Lecture 12: Summing up MPC and LQ(G)

- Recap (mostly): Model Predictive Control (MPC), finite and infinite horizon LQ
- Output feedback LQ: LQG
- Aspects related to implementation of MPC in industry

Reference: B&H Ch. 4.5-4.6

Linear MPC: repeated open loop dynamic optimization

$$\min_{z \in \mathbb{R}^n} f(z) = \sum_{t=0}^{N-1} \frac{1}{2} x_{t+1}^\top Q_{t+1} x_{t+1} + d_{xt+1} x_{t+1} + \frac{1}{2} u_t^\top R_t u_t + d_{ut} u_t + \frac{1}{2} \Delta u_t^\top S_t \Delta u_t$$

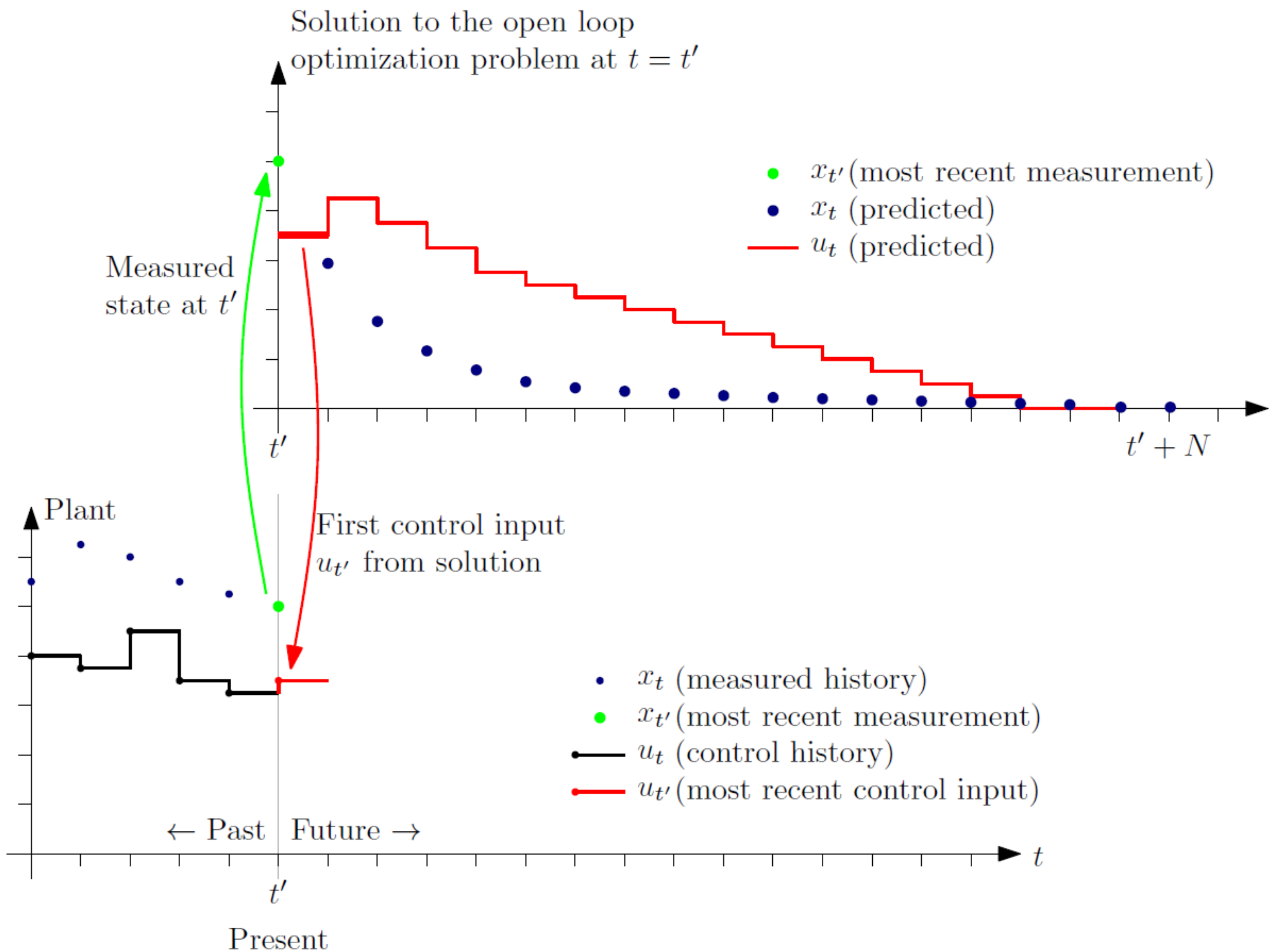
subject to

$$\begin{aligned}
x_{t+1} &= A_t x_t + B_t u_t, \quad t = \{0, \dots, N-1\} \\
x^{\text{low}} &\leq x_t \leq x^{\text{high}}, \quad t = \{1, \dots, N\} \\
u^{\text{low}} &\leq u_t \leq u^{\text{high}}, \quad t = \{0, \dots, N-1\} \\
-\Delta u^{\text{high}} &\leq \Delta u_t \leq \Delta u^{\text{high}}, \quad t = \{0, \dots, N-1\} \\
Q_t &\succeq 0 \quad t = \{1, \dots, N\} \\
R_t &\succeq 0 \quad t = \{0, \dots, N-1\} \\
S_t &\succeq 0 \quad t = \{0, \dots, N-1\}
\end{aligned}$$

where

$$\begin{aligned}
x_0 &\text{ and } u_{-1} \text{ is given} \\
\Delta u_t &:= u_t - u_{t-1} \\
z^\top &:= (u_0^\top, x_1^\top, \dots, u_{N-1}^\top, x_N^\top) \\
n &= N \cdot (n_x + n_u)
\end{aligned}$$

