Multi-Parametric Toolbox (MPT)

Michal Kvasnica and Colin Jones

Prototype & Deploy Optimization-Bazed Controllers

Motivating Example

JR-Maglev (581 km/h)

Motivating Example

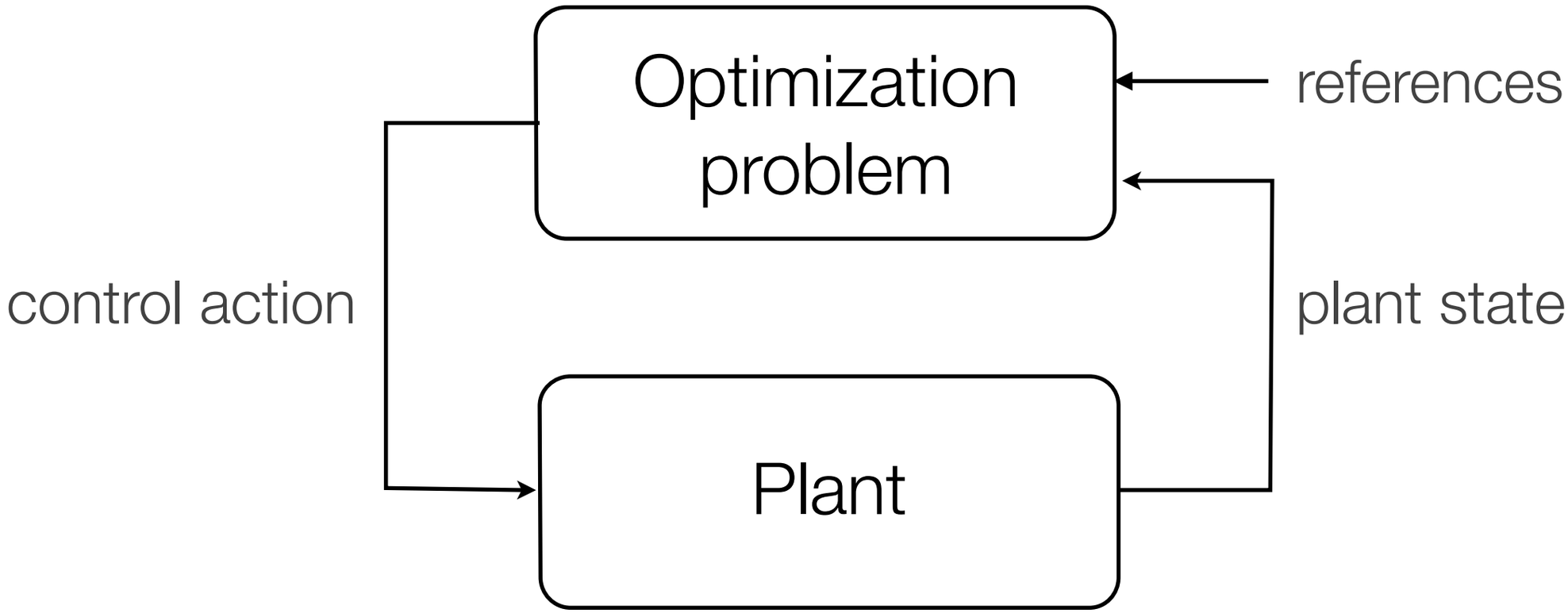
Fast linear dynamics

All states can be measured

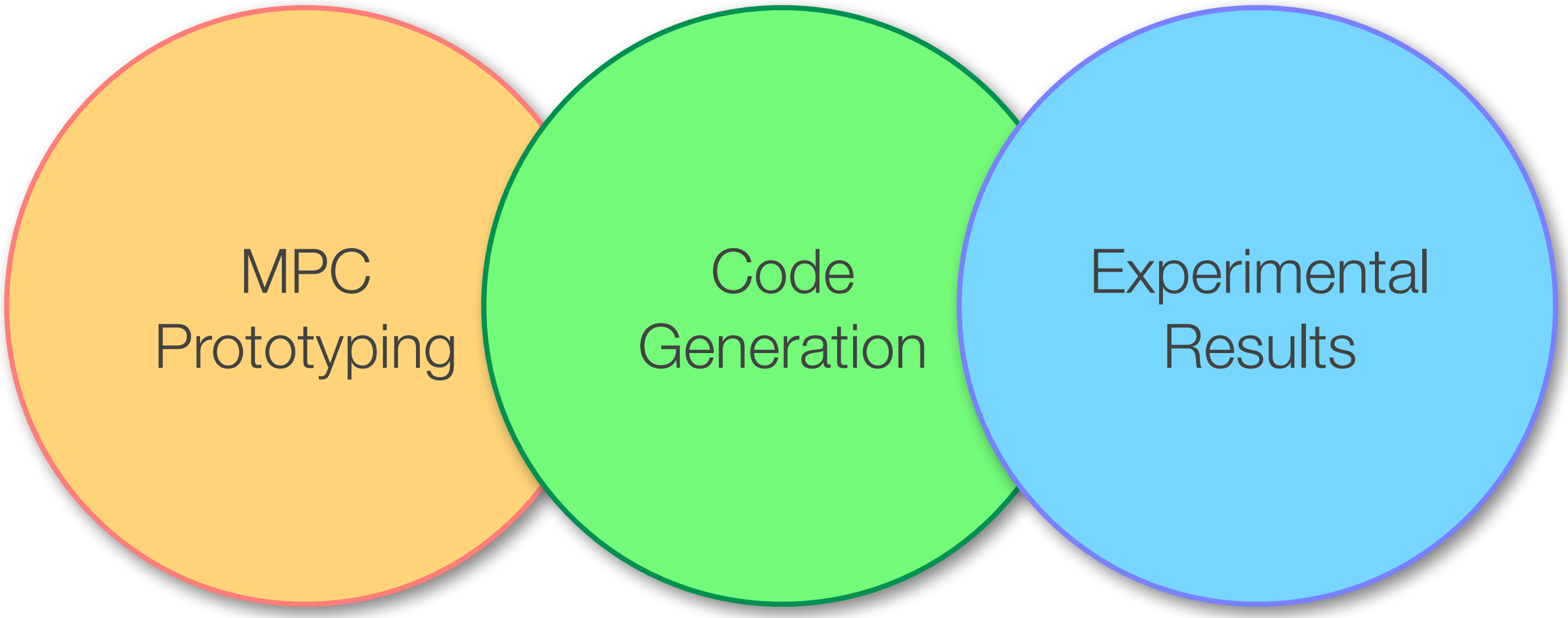
Constraints on inputs & outputs

Simple implementation hardware

Model Predictive Control



Agenda



MPC Prototyping

$$\underbrace{\begin{bmatrix} x_t \\ x_{t+1} \\ x_{t+2} \\ \vdots \\ x_{t+N-1} \end{bmatrix}}_{X} = \underbrace{\begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^{N-1} \end{bmatrix}}_{\tilde{A}} x_t + \underbrace{\begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ B & 0 & 0 & \cdots & 0 \\ AB & B & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ A^{N-2}B & A^{N-3}B & \cdots & B & 0 \end{bmatrix}}_{\tilde{B}} \underbrace{\begin{bmatrix} u_t \\ u_{t+1} \\ u_{t+2} \\ \vdots \\ u_{t+N-1} \end{bmatrix}}_{U}$$

MPC Prototyping

Regulation with disturbance rejection

Reference tracking

Soft constraints

Slew rate constraints

Custom setups

Disturbance Rejection

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q y_k^2 + r u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(q)
sys.u.penalty = QuadFunction(r)
N = 5
ctrl = MPCController(sys, N)
ctrl.construct()
```

Command-Line Implementation

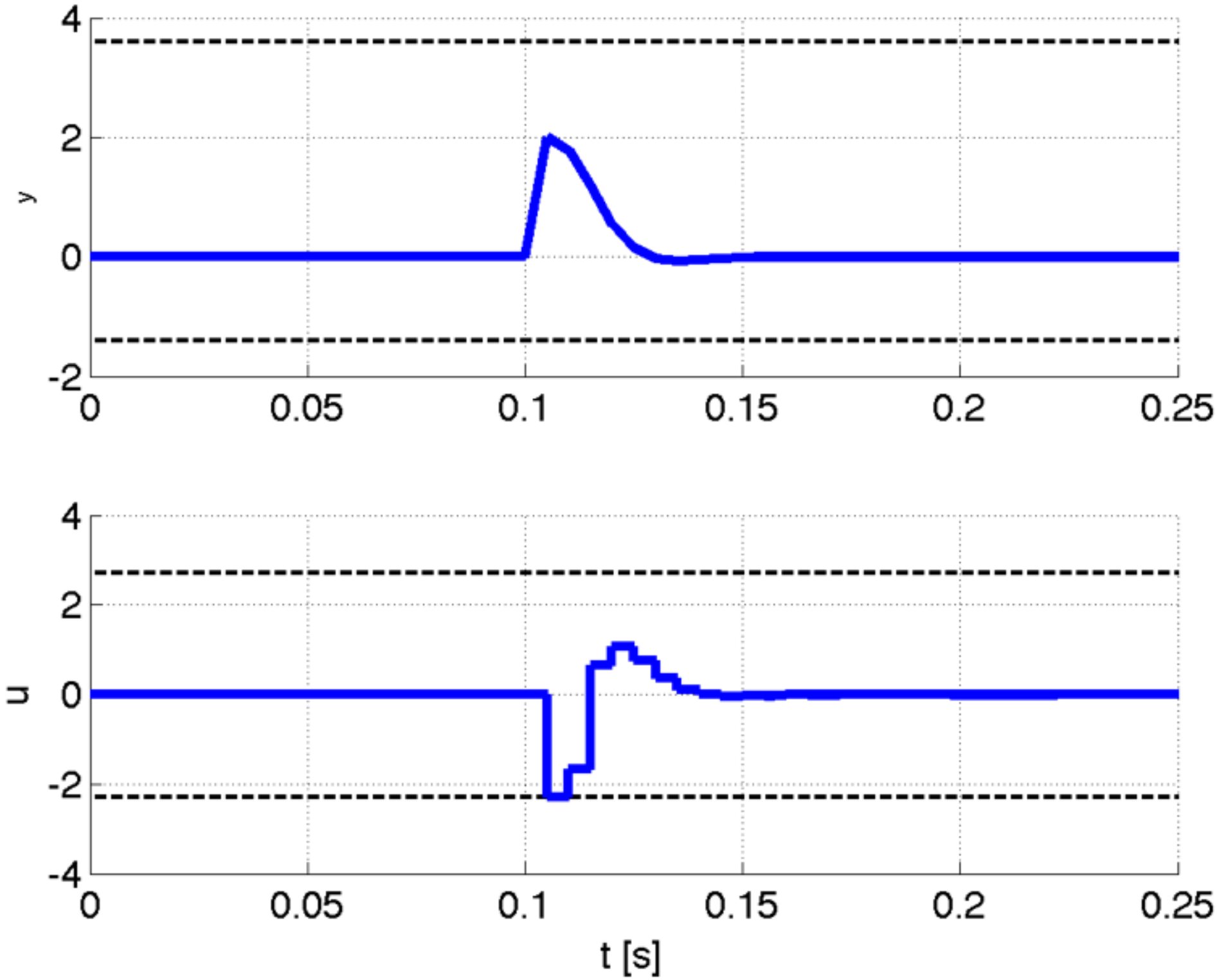
```
u0 = ctrl.evaluate(x0)

u0 =

   -2.2900
```

```
[u0, feasible, openloop] = ctrl.evaluate(x0)

openloop =

    cost: 96.5649
       U: [-2.2900 -1.7824 0.1978 0.2407 0]
       X: [2x6 double]
       Y: [2 1.7822 1.1774 0.4492 -0.2010]
```

Closed-Loop Implementation (Command Line)

```
loop = ClosedLoop(ctrl, sys)
data = loop.simulate(x0, Nsim)

data =

        X: [2x31 double]
        U: [1x30 double]
        Y: [1x30 double]
     cost: [1x30 double]
```

Disturbance Rejection



MPC Prototyping

Regulation with disturbance rejection

Reference tracking

Soft constraints

Slew rate constraints

Custom setups

Tracking of a Constant Reference

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(10)
sys.u.penalty = QuadFunction(1)


N = 5
ctrl = MPCController(sys, N)
ctrl.construct()
```

Tracking of a Constant Reference

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(10)
sys.u.penalty = QuadFunction(1)
sys.y.with('reference')

