qpOASES - Online Active Set Strategy for Fast Linear MPC

Moritz Diehl, Hans Joachim Ferreau, Lieboud Vanden Broeck, Jan Swevers
Dept. ESAT and Center of Excellence for Optimization in Engineering OPTEC
K.U. Leuven, Belgium

joint work with Hans Georg Bock,
Interdisciplinary Center for Scientific Computing, Univ. Heidelberg, Germany

Linear MPC (or QP subproblems in NMPC)

For

- linear dynamic system $\qquad x_{k+1} = Ax_k + Bu_k\,,$
- linear constraints
- quadratic cost

only *quadratic program (QP)* needs to be solved:

$$
\begin{aligned}
\min_{\substack{u_0,\ldots,u_{N-1}\\ x_1,\ldots,x_N}} \quad & x_N^T P x_N + \sum_{i=0}^{N-1} \left(x_i^T Q x_i + u_i^T R u_i\right) \\
\text{s.t.} \quad & x_{k+1} = Ax_k + Bu_k\,, \\
& (x_0 \text{ given})\,, \\
& \underline{c} \le Cx_k \le \overline{c}\,, \\
& \underline{d} \le Du_k \le \overline{d}\,, \\
& c_T \le C_T x_N\,,
\end{aligned}
$$

Linear MPC = parametric QP

Eliminate states via "condensing", obtain smaller scale *quadratic program (QP)* in variables $w := (u_0^T, \ldots, u_{N-1})^T$

$$\mathrm{QP}(x_0): \quad \min_{w} \; \frac{1}{2} w^T H w + w^T \underbrace{F^T x_0}_{=: g(x_0)}$$
$$\text{s.t.} \quad Gw \geq \underbrace{\bar{b} + E x_0}_{=: b(x_0)},$$

(assumption: H positive definite)

QP depends on x_0 via *affine functions* $g(x_0)$ and $b(x_0)$

Karush-Kuhn-Tucker (KKT) Conditions

Theorem

Let $\mathrm{QP}(x_0)$ be a strictly convex and feasible quadratic program. Then there exists a unique $w^* \in \mathbb{R}^n$ and at least one working set $\mathbb{A}$ and a vector $y^* \in \mathbb{R}^m$ which satisfy the following conditions:

$$
\begin{aligned}
Hw^* - G_{\mathbb{A}}^T y_{\mathbb{A}}^* &= -g(x_0), \\
G_{\mathbb{A}} w^* &= b_{\mathbb{A}}(x_0), \\
y_{\mathbb{I}}^* &= 0, \;\; (\mathbb{I} := \{1, \ldots, m\} \setminus \mathbb{A}), \\
G_{\mathbb{I}} w^* &\geq b_{\mathbb{I}}(x_0), \\
y_{\mathbb{A}}^* &\geq 0.
\end{aligned}
$$

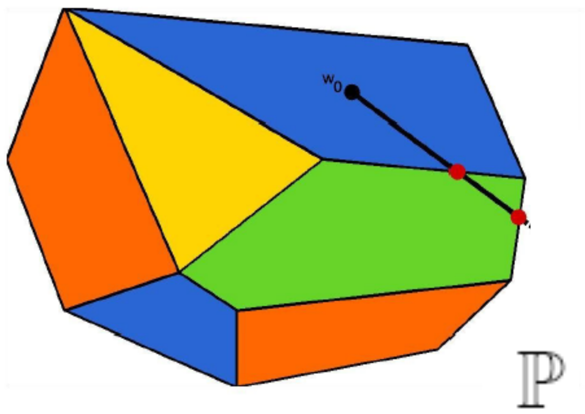
Parametric QP Solution Structure

Define set of “feasible parameters“:

$$\mathbb{P} := \{x_0 \in \mathbb{R}^{n_x} \mid \mathrm{QP}(x_0) \text{ is } \textit{feasible} \}$$

Well known:

THEOREM: Set $\mathbb{P}$ is convex, and can be partitioned into polyhedral „critical regions“ each corresponding to a different working set $\mathbb{A}$.
QP solution on each region is affine in x_0



Sketch of Proof for Polyhedral Critical Regions

Check KKT conditions for fixed working set $\mathbb{A}$:

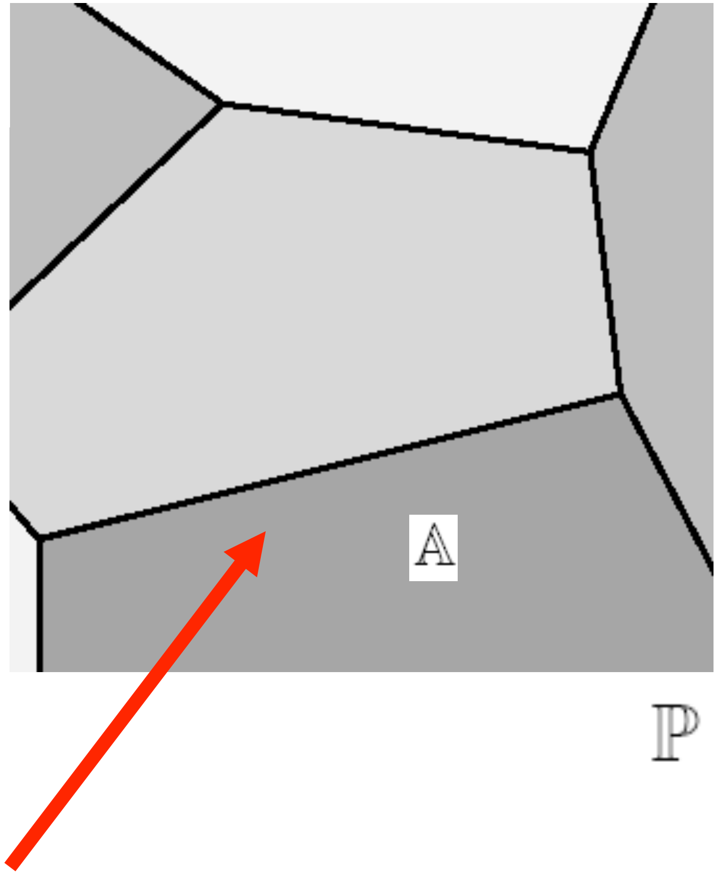
1. $g(x_0)$, $b(x_0)$ affine: then w^* , y^* affine, because solution of linear system:

$$
\begin{aligned}
Hw^* - G_{\mathbb{A}}^T y_{\mathbb{A}}^* &= -g(x_0), \\
G_{\mathbb{A}} w^* &= b_{\mathbb{A}}(x_0), \\
y_{\mathbb{I}}^* &= 0,
\end{aligned}
$$

2. w^*, y^* affine, therefore

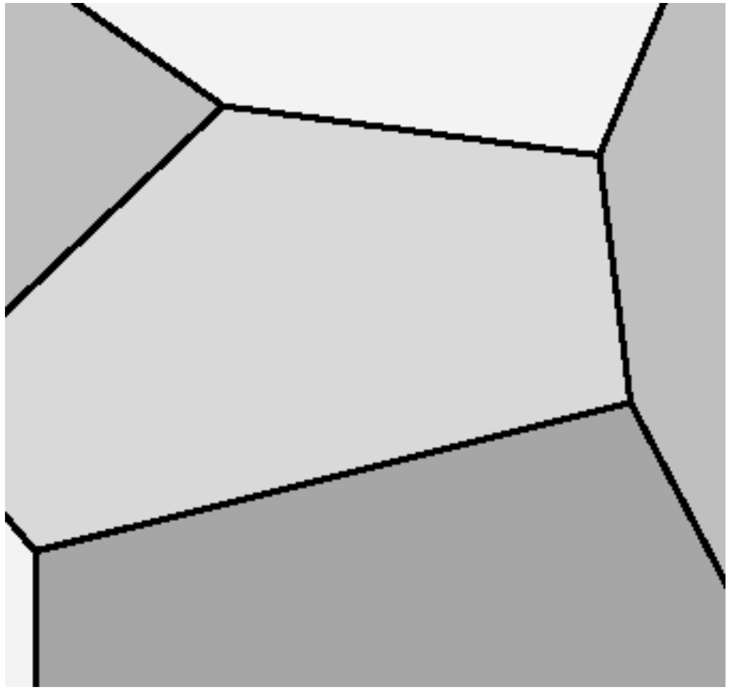
$$
\begin{aligned}
G_{\mathbb{I}} w^* &\geq b_{\mathbb{I}}(x_0), \\
y_{\mathbb{A}}^* &\geq 0.
\end{aligned}
$$

are **linear constraints** on x_0 that define polyhedral „critical region“ in $\mathbb{P}$



Explicit MPC: Precalculate Everything

Idea: Compute control on all critical regions in advance (Bemporad, Borrelli, Morari, 2002).



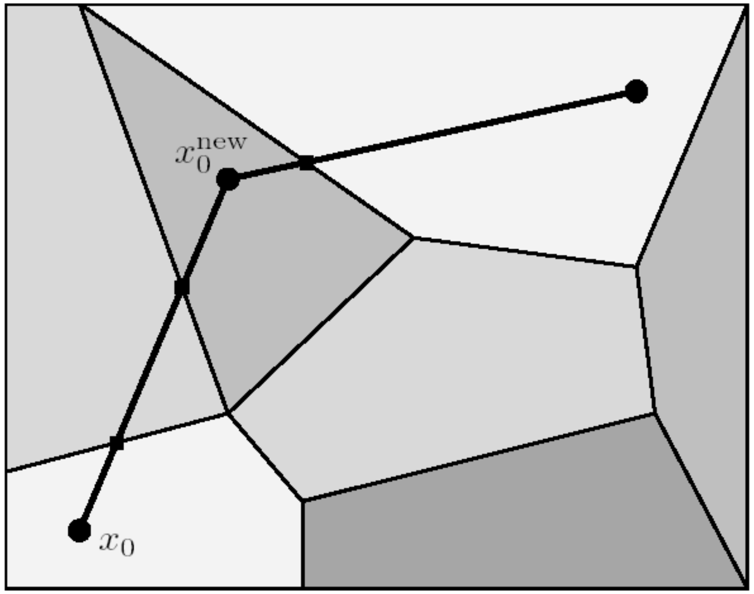
Pro: MPC in microsecs. possible
Contra: problem size limited

Example: 50 variables, lower and upper bounds:
$3^{50} = 10^{23}$ possible critical regions.
Prohibitive.