Model predictive control principle



LQ: MPC open loop problem without constraints

$$\min_{z \in \mathbb{R}^n} f(z) = \sum_{t=0}^{N-1} \frac{1}{2} x_{t+1}^\top Q_{t+1} x_{t+1} + \frac{1}{2} u_t^\top R_t u_t$$

subject to

$$\begin{aligned} x_{t+1} &= A_t x_t + B_t u_t, \quad t = \{0, \dots, N-1\} \\ x_0 &= \text{given} \\ Q_t &\succeq 0 \quad t = \{1, \dots, N\} \\ R_t &\succ 0 \quad t = \{0, \dots, N-1\} \end{aligned}$$

where

$$\begin{aligned} z^\top &:= (u_0^\top, x_1^\top, \dots, u_{N-1}^\top, x_N^\top) \\ n &= N \cdot (n_x + n_u) \end{aligned}$$

- Solution: LTV state feedback

$$u_t = -K_t x_t$$

where the feedback gain matrix is derived by

$$\begin{aligned} K_t &= R_t^{-1} B_t^\top P_{t+1} (I + B_t R_t^{-1} B_t^\top P_{t+1})^{-1} A_t, \qquad t = 0, \dots, N-1 \\ P_t &= Q_t + A_t^\top P_{t+1} (I + B_t R_t^{-1} B_t^\top P_{t+1})^{-1} A_t, \qquad t = 0, \dots, N-1 \\ P_N &= Q_N \end{aligned}$$

Linear quadratic control; some observations

- The optimal solution to LQ control is a linear, time-varying state feedback:

$$u_t = -K_t x_t$$

where the feedback gain matrix is derived by

$$\begin{aligned}
K_t &= R_t^{-1} B_t^\top P_{t+1}(I + B_t R_t^{-1} B_t^\top P_{t+1})^{-1} A_t, \qquad && t = 0, \dots, N-1 \\
P_t &= Q_t + A_t^\top P_{t+1}(I + B_t R_t^{-1} B_t^\top P_{t+1})^{-1} A_t, \qquad && t = 0, \dots, N-1 \\
P_N &= Q_N
\end{aligned}$$

- The matrix (difference) equation

$$\begin{aligned}
P_t &= Q_t + A_t^\top P_{t+1}(I + B_t R_t^{-1} B_t^\top P_{t+1})^{-1} A_t, \qquad && t = 0, \dots, N-1 \\
P_N &= Q_N
\end{aligned}$$

  is called the (discrete-time) *Riccati equation*.

- Note that the gain matrix K_t and the Riccati equation is independent of the states. It can therefore be computed in advance (knowing A_t, B_t, Q_t, R_t).

- Note that the “boundary condition” is given at the end of the horizon, and the P_t -matrices must be found iterating backwards in time.

Example

$$\min \sum_{t=0}^{10} \frac{1}{2} x_{t+1}^2 + \frac{1}{2} r\, u_t^2$$

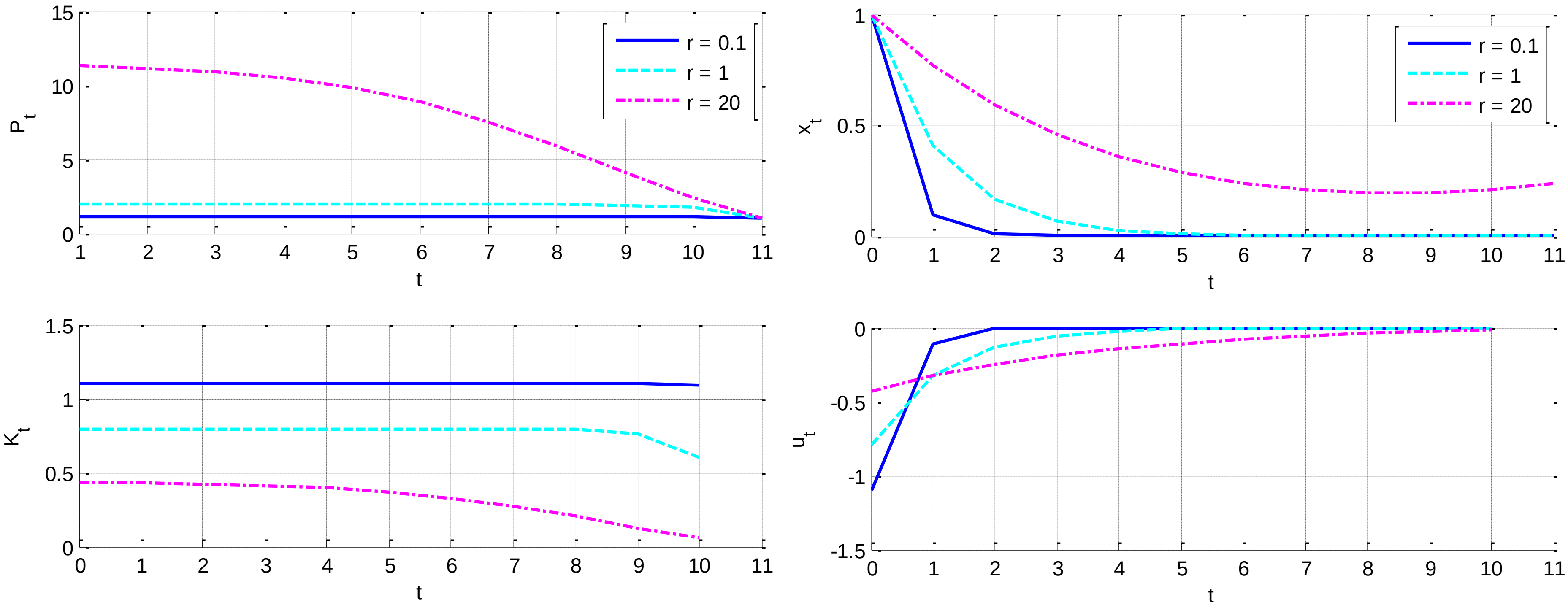
$$\text{s.t.} \quad x_{t+1} = 1.2 x_t + u_t, \quad t = 0, 1, \ldots, 10$$