N = 5
ctrl = MPCController(sys, N)
ctrl.construct()
```

Tracking of a Constant Reference

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

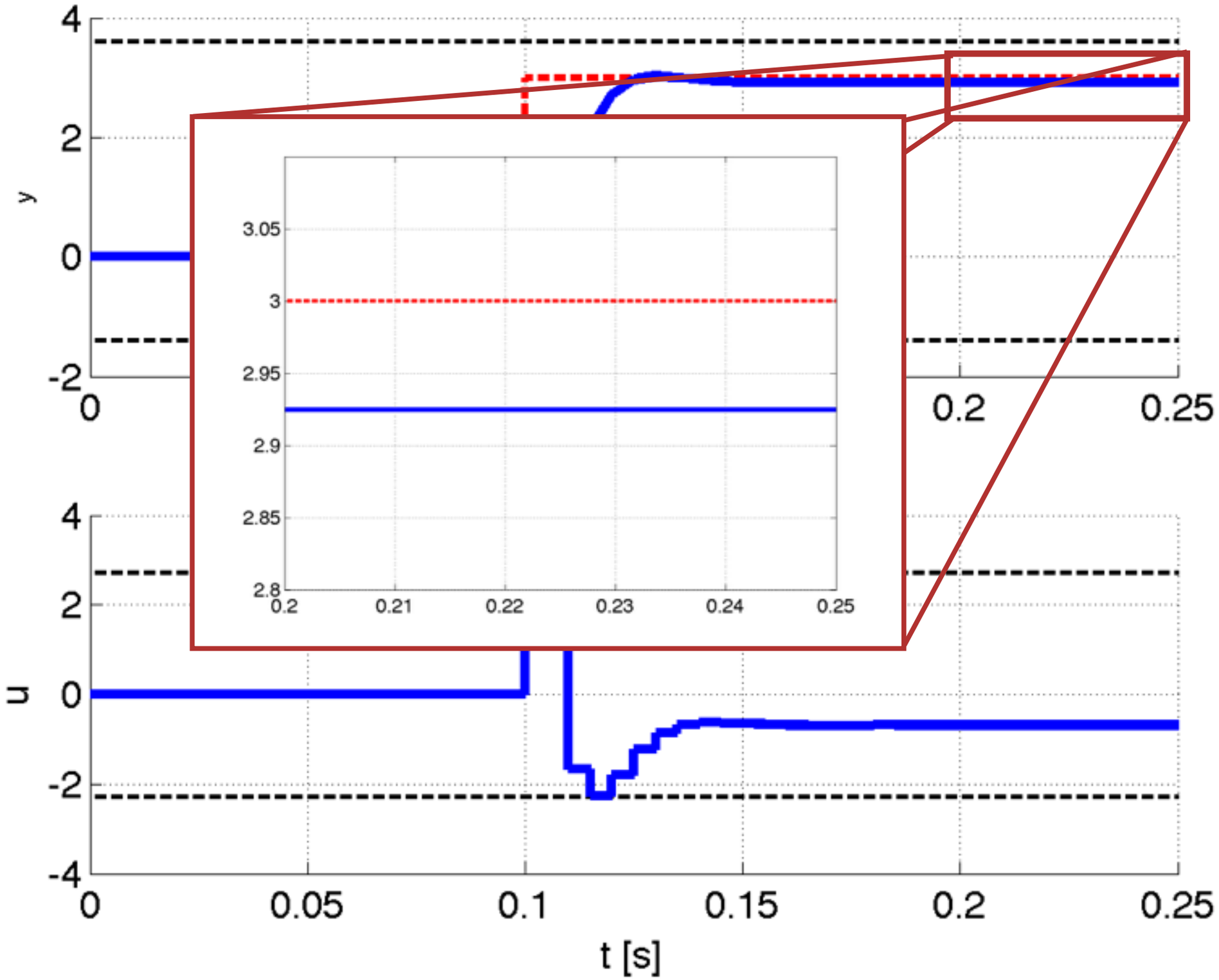
$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(10)
sys.u.penalty = QuadFunction(1)
sys.y.with('reference')
sys.y.reference = 3
N = 5
ctrl = MPCController(sys, N)
ctrl.construct()
```

Tracking of a Constant Reference



Tracking of a Constant Reference

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(10)
sys.u.penalty = QuadFunction(1)

sys.y.with('reference')
sys.y.reference = 3
N = 5
ctrl = MPCController(sys, N)
```

Tracking of a Constant Reference

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(10)
sys.u.penalty = QuadFunction(1)

sys.y.with('reference')
sys.y.reference = 3
N = 5
ctrl = MPCController(sys, N)
```

Tracking of a Constant Reference

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(10)
sys.u.with('deltaPenalty')

sys.y.with('reference')
sys.y.reference = 3
N = 5
ctrl = MPCController(sys, N)
```

Tracking of a Constant Reference

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

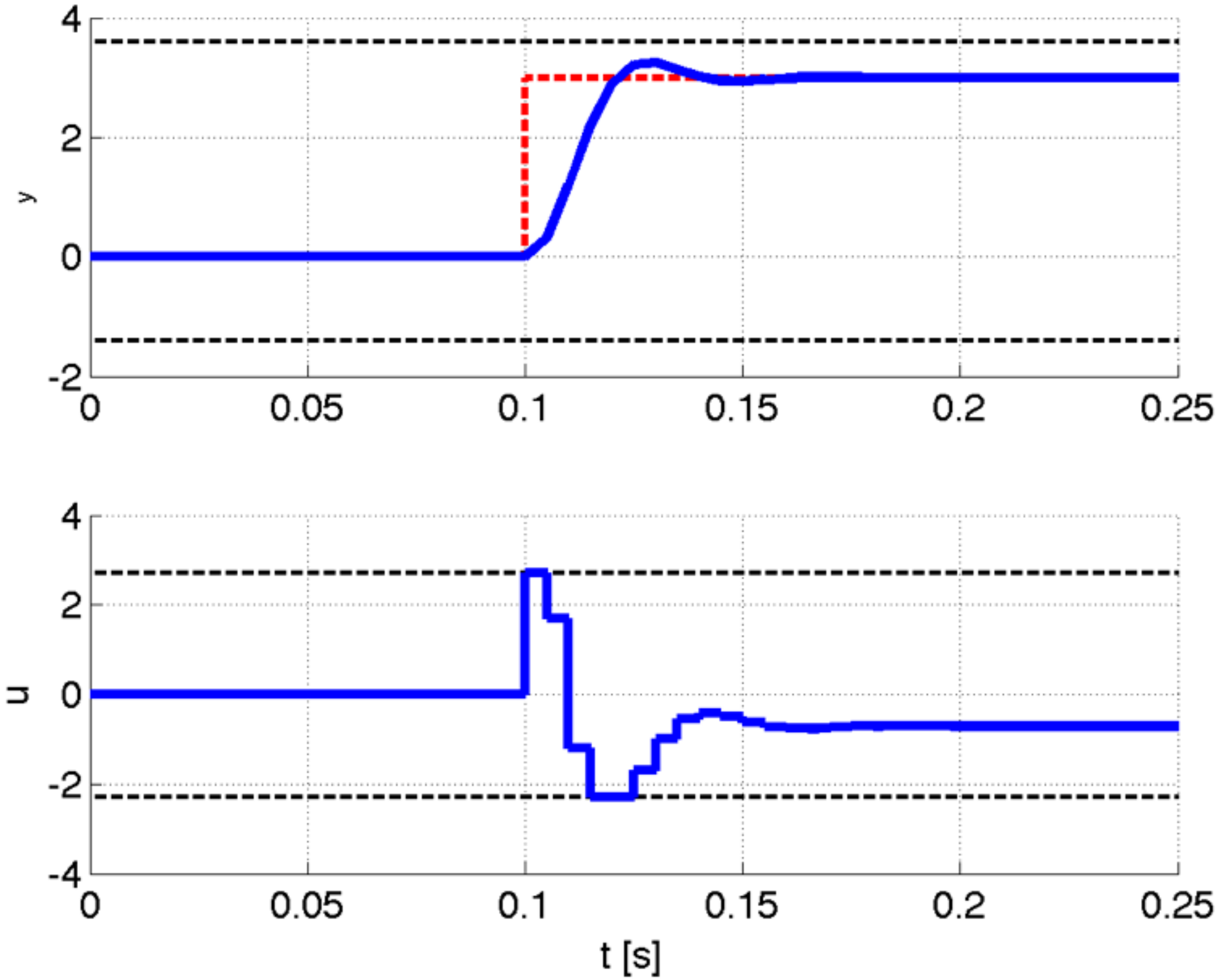
$$y_{\min} \leq y_k \leq y_{\max}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(10)
sys.u.with('deltaPenalty')
sys.u.deltaPenalty = QuadFunction(1)
sys.y.with('reference')
sys.y.reference = 3
N = 5
ctrl = MPCController(sys, N)
```

Tracking of a Constant Reference



Tracking of a Time-Varying References

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(1)
sys.u.with('deltaPenalty')
sys.u.deltaPenalty = QuadFunction(1)
sys.y.with('reference')
sys.y.reference = 3
N = 5
ctrl = MPCController(sys, N)
```

Tracking of a Time-Varying References

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

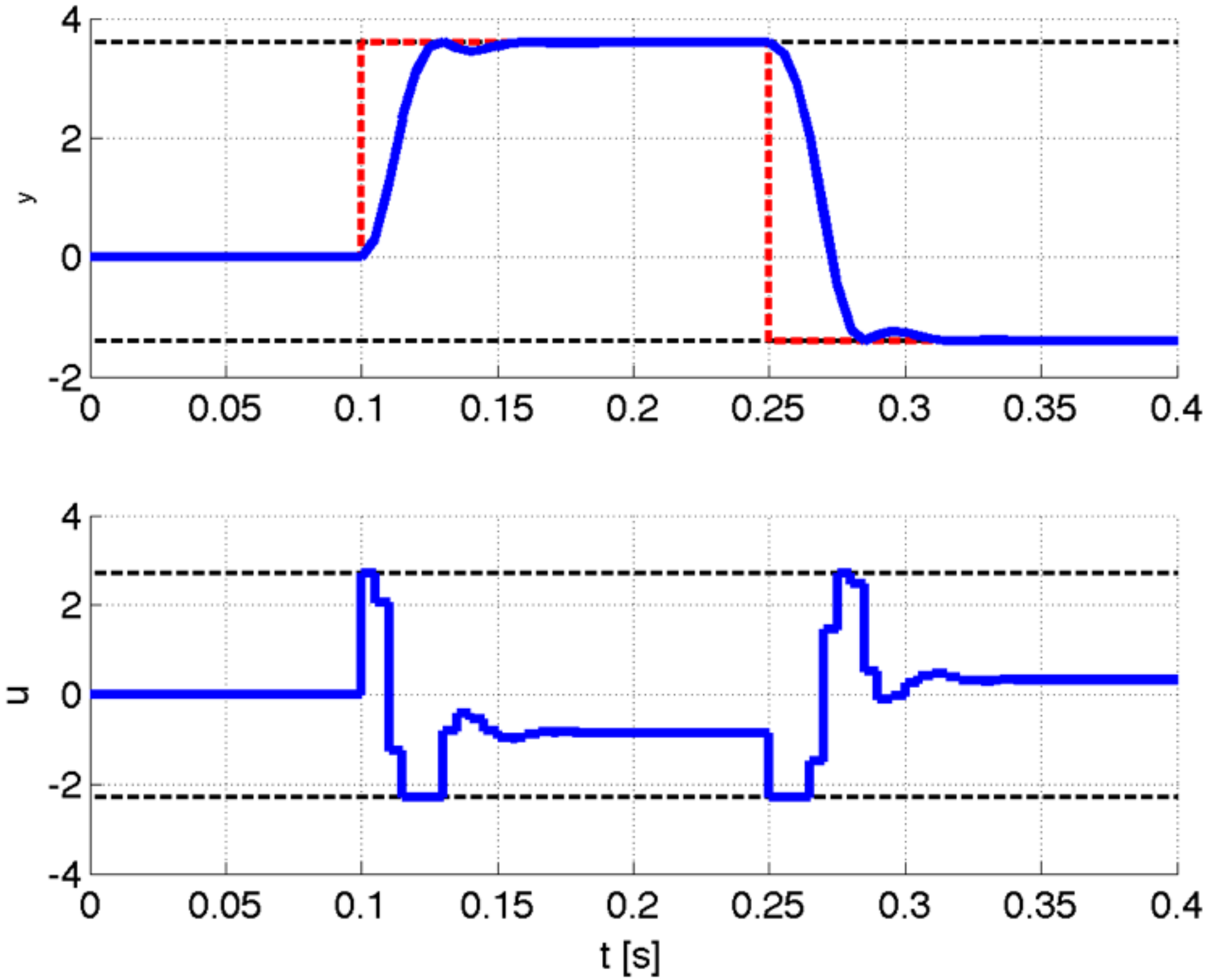
$$y_{\min} \leq y_k \leq y_{\max}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$x_0 = x(t)$$