Online Active Set Strategy (qpOASES)

Combine Explicit and Online MPC

- compute affine solution only on current critical region
- go on straight line from old to new problem data ($\mathbb{P}$ convex !)
- **solve each QP on path exactly (keep primal-dual feasibility)!**
- need to change working set only at boundaries of critical regions

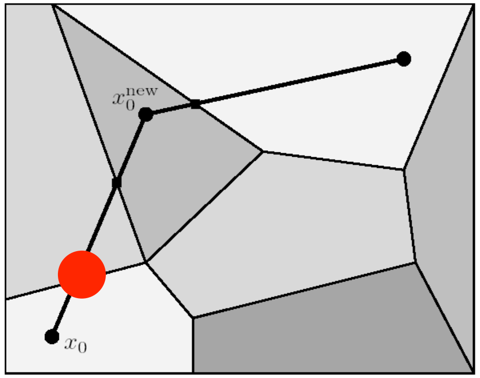


$\mathbb{P}$

How to compute each step?

- determine change in $g(x_0)$ and $b(x_0)$, solve KKT system

$$\begin{pmatrix} H & G_{\mathbb{A}}^T \\ G_{\mathbb{A}} & 0 \end{pmatrix} \begin{pmatrix} \Delta w^* \\ -\Delta y_{\mathbb{A}}^* \end{pmatrix} = \begin{pmatrix} -\Delta g \\ \Delta b_{\mathbb{A}} \end{pmatrix}$$



- choose steplength $\tau_{\max}$ maximal such that

$$G_i^T (w^* + \tau \Delta w^*) \geq b_i(x_0) + \tau \Delta b_i$$

still holds for all inactive (primal) constraints, and

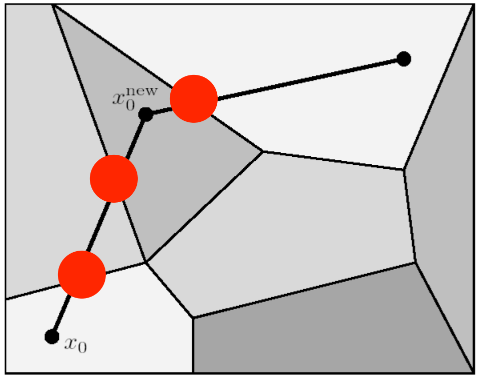
$$y_i^* + \tau \Delta y_i \geq 0$$

for all active dual variables: Set $\tau_{\max} := \min \{1, \tau_{\max}^{\text{prim}}, \tau_{\max}^{\text{dual}}\}$

with $\tau_{\max}^{\text{prim}} := \min\limits_{\substack{i \in \mathbb{I} \\ G_i^T \Delta w^* < \Delta b_i}} \dfrac{b_i(x_0) - G_i^T w^*}{G_i^T \Delta w^* - \Delta b_i}$ and $\tau_{\max}^{\text{dual}} := \min\limits_{\substack{i \in \mathbb{A} \\ \Delta y_i < 0}} -\dfrac{y_i^*}{\Delta y_i}$

How to change working set?

- add or remove constraints to/from working set when crossing borders of critical regions



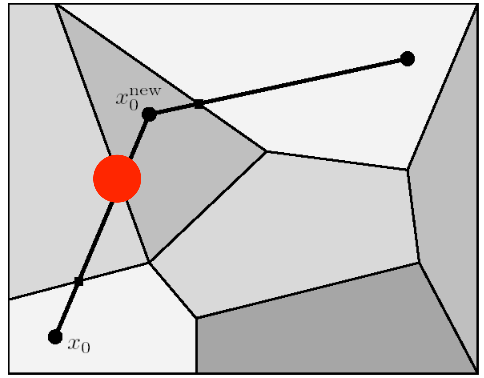
- use null space approach, keep QR-factorization of active constraint matrix, and Cholesky factorization of projected hessian:

$$G_{\mathbb{A}} = \begin{pmatrix} 0 & T \end{pmatrix} \begin{pmatrix} Z^T \\ Y^T \end{pmatrix},$$

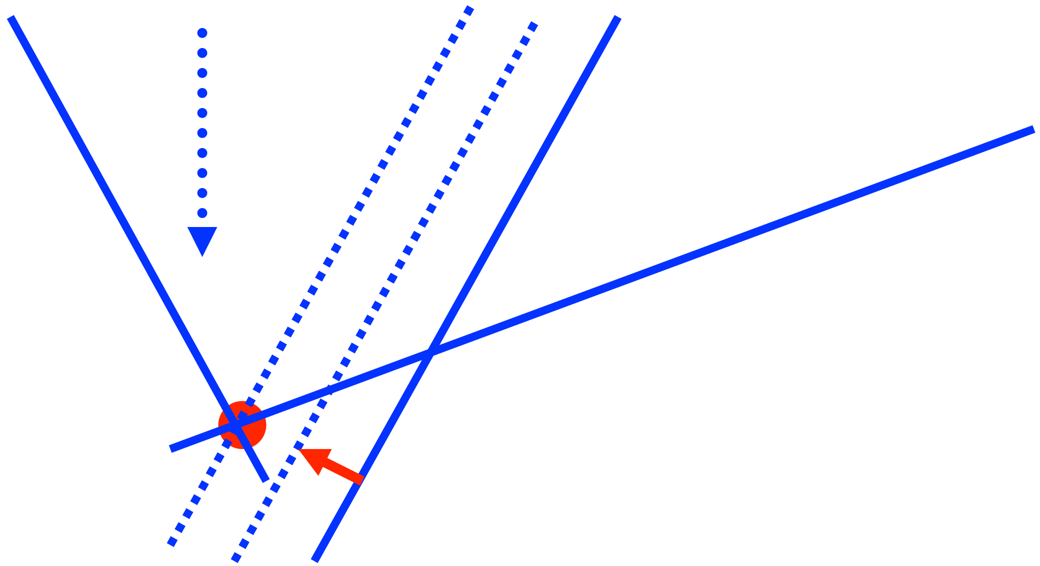
$$Z^T H Z = R^T R,$$

- each working set change costs only O(n^2) flops, exactly as one QP iteration in efficient QP solvers!

Extra difficulty: linear independence often violated



- during homotopy, often redundant constraints become active and cause degeneracy
- example of three active constraints in 2-D:



How to deal with degeneracy?

Without linear independence, KKT system becomes unsolvable! Addition of degenerate constraints must be avoided.

Remedy [Best 1996]: before adding extra row G_j to $G_{\mathbb{A}}$ solve auxiliary system

$$\begin{pmatrix} H & G_{\mathbb{A}}^T \\ G_{\mathbb{A}} & 0 \end{pmatrix} \begin{pmatrix} s \\ \xi \end{pmatrix} = \begin{pmatrix} G_j \\ 0 \end{pmatrix}$$

if $s = 0$, linear dependence is detected, and ξ helps to find a constraint from $\mathbb{A}$ that can be removed

Summary of qpOASES Algorithm

(1) Calculate Δx_0, Δg and Δb

(2) Calculate primal and dual step directions Δw^* and Δy^*

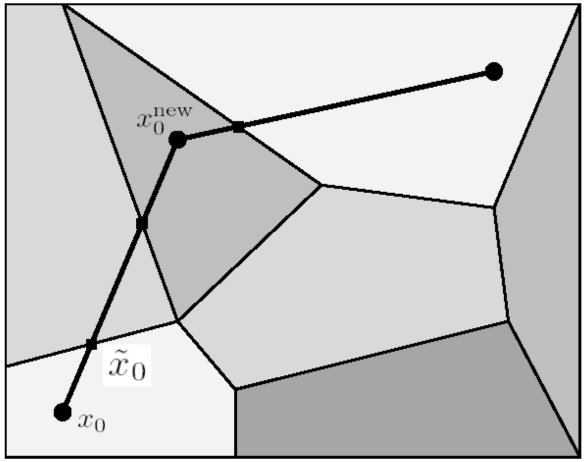
(3) Determine maximum homotopy step length $\tau_{\max} := \min\left\{1, \tau_{\max}^{\text{prim}}, \tau_{\max}^{\text{dual}}\right\}$

(4) Obtain optimal solution of $\text{QP}(\tilde{x}_0)$:

(a) $\tilde{x}_0 \leftarrow x_0 + \tau_{\max}\Delta x_0$,

(b) $\tilde{w}^* \leftarrow w^* + \tau_{\max}\Delta w^*$,

(c) $\tilde{y}^* \leftarrow y^* + \tau_{\max}\Delta y^*$.



(5) `if` $\tau_{\max} = 1$:

Optimal solution of $\text{QP}(x_0^{\text{new}})$ found.

(6) `if` $\tau_{\max} = \tau_{\max}^{\text{dual}}$:

Remove a dual blocking constraint j $\left(\tau_{\max}^{\text{dual}} = -\frac{y_j^*}{\Delta y_j}\right)$ from working set,

`elseif` $\tau_{\max} = \tau_{\max}^{\text{prim}}$:

Add a primal blocking constraint j $\left(\tau_{\max}^{\text{prim}} = \frac{b_j(x_0) - G_j^T w^*}{G_j^T \Delta w^* - \Delta b_j}\right)$

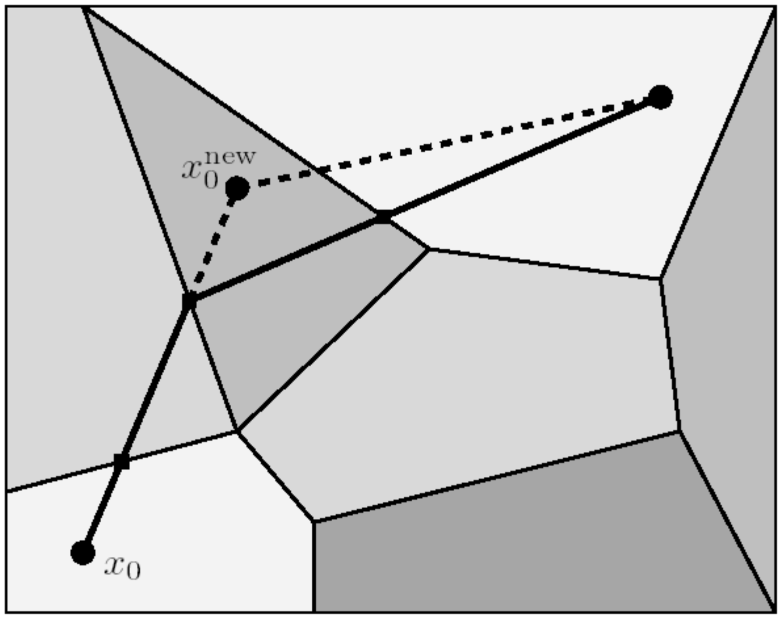
ensuring linear independence

(7) Set $x_0 \leftarrow \tilde{x}_0$, $w^* \leftarrow \tilde{w}^*$, $y^* \leftarrow \tilde{y}^*$ and continue with step (1).

