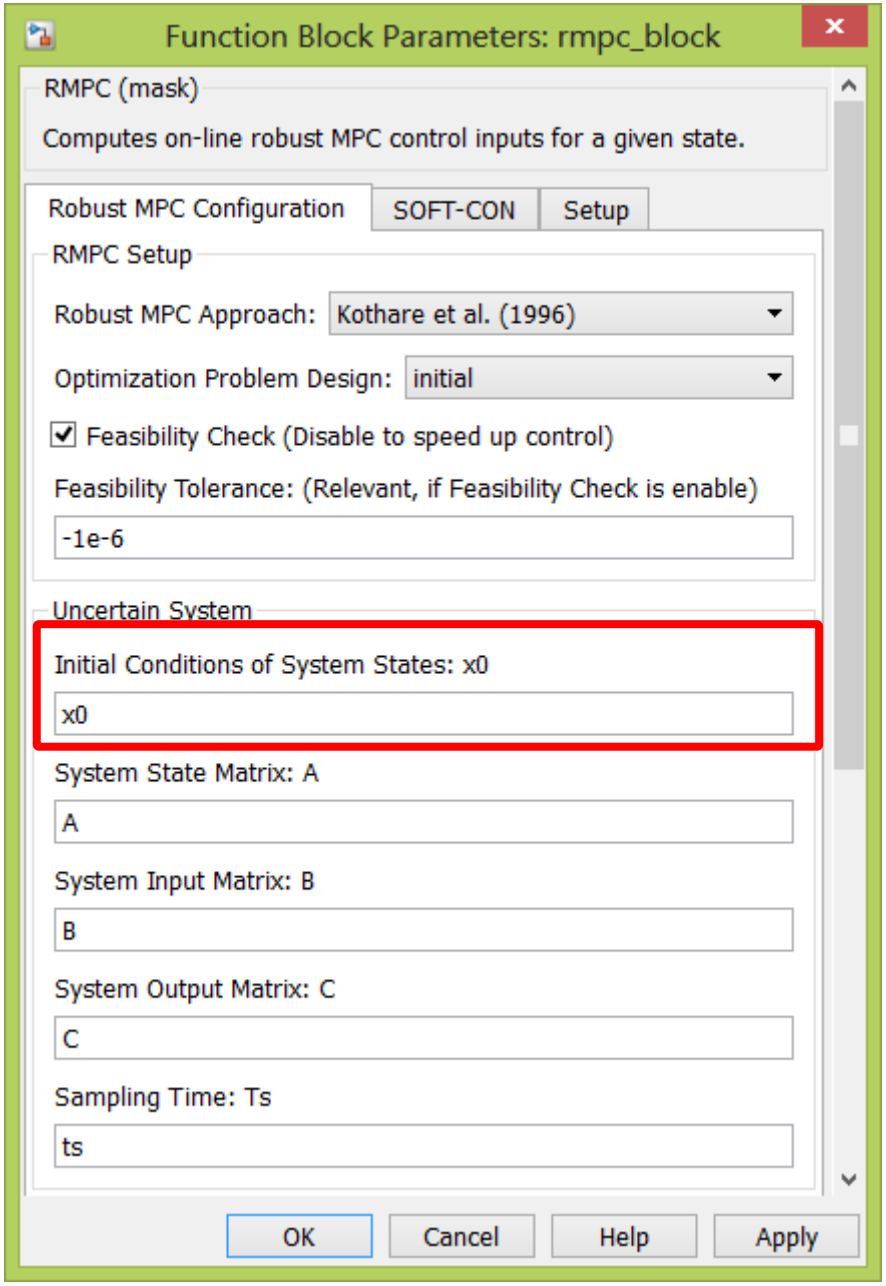


Uncertain System

$$x(k+1) = A\,x(k) + B\,u(k),$$

$$y(k) = C\,x(k), \quad x(0) = x_0,$$

$$[A, B] \in \text{convhull}\{[A_v, B_v]\}.$$

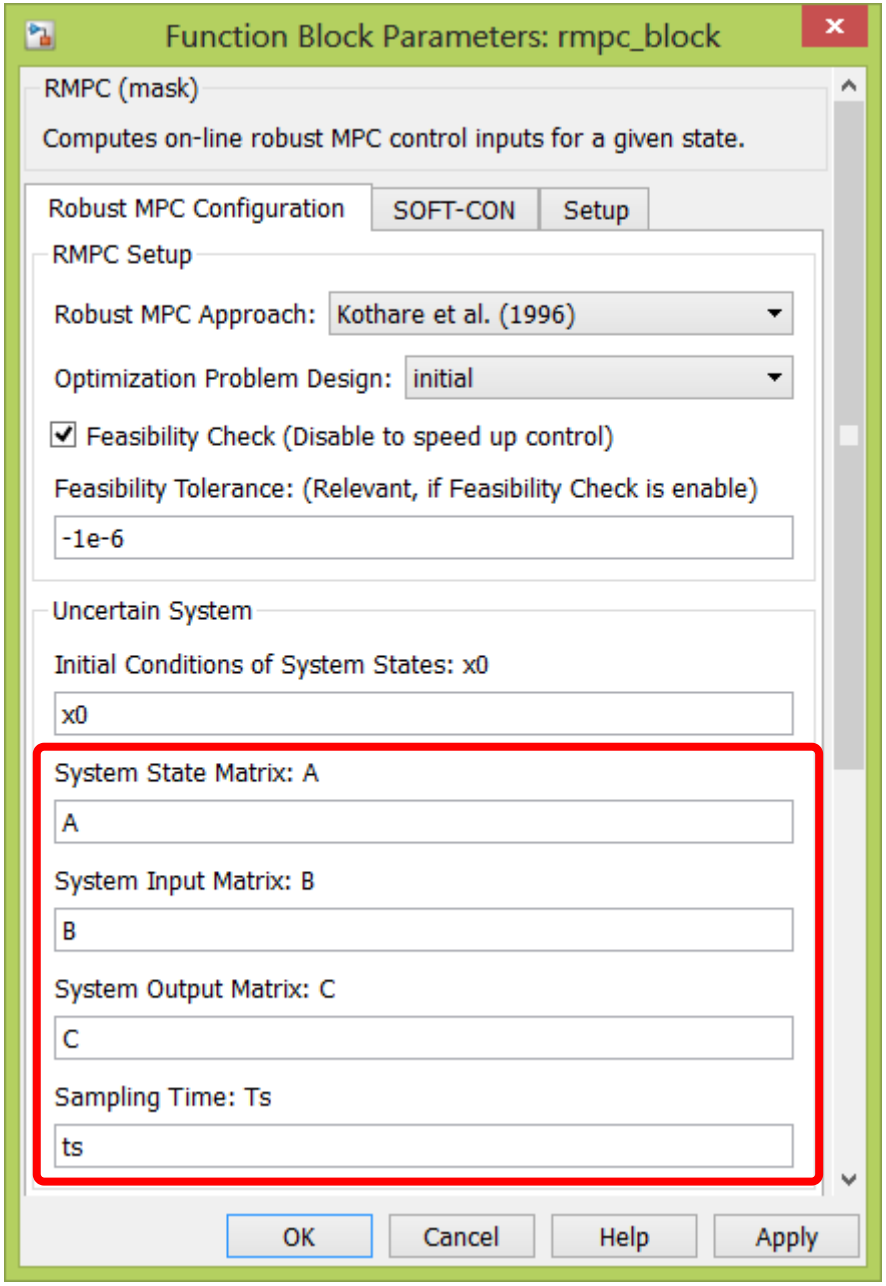


Uncertain System

$$x(k+1) = A\,x(k) + B\,u(k),$$

$$y(k) = C\,x(k), \quad x(0) = x_0,$$

$$[A, B] \in \text{convhull}\{[A_v, B_v]\}.$$

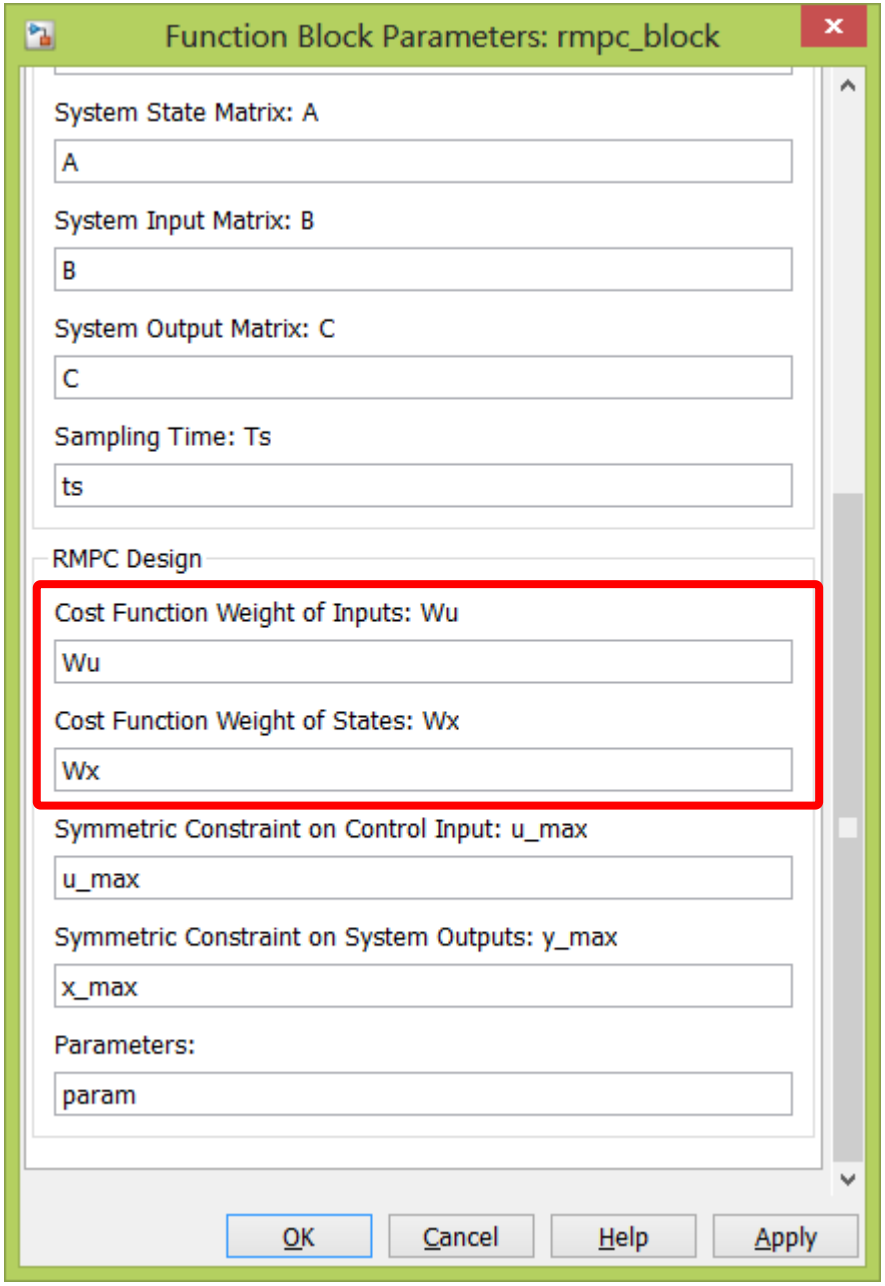


Uncertain System

$$x(k+1) = A\,x(k) + B\,u(k),$$

$$y(k) = C\,x(k), \quad x(0) = x_0,$$

$$[A, B] \in \text{convhull}\{[A_v, B_v]\}.$$

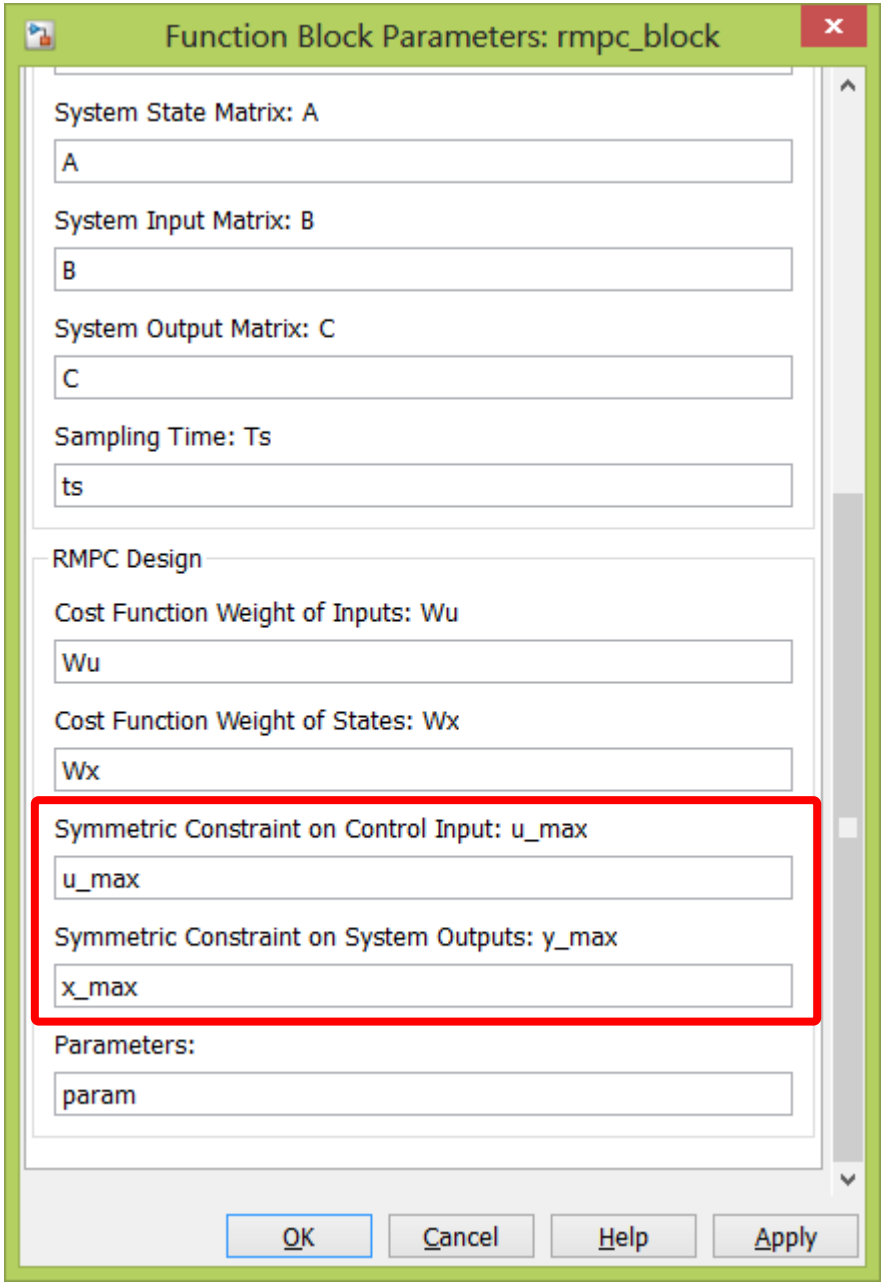


Quality criterion

$$J = \sum_{i=0}^{\infty} (\ell(x(k+i), u(k+i))) =$$

$$\sum_{i=0}^{\infty} (x(k+i)^\top W_\mathrm{x}\, x(k+i) +$$

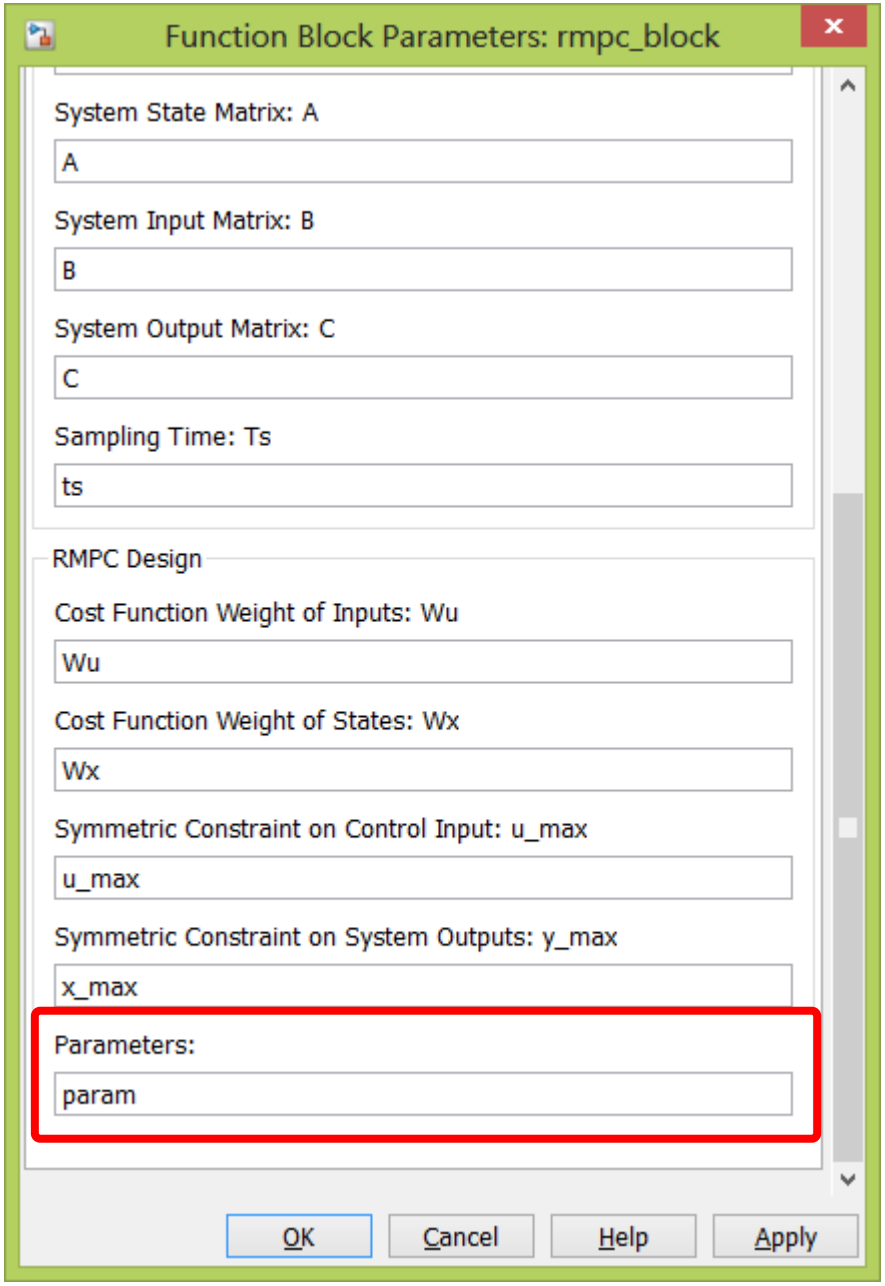
$$u(k+i)^\top W_\mathrm{u}\, u(k+i))$$



Quality criterion

$$\|u(k)\|_2^2 \preceq \|u_{\max}\|_2^2$$

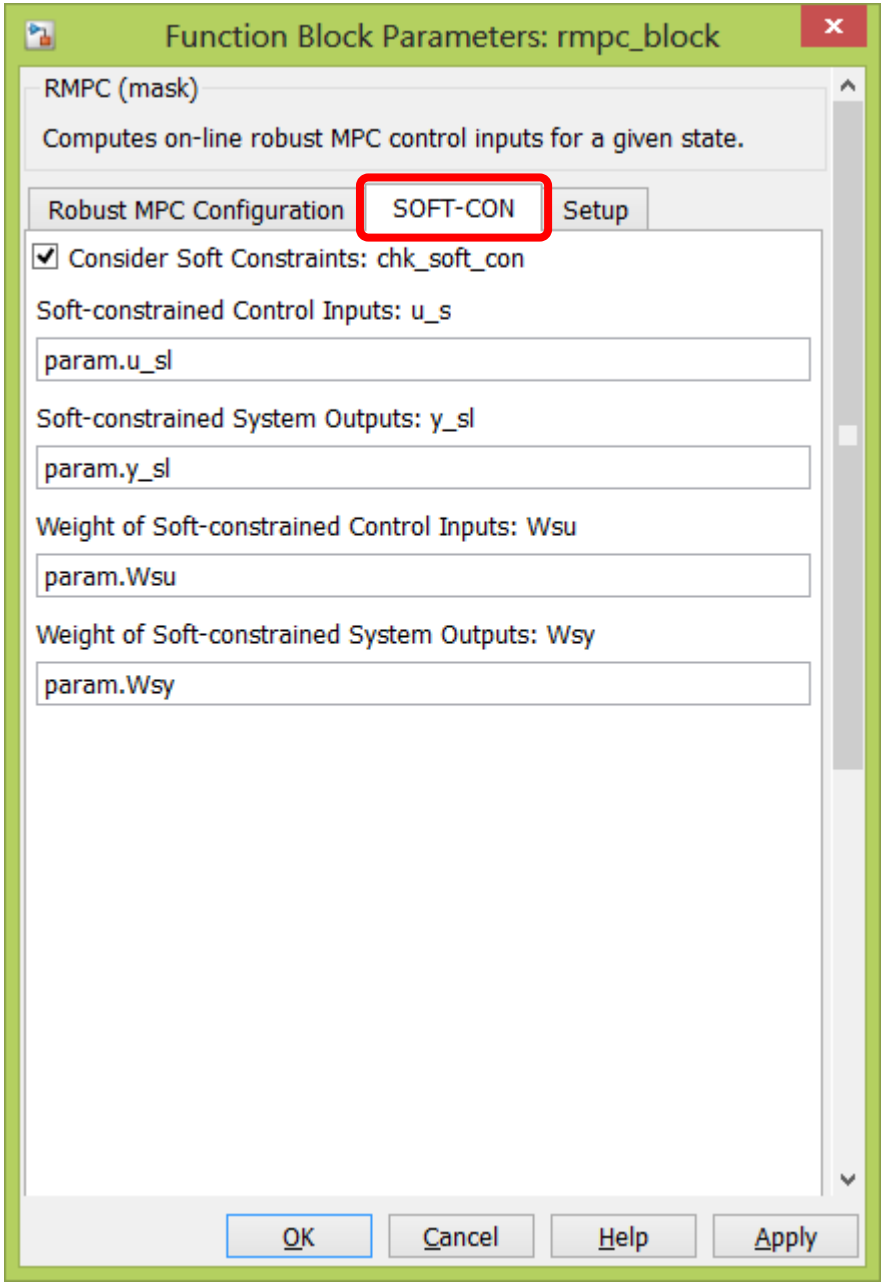
$$\|y(k)\|_2^2 \preceq \|y_{\max}\|_2^2$$