LQ solution

$$P_t = 1 + \frac{1.44 r P_{t+1}}{P_{t+1} + r}, \quad t = 10, \ldots, 1$$

$$P_{11} = 1$$

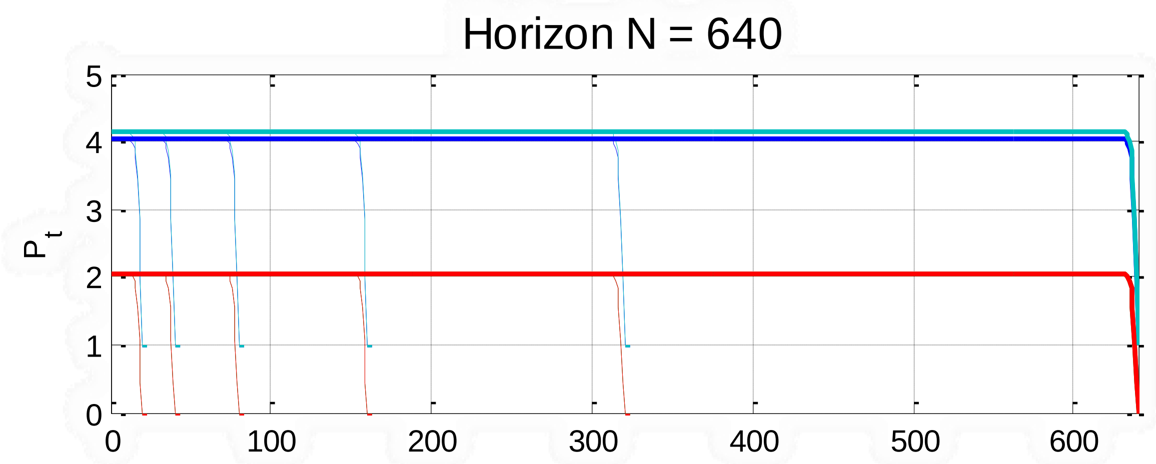
$$K_t = 1.2 \frac{P_{t+1}}{P_{t+1} + r}, \quad t = 0, \ldots, 10$$



Increasing LQ horizon

$$\min \sum_{t=0}^{N-1} \frac{1}{2} x_{t+1}^\top Q x_{t+1} + \frac{1}{2} u_t^\top R u_t$$
$$\text{s.t.} \quad x_{t+1} = A x_t + B u_t, \quad t = 0, 1, \ldots, N-1$$

$$A = \begin{pmatrix} 1 & 0.5 \\ 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 0.125 \\ 0.5 \end{pmatrix}, \quad Q = I, \quad R = 1.$$



Infinite horizon LQ solution is steady-state finite horizon LQ solution!

Infinite horizon LQ (LQR)

$$\min_{z\in\mathbb{R}^\infty} f(z) = \sum_{t=0}^{\infty} \frac{1}{2} x_{t+1}^\top Q x_{t+1} + \frac{1}{2} u_t^\top R u_t$$

subject to

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t, \quad t = \{0, 1, \ldots\} \\ x_0 &= \text{given} \\ Q &\succeq 0 \\ R &\succ 0 \end{aligned}$$

- This has a solution provided (*A*,*B*) is stabilizable
- Then the optimal solution is the LTI state feedback

  $$u_t = -Kx_t$$

  where the feedback gain matrix is derived by

  $$\begin{aligned} K &= R^{-1}B^\top P(I + BR^{-1}B^\top P)^{-1}A, \\ P &= Q + A^\top P(I + BR^{-1}B^\top P)^{-1}A; \quad P = P^\top \succ 0 \end{aligned}$$

- This solution is guaranteed to be closed-loop stable (eigenvalues of *A*-*BK* stable) if (*A*,*D*) is detectable, where $Q = D^T D$
- Being a state feedback solution, it implies some robustness (more on this later)

Controllability vs stabilizability
Observability vs detectability

- Stabilizable: All unstable modes are controllable
(that is: all uncontrollable modes are stable)

- Detectability: All unstable modes are observable
(that is: all unobservable modes are stable)

- Controllability implies stabilizability
- Observability implies detectability

Riccati equations

- Discrete-time Riccati equation in the note (and lecture)

$$P_t = Q_t + A_t^\top P_{t+1}(I + B_t R_t^{-1} B_t^\top P_{t+1})^{-1} A_t, \quad P_N = Q_N$$

- Typically, other forms are used, e.g. (from Wikipedia):

$$P_t = Q_t + A_t^\top P_{t+1} A_t - A_t^\top P_{t+1} B_t (R_t + B_t^\top P_{t+1} B_t)^{-1} B_t^\top P_{t+1} A_t, \quad P_N = Q_N$$

- The latter is more numerically stable due to “enforced symmetry”

- The trick used to get the different formulas is “the Matrix Inversion Lemma” (a very useful Lemma in control theory…)