Real-Time Variant of qpOASES Algorithm

Limit number of active set changes per sampling time:

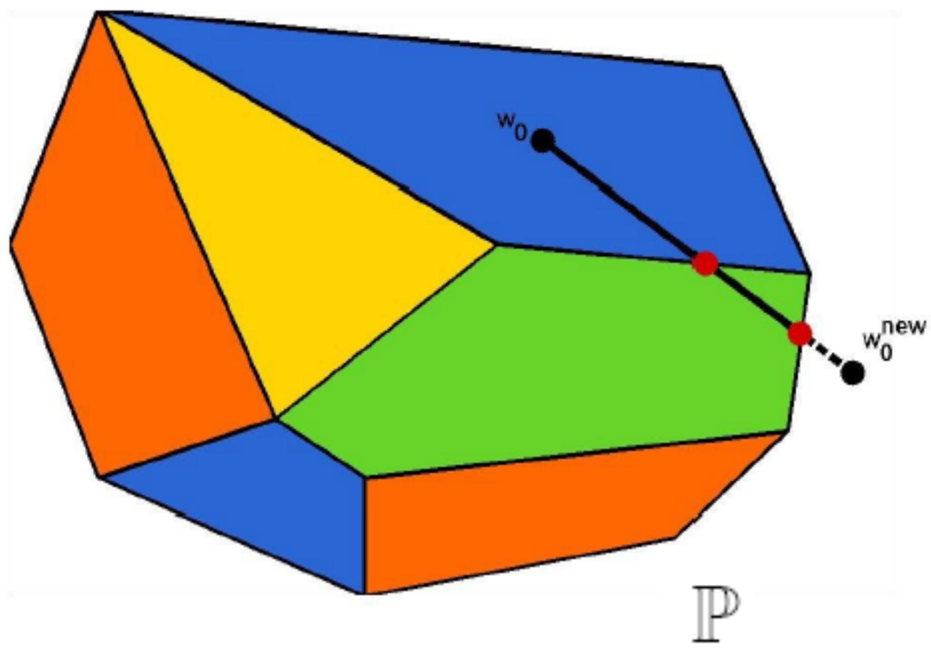
- lag behind, if too many changes necessary
- deliver solution of some problem between old and new
- make good for lag in later problems



Infeasibility treatment

$\mathbb{P}$ convex: QP on path infeasible <=> new QP infeasible

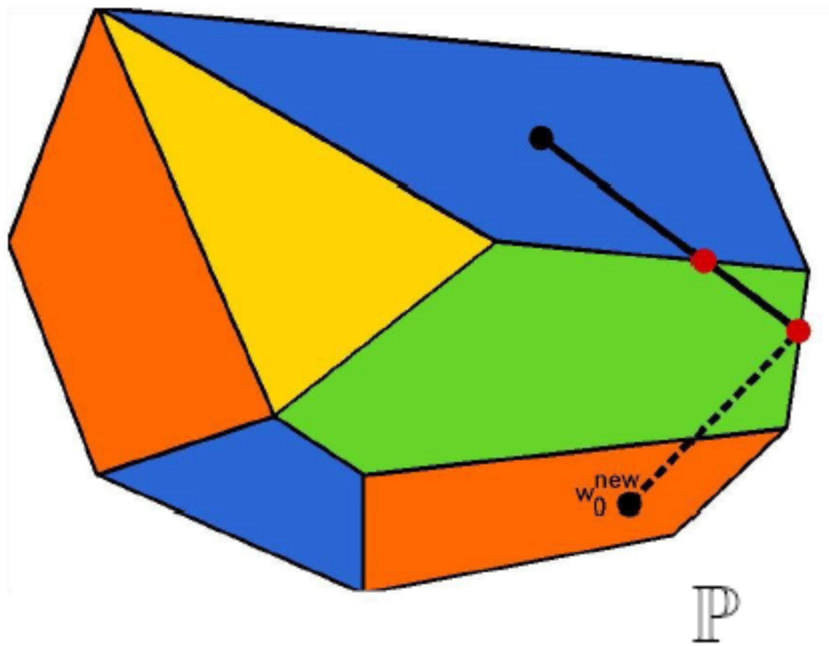
Stop at last feasible QP, wait for better posed problems



Infeasibility treatment

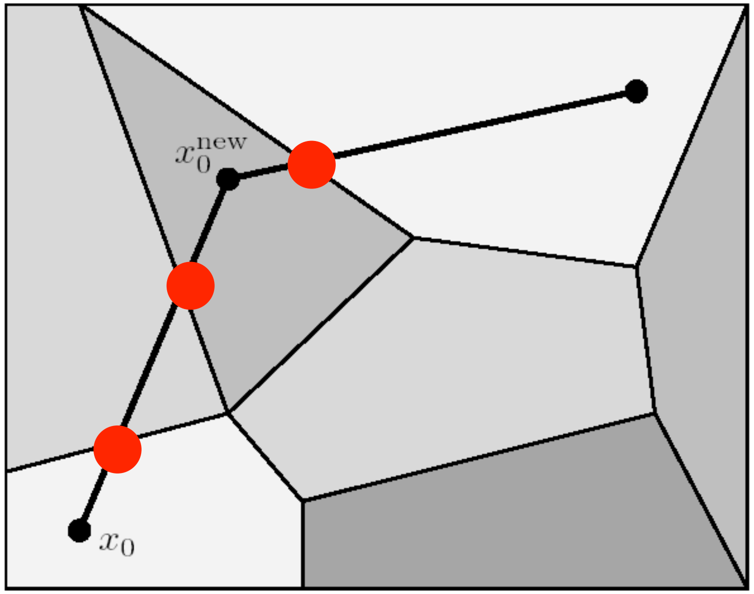
$\mathbb{P}$ convex: QP on path infeasible <=> new QP infeasible

Stop at last feasible QP, wait for better posed problems



Fortunately:
new QP feasible <=> full path is feasible, and strategy works again

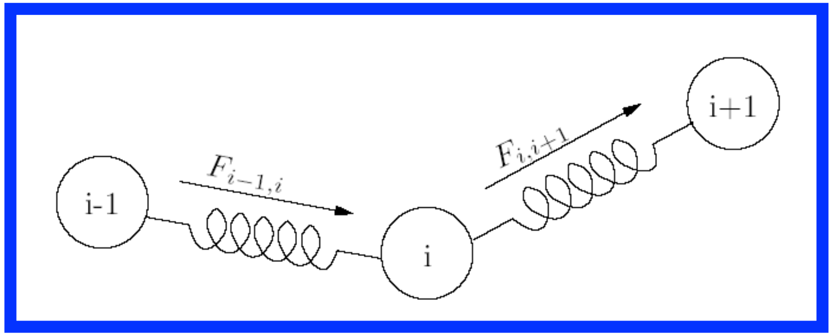
qpOASES: Open Code by Hans Joachim Ferreau



qpOASES:
open source C++ code by Hans Joachim Ferreau

http://www.kuleuven.be/optec/software/qpOASES

Application to Chain of Masses



- 10 balls connected by springs, No. 1 fixed
- 3-D velocity of ball No. 10 controlled: $\dot{x}_{N+1} = u(t)$
- 2nd order ODE for other balls:

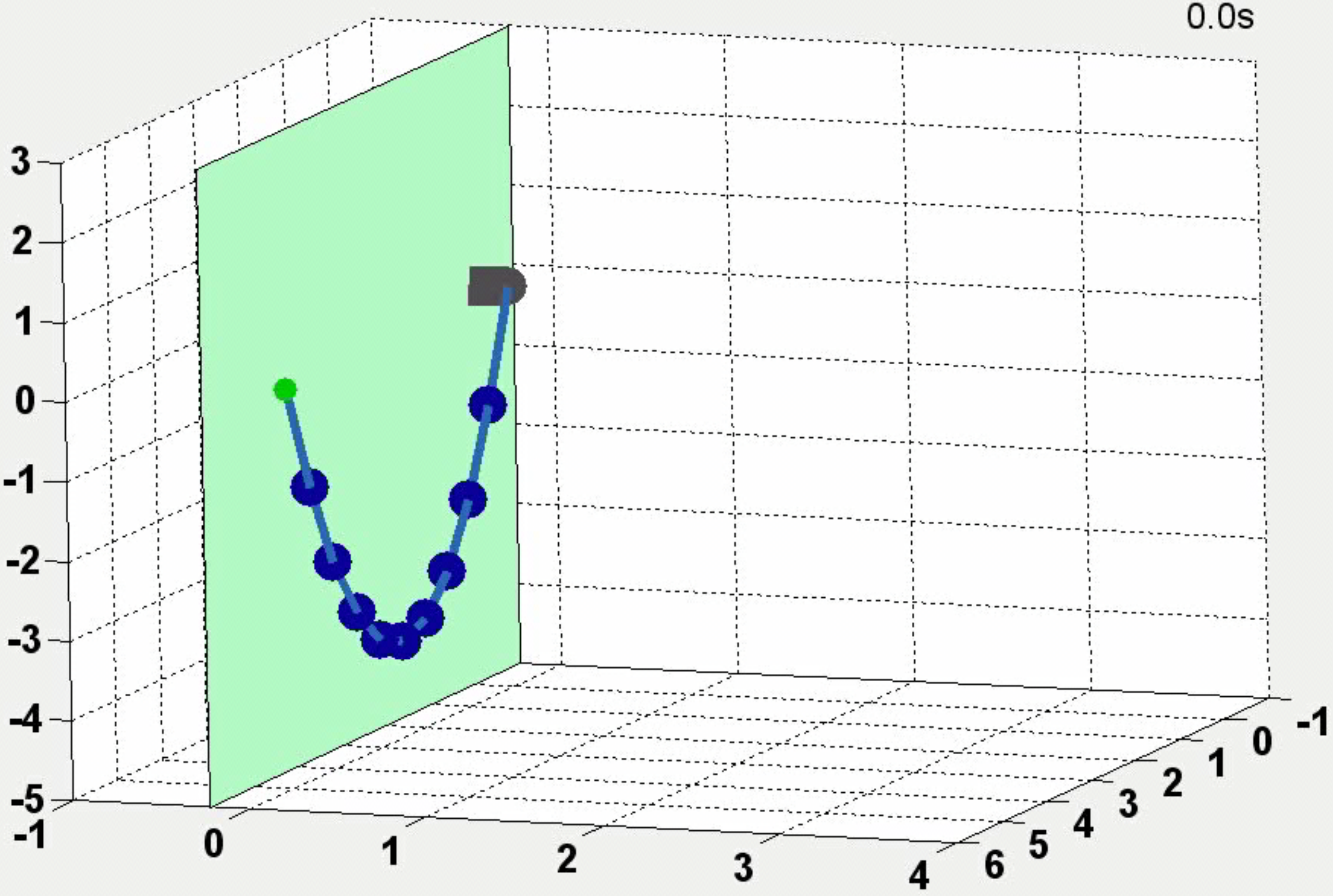
$$\ddot{x}_i = \frac{1}{m}\left(F_{i,i+1} - F_{i-1,i}\right) + g, \quad i = 1, \ldots, N$$

- Force according to Hooke's law strongly nonlinear:

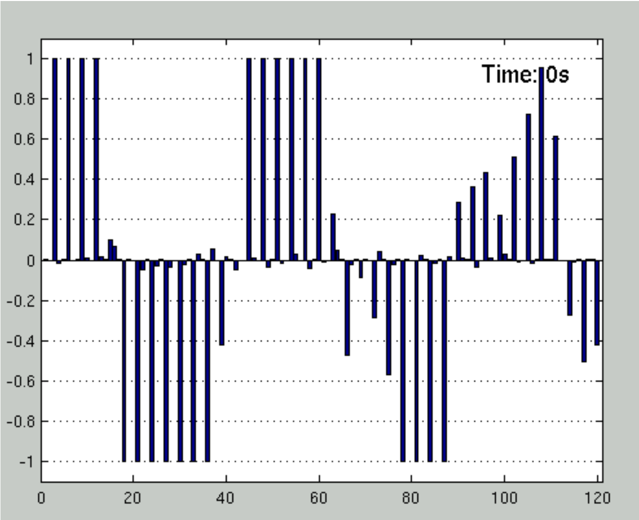
$$F_{i,i+1} = D\left(1 - \frac{L}{\|x_{i+1} - x_i\|}\right)(x_{i+1} - x_i),$$

- Together: 57 nonlinear ODEs, chaotic system

After disturbance, chain crashes into wall



MPC controller shall avoid crashing into wall

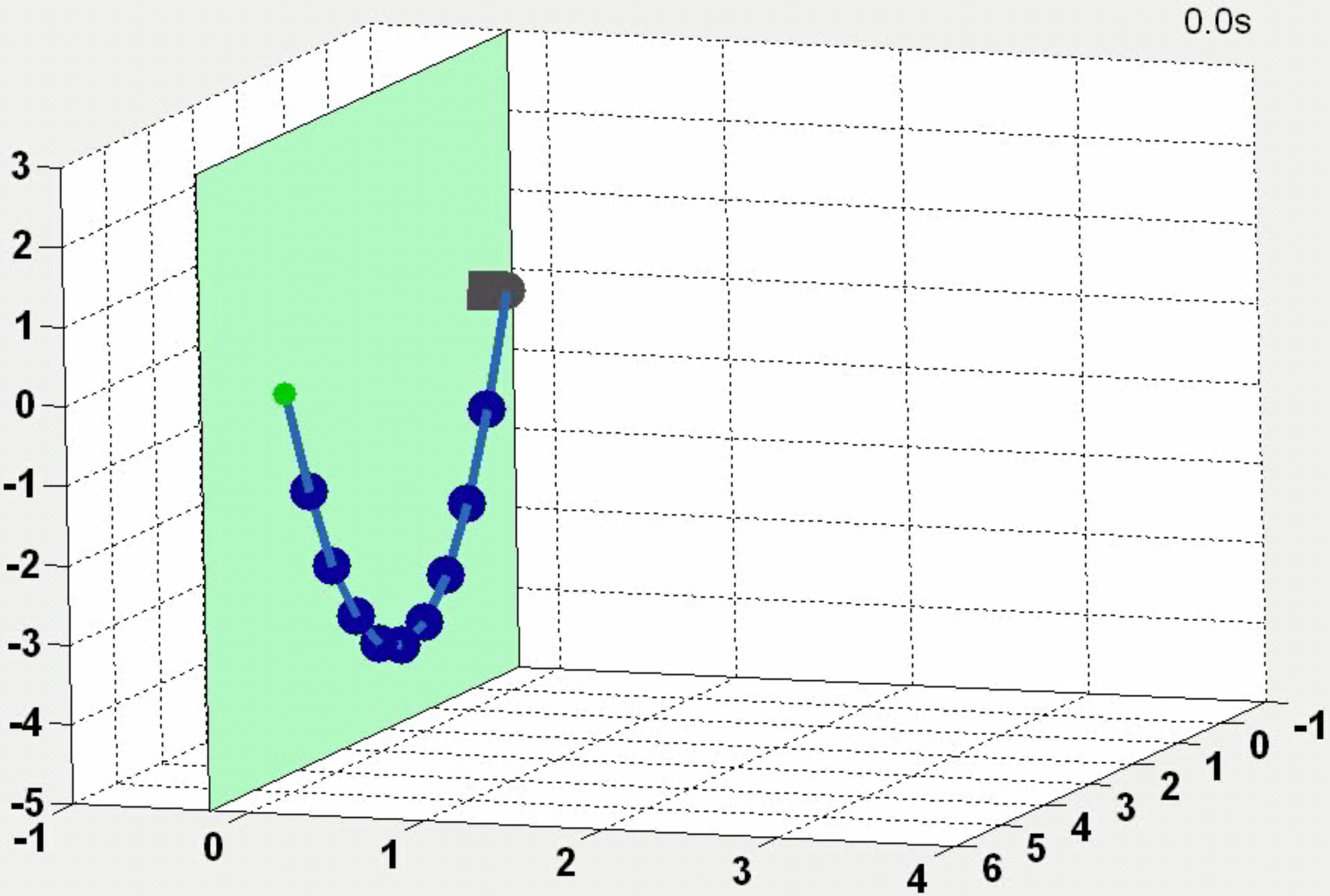


first half of MPC horizon

- linearize system at steady state
- choose 200 ms sampling time
- predict 80 samples:
  3 x 80 = **240 degrees of freedom**
- **bounds** (up/lo): 2 x 240 = **480**
- **state constraints** that avoid hitting the wall: 9 x 80 = **720**