```
sys = LTISystem('A', A, 'B', B, 'C', C)
sys.u.min = -2.29, sys.u.max = 2.71
sys.y.min = -1.40, sys.y.max = 3.60
sys.y.penalty = QuadFunction(1)
sys.u.with('deltaPenalty')
sys.u.deltaPenalty = QuadFunction(1)
sys.y.with('reference')
sys.y.reference = 'free'
N = 5
ctrl = MPCController(sys, N)
```

Tracking of a Time-Varying References



MPC Prototyping

Regulation with disturbance rejection

Reference tracking

Soft constraints

Slew rate constraints

Custom setups

Soft Constraints

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

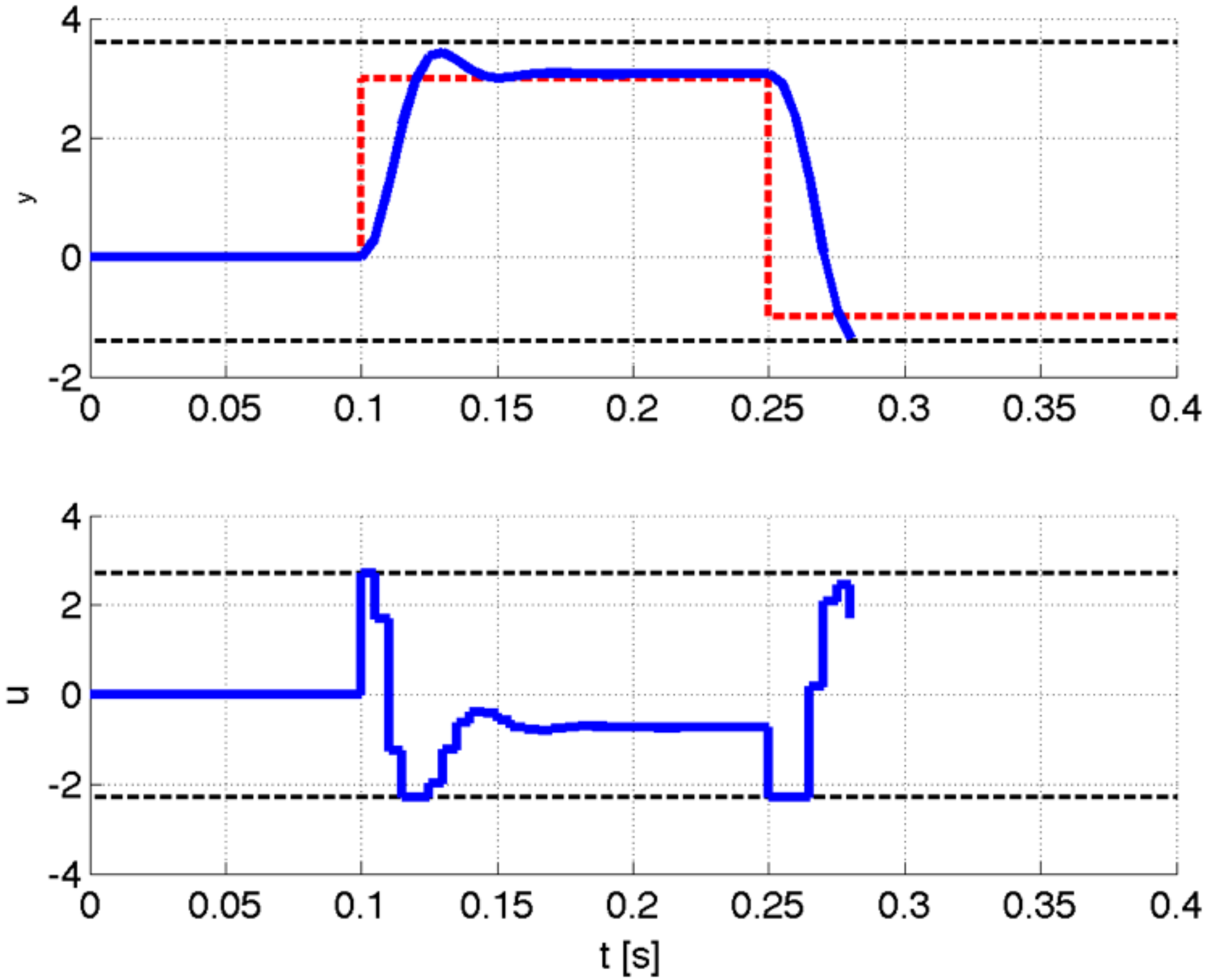
$$y_{\min} \leq y_k \leq y_{\max}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$x_0 = x(t)$$

```
simsys = LTISystem('A', A+0.01*A, 'B', B, 'C', C)
loop = ClosedLoop(ctrl, simsys)
```

Tracking of a Time-Varying References



Soft Constraints

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2 + sz_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} - z_k \leq y_k \leq y_{\max} + z_k$$

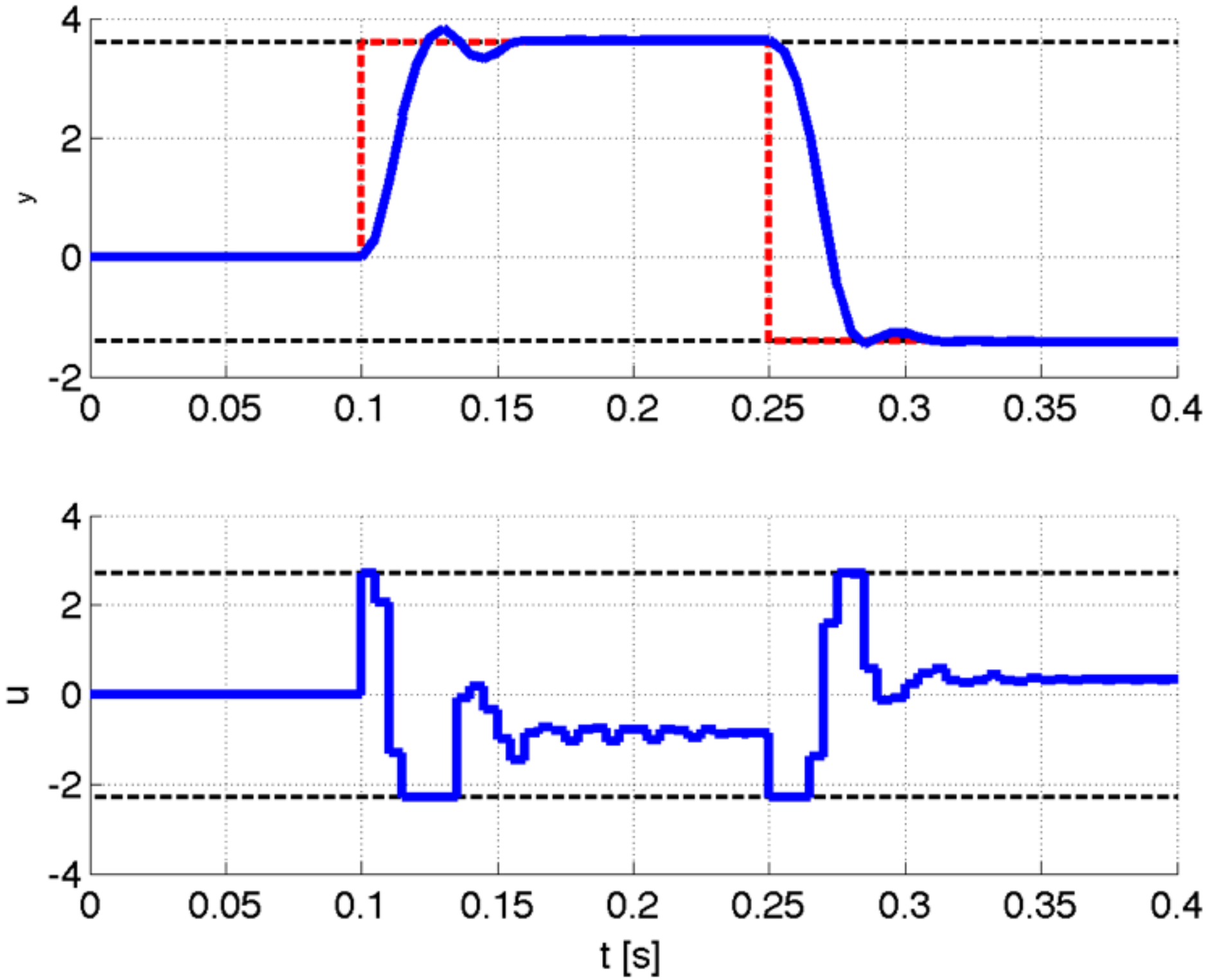
$$0 \leq z_k \leq z_{\max}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$x_0 = x(t)$$

```
sys.y.with('softMin')
sys.y.softMin.maximalViolation = zmax
sys.y.softMin.penalty = QuadFunction(s)