- Discrete-time Algebraic Riccati equation (DARE) in the note (and lecture)

$$P = Q + A^\top P (I + BR^{-1}B^\top P)^{-1} A$$

- Other form (e.g. Matlab)

$$P = Q + A^\top PA - A^\top PB(R + B^\top PB)^{-1} B^\top PA$$

  – Note: This is a quadratic equation with two solutions. The one we want is the positive definite solution (the “stabilizing” solution).

```
>> help dare
dare  Solve discrete-time algebraic Riccati equations.

   [X,L,G] = dare(A,B,Q,R,S,E) computes the unique
             stabilizing solution X of the discrete-time
             algebraic Riccati equation
```

LQR vs MPC

- LQR is MPC without constraints, solution is “linear state feedback”
  - MPC solution is “online optimization” (QP)

- Sometimes constraints can be active when far from setpoint, but become irrelevant close to setpoint.
  - MPC “reduces” to LQR close to setpoint (under some simple modifications of MPC problem)

- Consider double integrator example:

The double integrator, two integrators in series, discretized with sample interval T_s, can be written in state-space form as

$$A = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} T_s^2 \\ T_s \end{bmatrix}.$$

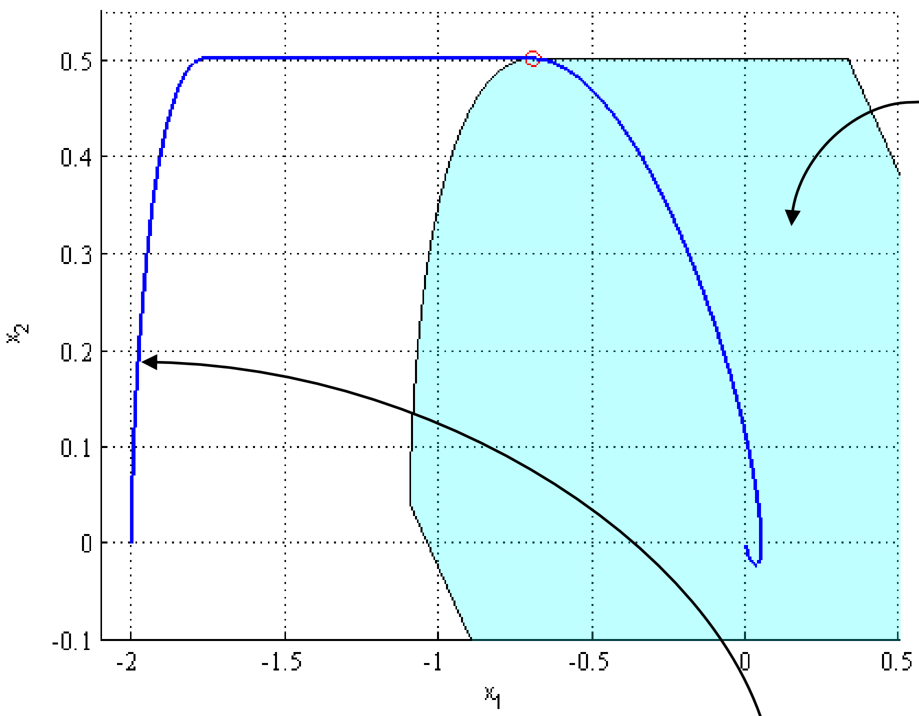
Consider an MPC cost function with

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad R = 1,$$

The constraints are $-0.5 \leq x_2 \leq 0.5$, and $-1 \leq u \leq 1$.

LQR vs MPC, II

Region (polytopic set) where LQR solution is optimal (where we can assume problem unconstrained)

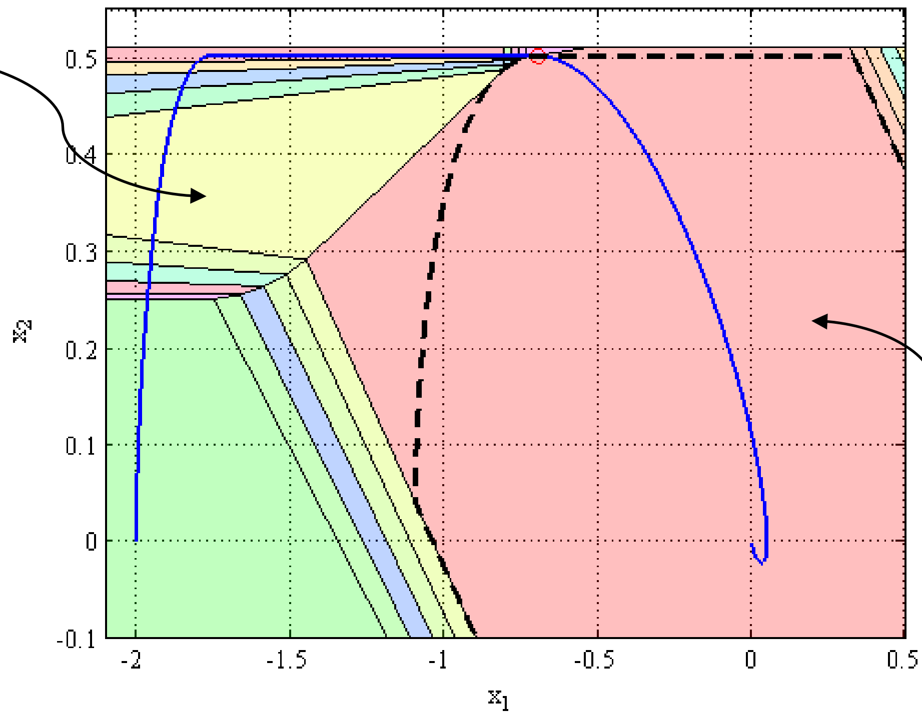


MPC solution has larger feasible region than LQR solution!