Note: large QP with ~1 MB data

MPC respects bounds and state constraint

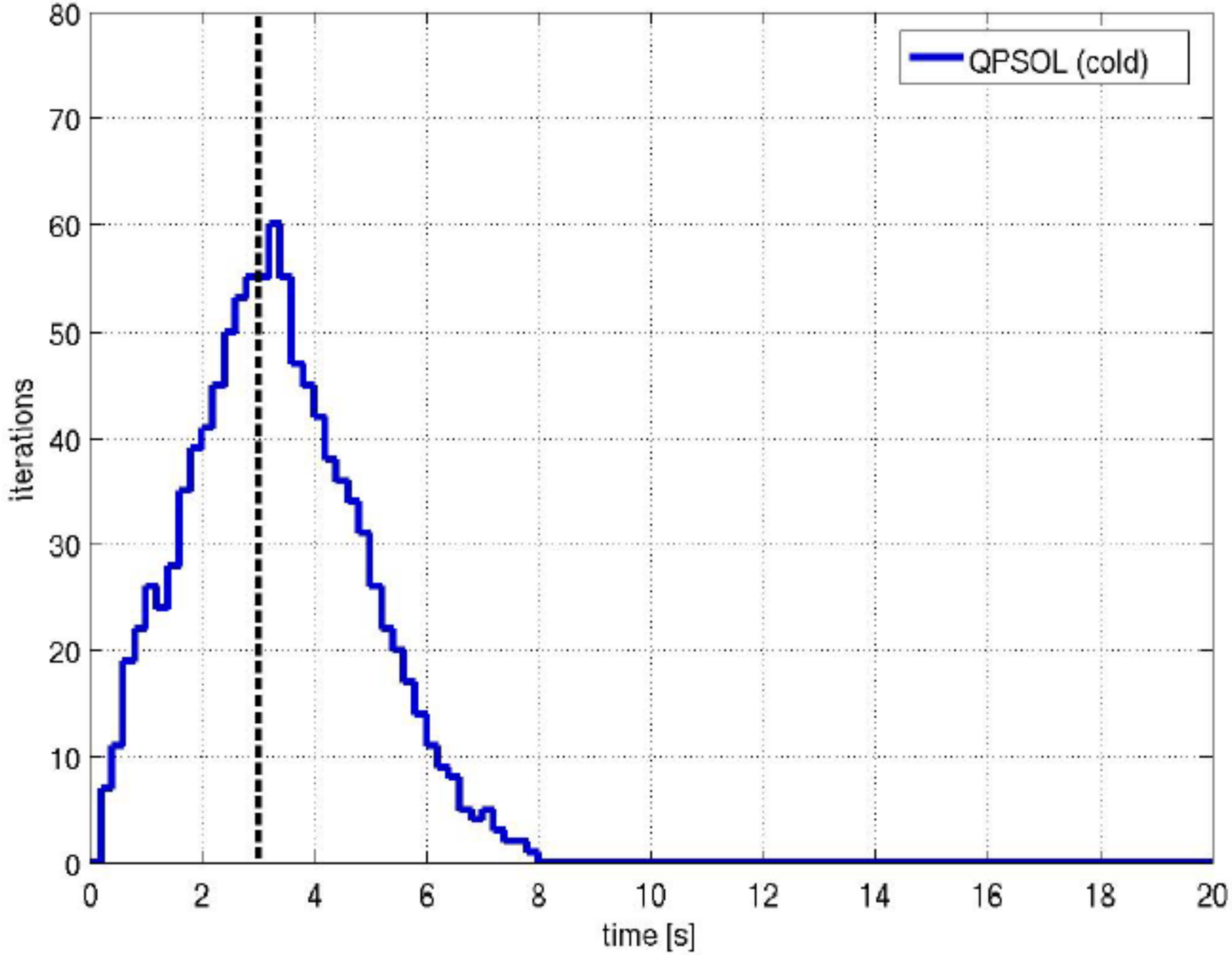


Compare four QP strategies

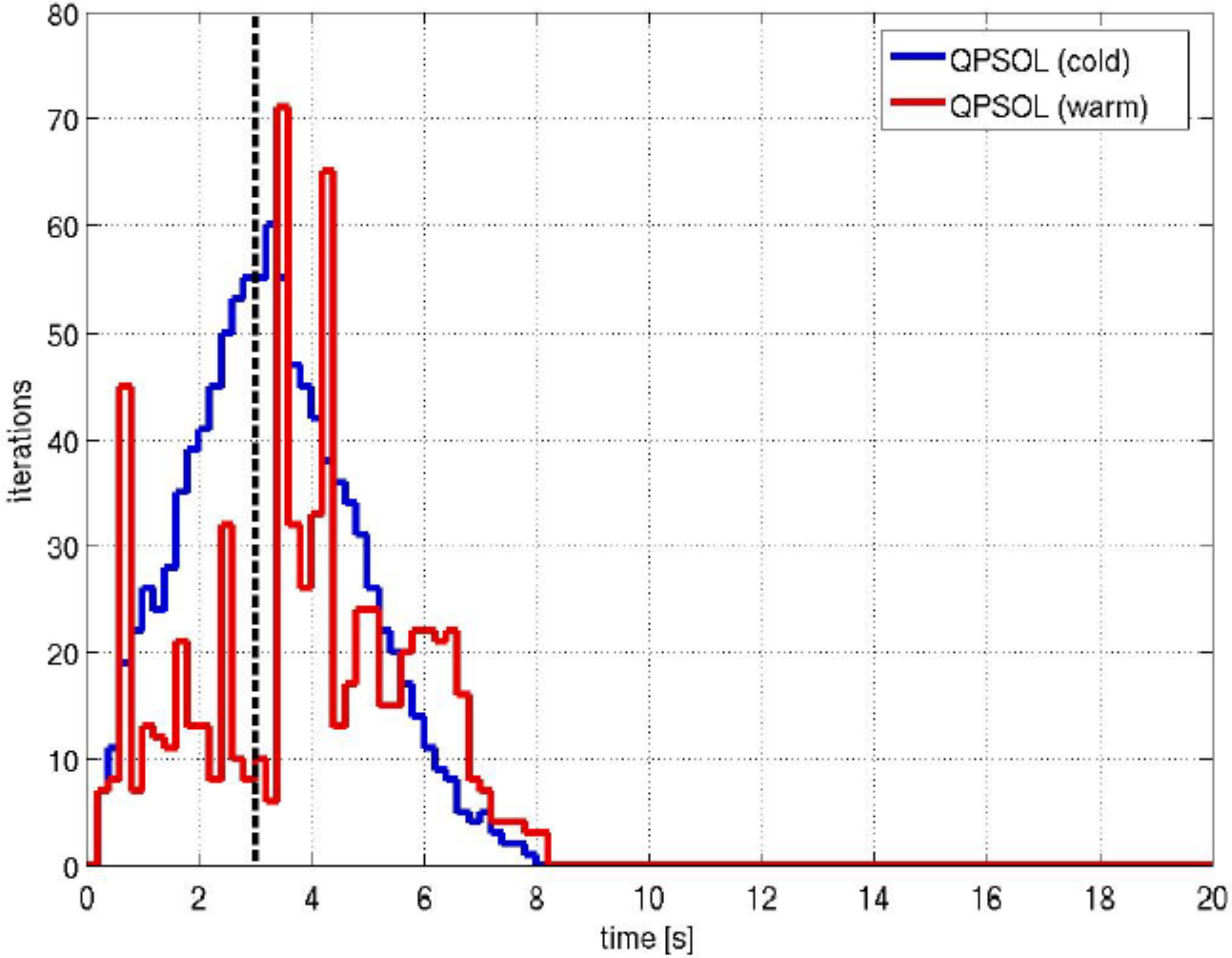
- Standard solver (QPSOL), cold start
- QPSOL, warm start
- Online Active Set Strategy (qpOASES), full convergence
- qpOASES, real-time variant with at most 10 QP Iterations

Note: Explicit MPC cannot be applied due to problem size

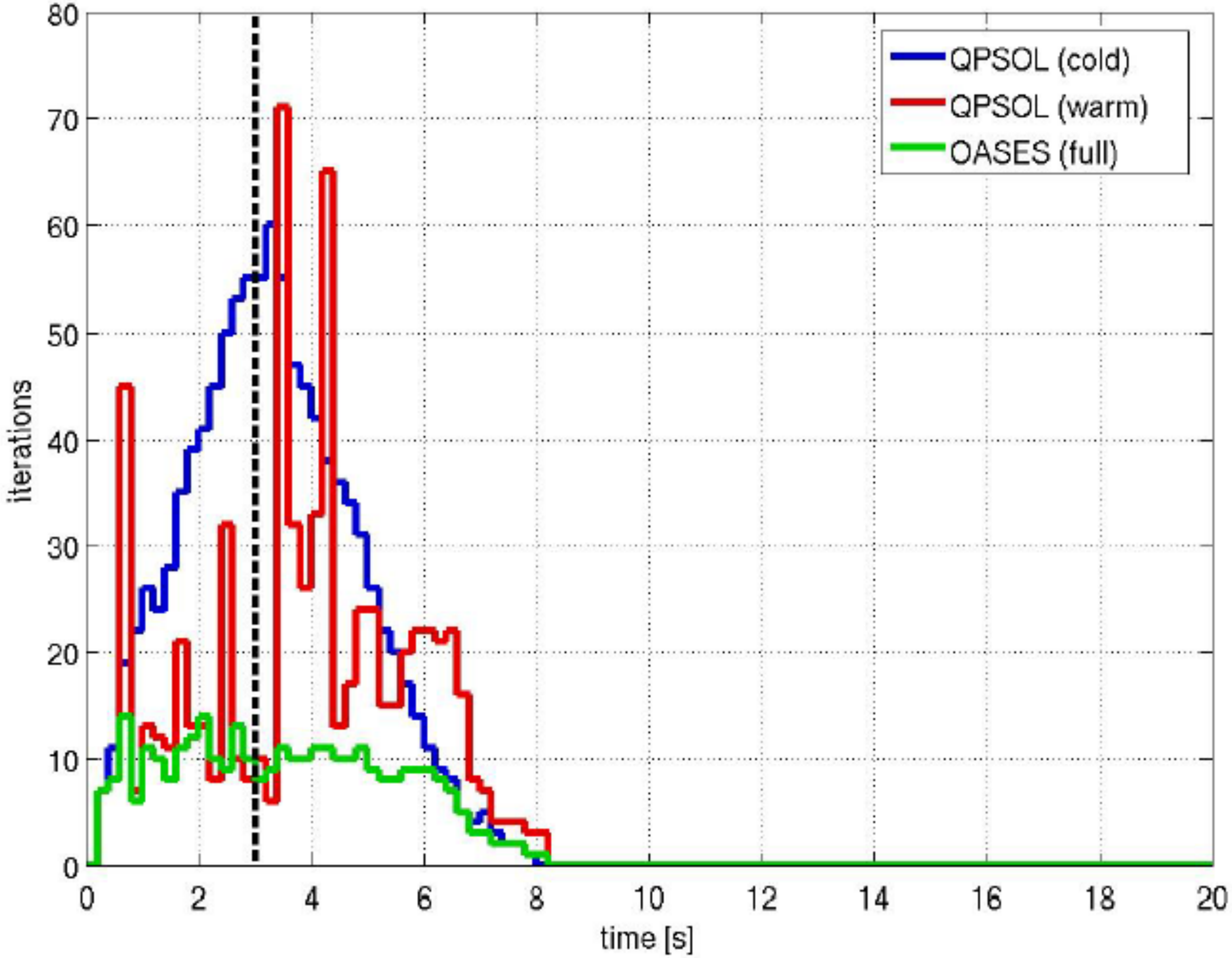
Number of QP Iterations (Working Set Changes)



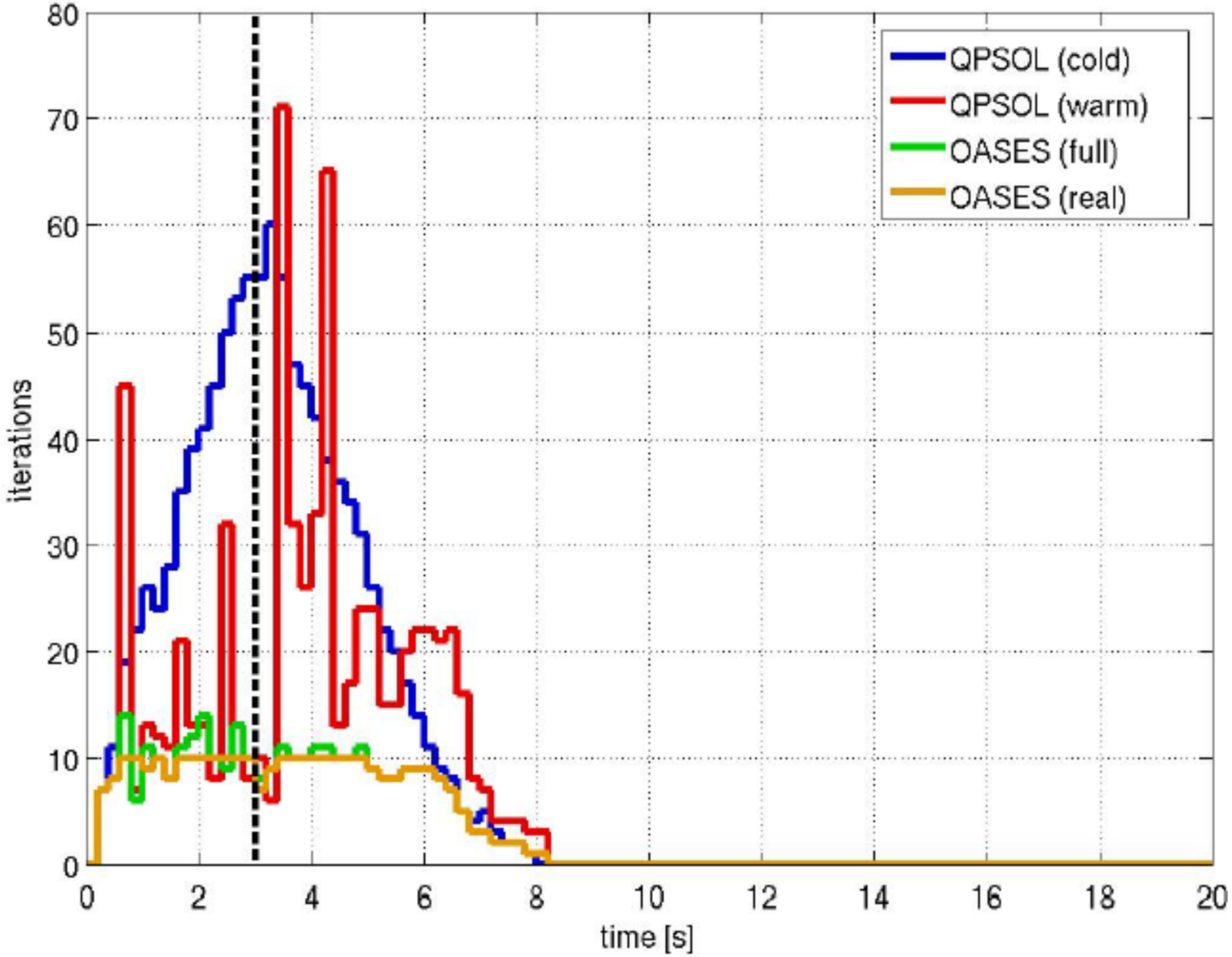
Number of QP Iterations (Working Set Changes)



Number of QP Iterations (Working Set Changes)



Number of QP Iterations (Working Set Changes)