sys.y.with('softMax')
sys.y.softMax.maximalViolation = zmax
sys.y.softMax.penalty = QuadFunction(s)
```

Soft Constraints



MPC Prototyping

Regulation with disturbance rejection

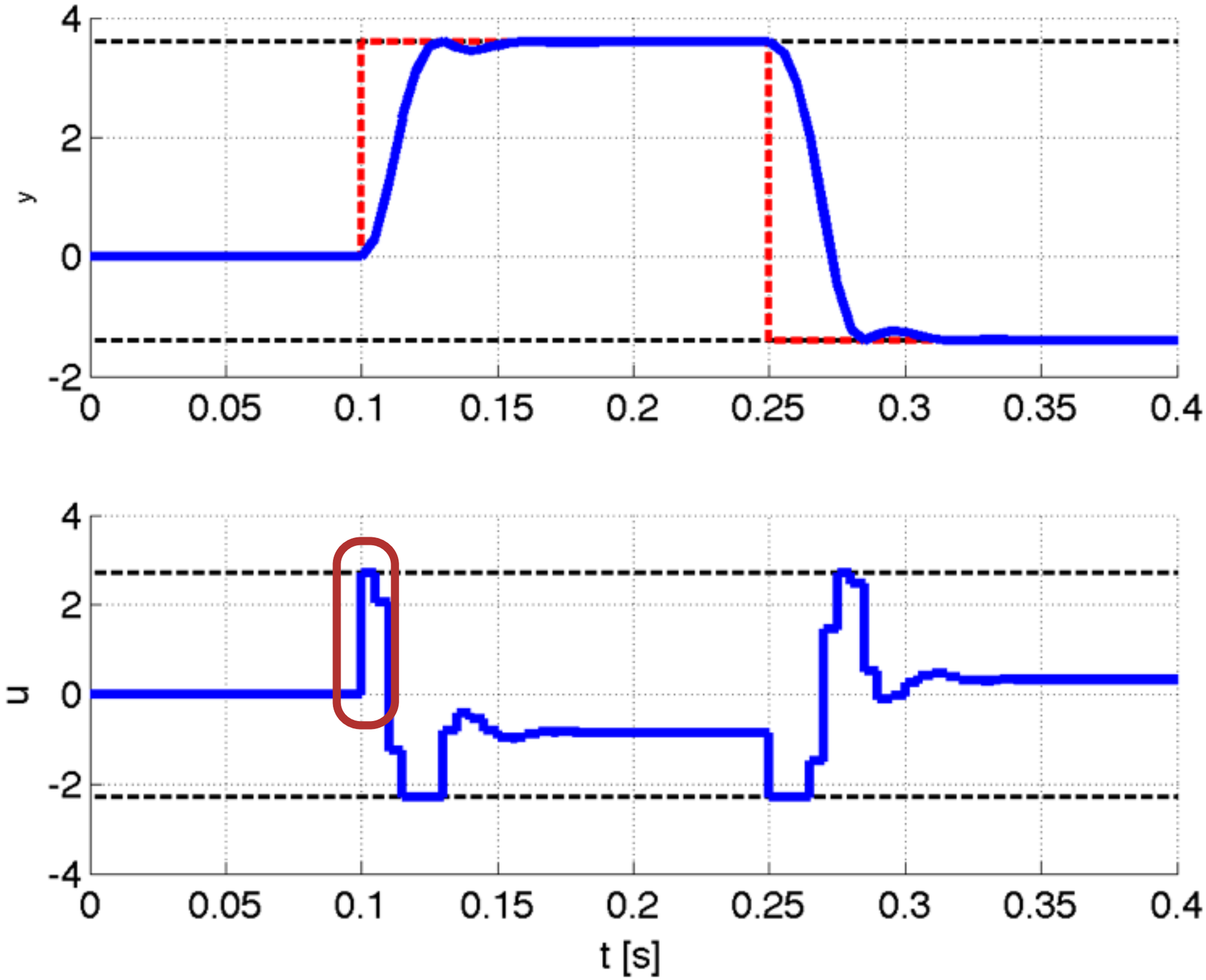
Reference tracking

Soft constraints

Slew rate constraints

Custom setups

Slew Rate Constraints



Slew Rate Constraints

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} q(y_k - y_{\text{ref}})^2 + r\Delta u_k^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ y_k = Cx_k$$

$$u_{\min} \leq u_k \leq u_{\max}$$

$$y_{\min} \leq y_k \leq y_{\max}$$

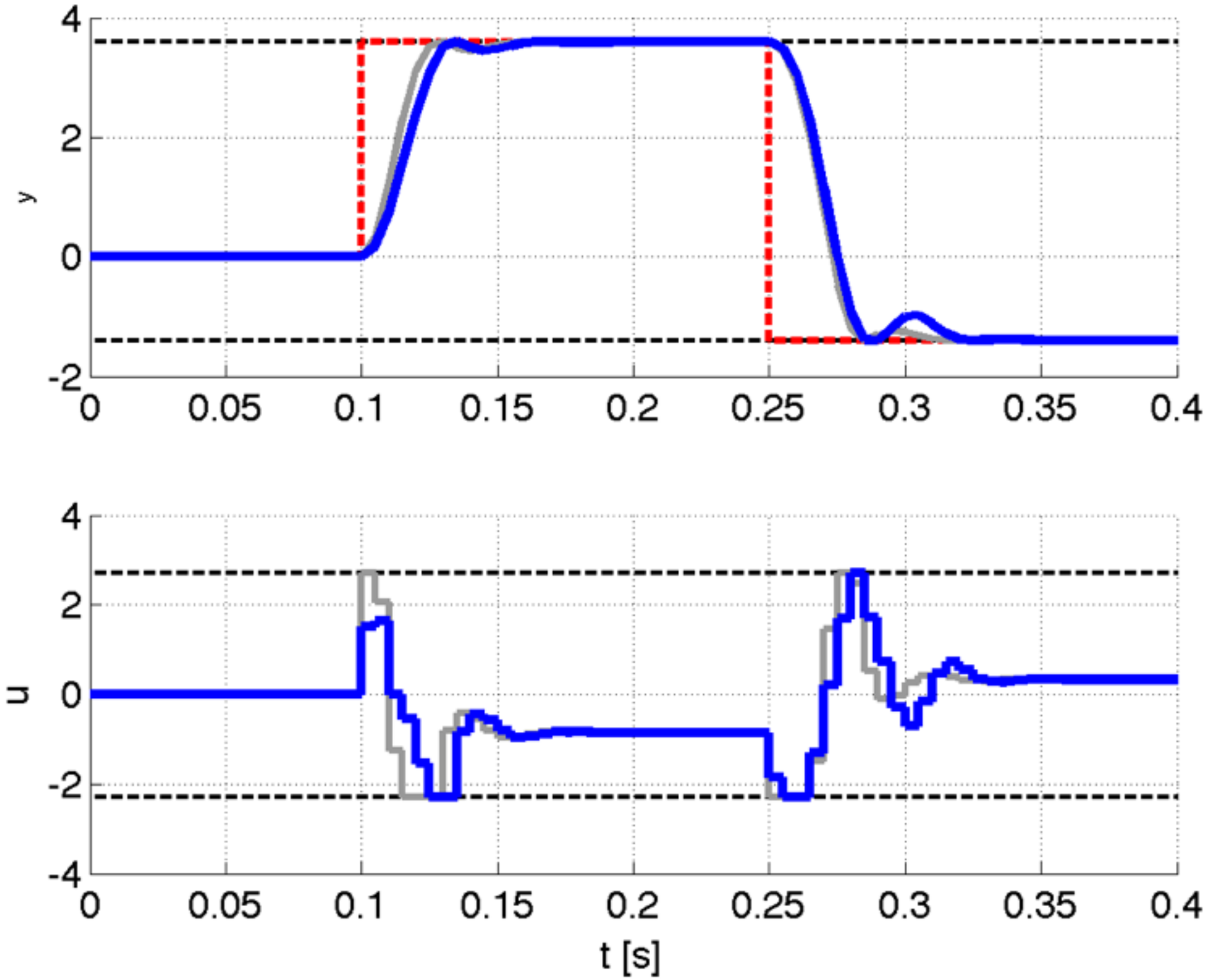
$$\Delta u_k = u_k - u_{k-1}$$

$$\Delta u_{\min} \leq \Delta u_k \leq \Delta u_{\max}$$

$$x_0 = x(t)$$

```
sys.u.with('deltaMin')
sys.u.deltaMin = -1
sys.u.with('deltaMax')
sys.u.deltaMax = 1.5
```

Slew Rate Constraints



MPC Prototyping

Regulation with disturbance rejection

Reference tracking

Soft constraints

Slew rate constraints

Custom setups

MPT is based on YALMIP


Berthelemy inv. et P.
anno 1802.

Johan Löfberg

YALMIP

Language for rapid prototyping of optimization problems

Matlab-based, textbook-like coding

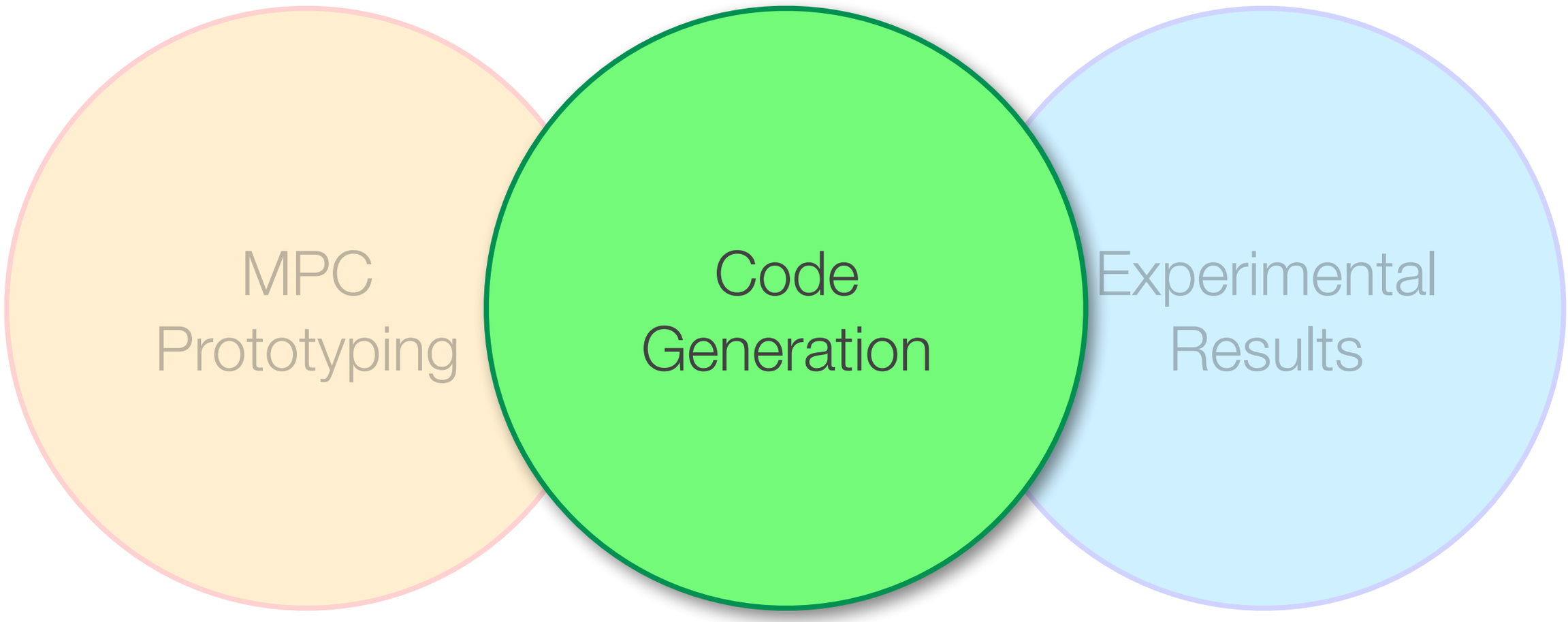
$$\min\ |x-y| + (y-3)^2$$
$$\text{s.t. } x^2 + y^2 \leq 1$$

$$\min\ z^T Q z + c^T z$$
$$\text{s.t. } z^T H_i z + A_i z \leq b_i$$

```
sdpvar x y
objective = abs(x-y) + (y-3)^2
constraints = [ x^2 + y^2 <= 1 ]
optimize(constraints, objective)
value([x y])
ans =