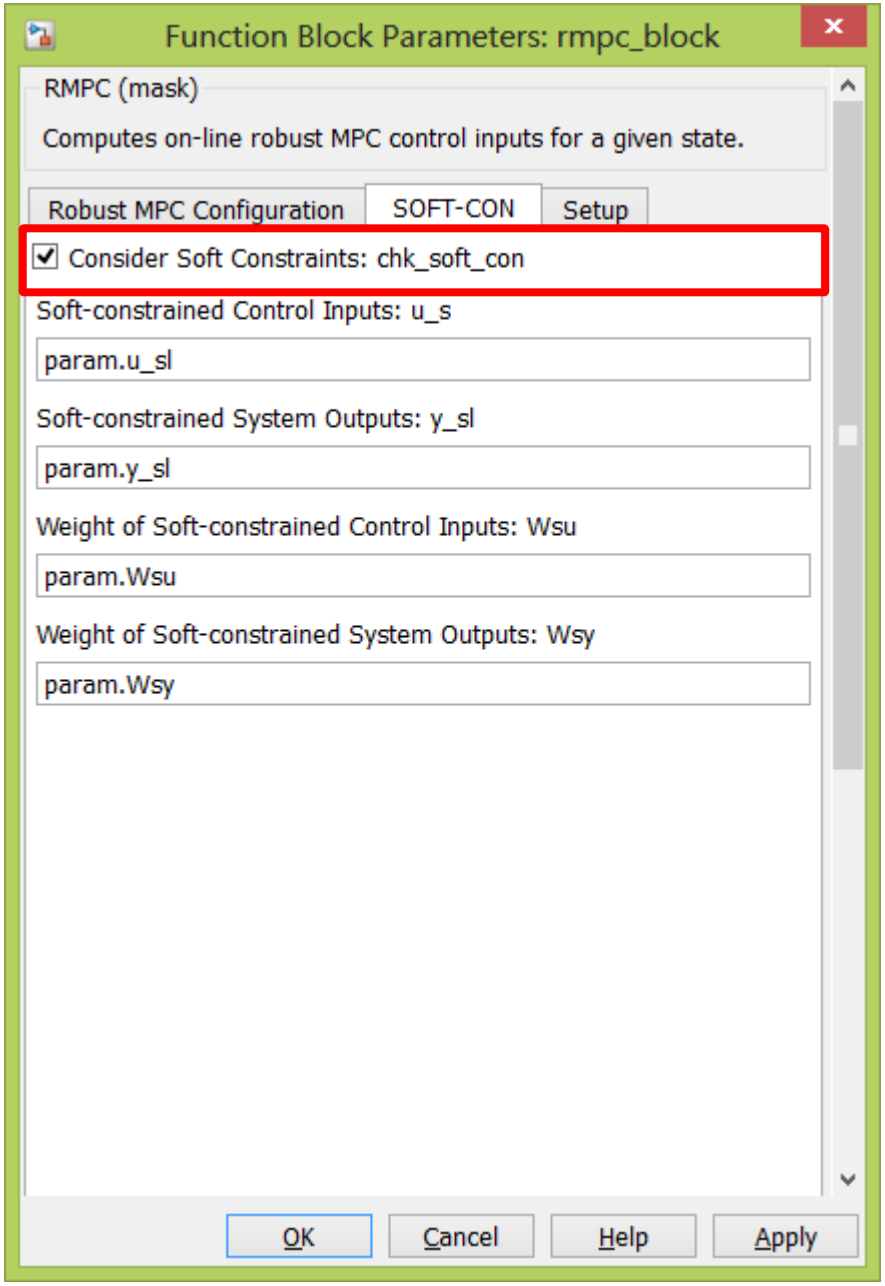
Parameters

optional RMPC design and tuning parameters

Soft Constraints

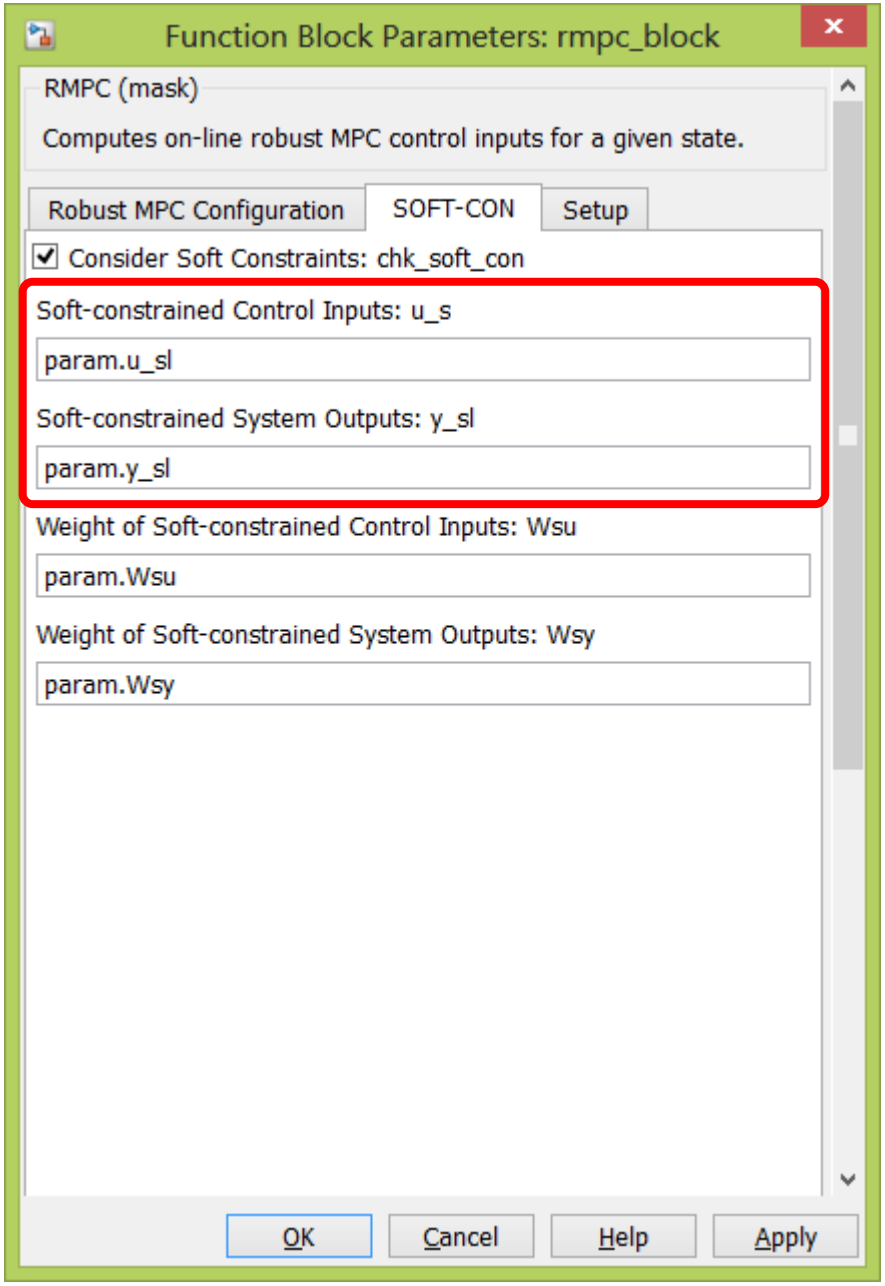


J. Oravec and M. Bakošová, *America Control Conference*, Boston, USA, 2016.



Soft Constraints

enable

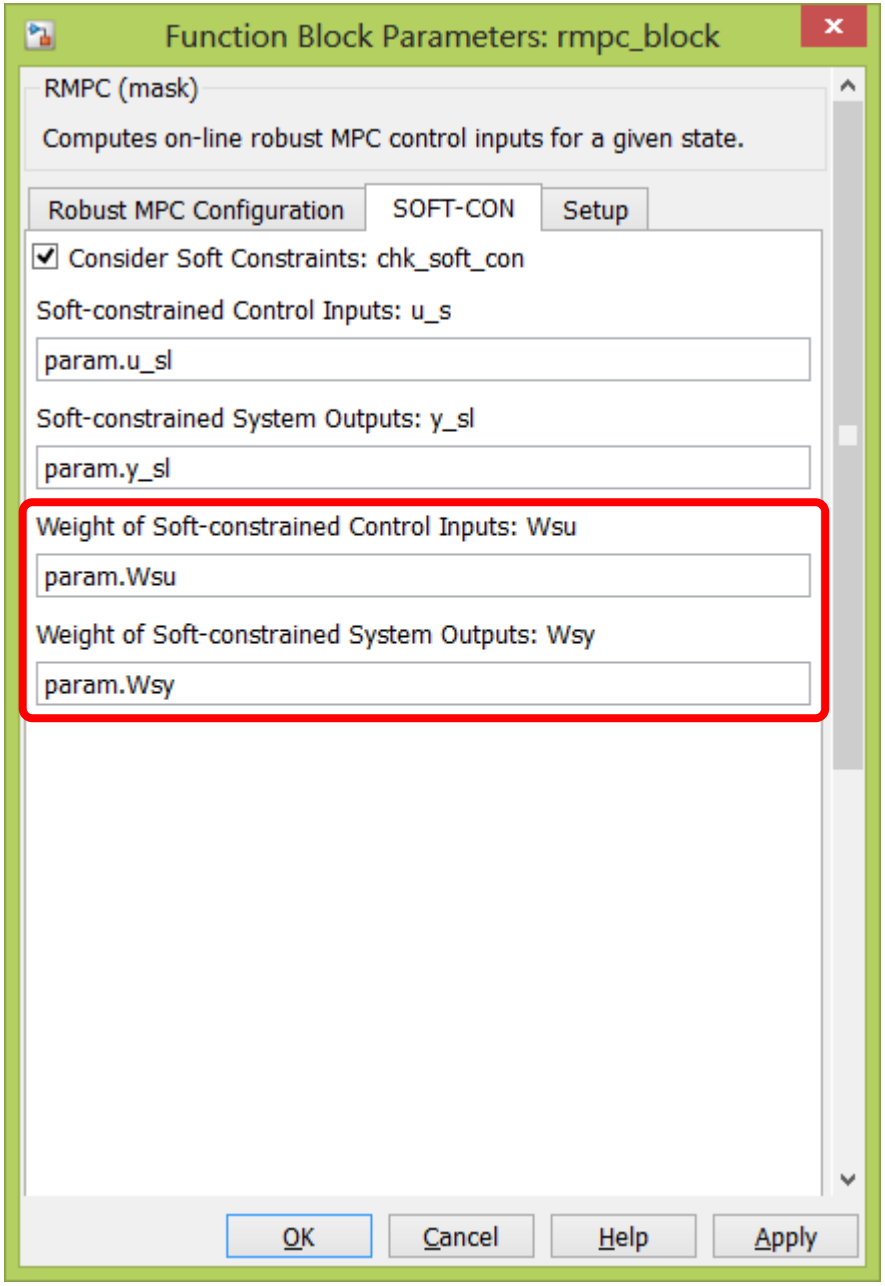


Soft Constraints

$$\|u(k)\|_2^2 \leq \|u_{\mathrm{sl}}\|_2^2 < \|u_{\max}\|_2^2$$

$$\|y(k)\|_2^2 \leq \|y_{\mathrm{sl}}\|_2^2 < \|y_{\max}\|_2^2$$

J. Oravec and M. Bakošová, *America Control Conference*, Boston, USA, 2016.



Soft Constraints

$$J = \sum_{i=0}^{\infty} (\ell(x(k+i), u(k+i))) =$$

$$\sum_{i=0}^{\infty} (x(k+i)^\top W_\mathrm{x}\, x(k+i) +$$

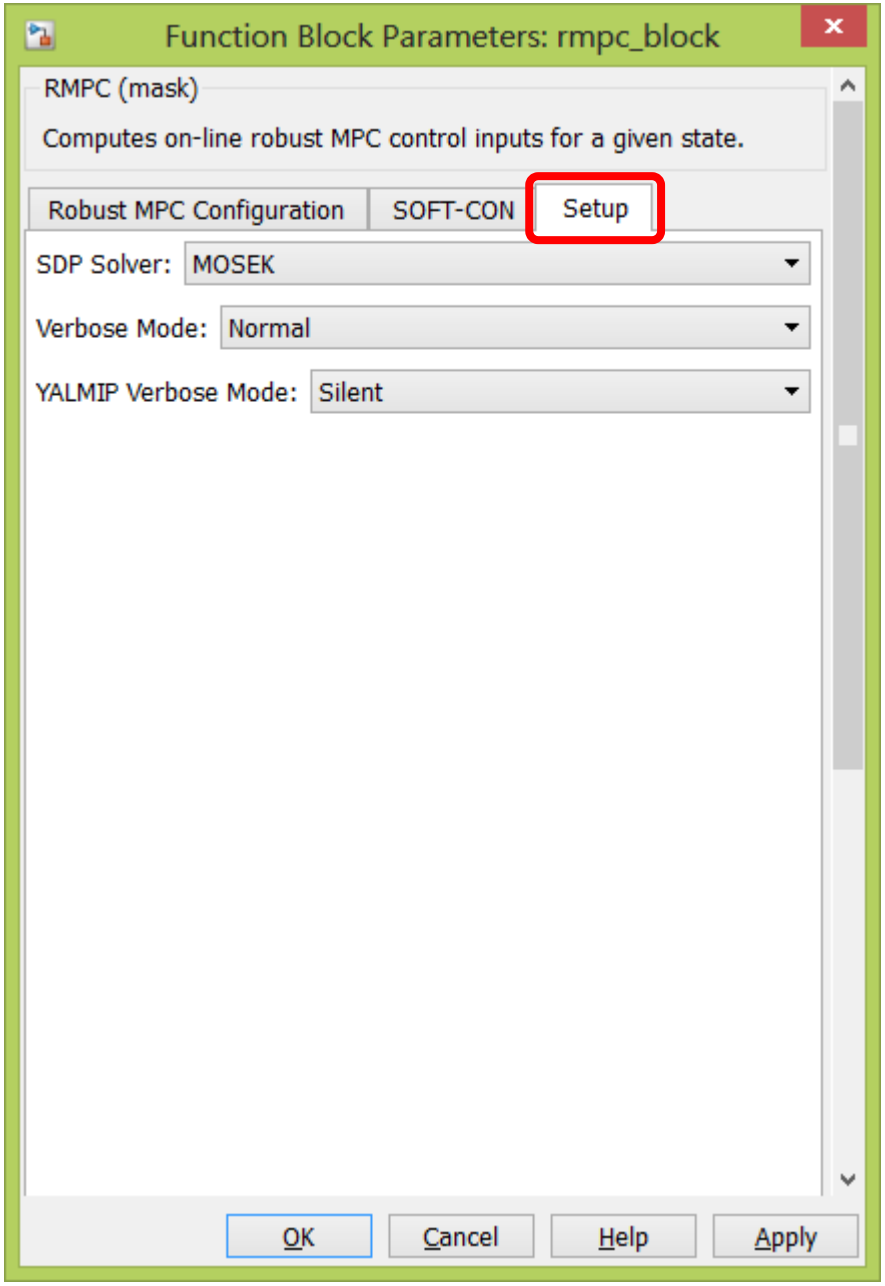
$$u(k+i)^\top W_\mathrm{u}\, u(k+i) +$$

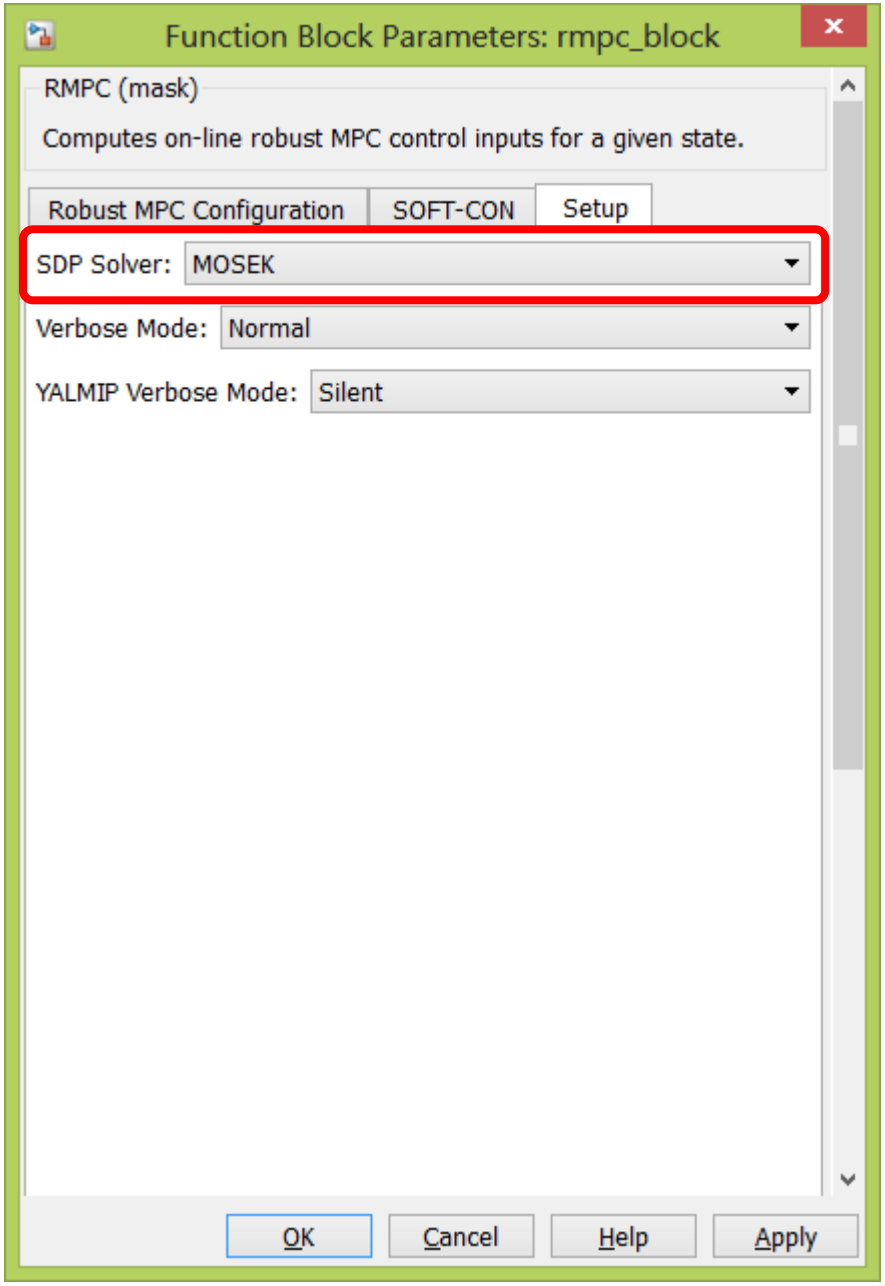
$$s_\mathrm{u}(k+i)^\top W_\mathrm{su}\, s_\mathrm{u}(k+i) +$$

$$s_\mathrm{y}(k+i)^\top W_\mathrm{sy}\, s_\mathrm{y}(k+i))$$

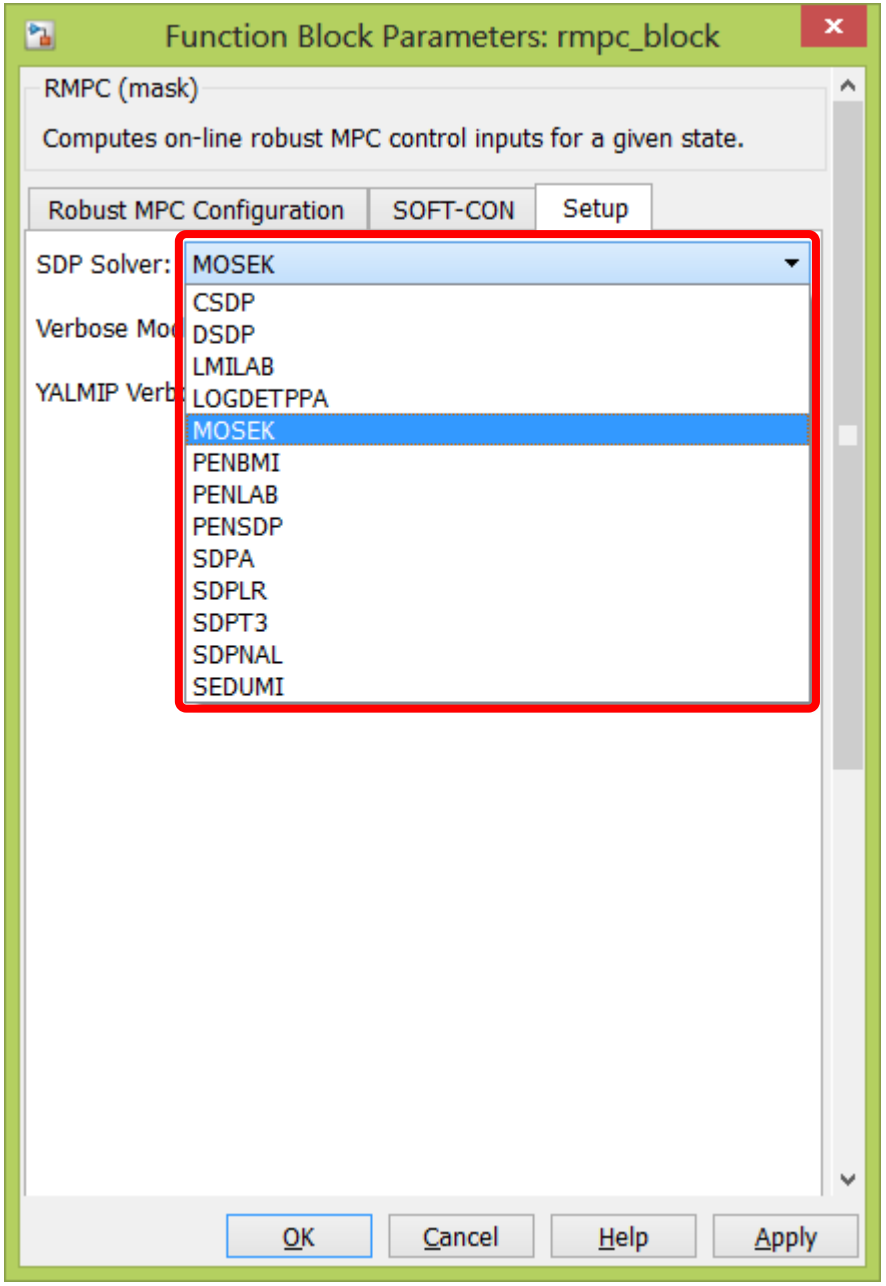
J. Oravec and M. Bakošová, *America Control Conference*, Boston, USA, 2016.