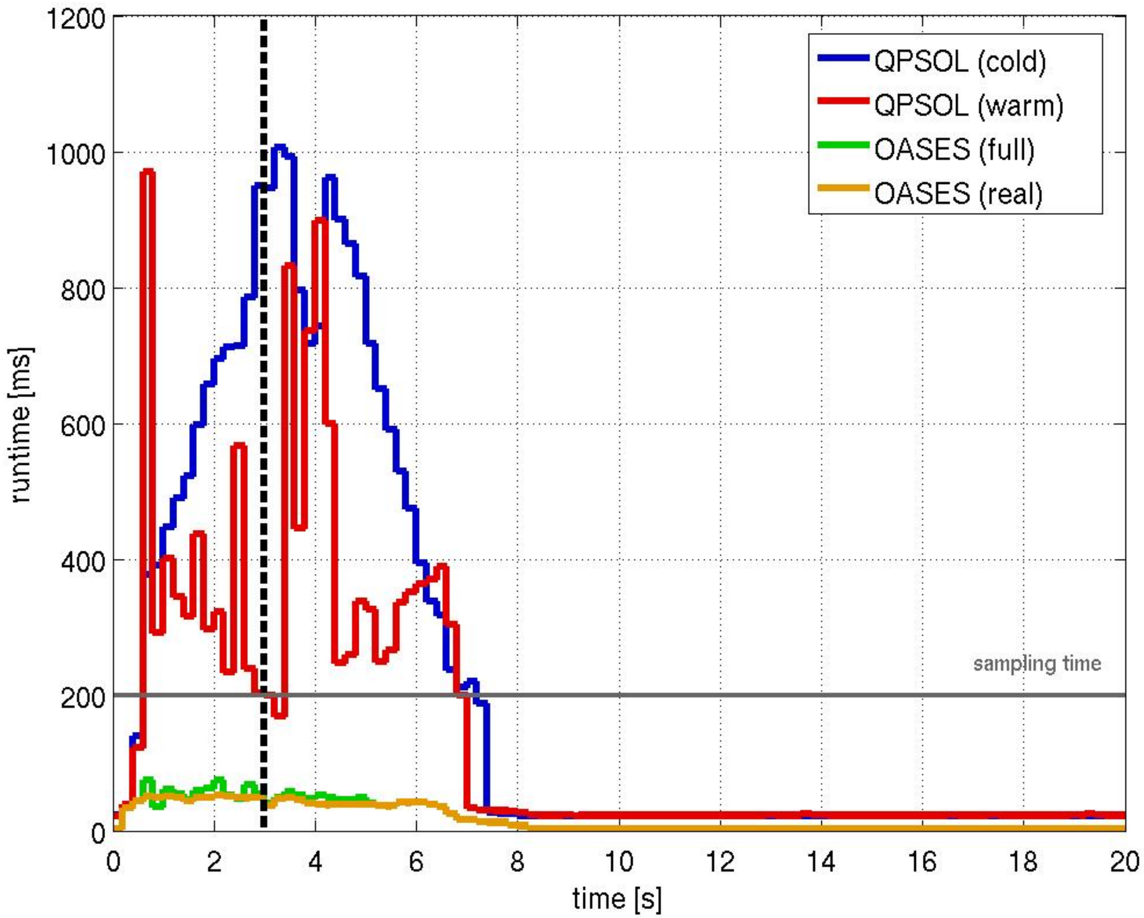
Performance of Online Active Set Strategy

- Number of QP working set changes 3-5 times lower than for QPSOL with warm starts
- Can limit maximum number without much suboptimality

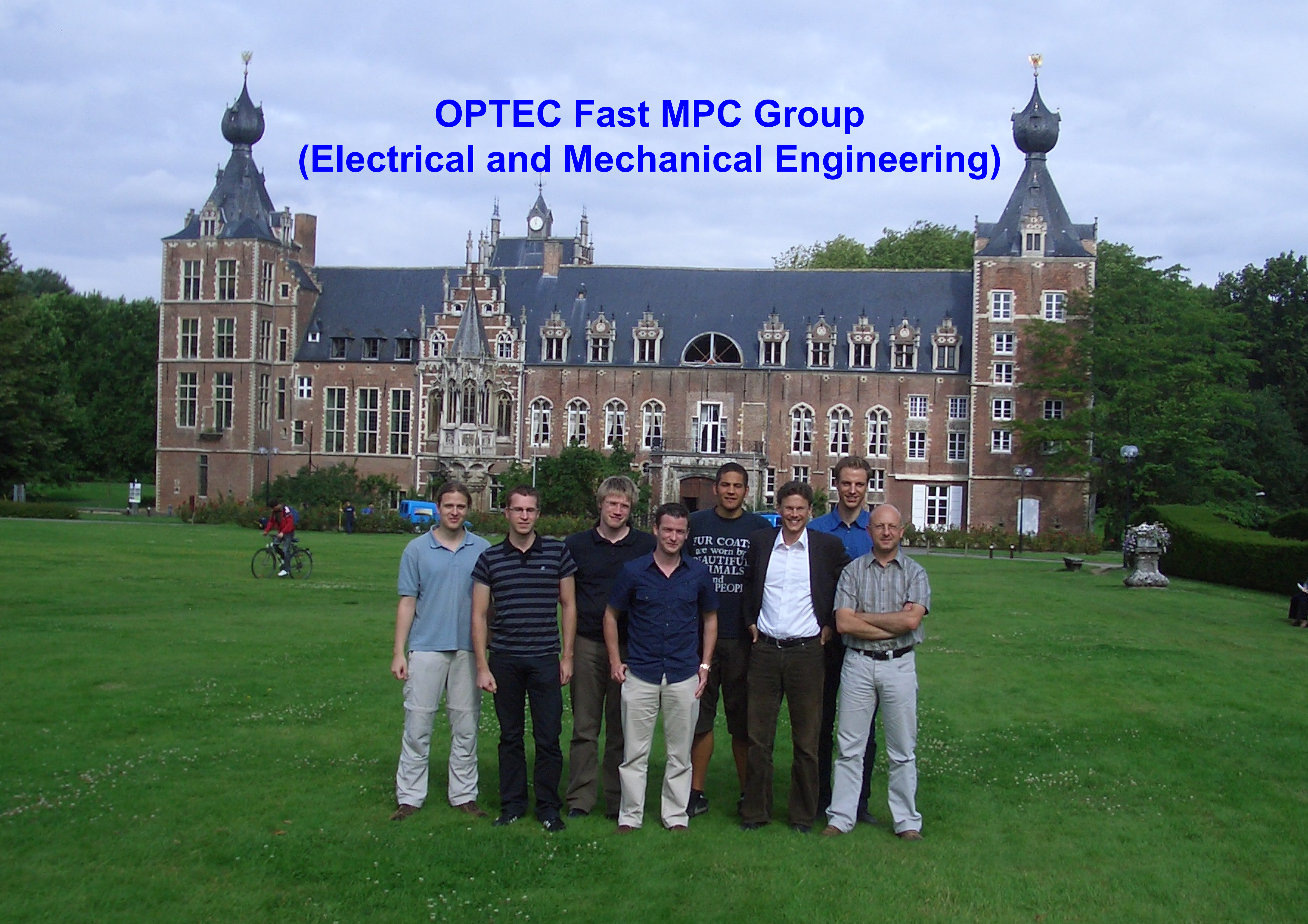
Additionally:

- QPSOL often even needs Phase1 LP Iterations
- qpOASES needs no new matrix factorizations
- CPU times compare even more favourably...

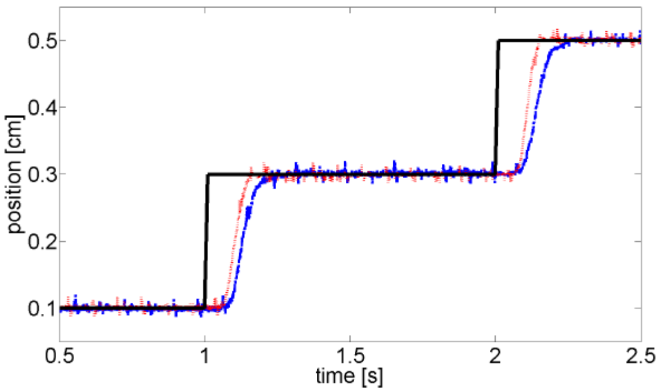
CPU Time Comparison: qpOASES Factor 10 Faster

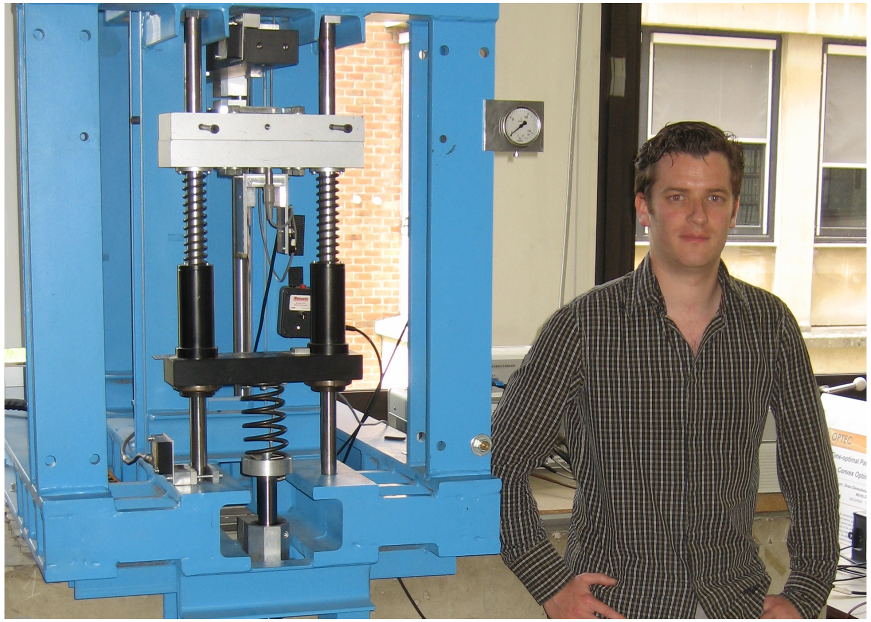


OPTEC Fast MPC Group
(Electrical and Mechanical Engineering)