    0.3074    0.9516
```

Endless possibilities…

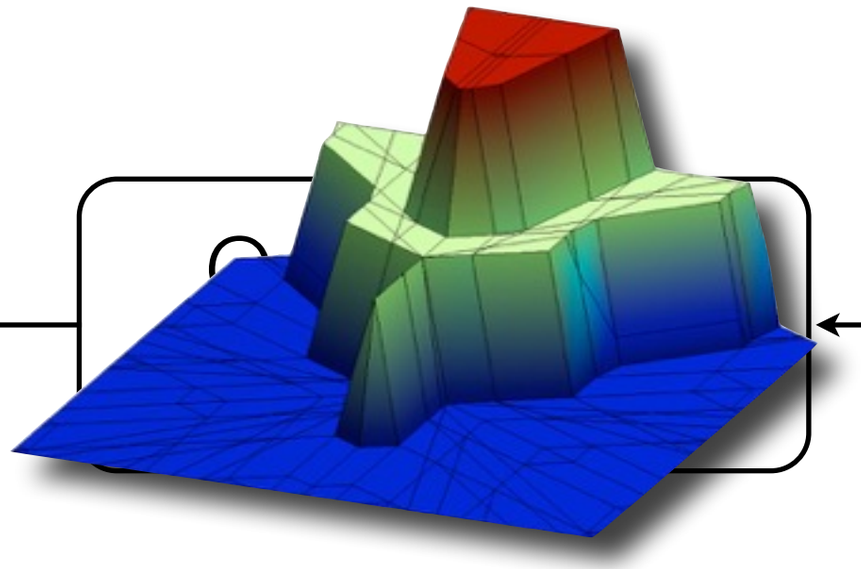
Agenda



Explicit MPC

Off-line

On-line

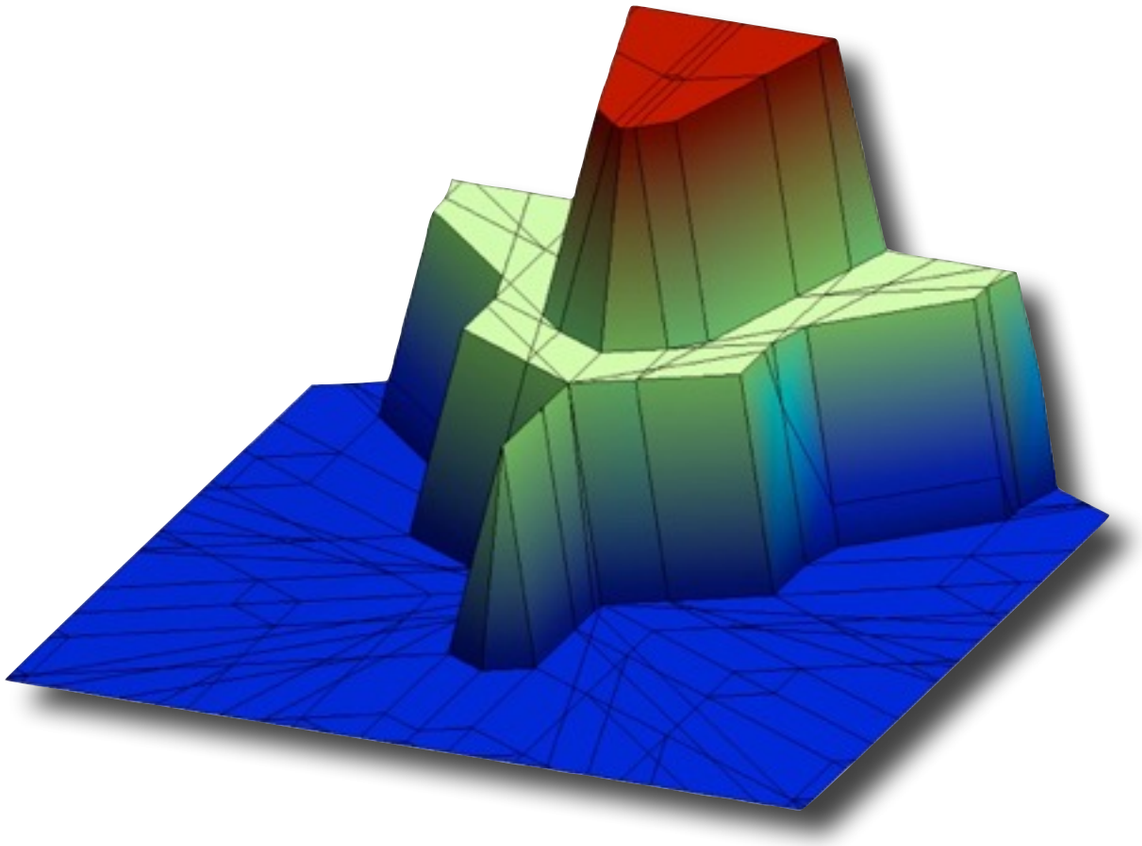


control action

plant state

Plant

Explicit MPC



Simple implementation

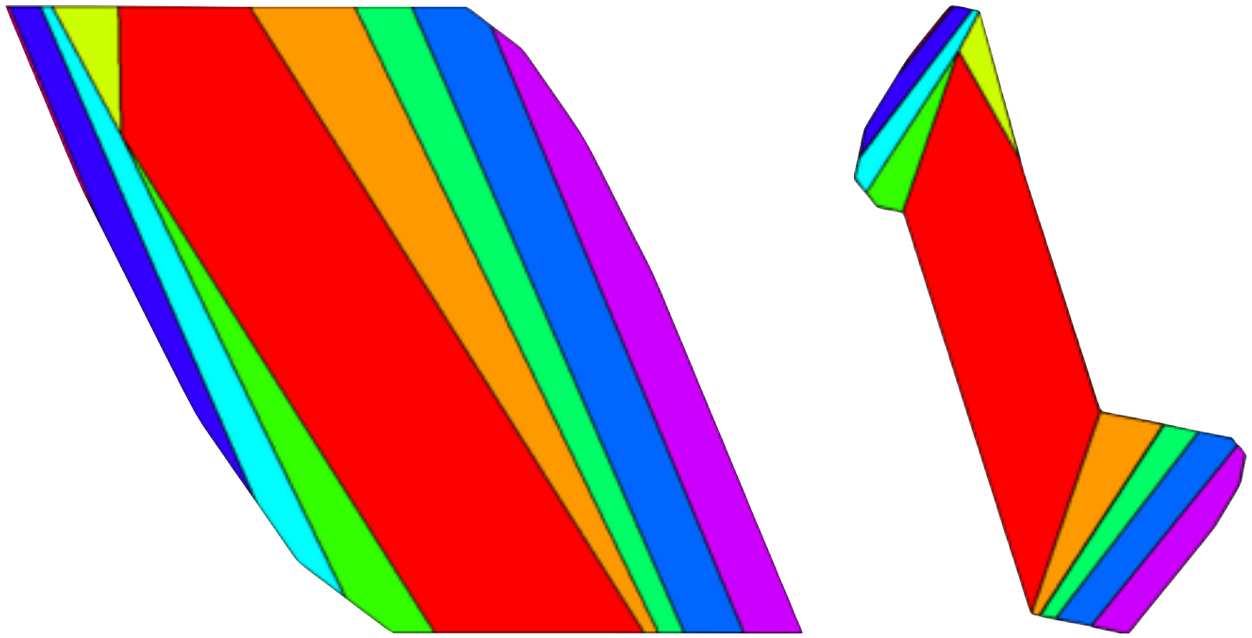
Verifiable performance

Predictable execution

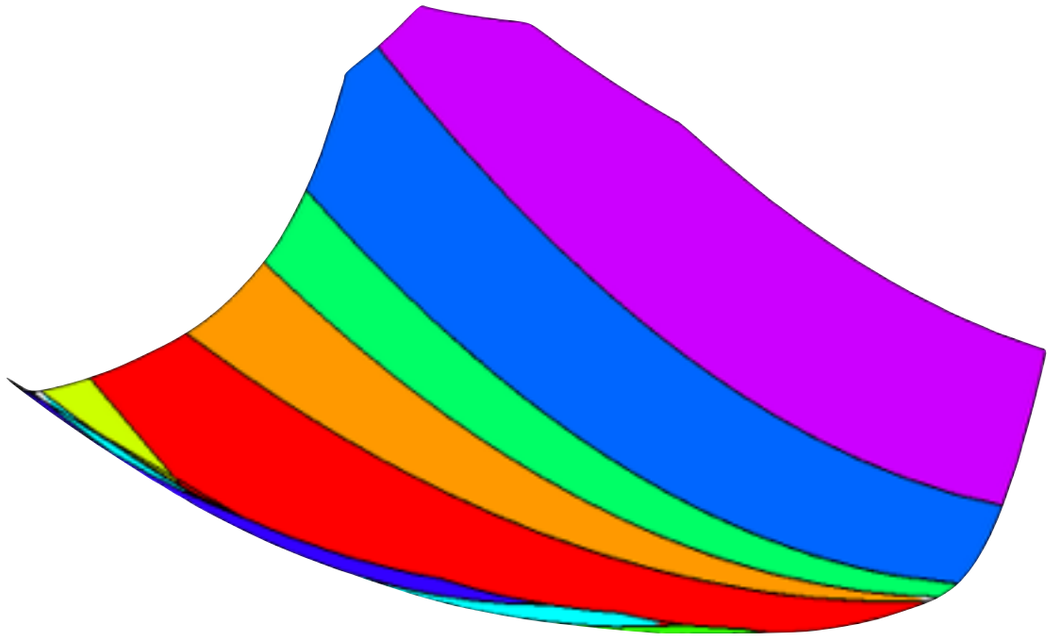
```
explicit = ctrl.toExplicit()
```

Command-Line Implementation

```
u0 = explicit.evaluate(x0)
data = ClosedLoop(explicit, sys).simulate(x0, Nsim)
```



`explicit.partition.plot()`



`explicit.cost.fplot()`

`explicit.feedback.fplot()`

Rigorous Analysis

Invariance computation

- is the MPC problem recursively feasible?
- if not, what is the largest set of initial conditions with such a property?

Stability analysis

- is the closed-loop system stable?
- what is a stabilizing terminal set/penalty?

Implementation analysis

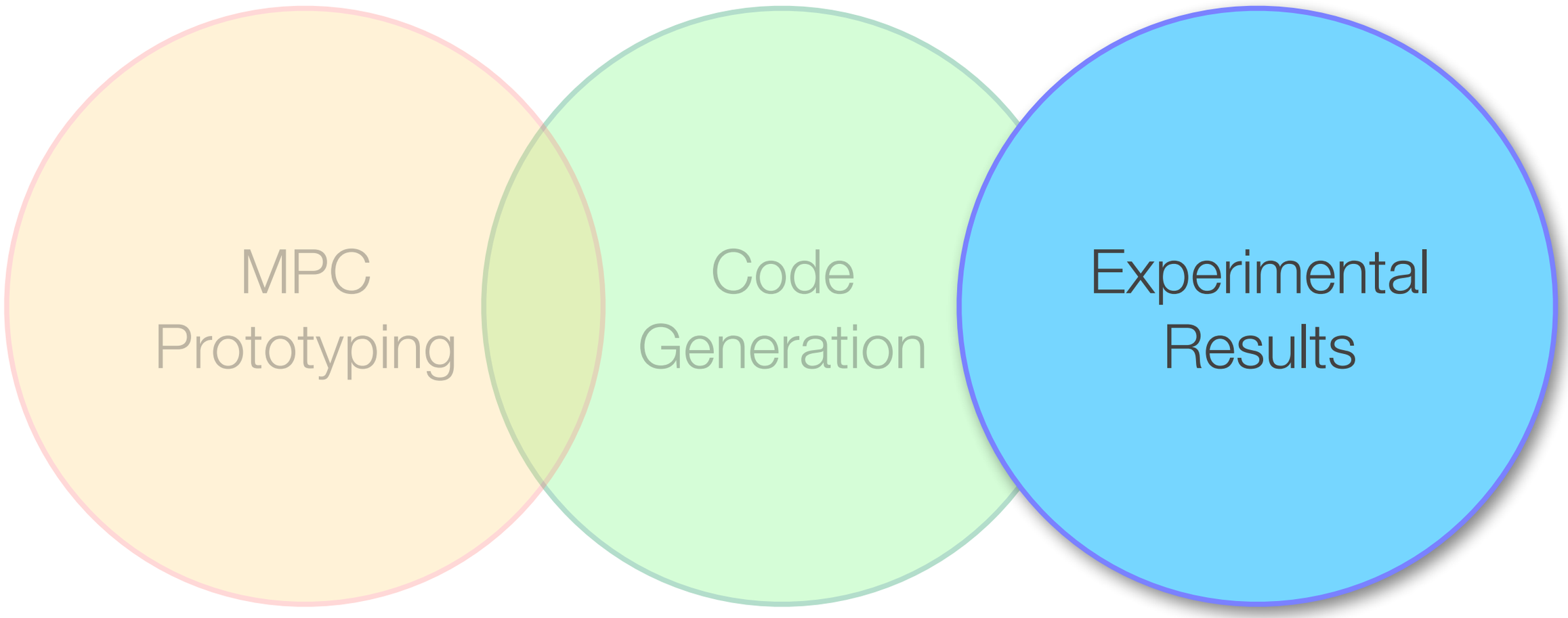
- how much memory and CPU power will I need?

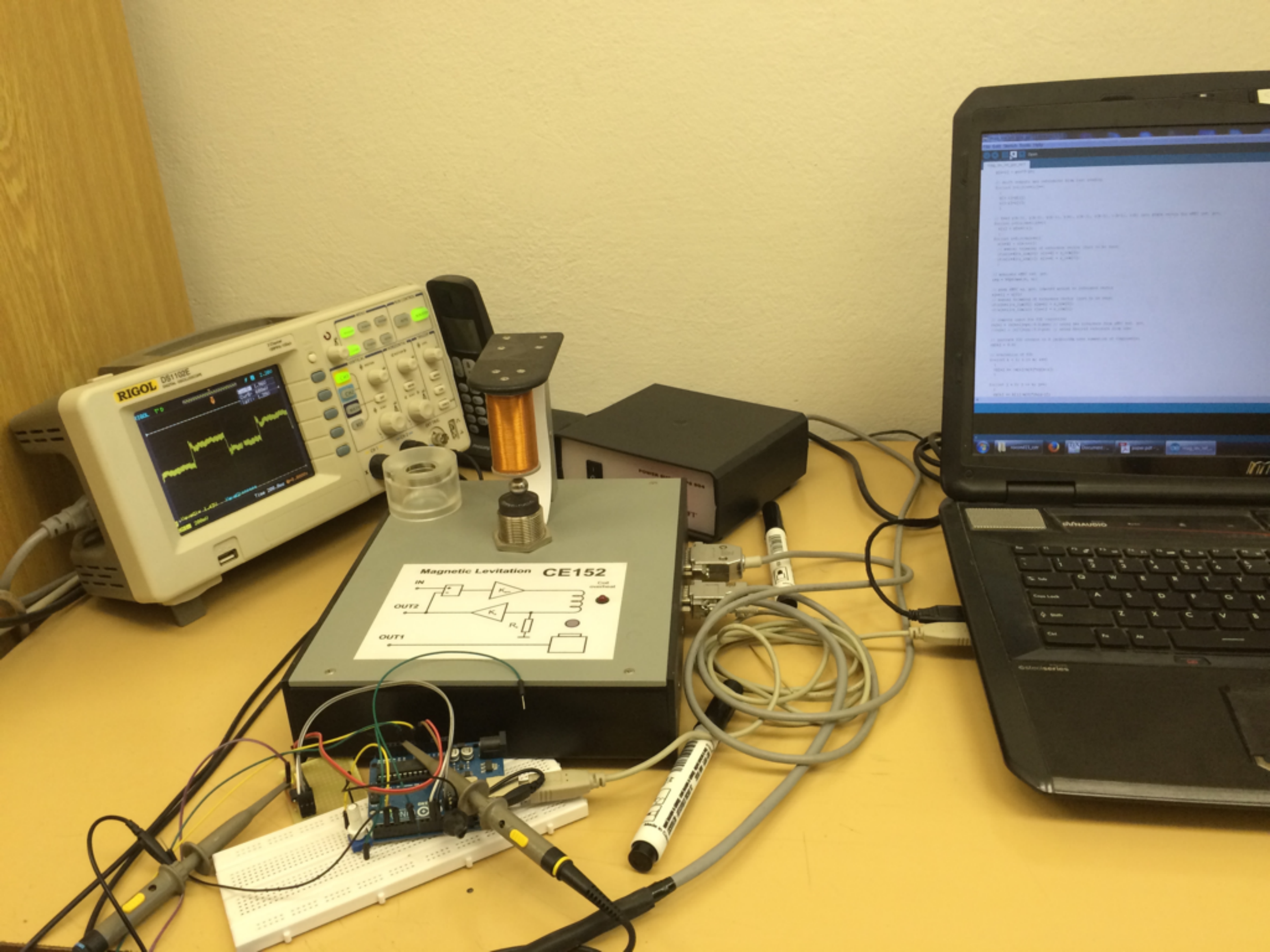
Code Generation

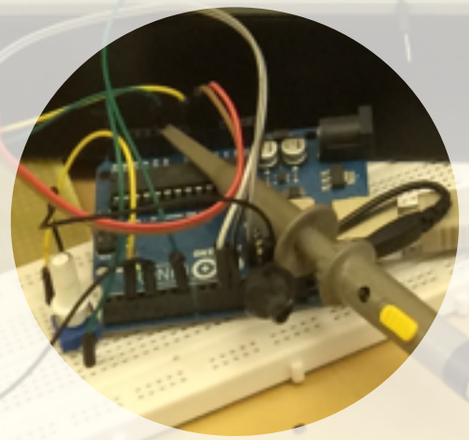
```
>> explicit.feedback.toC('my_controller')