Setup


Function Block Parameters: rmpc_block
RMPC (mask)
Computes on-line robust MPC control inputs for a given state.
Robust MPC Configuration
SOFT-CON
Setup
SDP Solver:
MOSEK
Verbose Mode:
Normal
YALMIP Verbose Mode:
Silent
OK
Cancel
Help
Apply

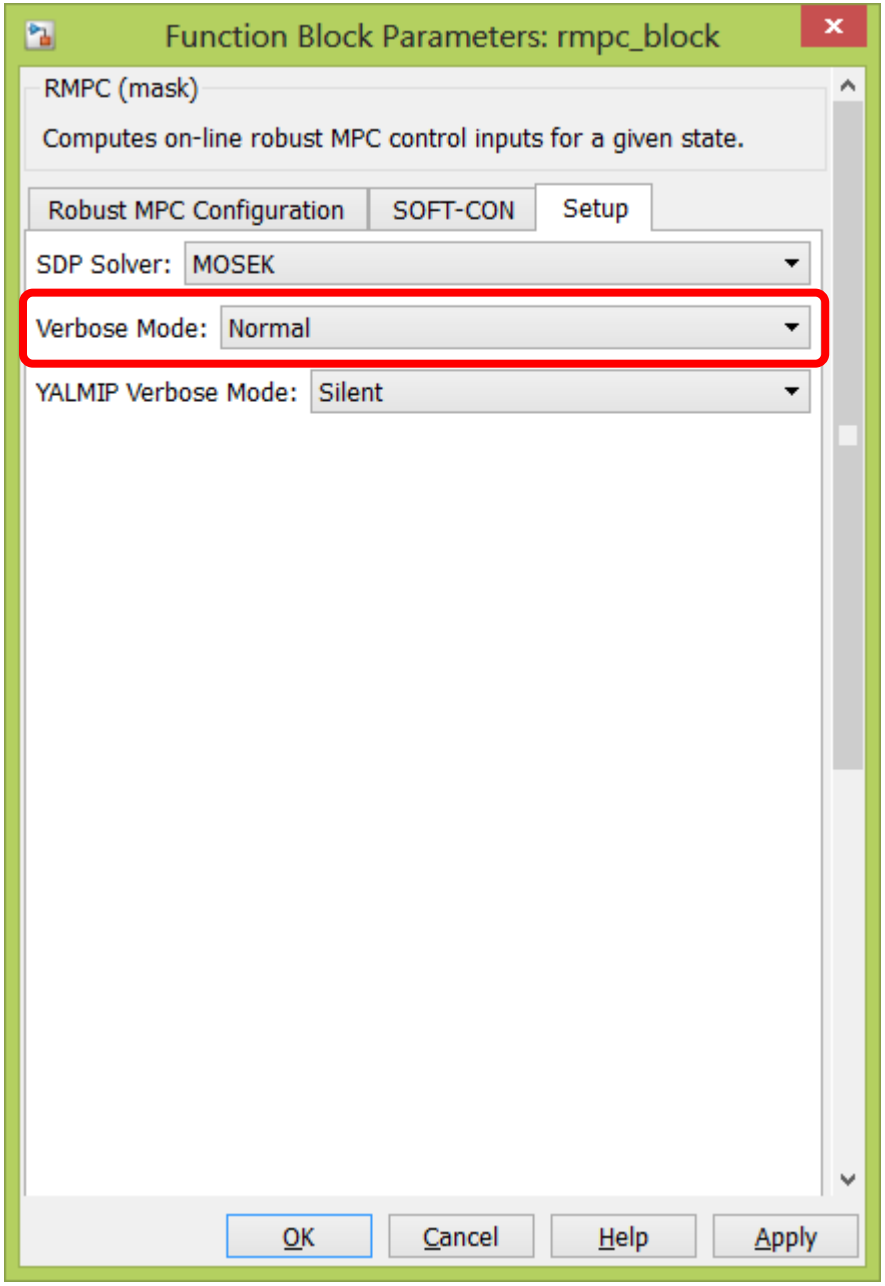

Function Block Parameters: rmpc_block
RMPC (mask)
Computes on-line robust MPC control inputs for a given state.
Robust MPC Configuration
SOFT-CON
Setup
SDP Solver: MOSEK
Verbose Mode: Normal
YALMIP Verbose Mode: Silent
OK
Cancel
Help
Apply

SDP Solver

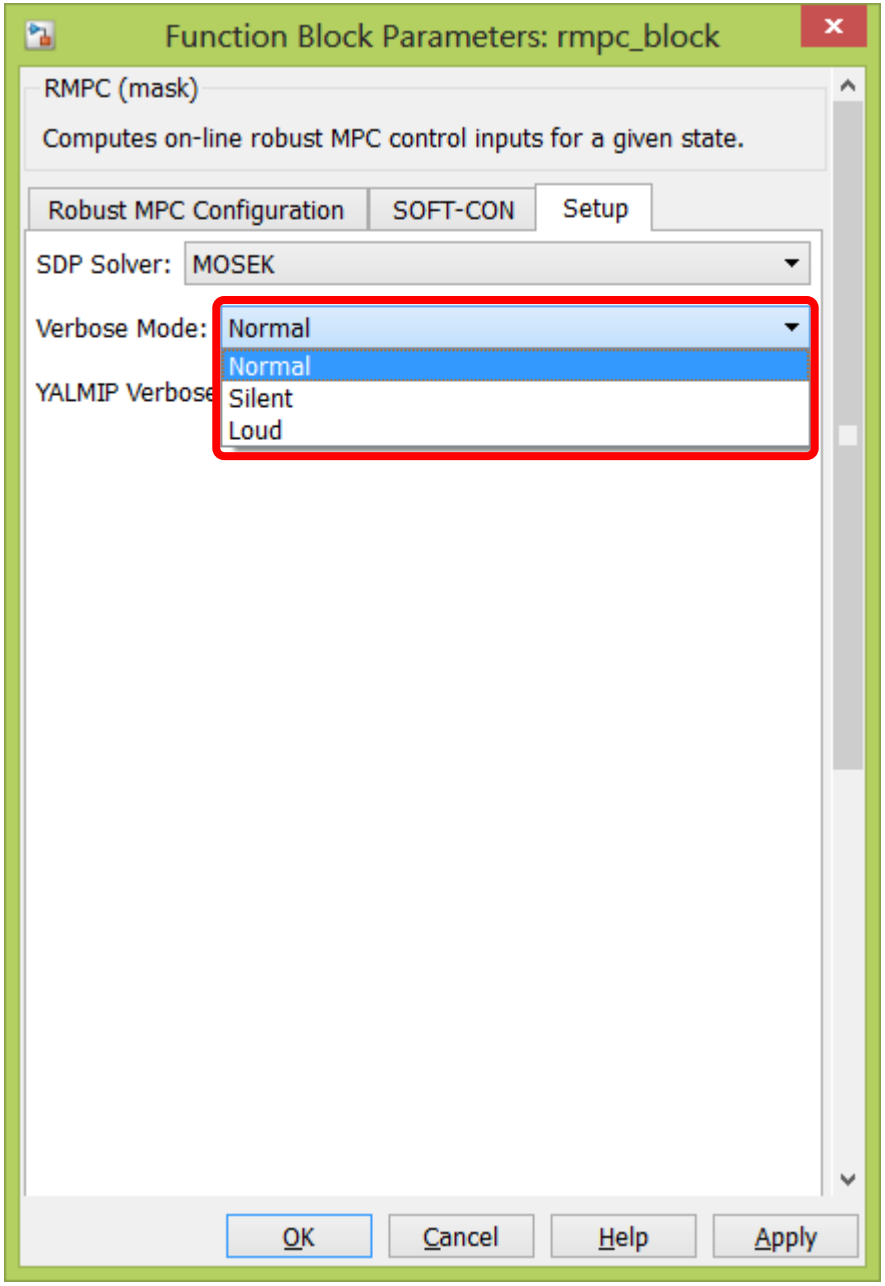


SDP Solver

- CSDP (*free*)
- DSDP (*free*)
- LMILAB (*comercial*)
- LOGDETPPA (*free*)
- MOSEK (*free for academia*)
- PENBMI (*comercial*)
- PENLAB (*free*)
- PENSDP (*free for academia*)
- SDPA (*free*)
- SDPLR (*free*)
- SDPT3 (*free*)
- SDPNAL (*free*)
- SEDUMI (*free*)

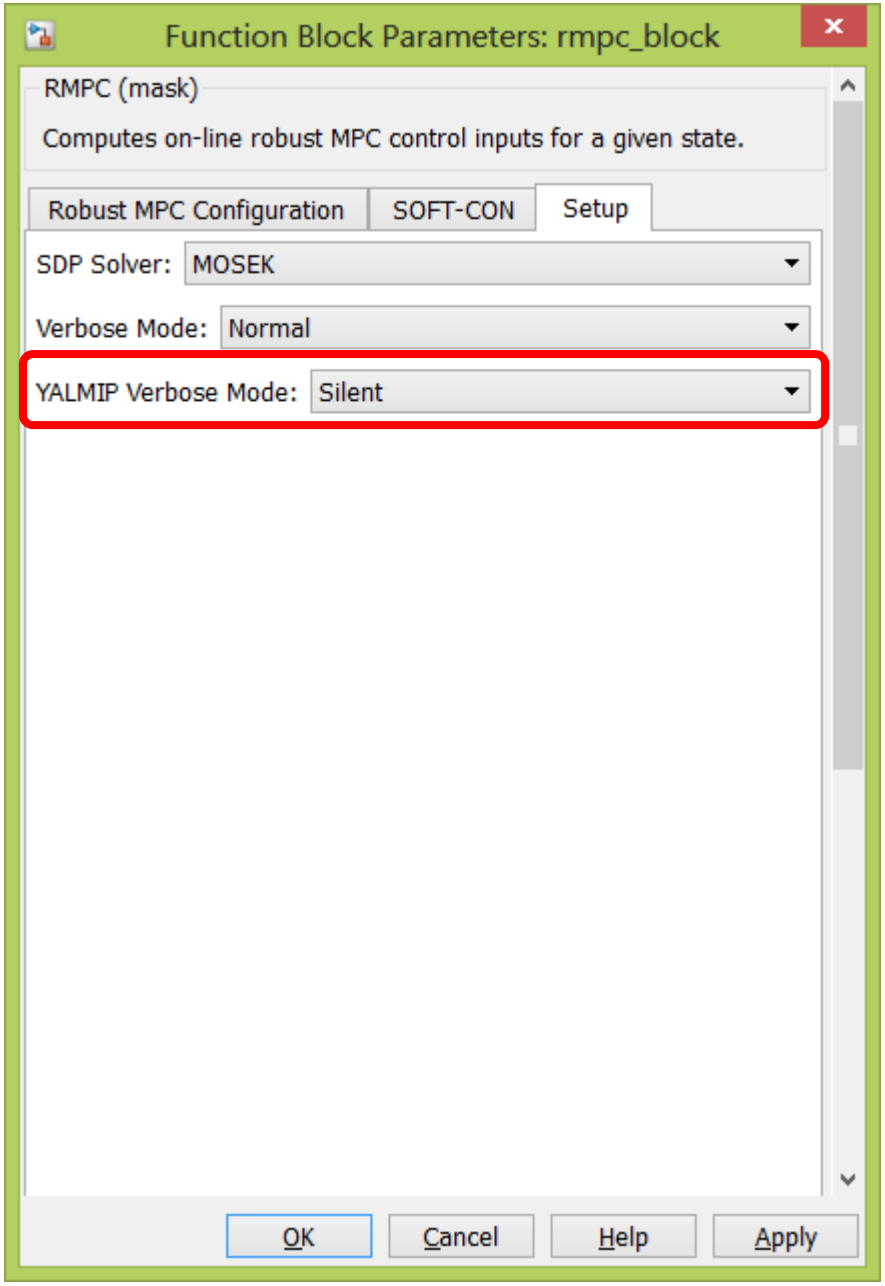


Verbose Mode

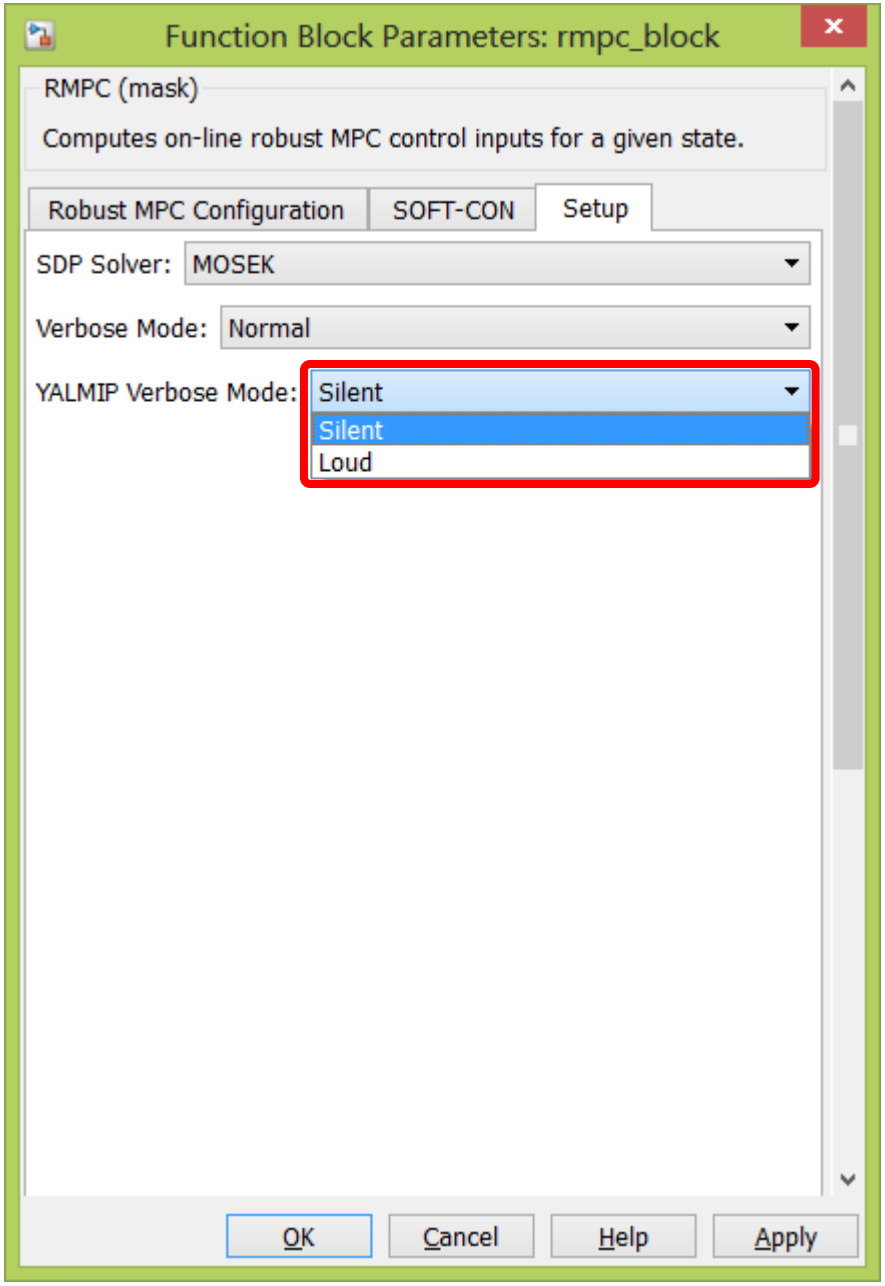


Verbose Mode

- Normal
- Silent
- Loud (*recommended*)



YALMIP Verbose Mode



YALMIP Verbose Mode

- Silent
- Loud

How to use it?

- MATLAB/Simulink: `rmpc_block` (*beginner*)
- MATLAB: CLI (*advanced*)

CLI

```
1: %% System
2: A{1}=...; A{2}=...; A{3}=...; A{4}=...;
3: B{1}=...; B{2}=...; B{3}=...; B{4}=...;
4: C{1}=...; C{2}=...; C{3}=...; C{4}=...;
5: ts=0.5;                    % Sampling time
6: x0=...;                    % Initial conditions
```

CLI

```
1: %% System
2: A{1}=...; A{2}=...; A{3}=...; A{4}=...;
3: B{1}=...; B{2}=...; B{3}=...; B{4}=...;
4: C{1}=...; C{2}=...; C{3}=...; C{4}=...;
5: ts=0.5;                    % Sampling time
6: x0=...;                    % Initial conditions
```

$$x(k+1) = A\,x(k) + B\,u(k),$$
$$y(k) = C\,x(k), \quad x(0) = x_0,$$
$$[A, B] \in \text{convhull}\{[A_v, B_v]\}.$$

CLI

```
1: %% System
2: A{1}=...; A{2}=...; A{3}=...; A{4}=...;
3: B{1}=...; B{2}=...; B{3}=...; B{4}=...;
4: C{1}=...; C{2}=...; C{3}=...; C{4}=...;
5: ts=0.5;                    % Sampling time
6: x0=...;                    % Initial conditions
```

$$x(k+1) = A\,x(k) + B\,u(k),$$
$$y(k) = C\,x(k), \quad x(0) = x_0,$$
$$[A, B] \in \text{convhull}\{[A_v, B_v]\}.$$

CLI

```
1: %% Symmetric constraints
2: u_max=...; % Input constraints
3: x_max=...; % State constraints
```

$$\|u(k)\|_2^2 \preceq \|u_{\max}\|_2^2$$

$$\|x(k)\|_2^2 \preceq \|x_{\max}\|_2^2$$

CLI

```
1: %% RMPC design
2: Wx=...; Wu=...;          % Weight matrices
3: [method,kwd] = mup_cli_rmpc_method; % Method
4: mup_sdp                  % SDP formulation
5: mup_rmpc_opt             % RMPC design
```

CLI

```
1: %% RMPC design
2: Wx=...; Wu=...;          % Weight matrices
3: [method,kwd] = mup_cli_rmpc_method; % Method
4: mup_sdp                  % SDP formulation
5: mup_rmpc_opt             % RMPC design
```

$$J(k) = \sum_{i=0}^{\infty} (x(k+i)^\top W_{\mathrm{x}}\, x(k+i) + u(k+i)^\top W_{\mathrm{u}}\, u(k+i))$$

CLI

```
1: %% RMPC design
2: Wx=...; Wu=...;          % Weight matrices
3: [method,kwd] = mup_cli_rmpc_method; % Method
4: mup_sdp                  % SDP formulation
5: mup_rmpc_opt             % RMPC design
```

CLI

```
1: %% RMPC design
2: Wx=...; Wu=...;          % Weight matrices
3: [method,kwd] = mup_cli_rmpc_method; % Method
4: mup_sdp                  % SDP formulation
5: mup_rmpc_opt             % RMPC design
```

```
List of avaliable methods:
  1: Cao et al. (2005)
  2: Cao et al. (2005) - Soft-Con
  3: Cuzzola et al. (2002)
```

CLI

```
1: %% RMPC design
2: Wx=...; Wu=...;          % Weight matrices
3: [method,kwd] = mup_cli_rmpc_method; % Method
4: mup_sdp                  % SDP formulation
5: mup_rmpc_opt             % RMPC design
```

$$\begin{aligned} &\min\ \gamma, \\ &\text{s.t.:}\ \ x \preceq_{\mathcal{K}} 0 \Rightarrow M(x) \prec 0 \end{aligned}$$

CLI

```
1: %% RMPC design
2: Wx=...; Wu=...;          % Weight matrices
3: [method,kwd] = mup_cli_rmpc_method; % Method
4: mup_sdp                  % SDP formulation
5: mup_rmpc_opt             % RMPC design
```

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

CLI

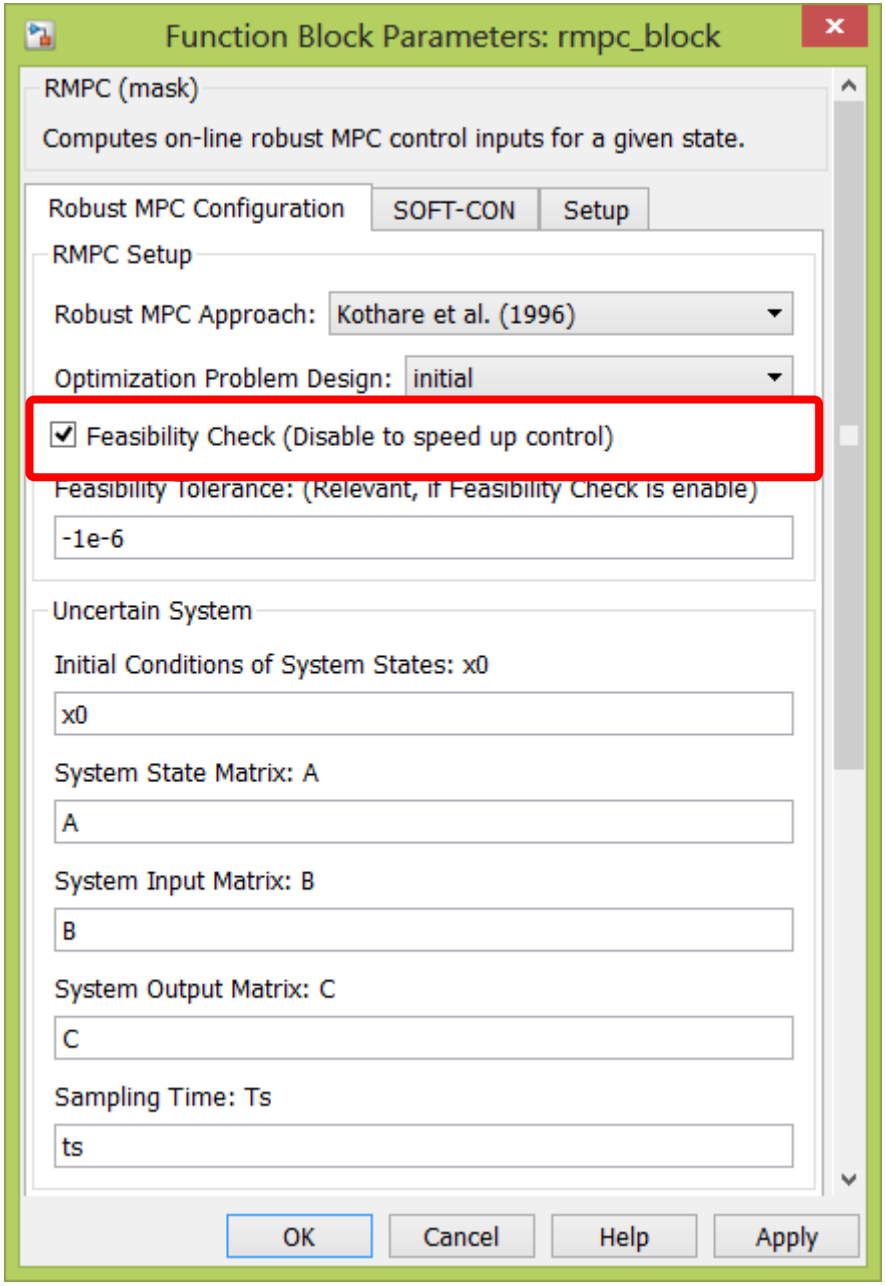
```
1:  %% Closed-loop control in k-th step
2:  u = mup_rmpc{x,design};   % RMPC control law
```