Time Optimal MPC: a 100 Hz Application

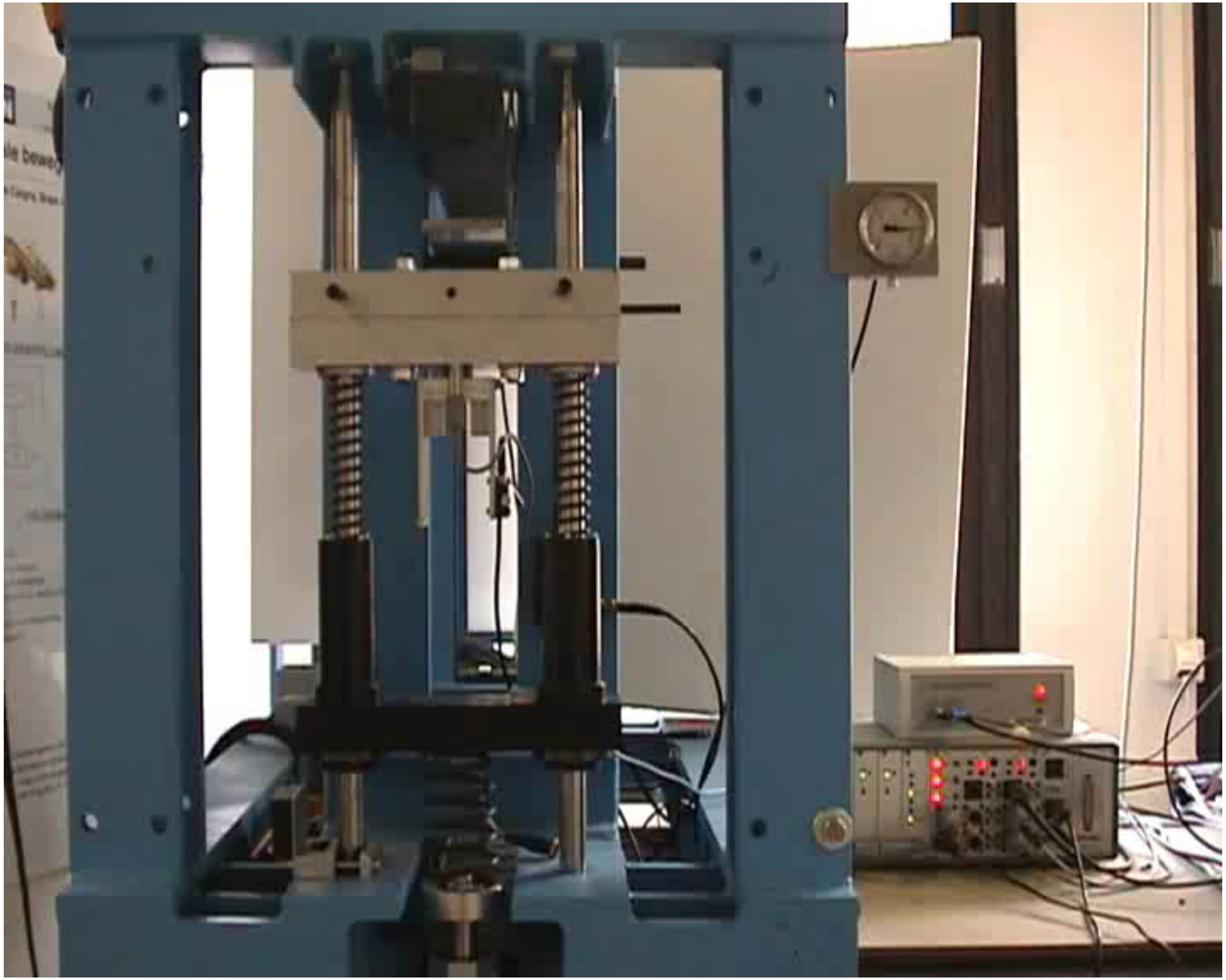




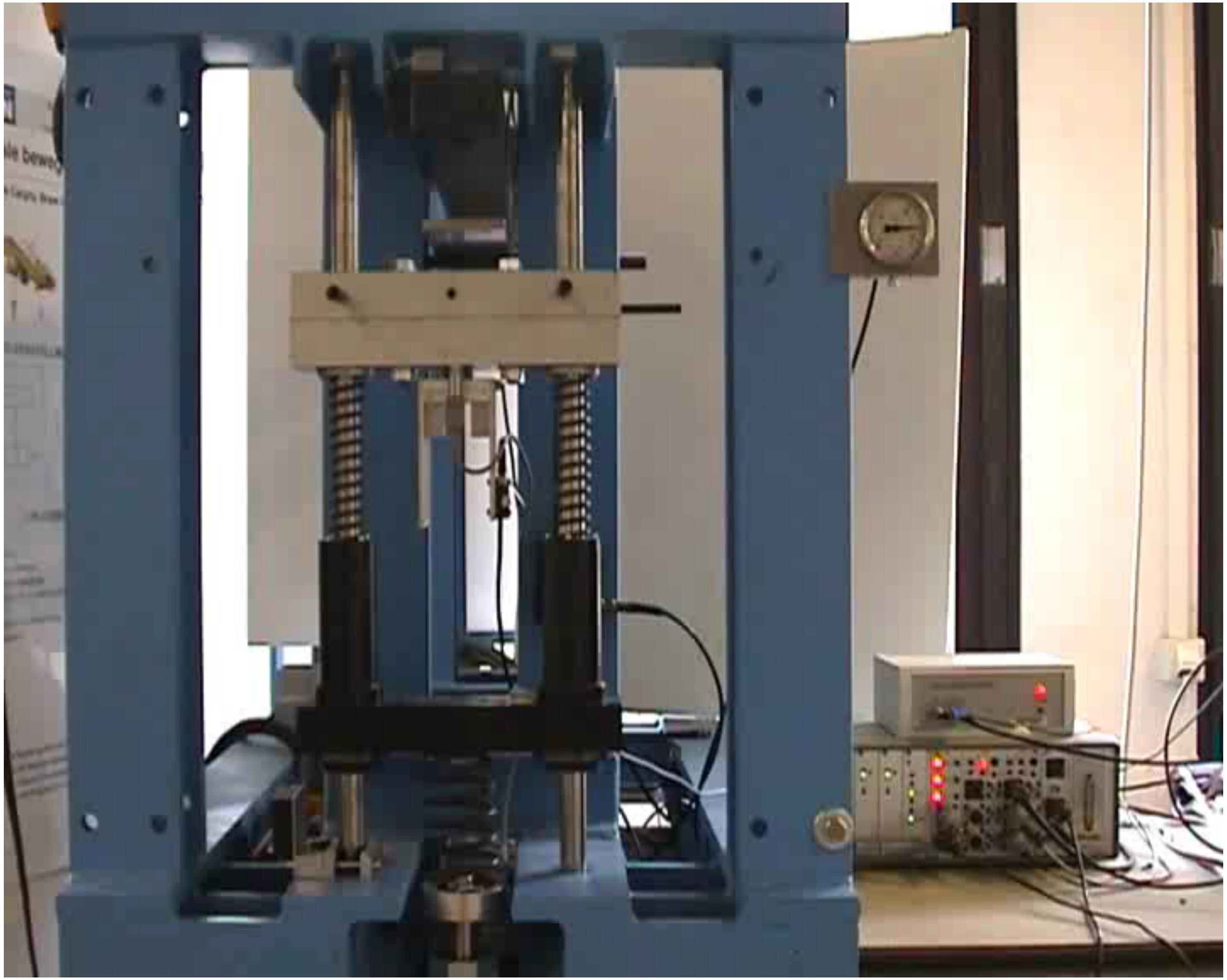
- Quarter car: oscillating spring damper system
- MPC Aim: settle at any new setpoint in ***in minimal time***
- Two level algorithm: MIQP
  - 6 online data
  - 40 variables + one integer
  - 242 constraints (in-&output)
- use qpOASES on dSPACE
- **CPU time: <10 ms**

Lieboud Van den Broeck in front of quarter car experiment

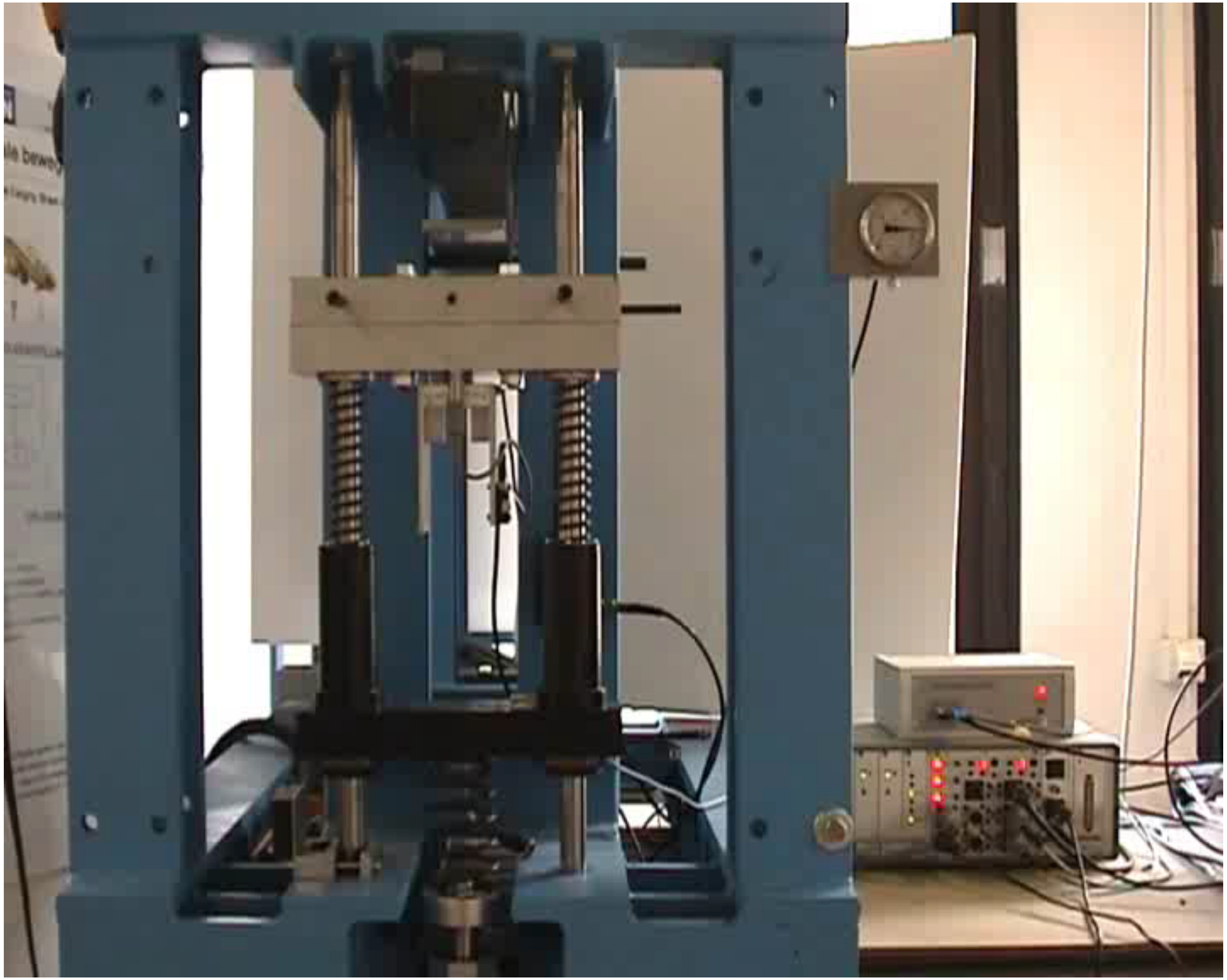
Setpoint change without control: oscillations