LQR vs MPC, III

- In fact, MPC solution is piecewise linear, defined on polytopic regions

$$u_k = K^i x_k + k^i$$



$$u_k = -Kx_k$$

- Proof/computation of this is an exercise in studying KKT conditions
  - (Not very difficult, but was not realized before ca. 2000)
  - But: quickly becomes very complex, except for very small systems.

Bemporad, A., Morari, M., Dua, V., Pistikopoulos, E.N.: The explicit linear quadratic regulator for constrained systems. Automatica 38(1), 3–20 (2002)
Tøndel, P., Johansen, T.A., Bemporad, A.: An algorithm for multi-parametric quadratic programming and explicit MPC solutions. Automatica 39(3), 489–497 (2003)

LQ regulator (LQR)

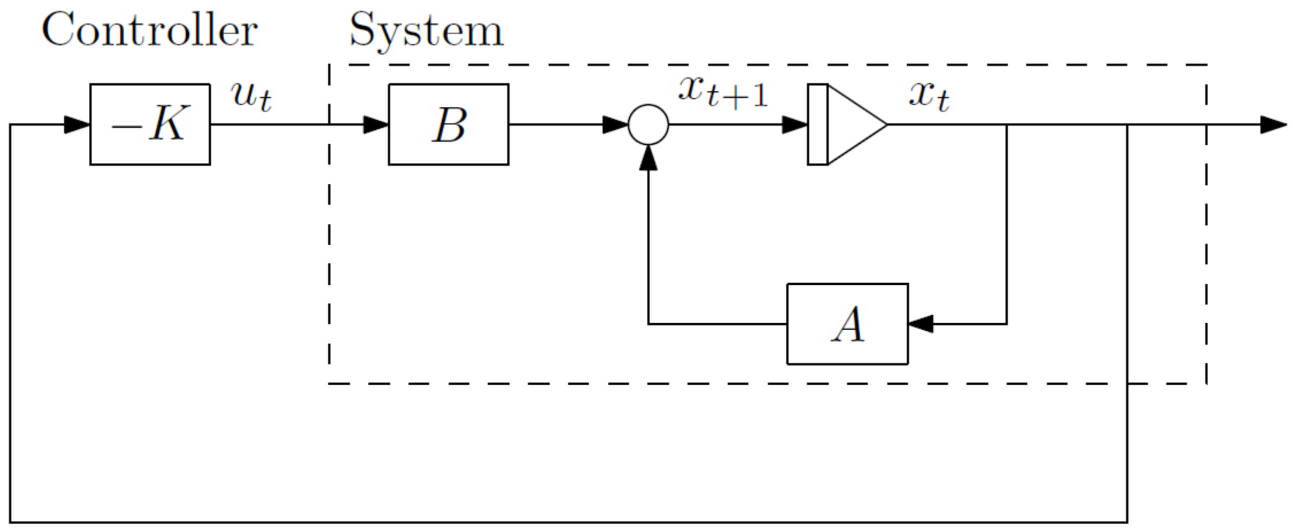


Figure 4.4: Solution of the LQ control problem, i.e., with state feedback.

LQ and robustness

- SISO LQ regulators have 60 degrees phase margin and 6dB gain margin
- Can be extended to MIMO systems

IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. AC-22, NO. 2, APRIL 1977 173

Gain and Phase Margin for Multiloop ~~LQG~~ Regulators

LQ

MICHAEL G. SAFONOV, STUDENT MEMBER, IEEE, AND MICHAEL ATHANS, FELLOW, IEEE

***Abstract*—Multiloop linear-quadratic state-feedback (LQSF) regulators are shown to be robust against a variety of large dynamical linear time-invariant and memoryless nonlinear time-varying variations in open-loop dynamics. The results are interpreted in terms of the classical concepts of gain and phase margin, thus strengthening the link between classical and modern feedback theory.**

measured in terms of multiloop generalizations of the classical notions of *gain and phase margin*. Like classical gain and phase margin, the present results consider robustness as an input-output property characterizing variations in open-loop transfer functions which will not