CLI

```
1:  %% Closed-loop control in k-th step
2:  u = mup_rmpc{x,design};  % RMPC control law
```

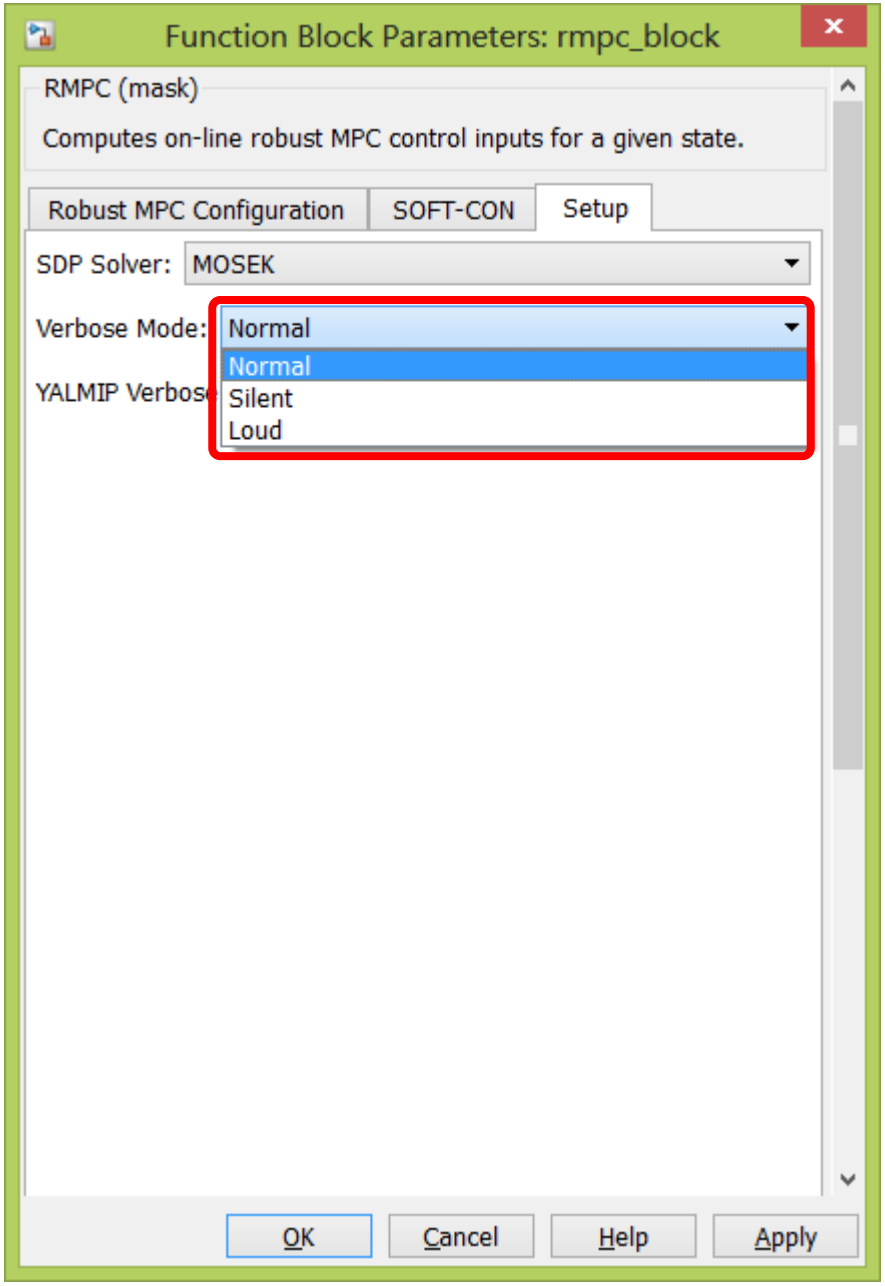
$$u(k) = F(x(k))\,x(k)$$

Feasibility Check



Feasibility Check

enable



Verbose Mode

- Normal
- Silent
- Loud (*recommended*)

Feasibility Check / Normal

MUP:BLOCK:RMPC: FEASIBILITY_CHECK initialized...
MUP:BLOCK:RMPC: Valid. (0.12)
MUP:BLOCK:RMPC: FEASIBILITY_CHECK: Termined. (0.03 sec)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Normal

MUP:BLOCK:RMPC: FEASIBILITY_CHECK initialized...
MUP:BLOCK:RMPC: Valid. (0.12)
MUP:BLOCK:RMPC: FEASIBILITY_CHECK: Termined. (0.03 sec)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Normal

MUP:BLOCK:RMPC: FEASIBILITY_CHECK initialized...
FEAS_CHK: VTX:2|I:1: Stability condition violated!
FEAS_CHK: VTX:2|I:2: Stability condition violated!
FEAS_CHK: VTX:1|I:1: State contraints violated!
FEAS_CHK: VTX:1|I:2: State contraints violated!
FEAS_CHK: VTX:2|I:1: State contraints violated!
FEAS_CHK: VTX:2|I:2: State contraints violated!
FEAS_CHK: VTX:3|I:1: State contraints violated!
FEAS_CHK: VTX:3|I:2: State contraints violated!
FEAS_CHK: Failed! (8 infeasible constraint(s) found) (0.03)
MUP:BLOCK:RMPC: Feasibility check failed!
MUP:BLOCK:RMPC: Instead of Robust MPC has been designed LQ controller
MUP:BLOCK:RMPC: FEASIBILITY_CHECK: Termined. (0.05 sec)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Normal

MUP:BLOCK:RMPC: FEASIBILITY_CHECK initialized...
FEAS_CHK: VTX:2|I:1: Stability condition violated!
FEAS_CHK: VTX:2|I:2: Stability condition violated!
FEAS_CHK: VTX:1|I:1: State contraints violated!
FEAS_CHK: VTX:1|I:2: State contraints violated!
FEAS_CHK: VTX:2|I:1: State contraints violated!
FEAS_CHK: VTX:2|I:2: State contraints violated!
FEAS_CHK: VTX:3|I:1: State contraints violated!
FEAS_CHK: VTX:3|I:2: State contraints violated!
FEAS_CHK: Failed! (8 infeasible constraint(s) found) (0.03)
MUP:BLOCK:RMPC: Feasibility check failed!
MUP:BLOCK:RMPC: Instead of Robust MPC has been designed LQ controller
MUP:BLOCK:RMPC: FEASIBILITY_CHECK: Termined. (0.05 sec)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Normal

MUP:BLOCK:RMPC: FEASIBILITY_CHECK initialized...
FEAS_CHK: VTX:2|I:1: Stability condition violated!
FEAS_CHK: VTX:2|I:2: Stability condition violated!
FEAS_CHK: VTX:1|I:1: State contraints violated!
FEAS_CHK: VTX:1|I:2: State contraints violated!
FEAS_CHK: VTX:2|I:1: State contraints violated!
FEAS_CHK: VTX:2|I:2: State contraints violated!
FEAS_CHK: VTX:3|I:1: State contraints violated!
FEAS_CHK: VTX:3|I:2: State contraints violated!
FEAS_CHK: Failed! (8 infeasible constraint(s) found) (0.03)
MUP:BLOCK:RMPC: Feasibility check failed!
MUP:BLOCK:RMPC: Instead of Robust MPC has been designed LQ controller
MUP:BLOCK:RMPC: FEASIBILITY_CHECK: Termined. (0.05 sec)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Normal

MUP:BLOCK:RMPC: FEASIBILITY_CHECK initialized...
FEAS_CHK: VTX:2|I:1: Stability condition violated!
FEAS_CHK: VTX:2|I:2: Stability condition violated!
FEAS_CHK: VTX:1|I:1: State contraints violated!
FEAS_CHK: VTX:1|I:2: State contraints violated!
FEAS_CHK: VTX:2|I:1: State contraints violated!
FEAS_CHK: VTX:2|I:2: State contraints violated!
FEAS_CHK: VTX:3|I:1: State contraints violated!
FEAS_CHK: VTX:3|I:2: State contraints violated!
FEAS_CHK: Failed! (8 infeasible constraint(s) found) (0.03)
MUP:BLOCK:RMPC: Feasibility check failed!
MUP:BLOCK:RMPC: Instead of Robust MPC has been designed LQ controller
MUP:BLOCK:RMPC: FEASIBILITY_CHECK: Termined. (0.05 sec)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Normal

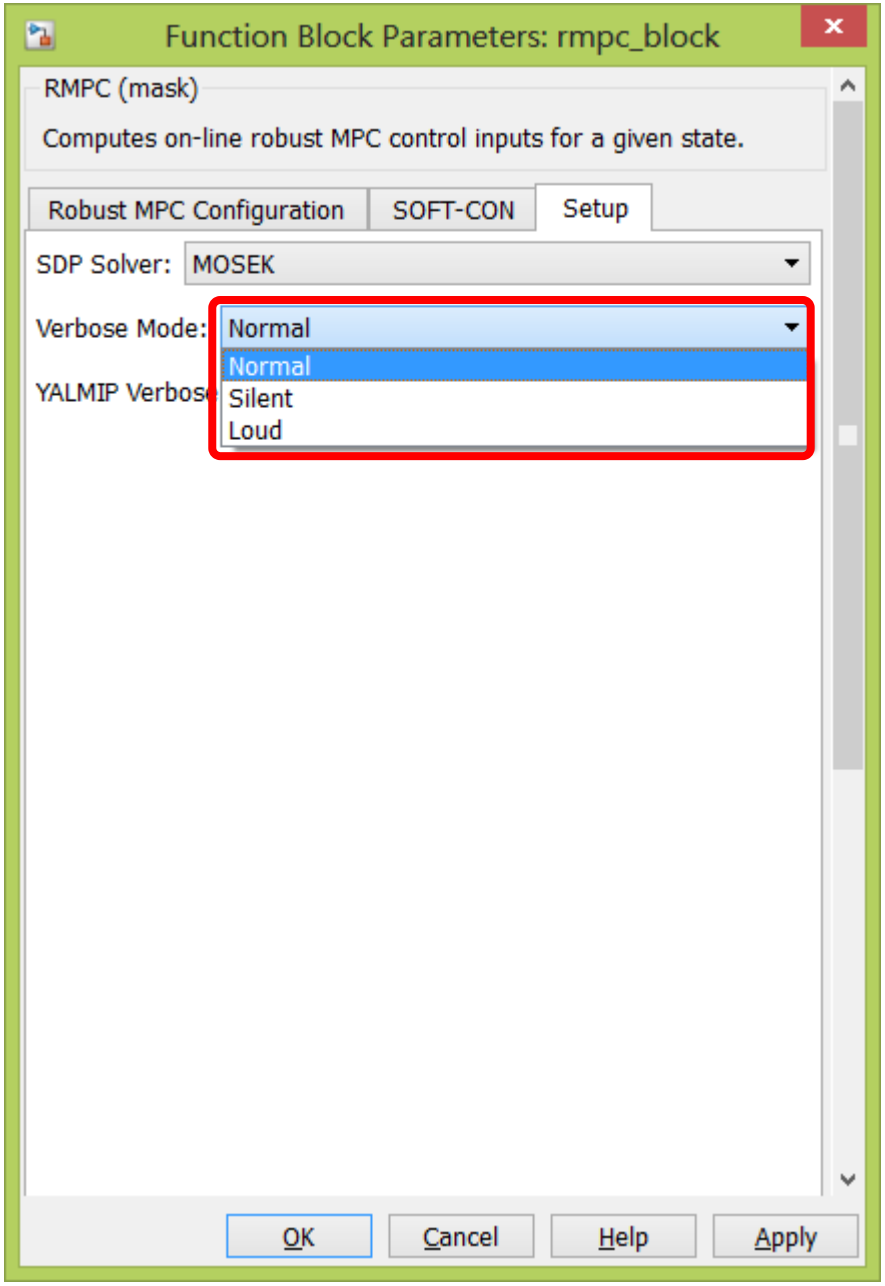
MUP:BLOCK:RMPC: FEASIBILITY_CHECK initialized...
FEAS_CHK: VTX:2|I:1: Stability condition violated!
FEAS_CHK: VTX:2|I:2: Stability condition violated!
FEAS_CHK: VTX:1|I:1: State contraints violated!
FEAS_CHK: VTX:1|I:2: State contraints violated!
FEAS_CHK: VTX:2|I:1: State contraints violated!
FEAS_CHK: VTX:2|I:2: State contraints violated!
FEAS_CHK: VTX:3|I:1: State contraints violated!
FEAS_CHK: VTX:3|I:2: State contraints violated!
FEAS_CHK: Failed! (8 infeasible constraint(s) found) (0.03)
MUP:BLOCK:RMPC: Feasibility check failed!
MUP:BLOCK:RMPC: Instead of Robust MPC has been designed LQ controller
MUP:BLOCK:RMPC: FEASIBILITY_CHECK: Termined. (0.05 sec)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Normal

MUP:BLOCK:RMPC: FEASIBILITY_CHECK initialized...
FEAS_CHK: VTX:2|I:1: Stability condition violated!
FEAS_CHK: VTX:2|I:2: Stability condition violated!
FEAS_CHK: VTX:1|I:1: State contraints violated!
FEAS_CHK: VTX:1|I:2: State contraints violated!
FEAS_CHK: VTX:2|I:1: State contraints violated!
FEAS_CHK: VTX:2|I:2: State contraints violated!
FEAS_CHK: VTX:3|I:1: State contraints violated!
FEAS_CHK: VTX:3|I:2: State contraints violated!
FEAS_CHK: Failed! (8 infeasible constraint(s) found) (0.03)
MUP:BLOCK:RMPC: Feasibility check failed!
MUP:BLOCK:RMPC: Instead of Robust MPC has been designed LQ controller
MUP:BLOCK:RMPC: FEASIBILITY_CHECK: Termined. (0.05 sec)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK



Verbose Mode

- Normal
- Silent
- Loud (*recommended*)

Feasibility Check / Loud

FEAS_CHK: Inverse Lyapunov matrix MIN_EIG = 3.275738e+00.
FEAS_CHK: I:1: Inverse SDLF matrix W MIN_EIG = 3.115516e-01.
FEAS_CHK: I:2: Inverse SDLF matrix W MIN_EIG = 3.115516e-01.
FEAS_CHK: I:1: Inverse SDLF matrix Q MIN_EIG = 1.482989e-06.
FEAS_CHK: I:2: Inverse SDLF matrix Q MIN_EIG = 1.480064e-06.
FEAS_CHK: I:1: Invariant ellipsoid matrix MIN_EIG = 1.566609e-08.
FEAS_CHK: I:2: Invariant ellipsoid matrix MIN_EIG = 1.439442e-08.
FEAS_CHK: VTX:1|I:1: Stability condition matrix MIN_EIG = 8.704402e-06.
FEAS_CHK: VTX:1|I:2: Stability condition matrix MIN_EIG = 8.705740e-06.
FEAS_CHK: VTX:2|I:1: Stability condition matrix MIN_EIG = -5.363964e-06.
FEAS_CHK: VTX:2|I:1: Stability condition violated!
… (*shortened*)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Loud

FEAS_CHK: Inverse Lyapunov matrix MIN_EIG = 3.275738e+00.
FEAS_CHK: I:1: Inverse SDLF matrix W MIN_EIG = 3.115516e-01.
FEAS_CHK: I:2: Inverse SDLF matrix W MIN_EIG = 3.115516e-01.
FEAS_CHK: I:1: Inverse SDLF matrix Q MIN_EIG = 1.482989e-06.
FEAS_CHK: I:2: Inverse SDLF matrix Q MIN_EIG = 1.480064e-06.
FEAS_CHK: I:1: Invariant ellipsoid matrix MIN_EIG = 1.566609e-08.
FEAS_CHK: I:2: Invariant ellipsoid matrix MIN_EIG = 1.439442e-08.
FEAS_CHK: VTX:1|I:1: Stability condition matrix MIN_EIG = 8.704402e-06.
FEAS_CHK: VTX:1|I:2: Stability condition matrix MIN_EIG = 8.705740e-06.
FEAS_CHK: VTX:2|I:1: Stability condition matrix MIN_EIG = -5.363964e-06.
FEAS_CHK: VTX:2|I:1: Stability condition violated!
… (*shortened*)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Loud

FEAS_CHK: Inverse Lyapunov matrix MIN_EIG = 3.275738e+00.
FEAS_CHK: I:1: Inverse SDLF matrix W MIN_EIG = 3.115516e-01.
FEAS_CHK: I:2: Inverse SDLF matrix W MIN_EIG = 3.115516e-01.
FEAS_CHK: I:1: Inverse SDLF matrix Q MIN_EIG = 1.482989e-06.
FEAS_CHK: I:2: Inverse SDLF matrix Q MIN_EIG = 1.480064e-06.
FEAS_CHK: I:1: Invariant ellipsoid matrix MIN_EIG = 1.566609e-08.
FEAS_CHK: I:2: Invariant ellipsoid matrix MIN_EIG = 1.439442e-08.
FEAS_CHK: VTX:1|I:1: Stability condition matrix MIN_EIG = 8.704402e-06.
FEAS_CHK: VTX:1|I:2: Stability condition matrix MIN_EIG = 8.705740e-06.
FEAS_CHK: VTX:2|I:1: Stability condition matrix MIN_EIG = -5.363964e-06.
FEAS_CHK: VTX:2|I:1: Stability condition violated!
… (*shortened*)

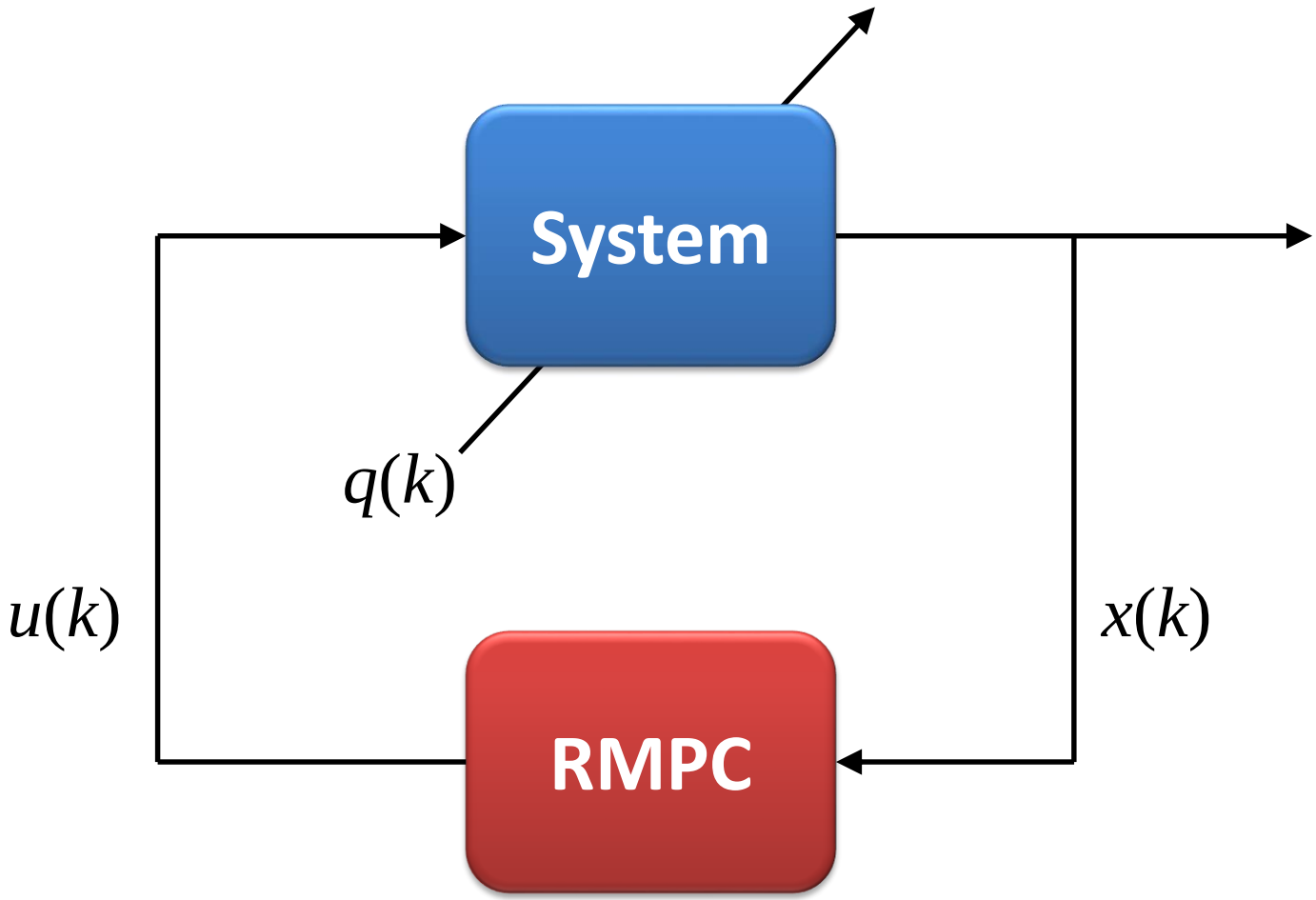
★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Feasibility Check / Loud

FEAS_CHK: Inverse Lyapunov matrix MIN_EIG = 3.275738e+00.
FEAS_CHK: I:1: Inverse SDLF matrix W MIN_EIG = 3.115516e-01.
FEAS_CHK: I:2: Inverse SDLF matrix W MIN_EIG = 3.115516e-01.
FEAS_CHK: I:1: Inverse SDLF matrix Q MIN_EIG = 1.482989e-06.
FEAS_CHK: I:2: Inverse SDLF matrix Q MIN_EIG = 1.480064e-06.
FEAS_CHK: I:1: Invariant ellipsoid matrix MIN_EIG = 1.566609e-08.
FEAS_CHK: I:2: Invariant ellipsoid matrix MIN_EIG = 1.439442e-08.
FEAS_CHK: VTX:1|I:1: Stability condition matrix MIN_EIG = 8.704402e-06.
FEAS_CHK: VTX:1|I:2: Stability condition matrix MIN_EIG = 8.705740e-06.
FEAS_CHK: VTX:2|I:1: Stability condition matrix MIN_EIG = -5.363964e-06.
FEAS_CHK: VTX:2|I:1: Stability condition violated!
… (*shortened*)