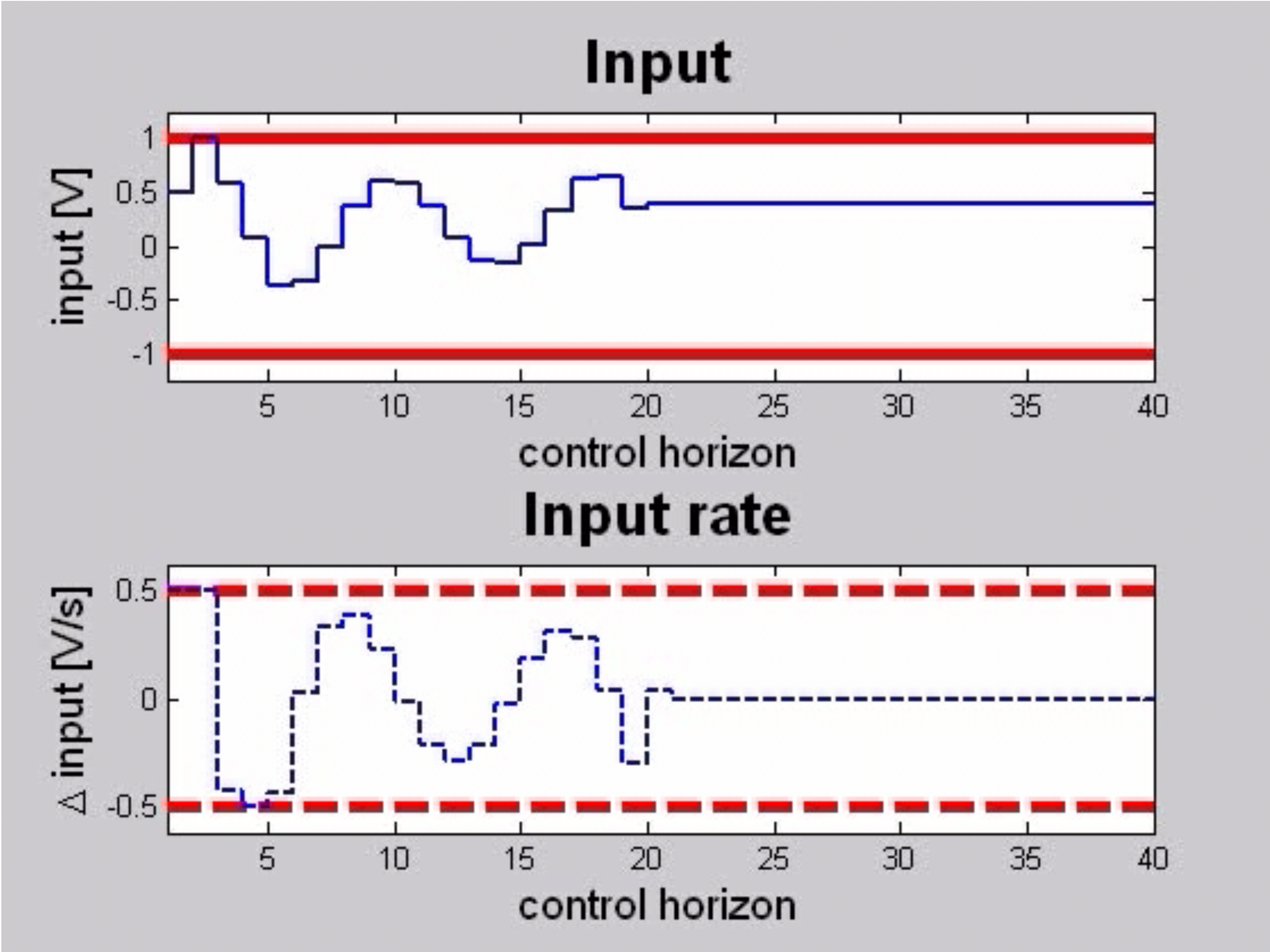
With LQR control: inequalities violated



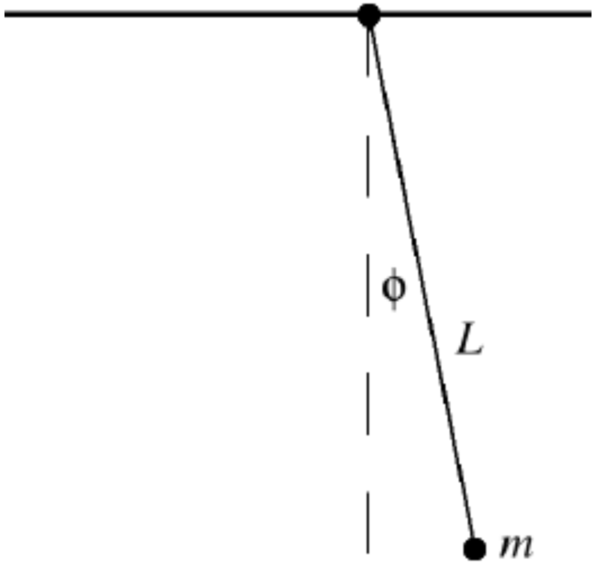
With Time Optimal MPC

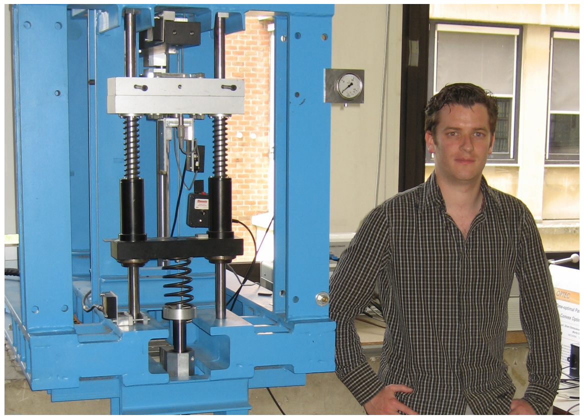


Time Optimal MPC: qpOASES Optimizer Contents



Time Optimal MPC: a 60 Hz Application





Lieboud Van den Broeck

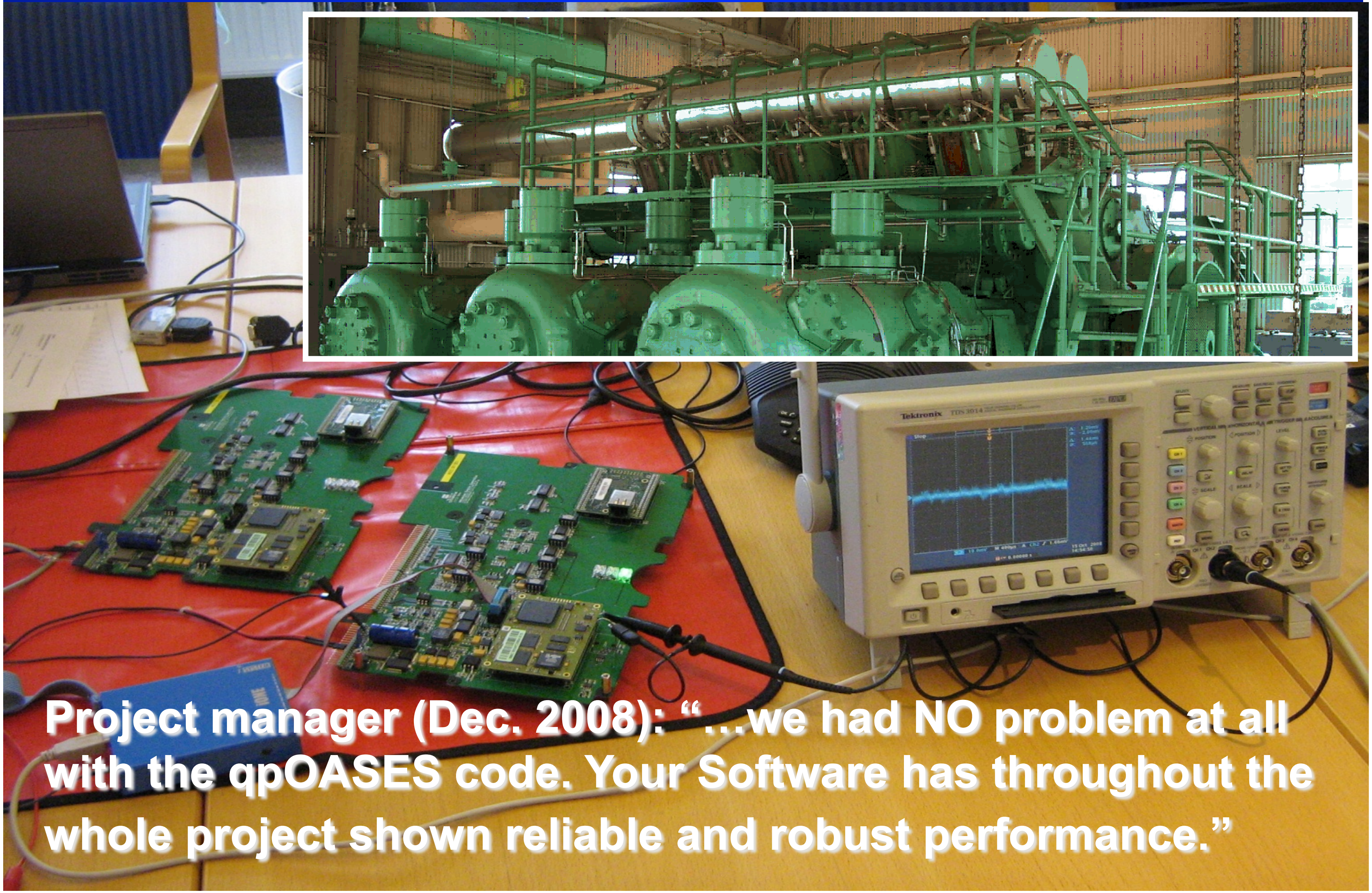
- Overhead crane
- MPC Aim: settle at any new setpoint in ***in minimal time***
- Two level algorithm: MIQP
  - 6 online data
  - 40 variables + one integer
  - 242 constraints (in-&output)
- use qpOASES on dSPACE
- **CPU time: <10 ms**

qpOASES: open code, but direct industrial funding

Hoerbiger: MPC of Large Bore Gas Engines for US Pipeline Compressors

IPCOS: Efficient QP for Process Control

qpOASES running on Industrial Control Hardware (20 ms)



Project manager (Dec. 2008): "…we had NO problem at all with the qpOASES code. Your Software has throughout the whole project shown reliable and robust performance."

Conclusions

- Linear and Nonlinear MPC need reliable QP solution
- Explicit MPC prohibitive for nontrivial problem dimensions
- Online Active Set Strategy (qpOASES) is one order of magnitude faster than conventional QP with warmstarts
- Linear MPC in kHz range realizable even for larger QPs
- Time Optimal MPC interesting alternative to tracking