- However, usually one does not measure all the states…

Output feedback MPC

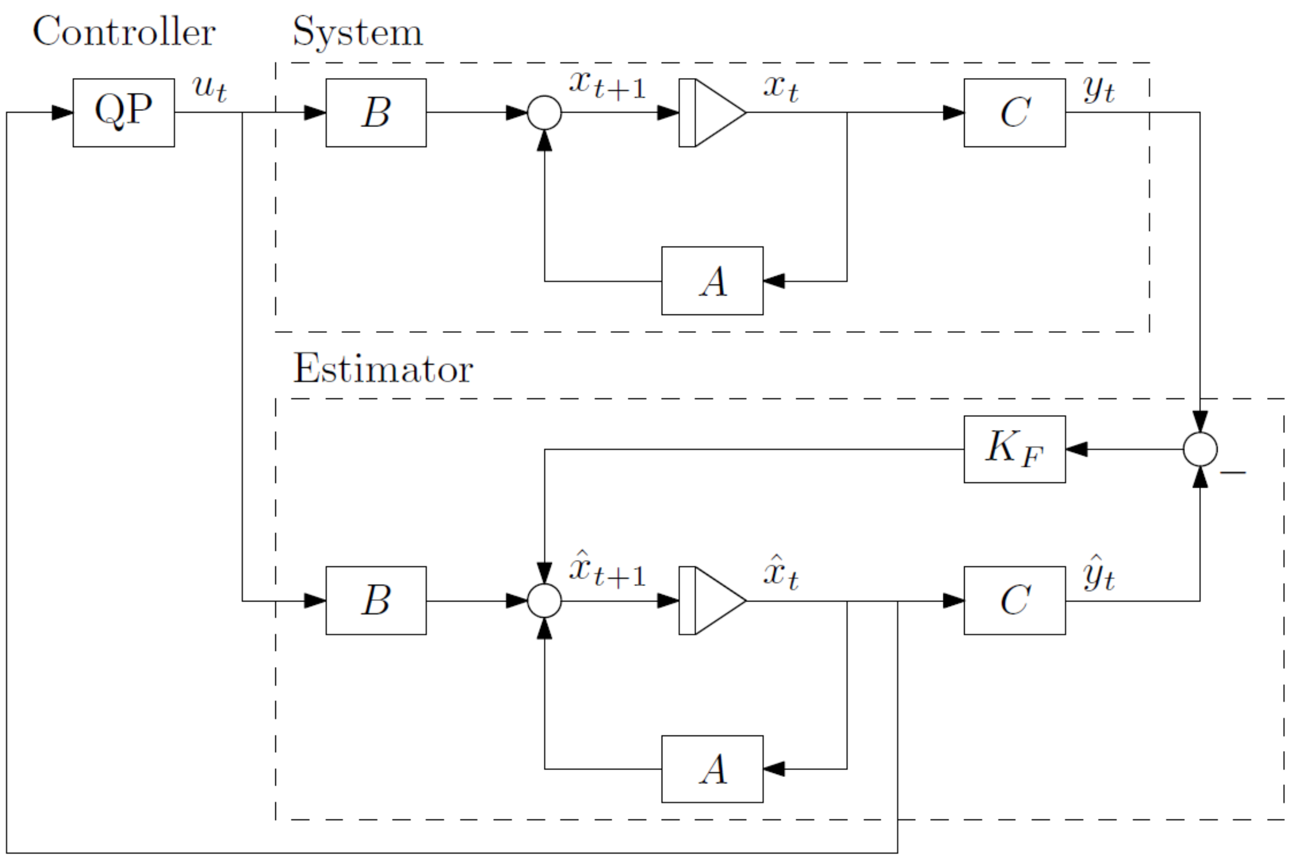


Figure 4.3: The structure of an output feedback linear MPC.

LQG: Linear Quadratic Gaussian

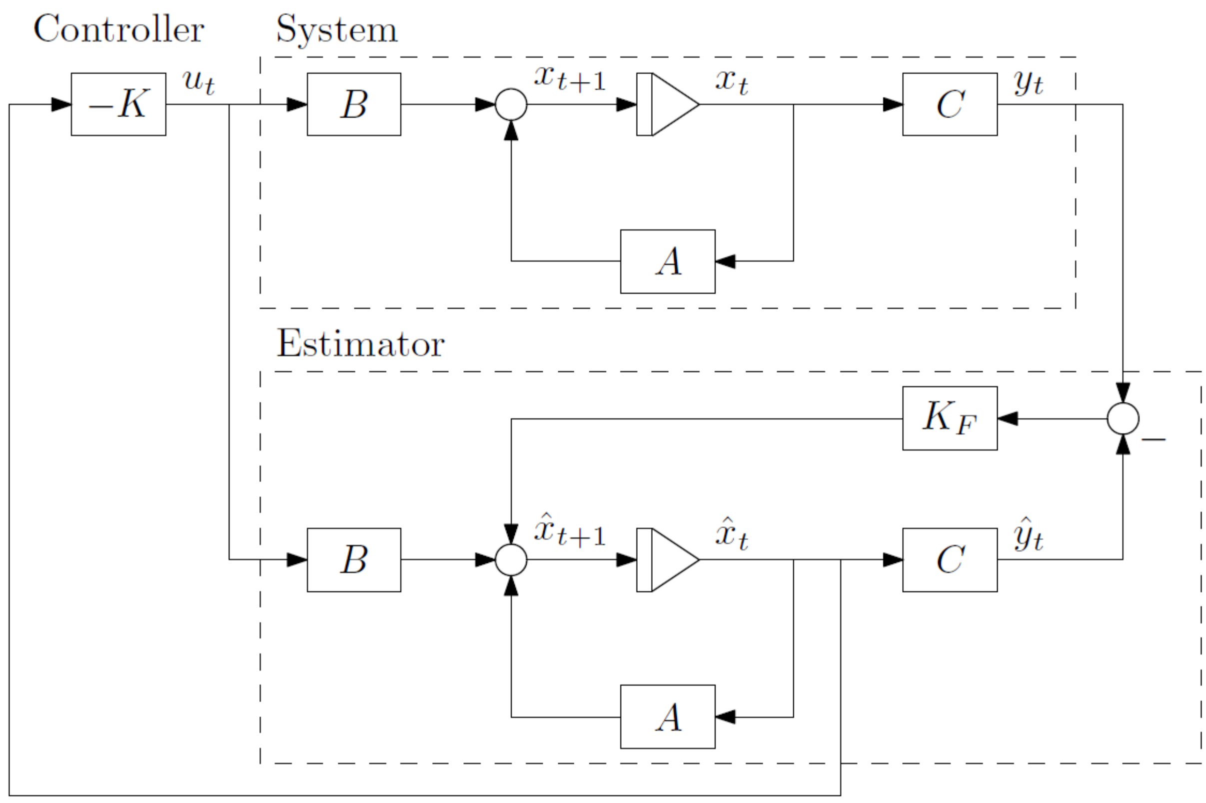
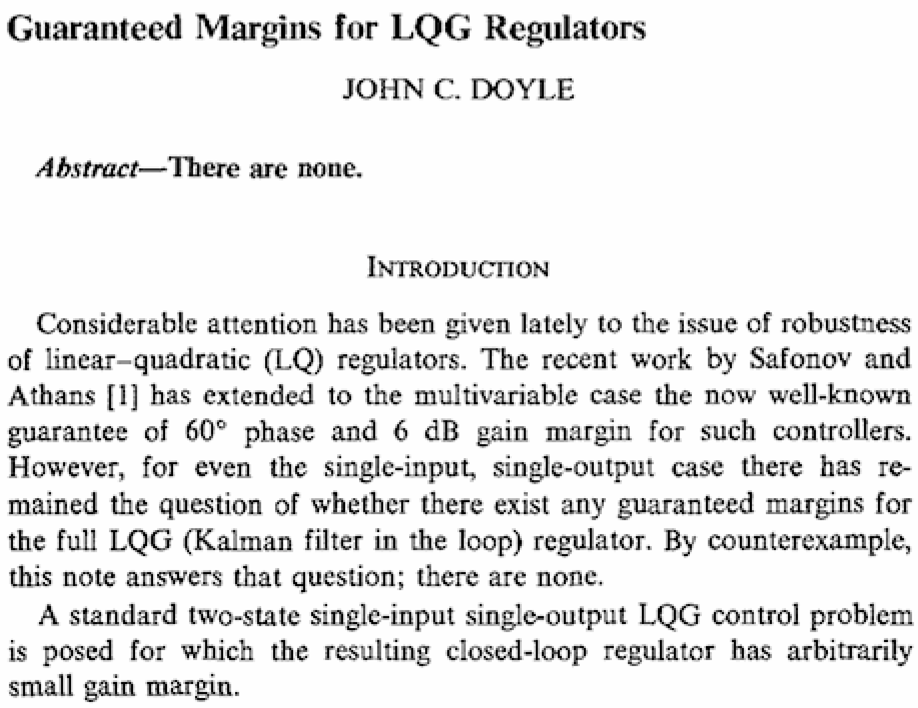