★ 2.5 GHz CPU, 4GB RAM, MATLAB 2013a, MOSEK

Case Study



Case Study

$u(k)$

$x(k)$

RMPC

★ *Oravec and Bakošová, American Control Conference, 2014.*

Case Study

physical plant

$u(k)$

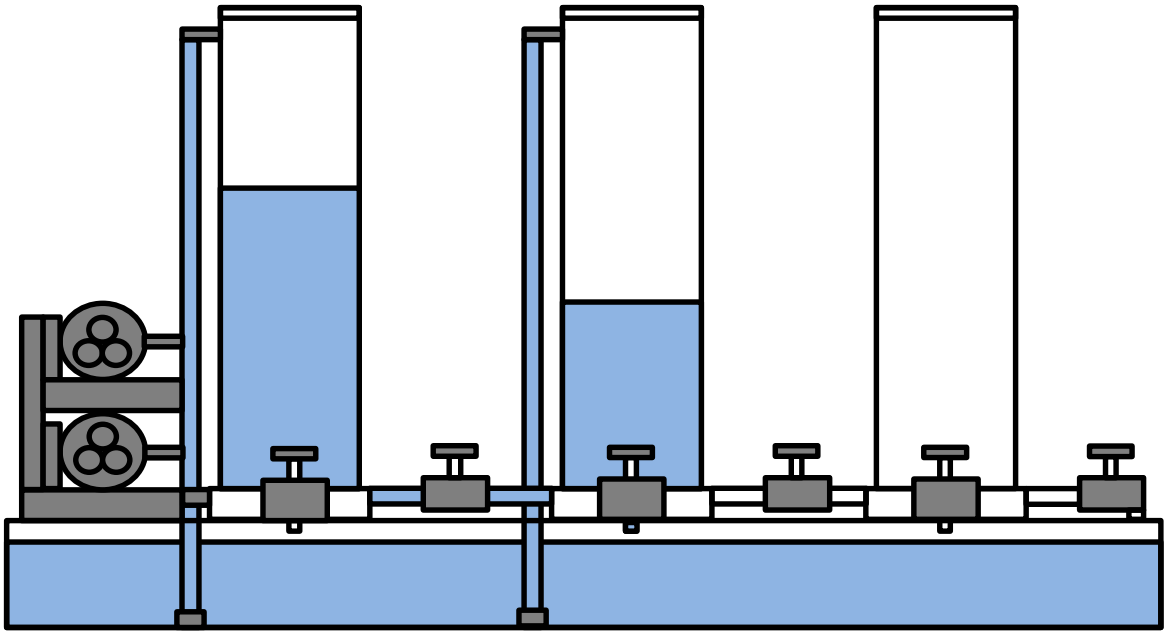
$x(k)$

MATLAB

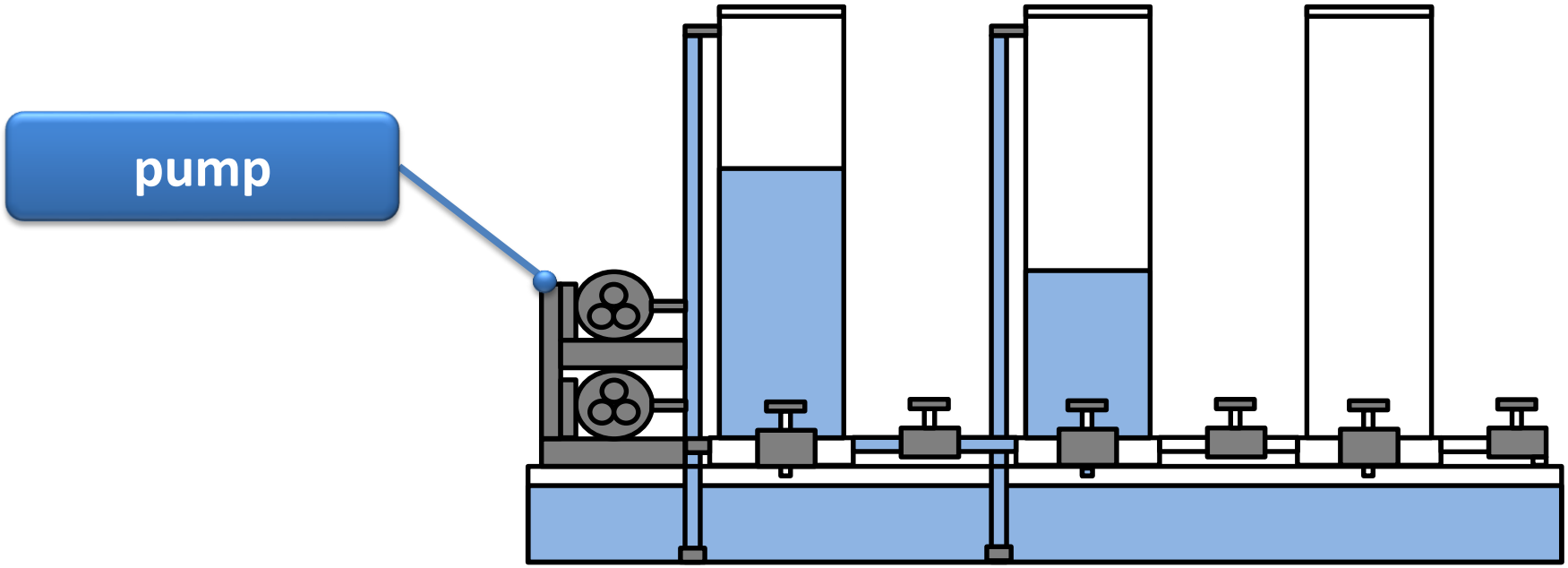
RMPC

★ *Oravec and Bakošová, American Control Conference, 2014.*

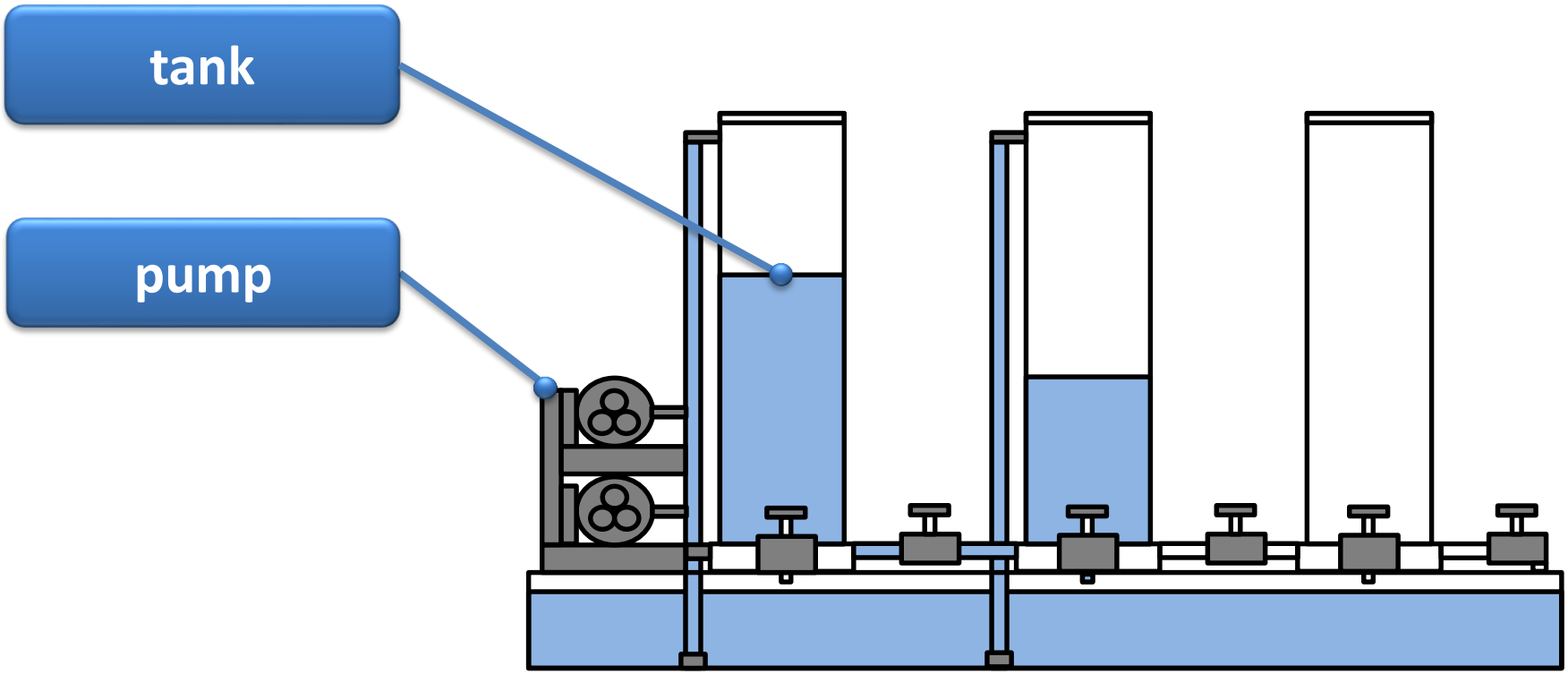
Liquid Tanks



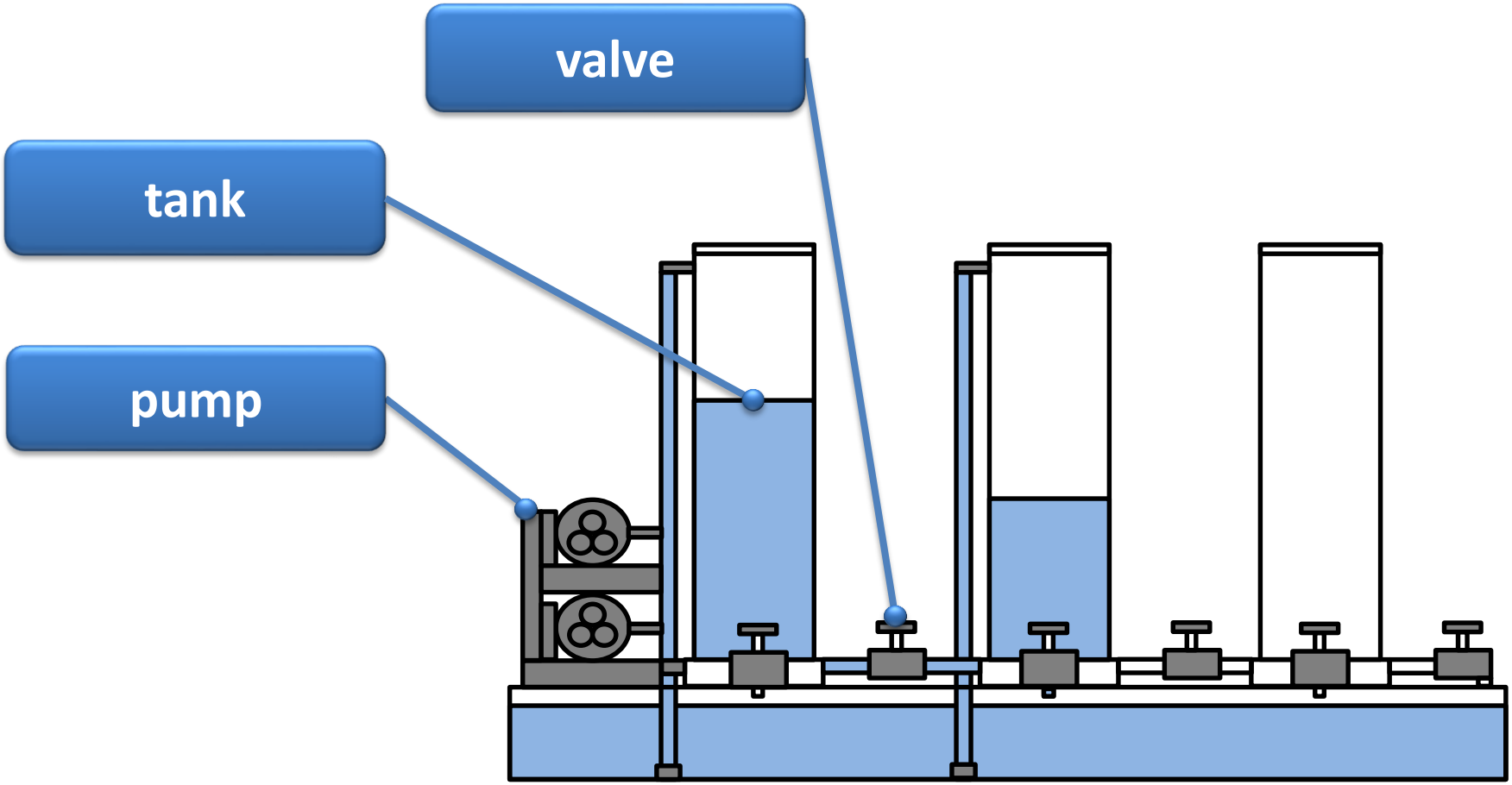
Liquid Tanks



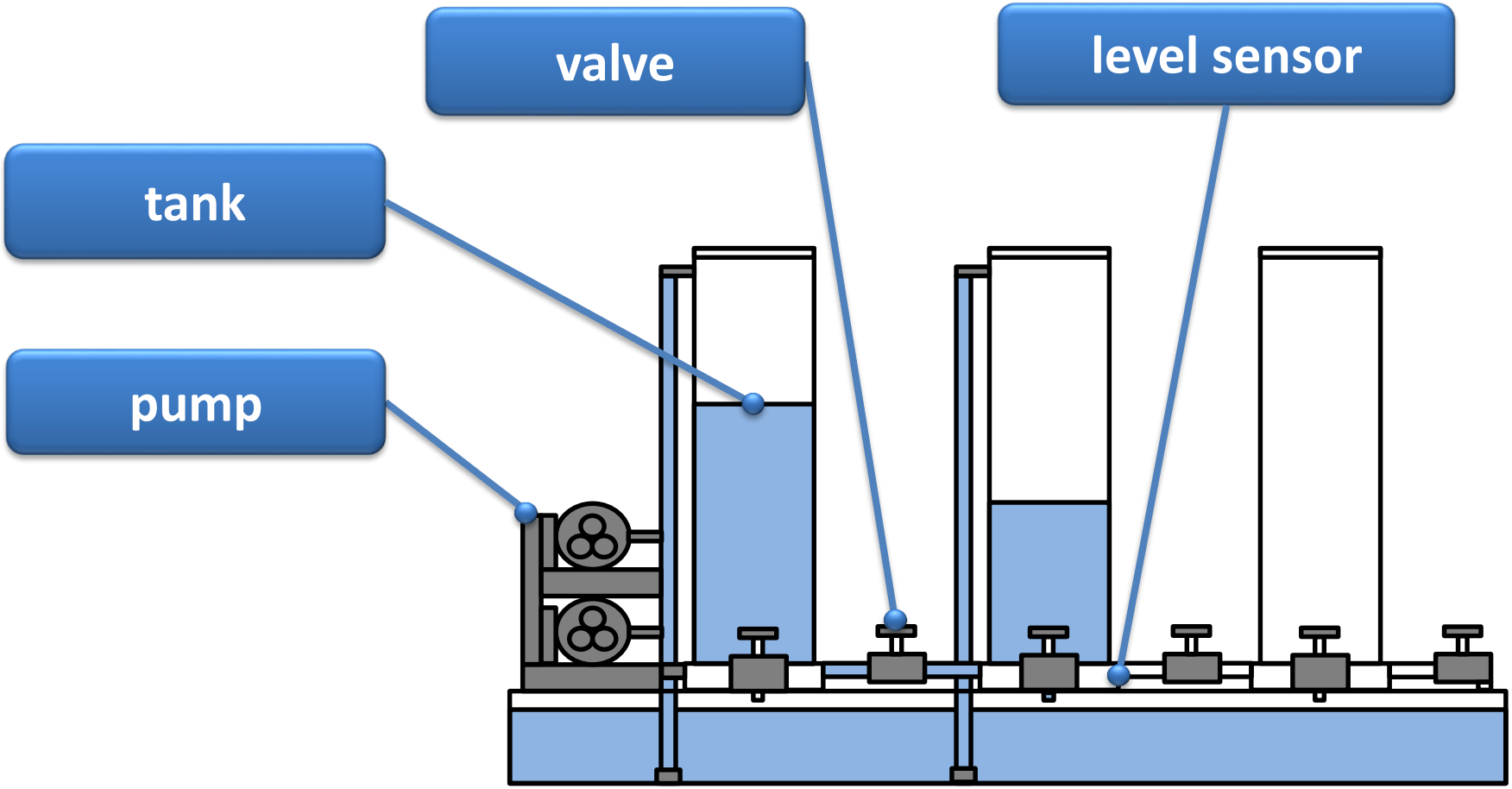
Liquid Tanks



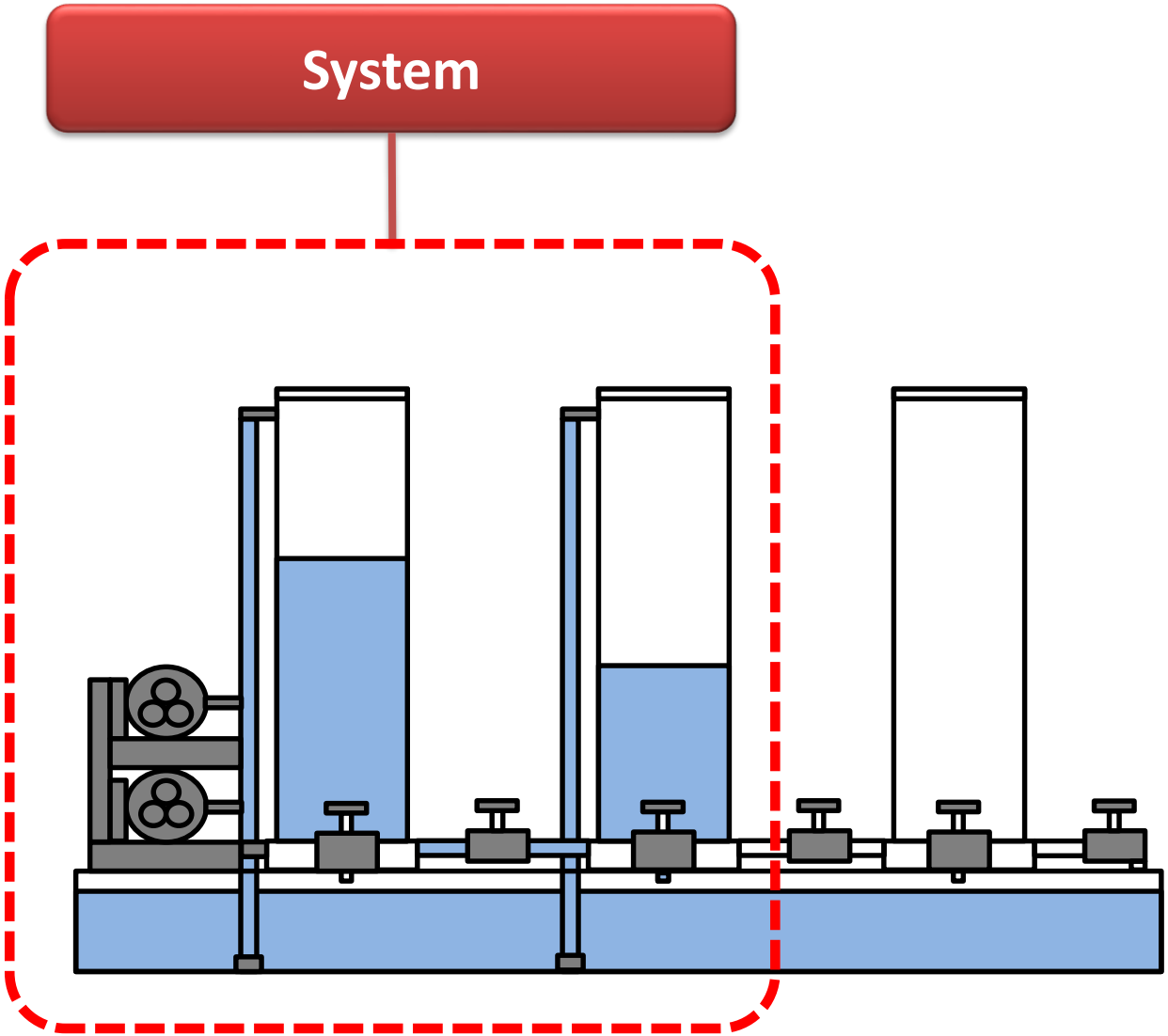
Liquid Tanks



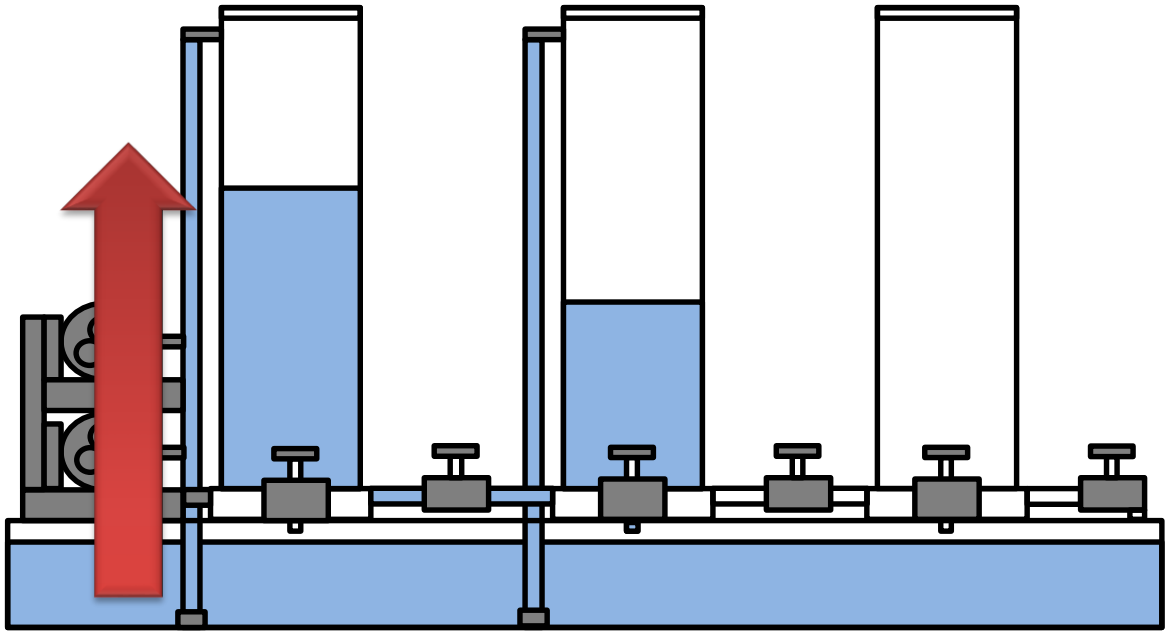
Liquid Tanks



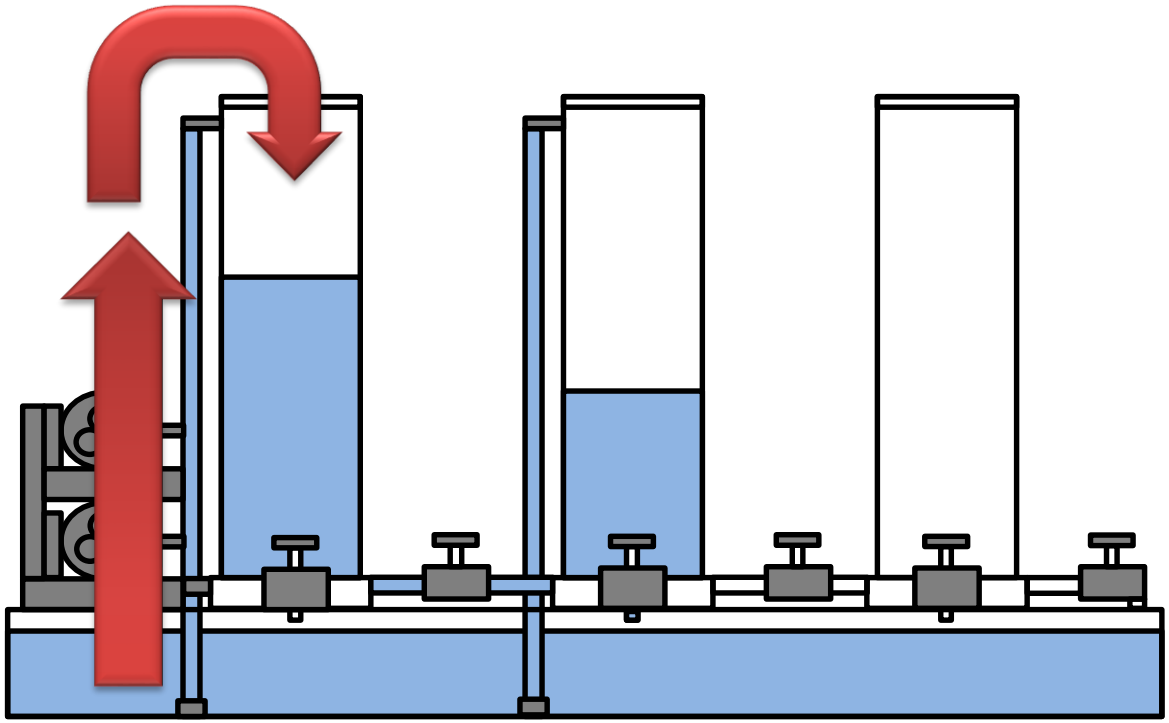
Liquid Tanks



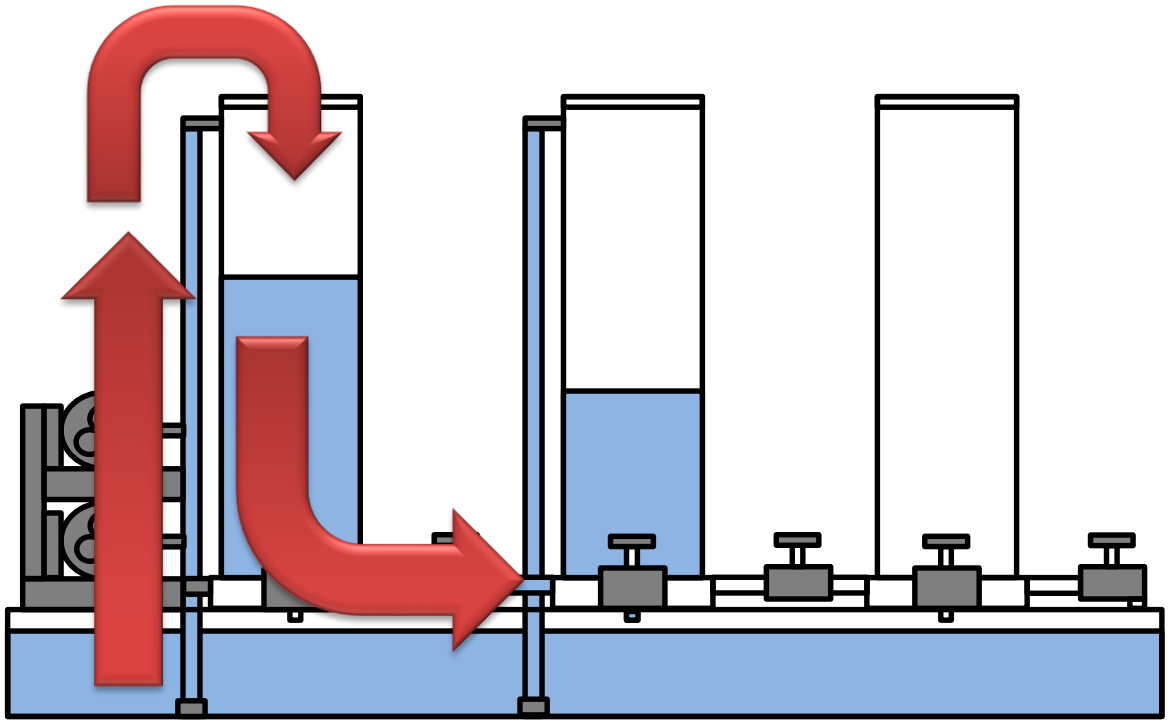
Liquid Tanks



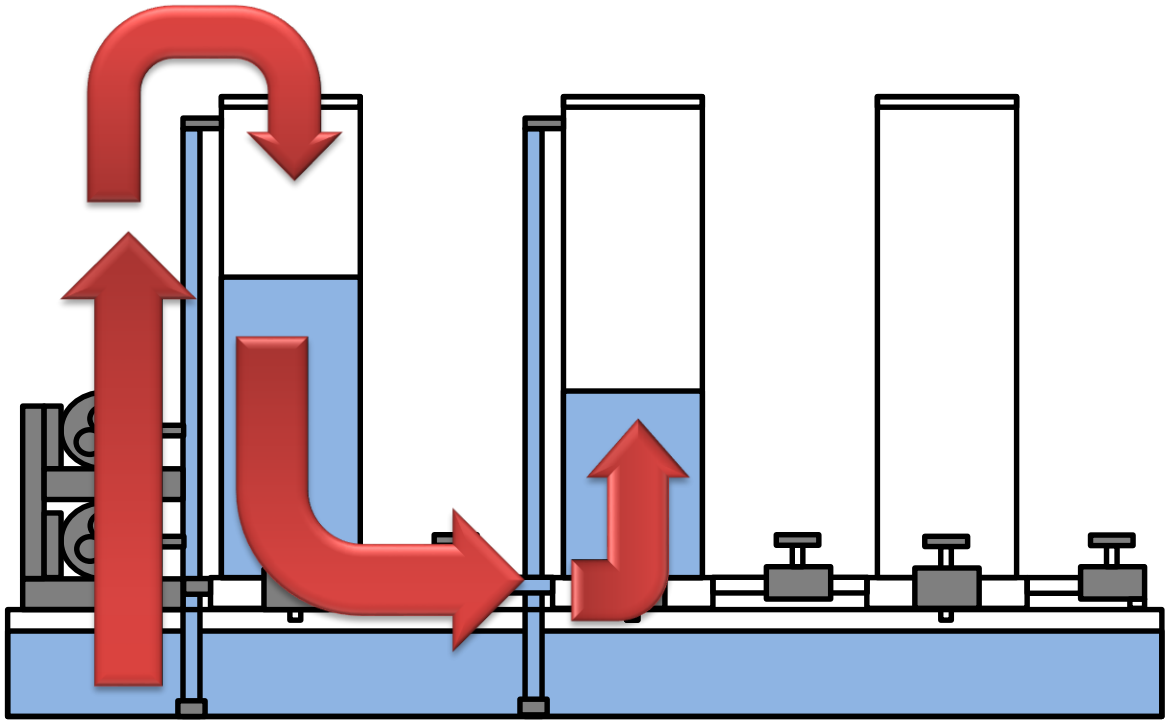
Liquid Tanks