Output written to "mycontroller.c".
C-mex function written to "mycontroller_mex.c".
```

```
static unsigned long mycontroller( double *X, double *U) {
    int ix, iu, ic, nc, isinside; unsigned long ireg, abspos, region; double hx;
    abspos = 0; region = 0;
    for (iu=0; iu<MPT_RANGE; iu++) { U[iu] = 0; }
    for (ireg=0; ireg<MPT_NR; ireg++) {
        isinside = 1; nc = MPT_NC[ireg];
        for (ic=0; ic<nc; ic++) {
            hx = 0; for (ix=0; ix<MPT_DOMAIN; ix++) {
           hx = hx + MPT_A[abspos*MPT_DOMAIN+ic*MPT_DOMAIN+ix]*X[ix]; }
            if ((hx - MPT_B[abspos+ic]) > MPT_ABSTOL) { isinside = 0; break; }
        }
        if (isinside==1) {
            region = ireg + 1; for (iu=0; iu<MPT_RANGE; iu++) {
                for (ix=0; ix<MPT_DOMAIN; ix++) {
             U[iu]=U[iu]+MPT_F[ireg*MPT_DOMAIN*MPT_RANGE+iu*MPT_DOMAIN+ix]*X[ix];
            }
                U[iu] = U[iu] + MPT_G[ireg*MPT_RANGE + iu];
            }
            return region;
        } abspos = abspos + MPT_NC[ireg];
    }
    return region;
}
```

Agenda




RIGOL
DS1102E
Magnetic Levitation
CE152
IN
OUT2
OUT1



Embedded Hardware

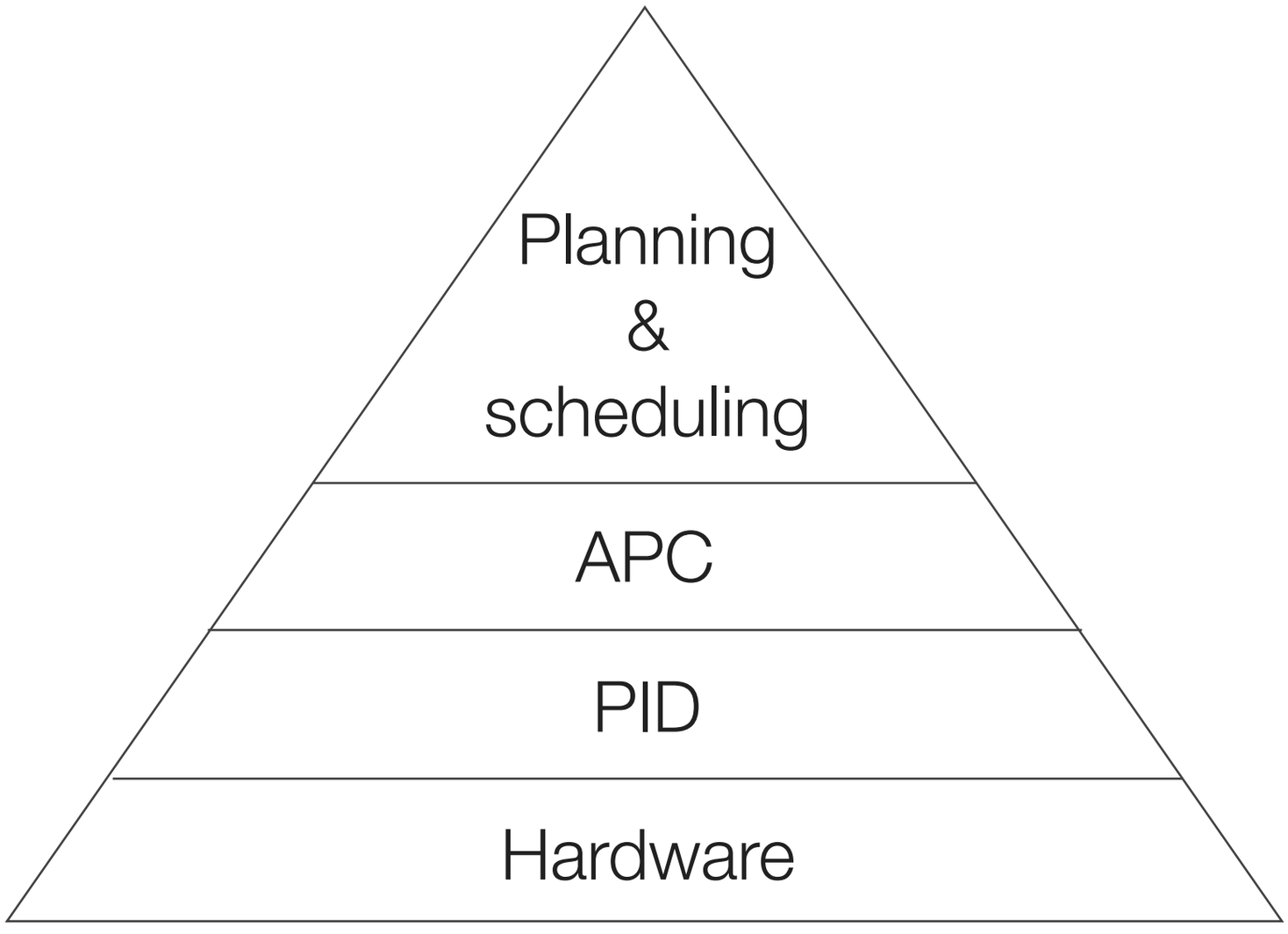
Atmel ATMega328p

8-bit, 16 MHz CPU

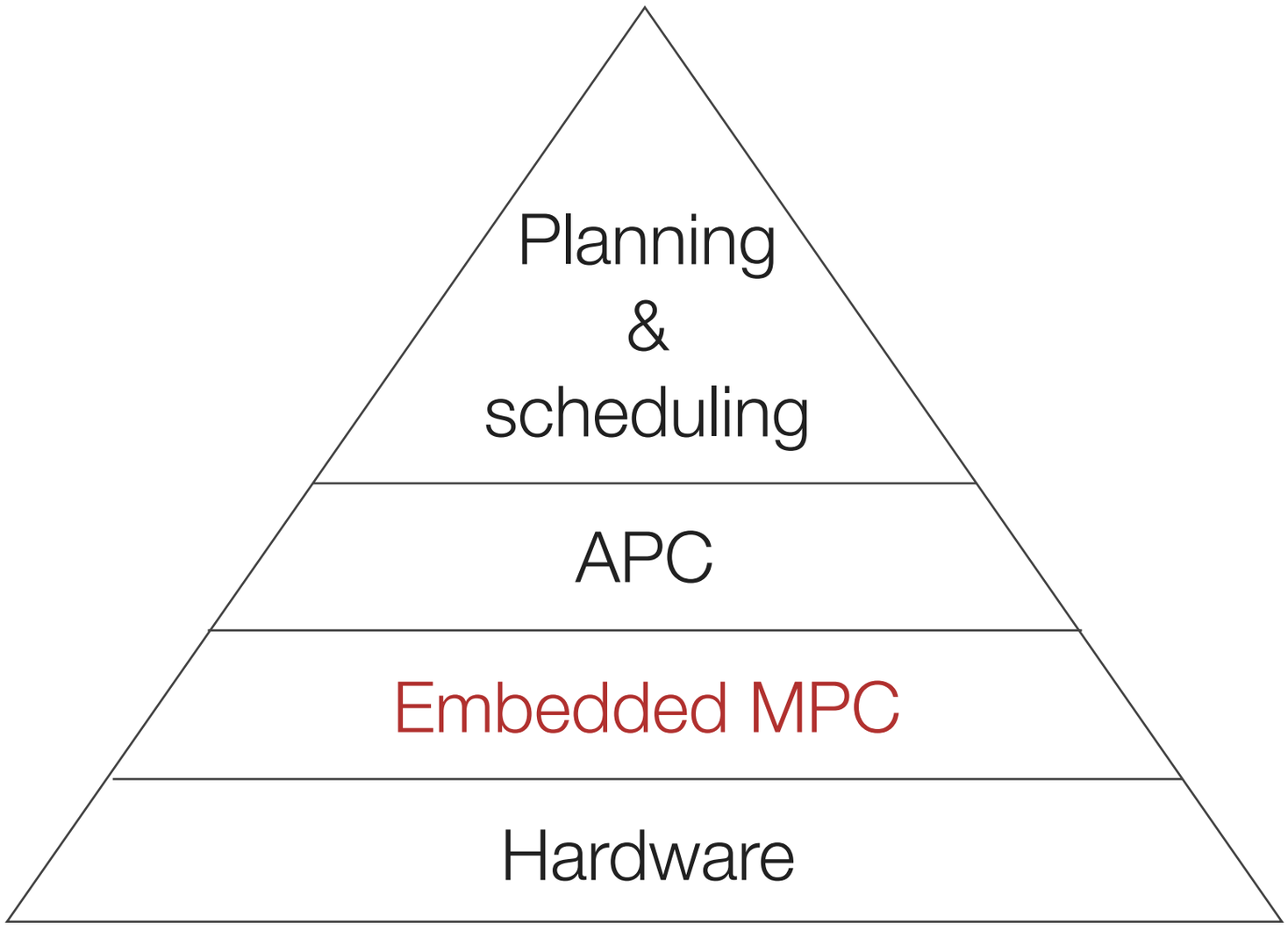
60 kFLOPs/sec

2 kB of SRAM for data

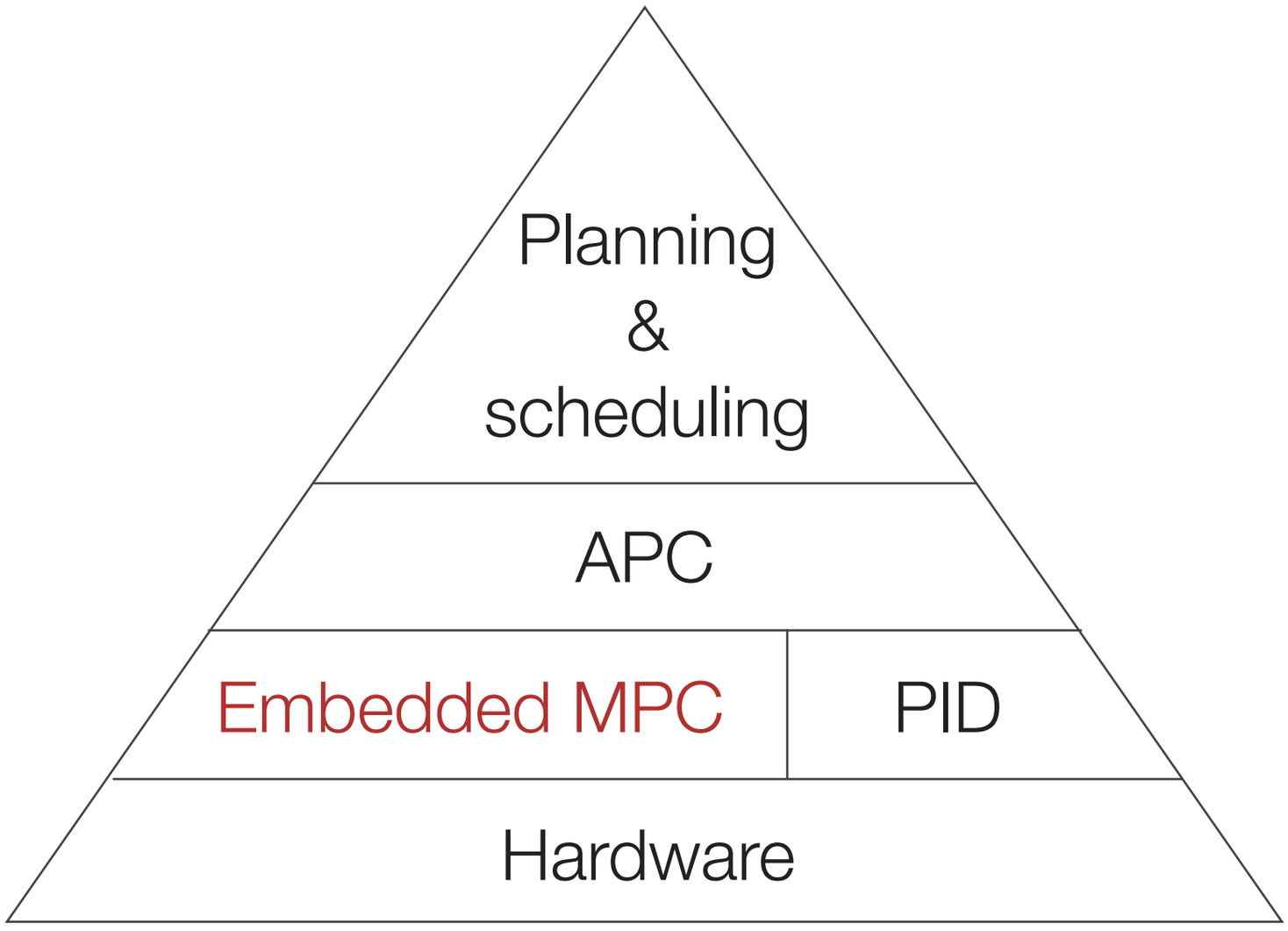
MPC in Process Industries



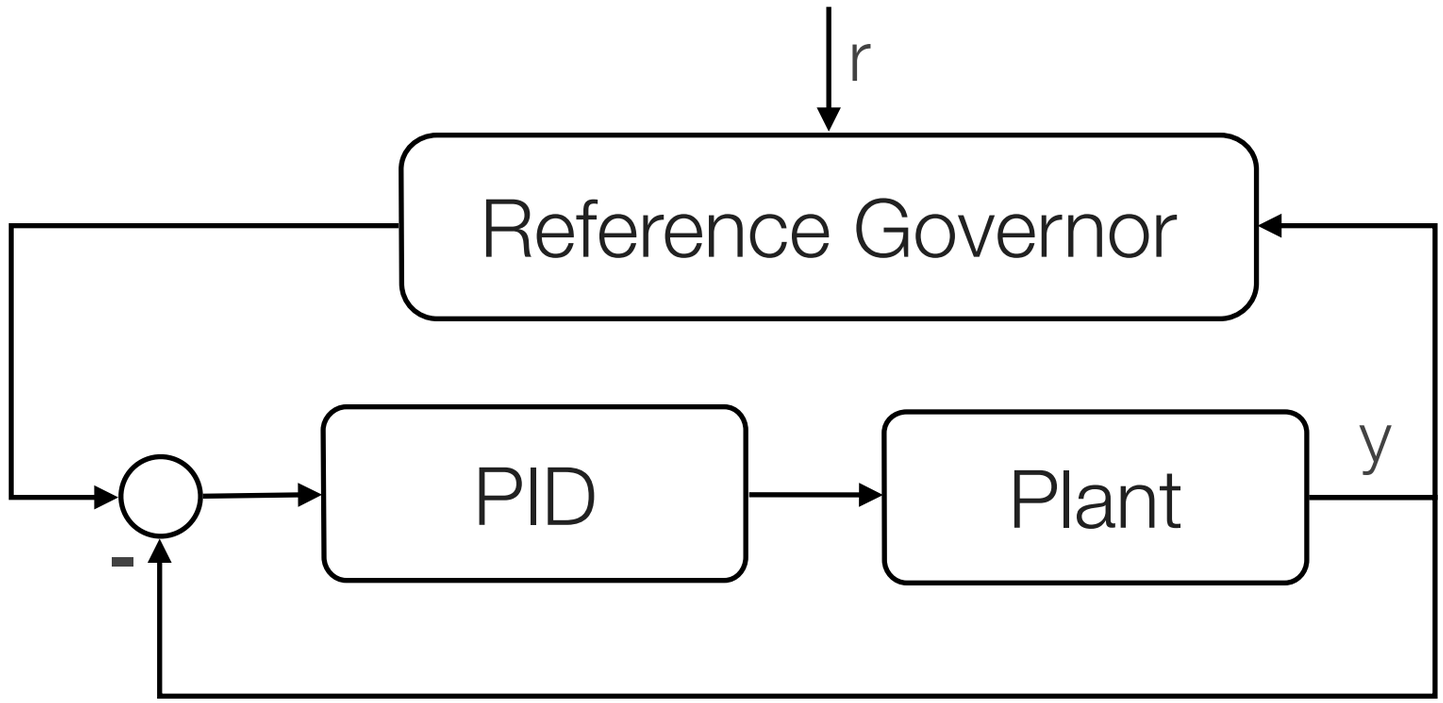
MPC in Process Industries



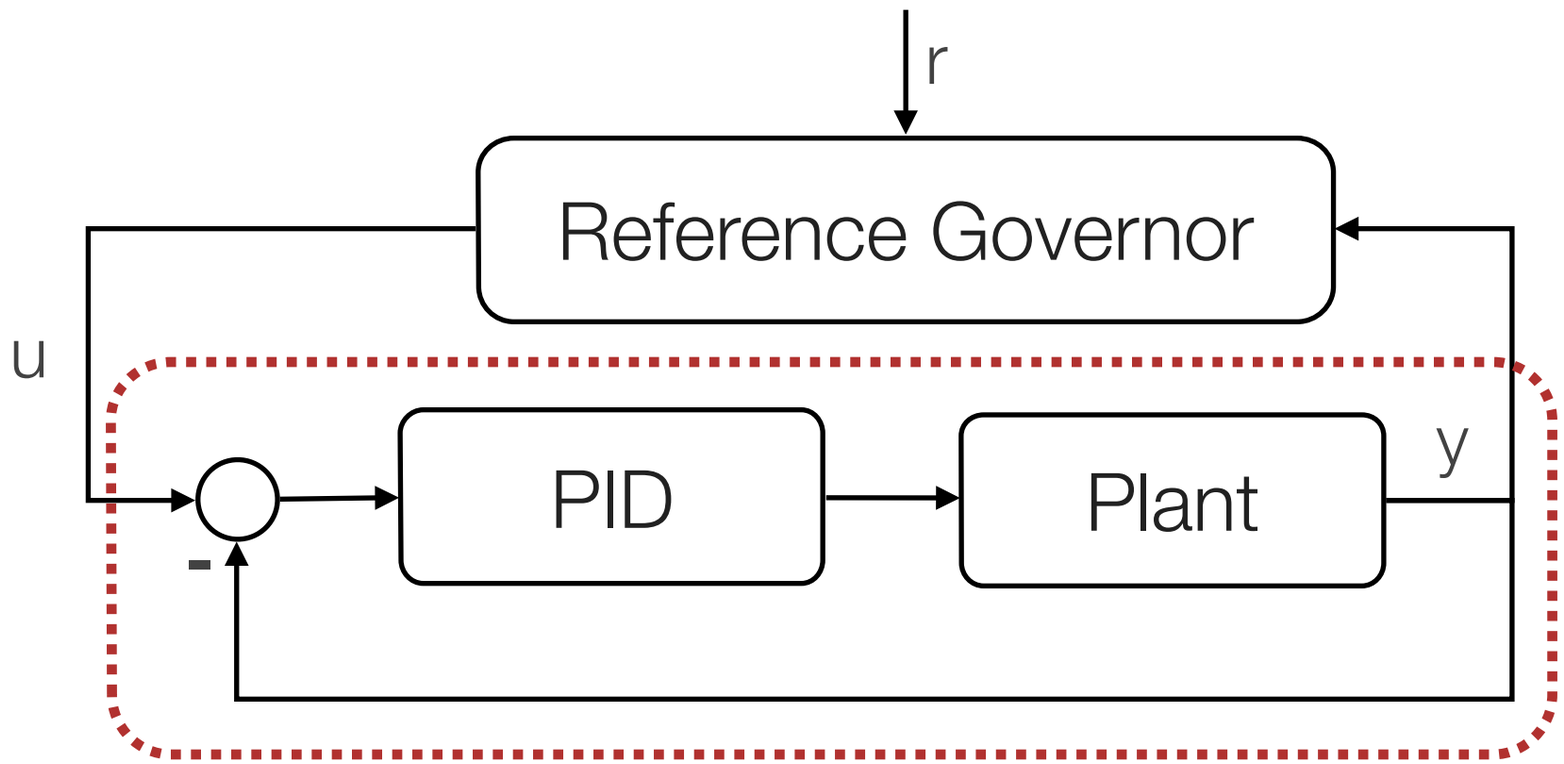
MPC in Process Industries



Reference Governor



Reference Governor



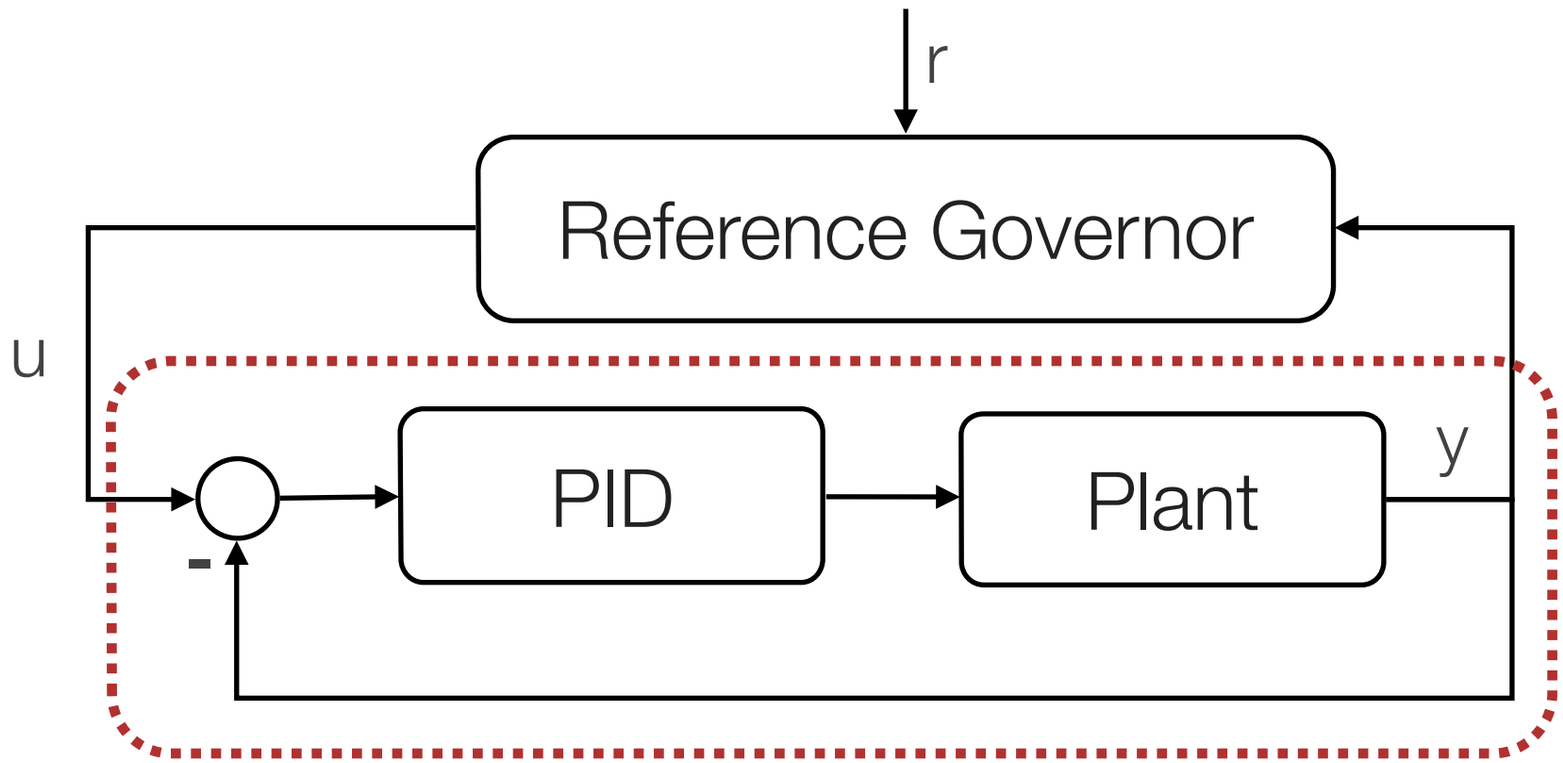
$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} Q_{\mathrm{d}} \Delta u_k^2 + Q_{\mathrm{u}}(u_k - r)^2 + Q_{\mathrm{y}}(y_k - u_k)^2$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k$$