Figure 4.7: Structure of the LQG controller, i.e., output feedback LQ control.

LQG and robustness

- Doyle, 1978:



Guaranteed Margins for LQG Regulators

JOHN C. DOYLE

Abstract—There are none.

INTRODUCTION

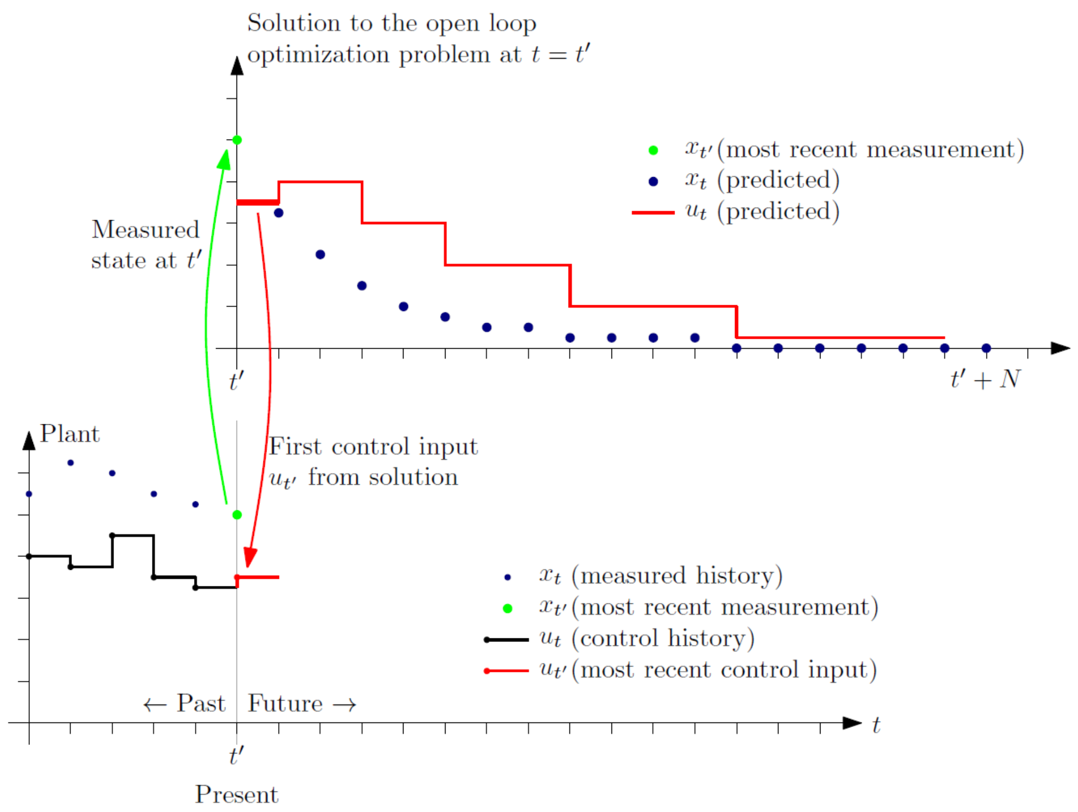
Considerable attention has been given lately to the issue of robustness of linear–quadratic (LQ) regulators. The recent work by Safonov and Athans [1] has extended to the multivariable case the now well-known guarantee of 60° phase and 6 dB gain margin for such controllers. However, for even the single-input, single-output case there has remained the question of whether there exist any guaranteed margins for the full LQG (Kalman filter in the loop) regulator. By counterexample, this note answers that question; there are none.

A standard two-state single-input single-output LQG control problem is posed for which the resulting closed-loop regulator has arbitrarily small gain margin.