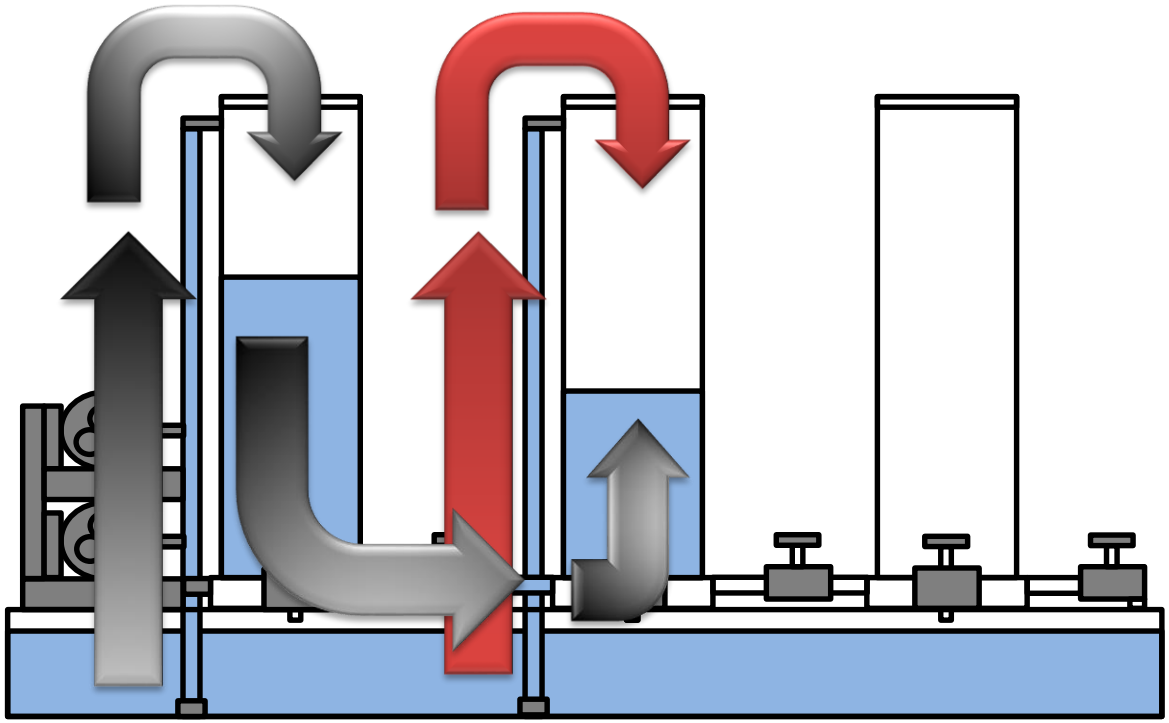
Liquid Tanks



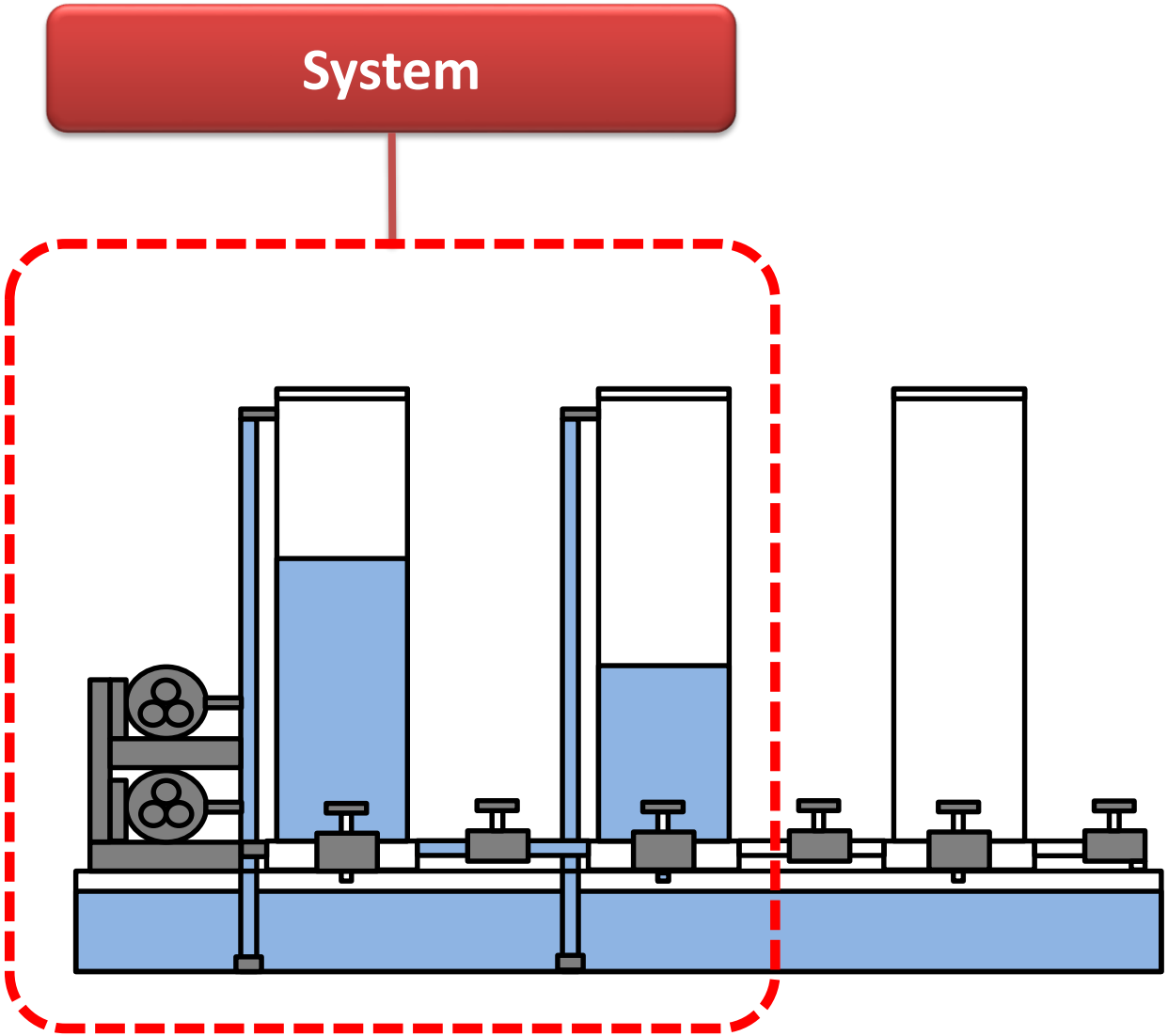
Liquid Tanks



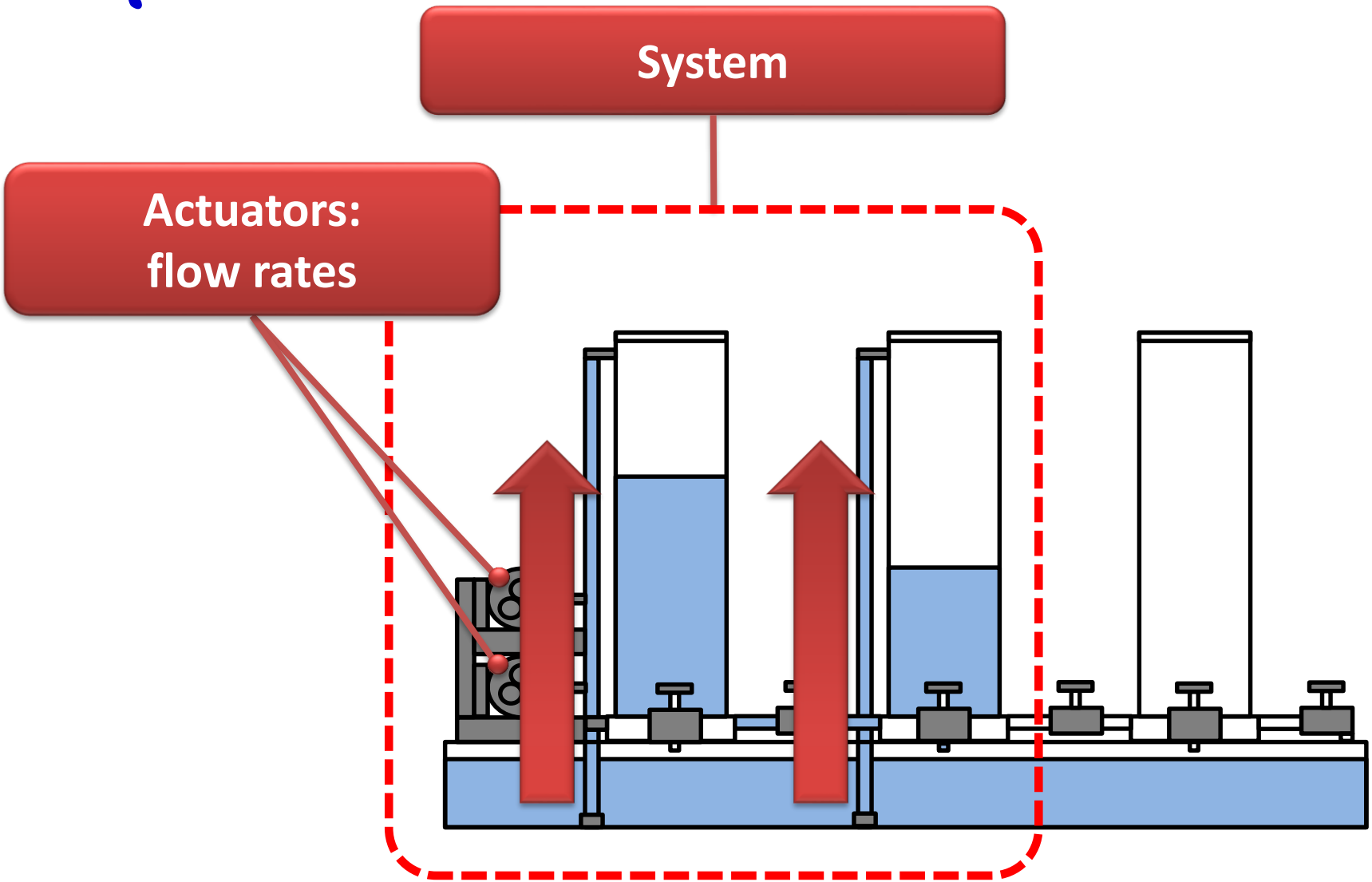
Liquid Tanks



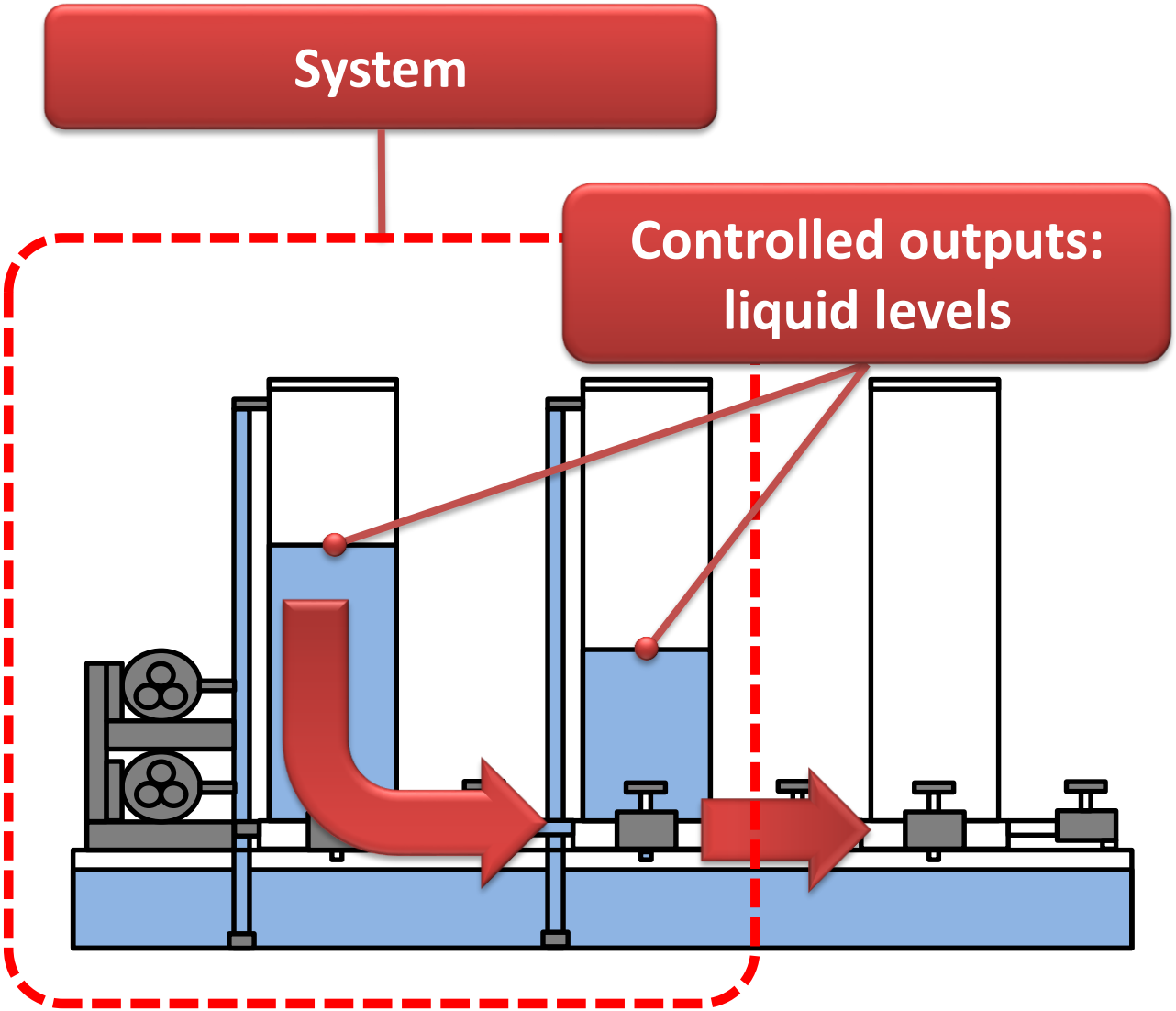
Liquid Tanks



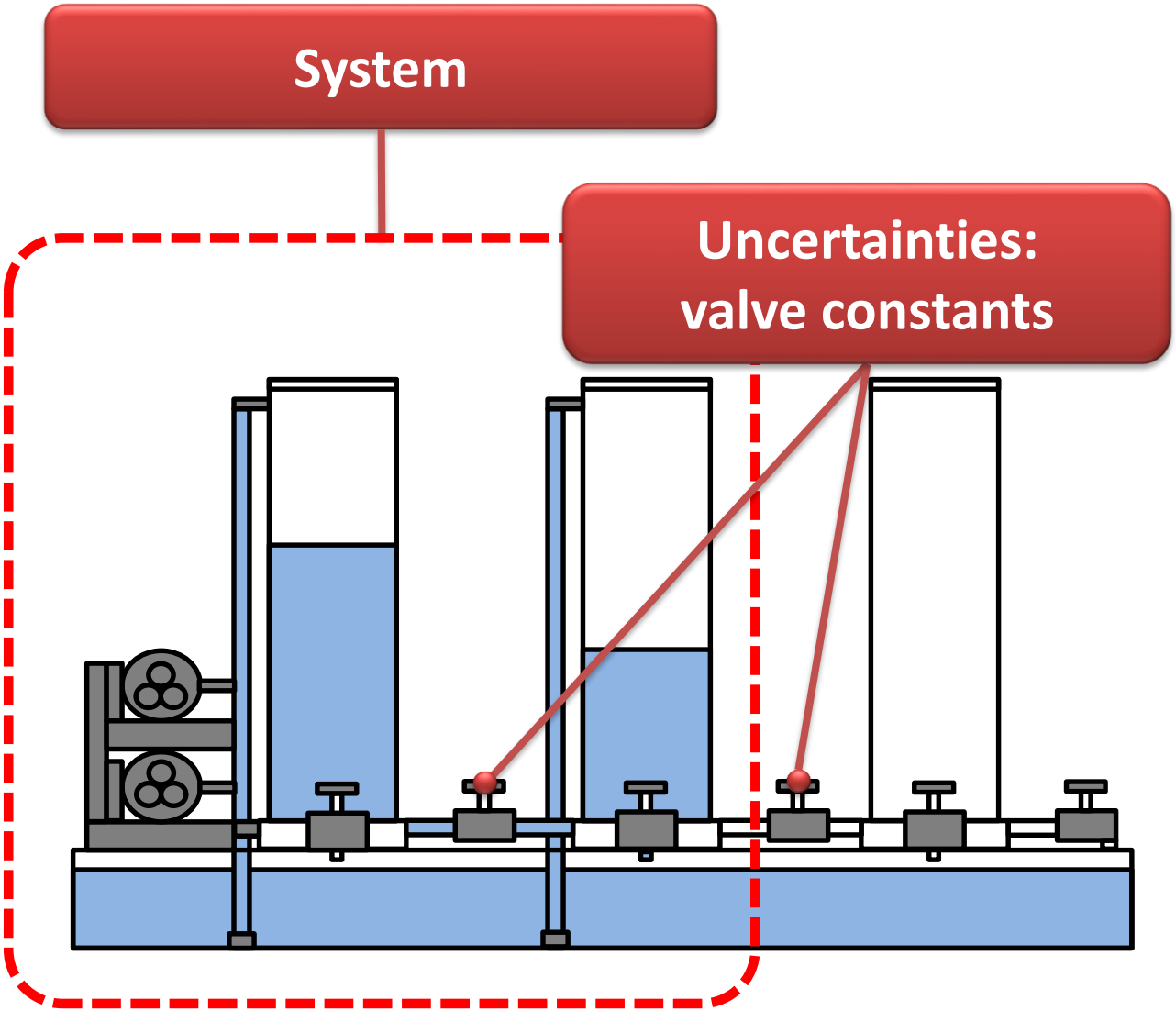
Liquid Tanks



Liquid Tanks



Liquid Tanks



Case Study

physical plant

$u(k)$

$x(k)$

mup

RMPC

Control Performance

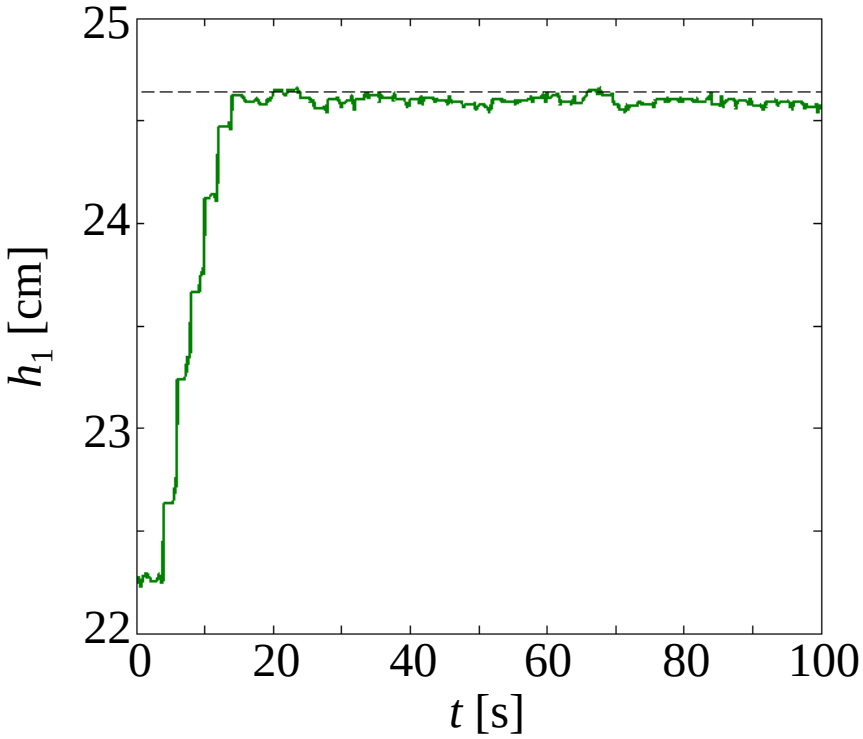
Approach	Lyapunov function	diag(W_x)	diag(W_u)
$RMPC_1$	single LF★	[500, 50]	[1, 0.1]
$RMPC_2$	single LF★	[5, 5]	[1, 1]
$RMPC_3$	PDLF★★	[5000, 50]	[0.1, 1]
$RMPC_4$	PDLF★★	[5000, 50]	[1, 1]

$$J = \sum_{t=0}^{\infty} (h - h_{\text{ref}})^{\top} W_{\text{x}} (h - h_{\text{ref}}) + (q - q_{\text{ref}})^{\top} W_{\text{u}} (q - q_{\text{ref}}))$$

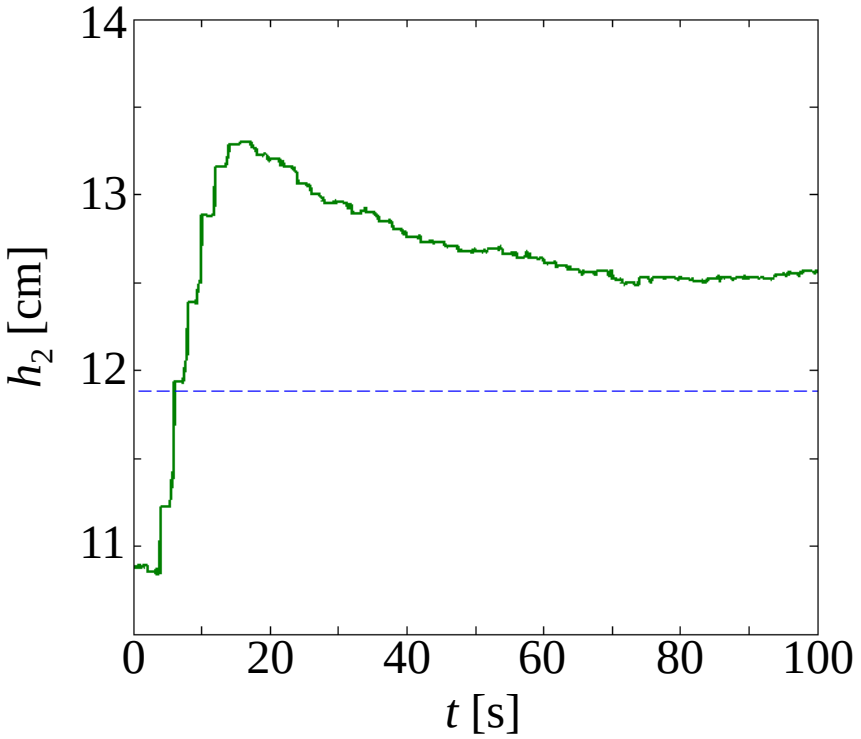