$$y_k = Cx_k + Du_k$$

$$y_{\min} \leq y_k \leq y_{\max}$$

Reference Governor



MPC setup:

- sampling time 2ms
- prediction horizon 3
- control horizon 1

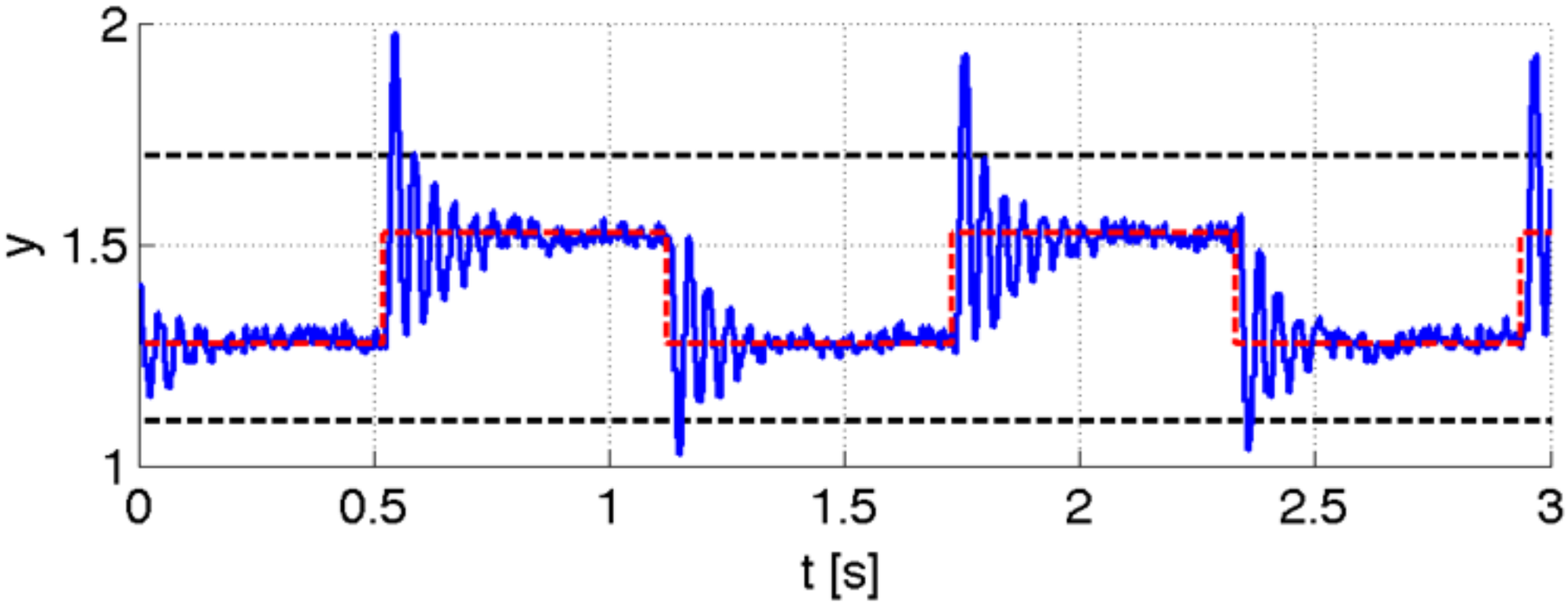
Explicit solution:

- 5 regions in 8D
- binary search tree
- 10 lines of C code

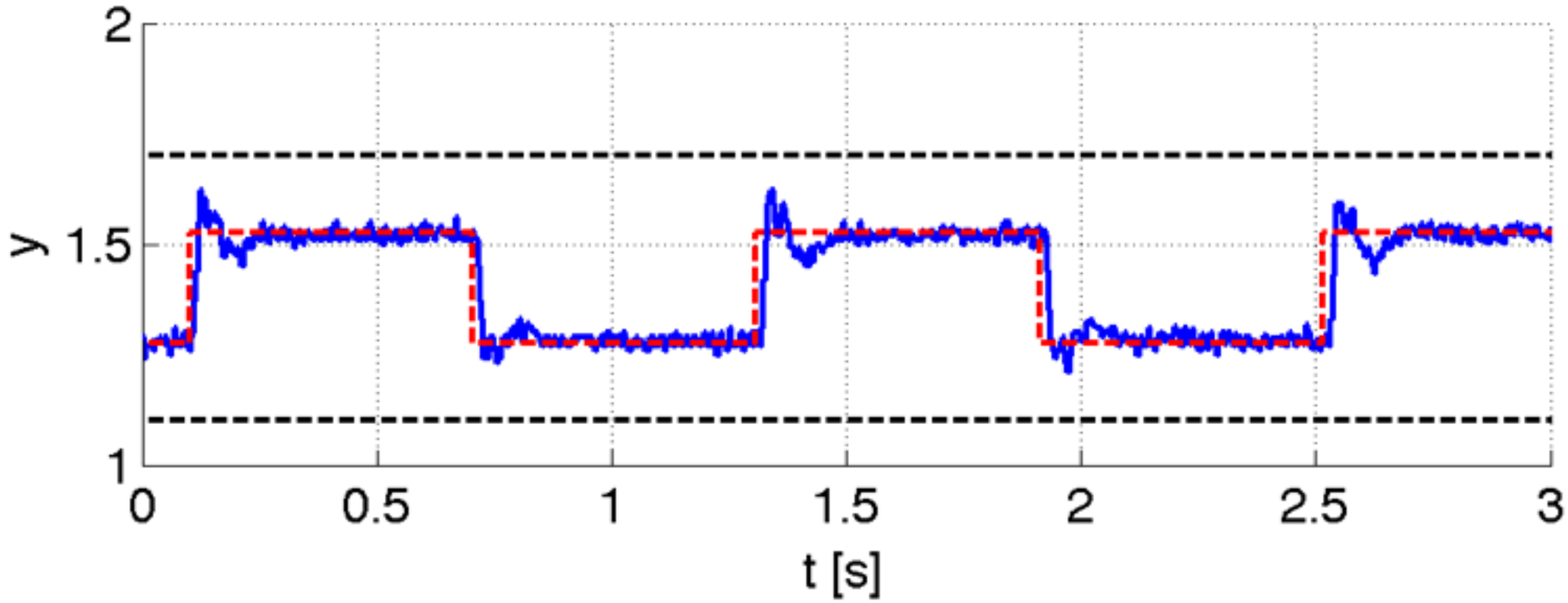
Implementation:

- 621 bytes
- 0.64 ms execution
- 0.50 ms for PID

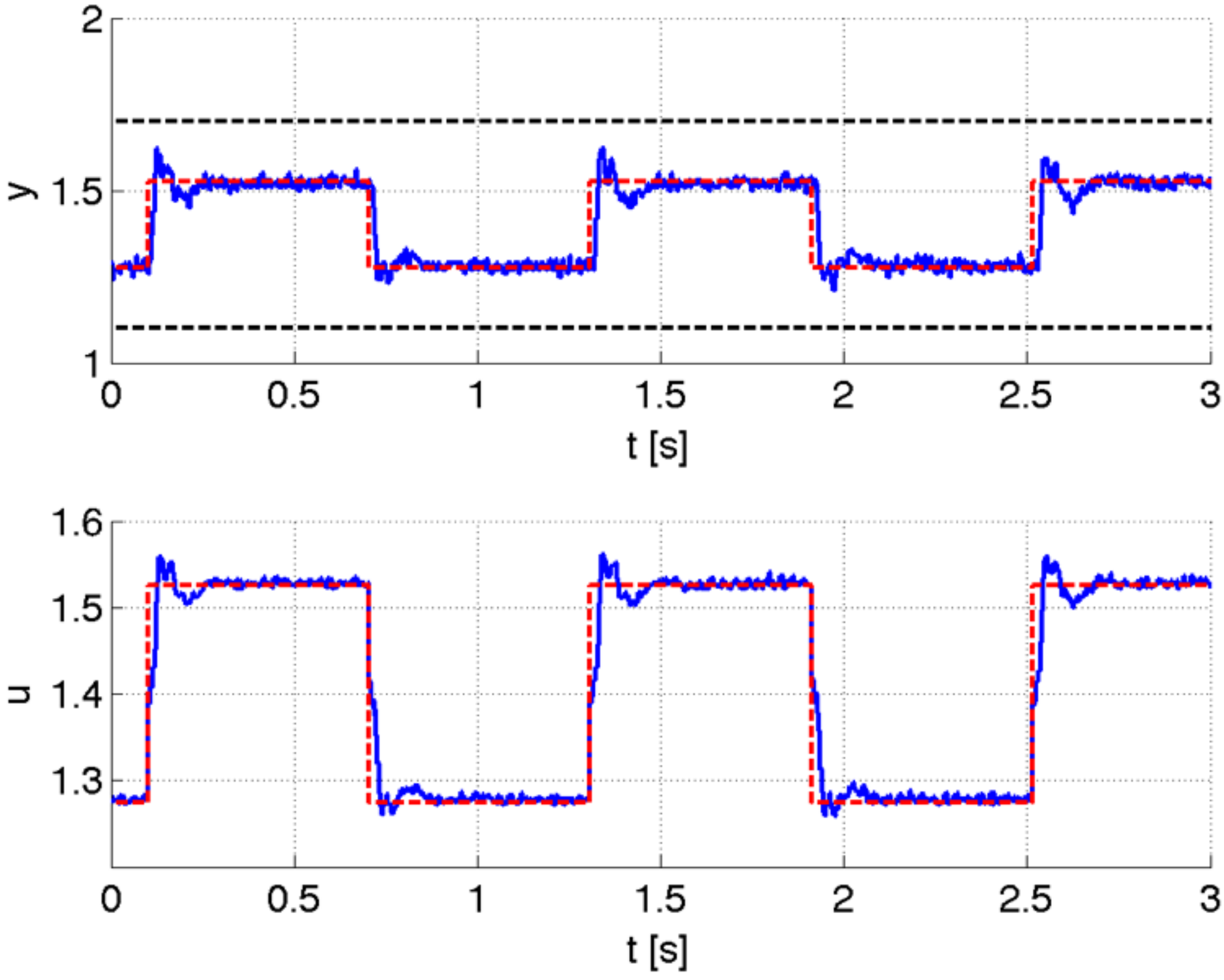
Experimental Results (PID)



Experimental Results (Reference Governor)

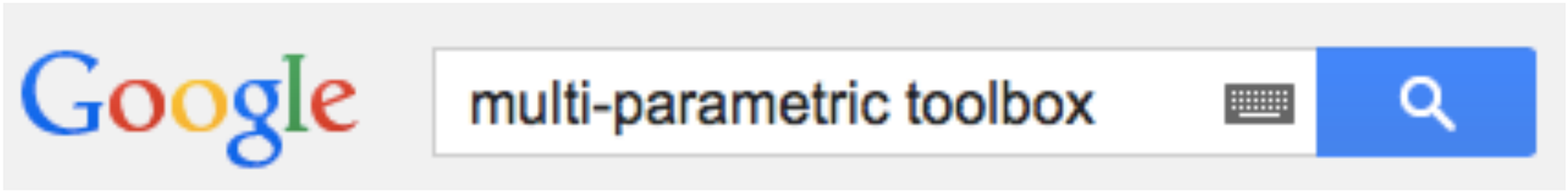


Experimental Results (Reference Governor)



Prototype & Deploy Optimization-Bazed Controllers


Whosoever holds
this hammer, if
he be worthy,
shall possess the
power of...
THOR


Google
multi-parametric toolbox