★*Kothare et al., Automatica, 1996;* ★★*Cuzzola et al., Automatica, 2002*

Control Performance

Liquid level in Tank #1
h_1 [cm]



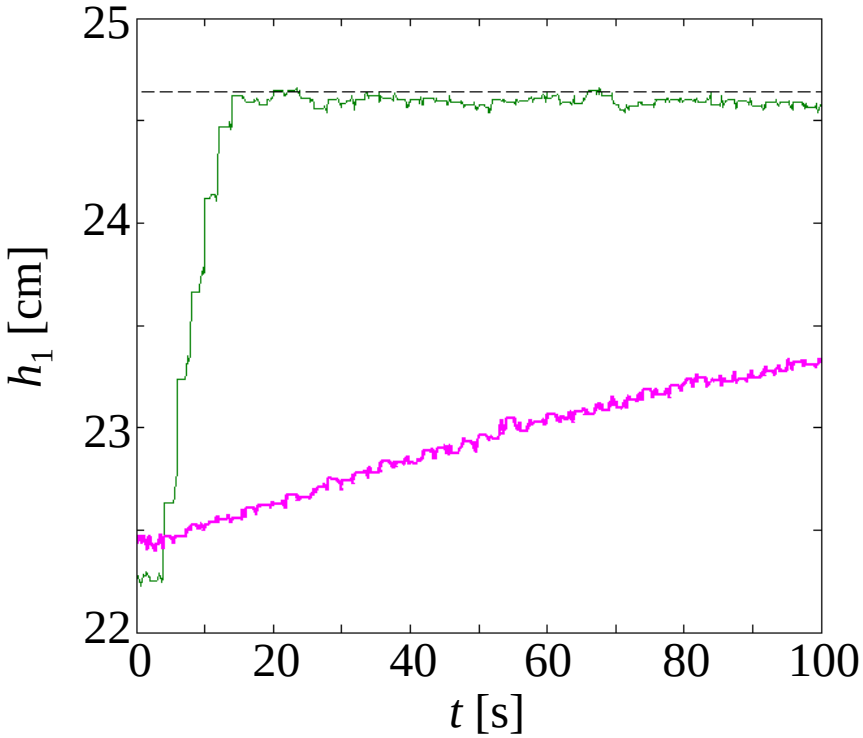
Liquid level in Tank #2
h_2 [cm]



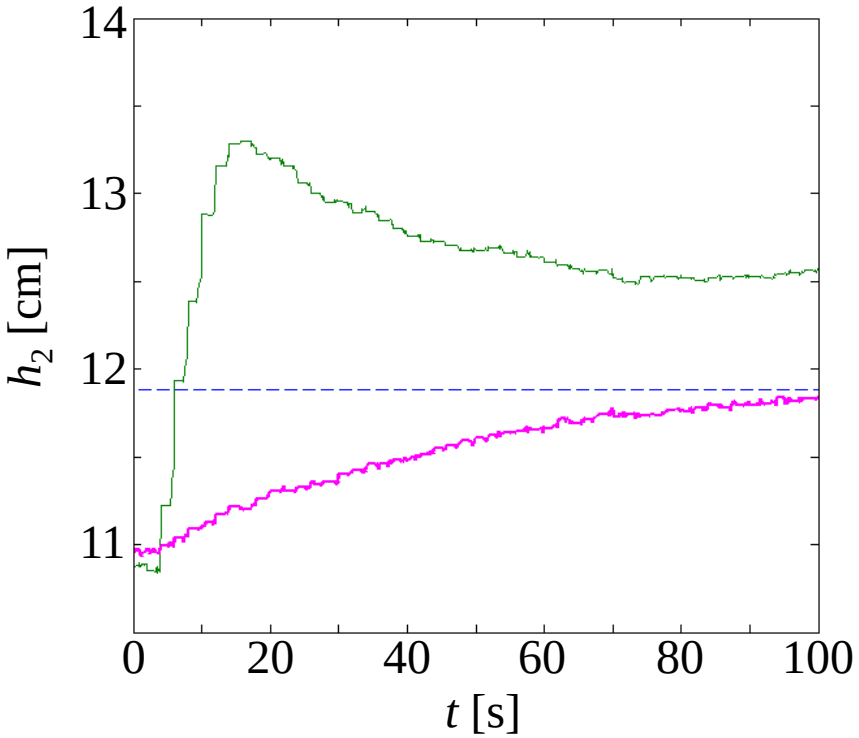
RMPC$_1$

Control Performance

Liquid level in Tank #1
h_1 [cm]



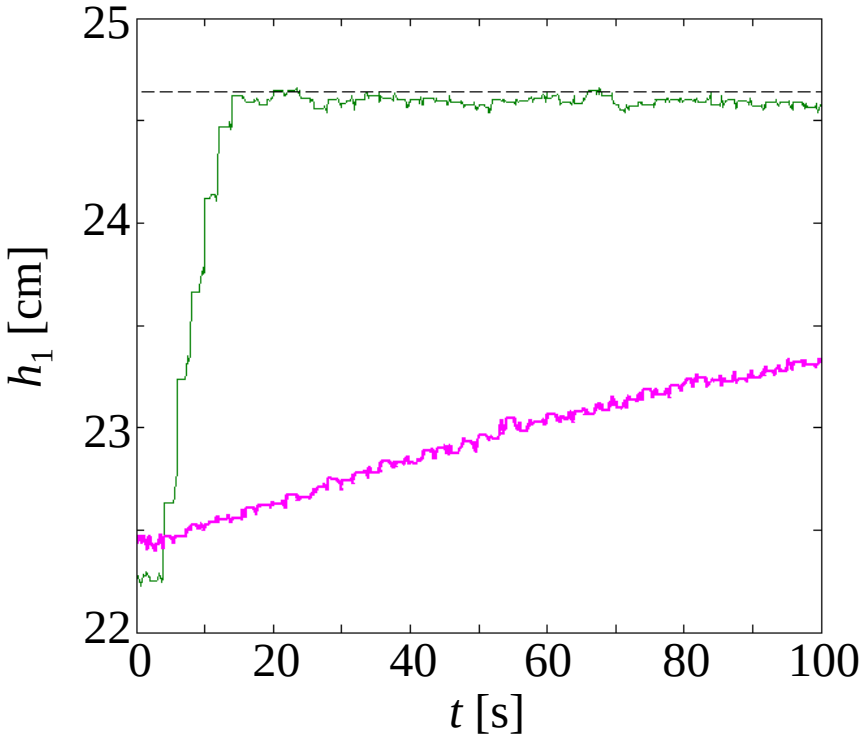
Liquid level in Tank #2
h_2 [cm]



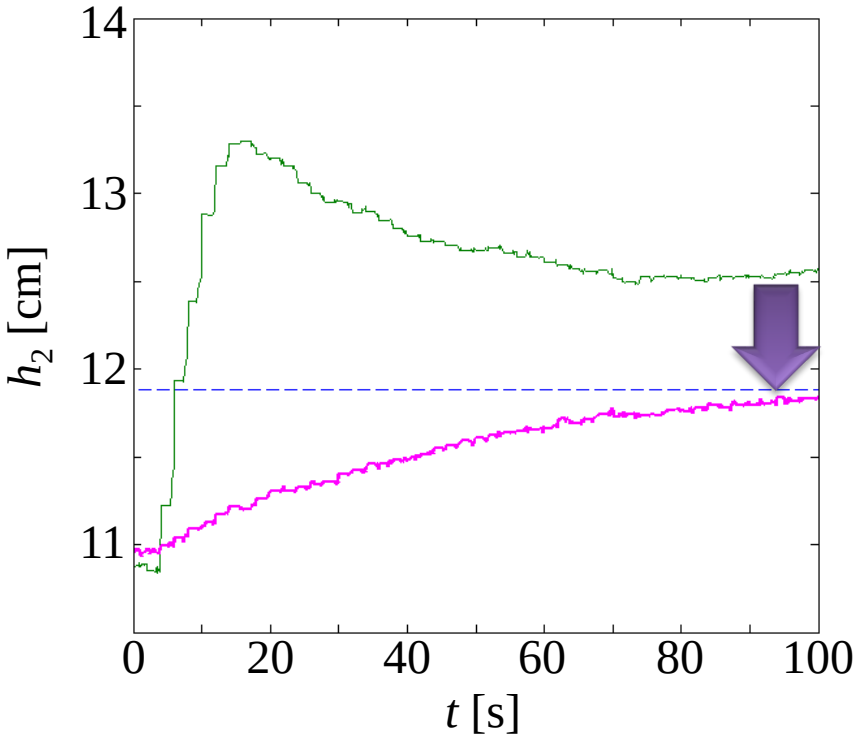
RMPC$_1$ **RMPC$_2$**

Control Performance

Liquid level in Tank #1
h_1 [cm]



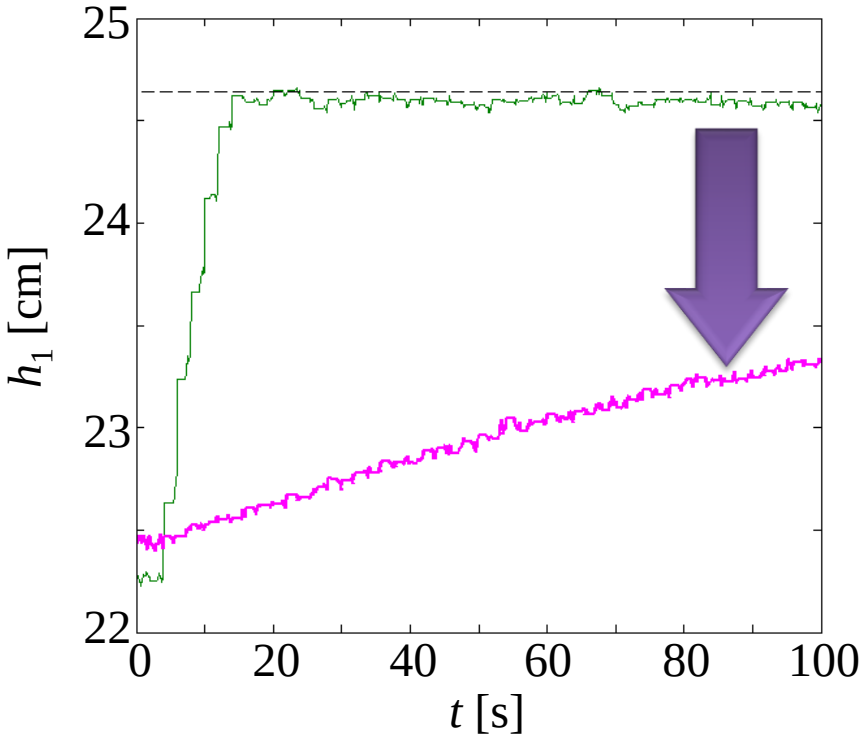
Liquid level in Tank #2
h_2 [cm]



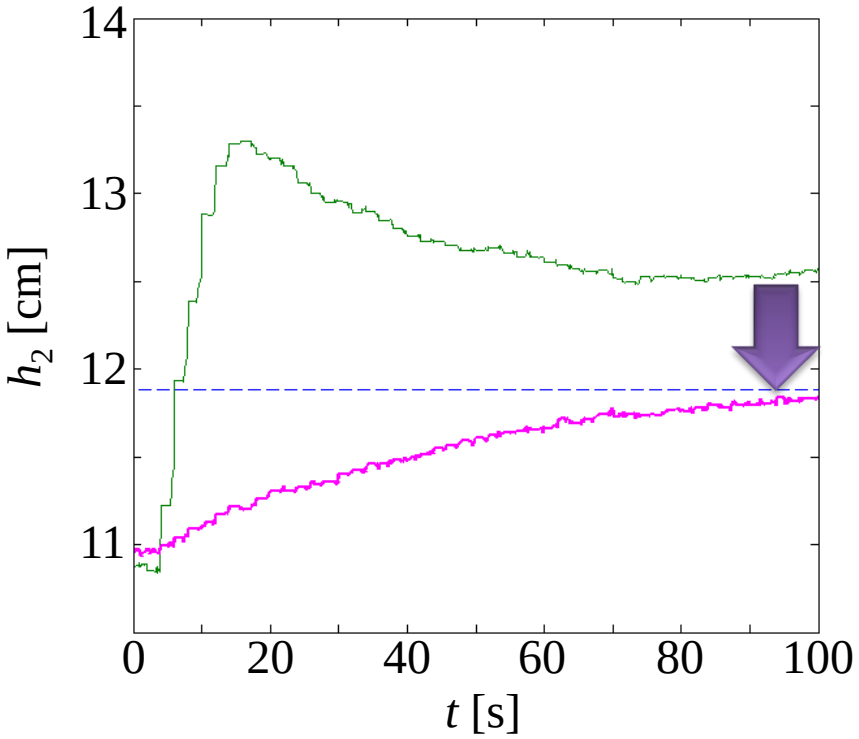
RMPC$_1$ **RMPC$_2$**

Control Performance

Liquid level in Tank #1
h_1 [cm]



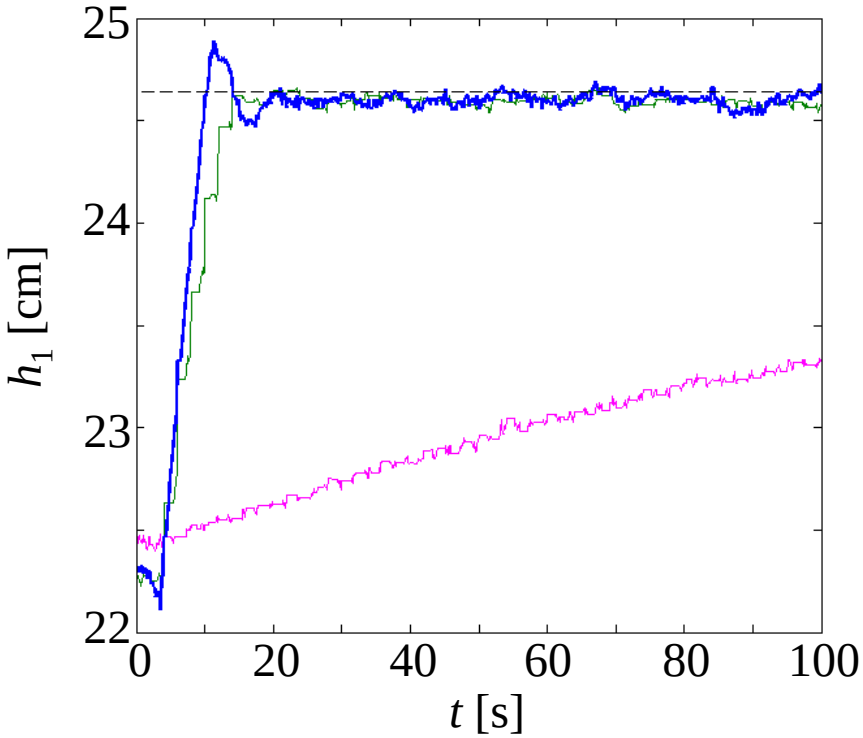
Liquid level in Tank #2
h_2 [cm]



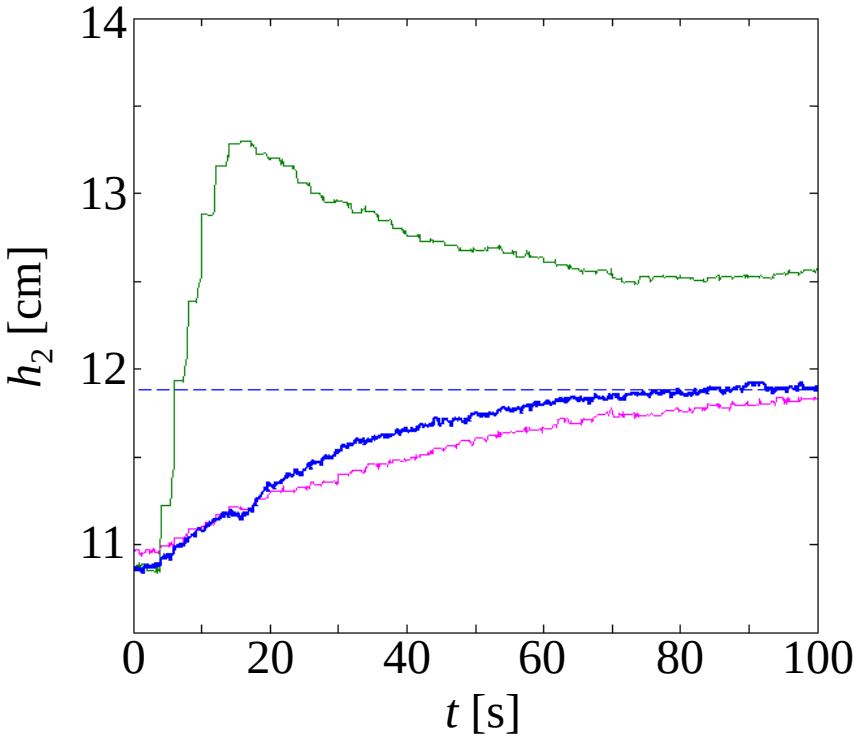
RMPC$_1$ **RMPC$_2$**

Control Performance

Liquid level in Tank #1
h_1 [cm]



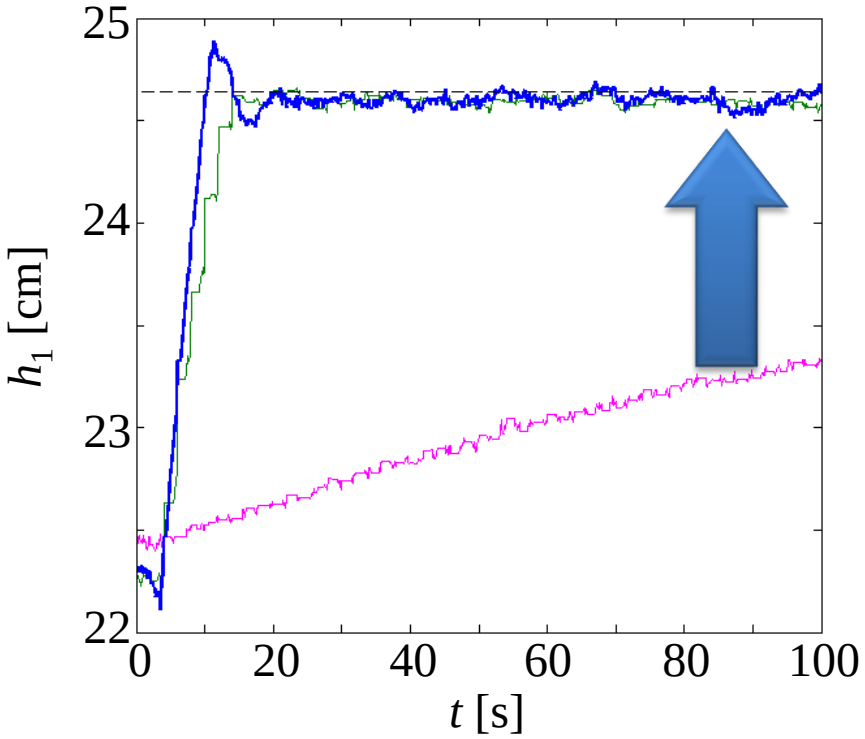
Liquid level in Tank #2
h_2 [cm]



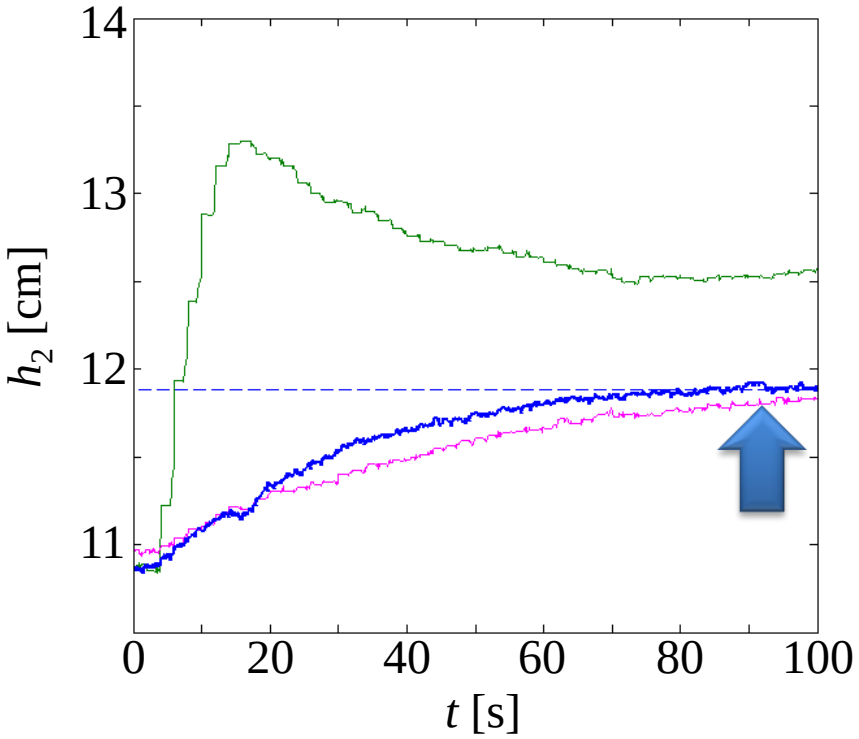
RMPC$_1$ **RMPC$_2$** **RMPC$_3$**

Control Performance

Liquid level in Tank #1
h_1 [cm]



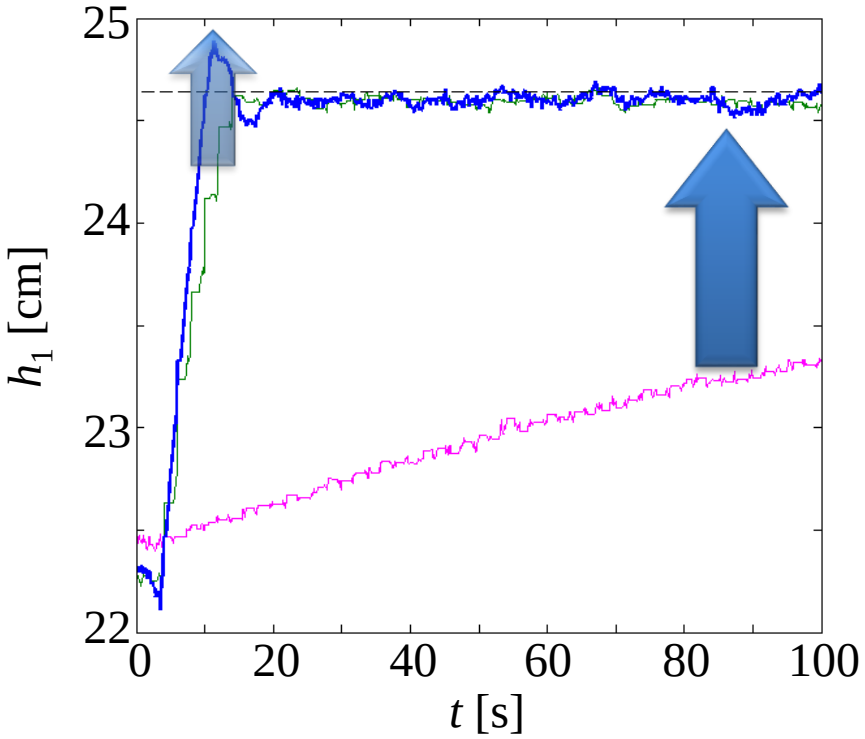
Liquid level in Tank #2
h_2 [cm]



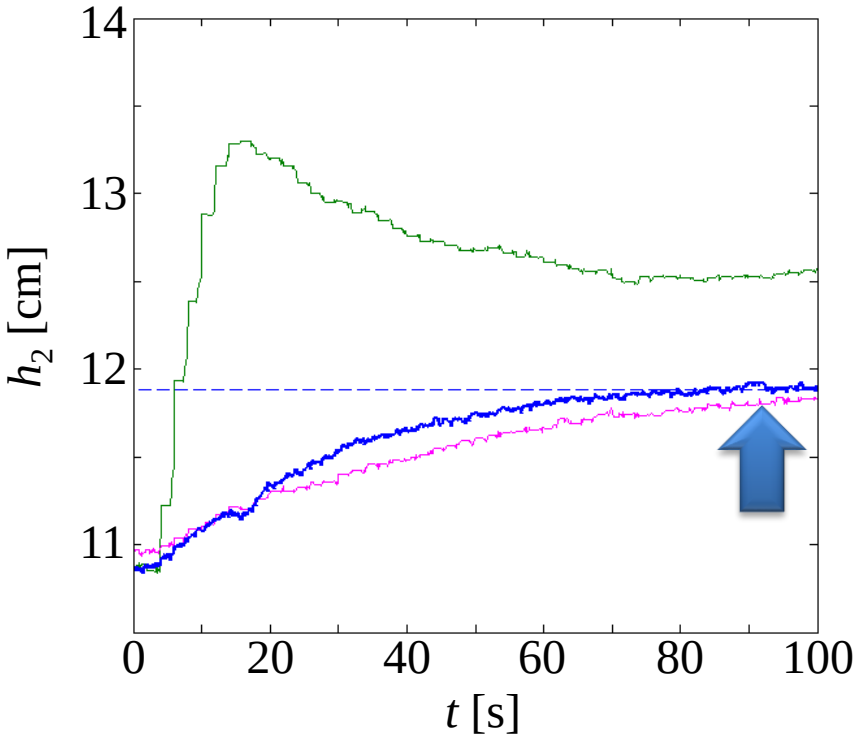
RMPC$_1$ **RMPC$_2$** **RMPC$_3$**

Control Performance

Liquid level in Tank #1
h_1 [cm]



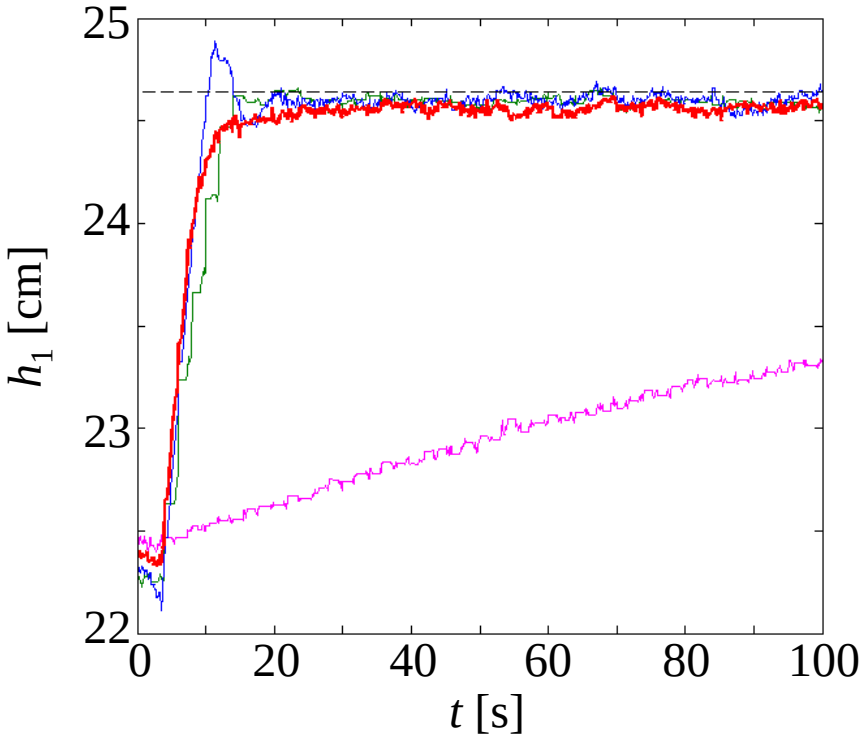
Liquid level in Tank #2
h_2 [cm]



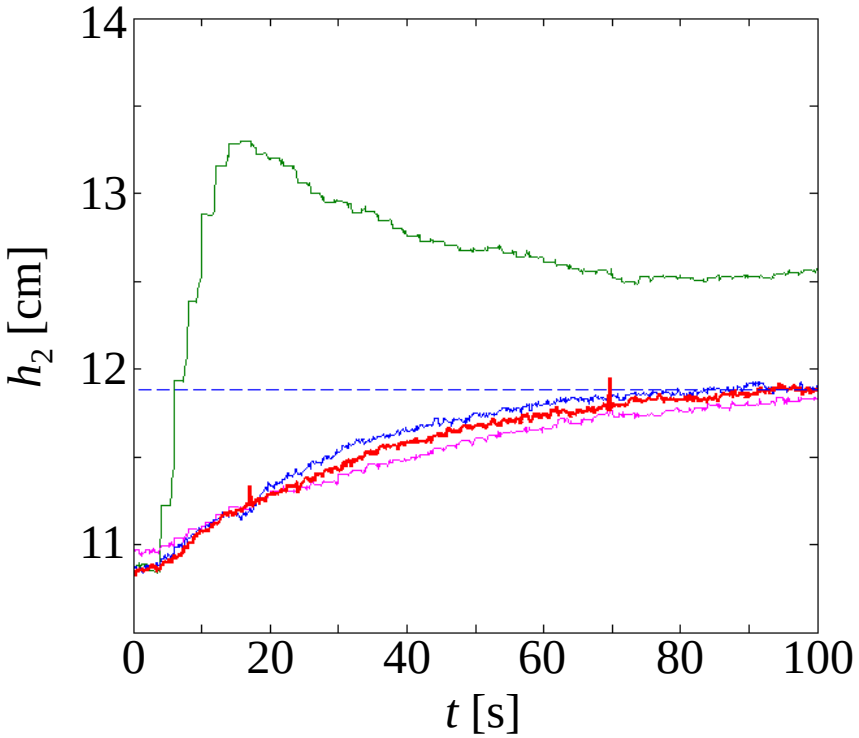
RMPC$_1$ **RMPC$_2$** **RMPC$_3$**

Control Performance

Liquid level in Tank #1
h_1 [cm]



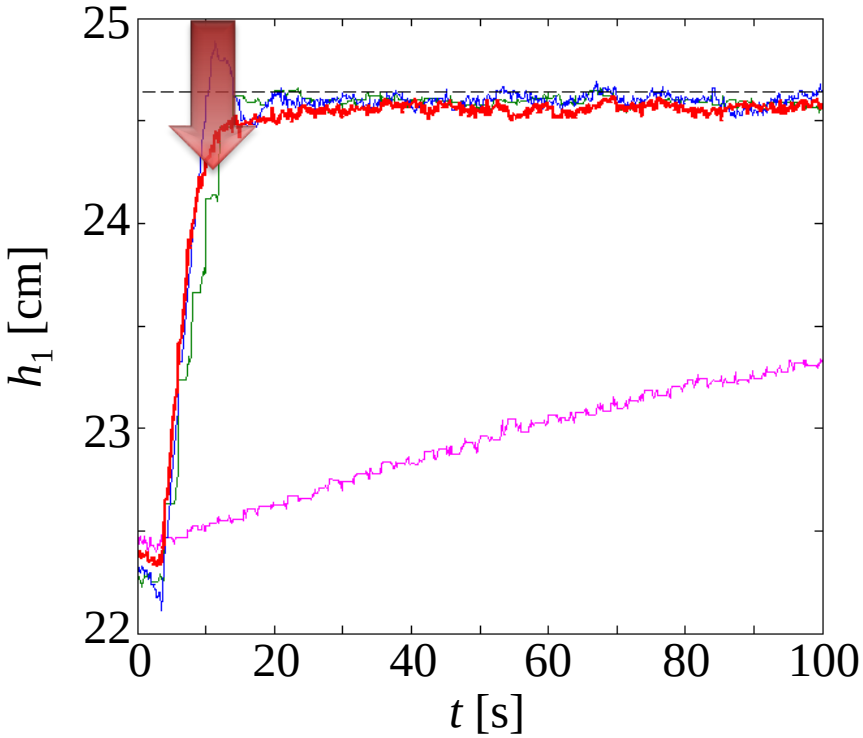
Liquid level in Tank #2
h_2 [cm]



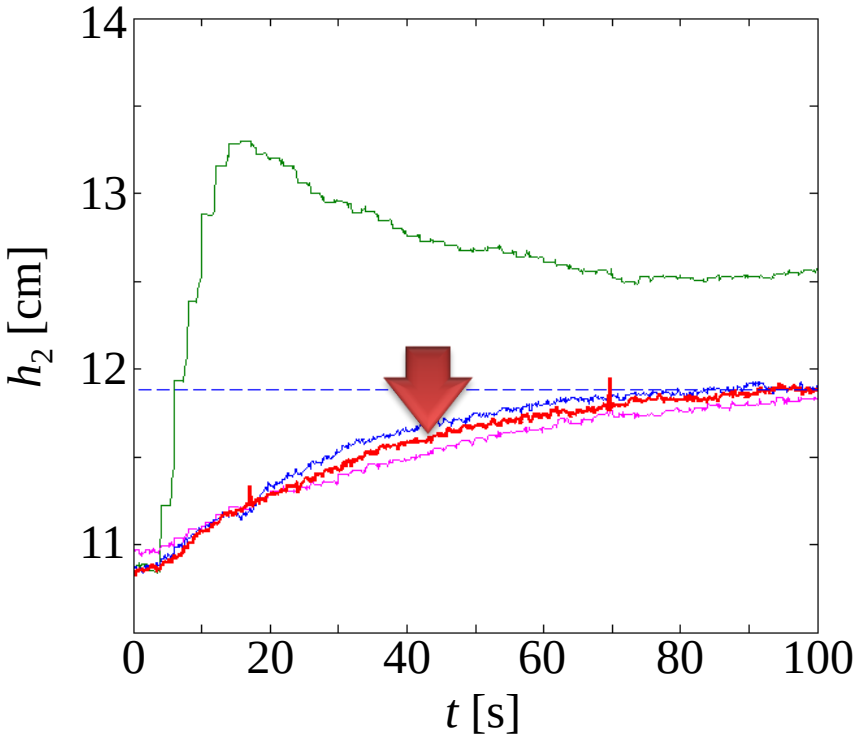
RMPC$_1$ **RMPC$_2$** **RMPC$_3$** **RMPC$_4$**

Control Performance

Liquid level in Tank #1
h_1 [cm]

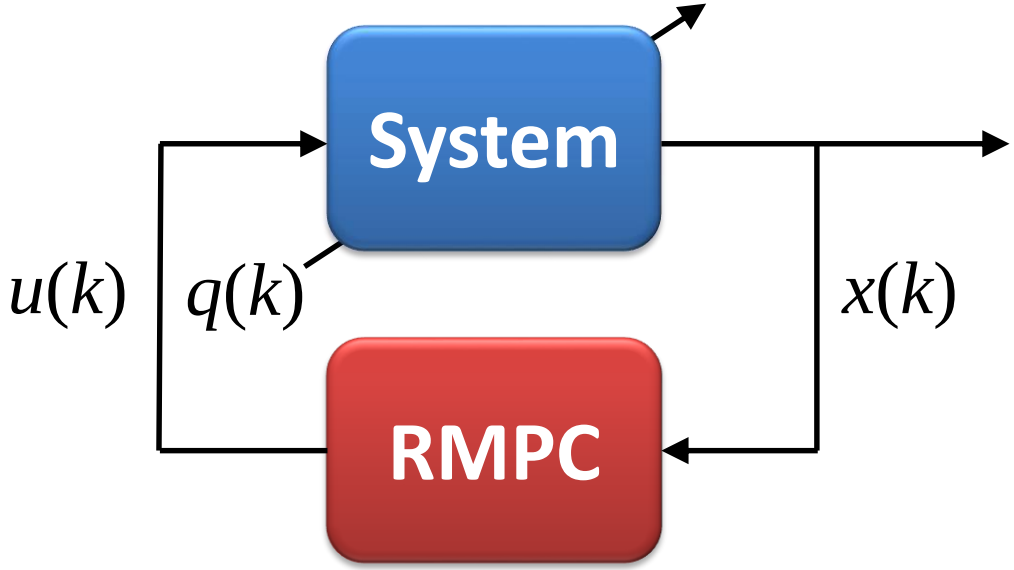


Liquid level in Tank #2
h_2 [cm]



RMPC$_1$ **RMPC$_2$** **RMPC$_3$** **RMPC$_4$**

Toolbox



MATLAB toolbox for on-line RMPC design by LMIs

- speed up RMPC design and tuning
- advanced RMPC design analysis
- user-friendly tool

★ ***bitbucket.org/oravec/mup***

References

[1] *J. Löfberg* (2004)*: YALMIP: A Toolbox for Modeling and Optimization in MATLAB. IEEE International Symposium on Computer Aided Control Systems Design, Taipei, Taiwan,* 284-289*.*

[2] *J. F. Sturm* (1999)*: Using SeDuMi* 1.02*, A MATLAB Toolbox for Optimization Over Symmetric Cones.* 1-30*.*

[3] *M. Bakošová, J. Oravec* (2014)*: Robust Model Predictive Control of a Laboratory Two-Tank System. In American Control Conference, Portland, Oregon, USA,* 5242–5247*.*

[4] *J. Oravec, M. Bakošová* (2015)*: Software for Efficient LMI-based Robust MPC Design. In IEEE International Conference on Process Control, Slovakia,* 272–277*.*

[5] *J. Oravec, M. Bakošová* (2016)*: Soft Constraints in the Robust MPC Design via LMIs. In American Control Conference, Boston, USA.*