- Lead to a lot of research in robust control in the 80's (and later), not topic of this course

Complexity reduction strategies in MPC

- Input blocking (or move blocking) – reduce number of QP variables



- "Incident points" – reduce number of QP constraints
  - Only check constraints at certain time instants, rather than at all times on horizon

Production optimization in process industry

"Real-time optimization", "Steady-state optimization"

- In a plant we (usually) have
  - Degrees of freedom (*u* – "valve openings")
  - States (*x* – "temperatures and pressures")
  - Disturbances (*d*)

- How to use the DOFs? What should we optimize?

$$E(x,u,d) = \text{cost of feed} + \text{cost of energy} - \text{value of product}$$

  - Often translates to throughput maximization – active constraints!

- Mathematical formulation

$$\begin{aligned} \min_{u} \quad & E(x,u,d) & \\ \text{subject to} \quad & f_{ss}(x,u,d) = 0 & \quad \text{(model equations)} \\ & g(x,u,d) \leq 0 & \quad \text{(process constraints)} \end{aligned}$$

Control and optimization hierarchy

Production optimization (RTO)

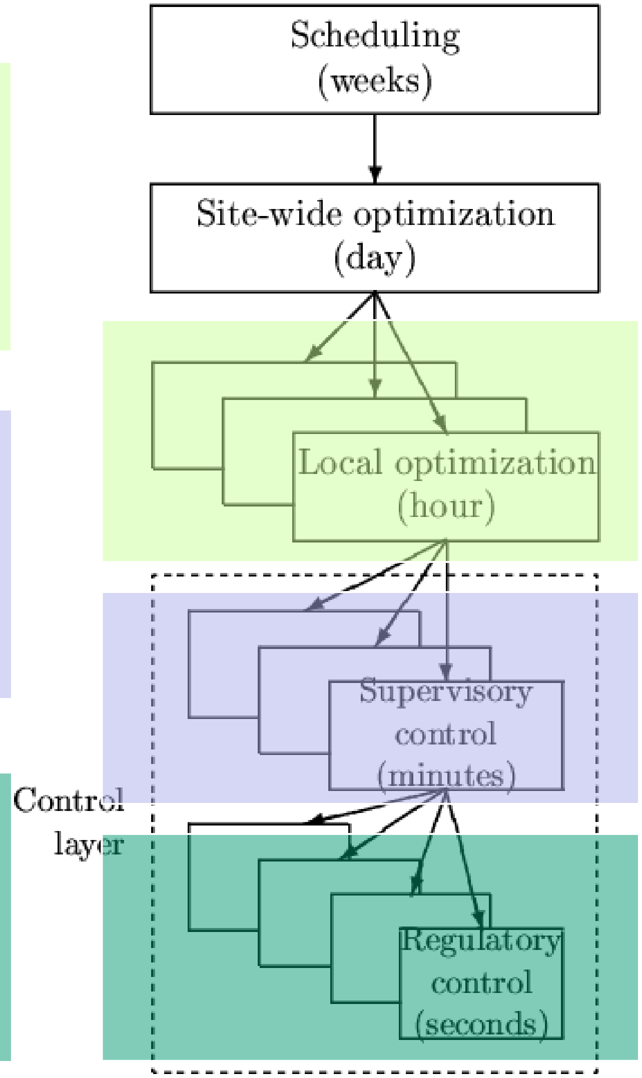
- Minimize *E* to obtain setpoints (reference trajectories) for control layer

Model Predictive Control (MPC)

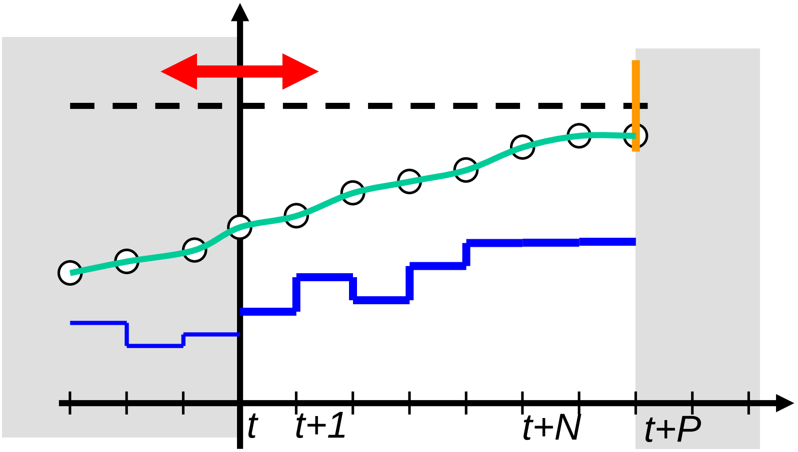
- Follow trajectories,
  keep process within constraints

Base control (PID)

- Stabilize integrating and unstable process parts



NMPC Discrete-time Open Loop Optimal Control Problem



Solve

$$\min_{\{\Delta u_k\}_{k=t}^{t+N-1}} J(x_t, \{\Delta u_k\}_{k=t}^{t+N-1})$$

subject to

$$\begin{aligned}
x_{k+1} &= f(x_k, u_k), & k &= t, \ldots, t+P-1 \\
u_k &= u_{k-1} + \Delta u_k & k &= t, \ldots, t+N-1 \\
\underline{\Delta u} \le \Delta u_k &\le \overline{\Delta u}, & k &= t, \ldots, t+N-1 \\
\underline{u} \le u_k &\le \overline{u}, & k &= t, \ldots, t+N-1 \\
\underline{z} \le z_k = g(x_k, u_k) &\le \overline{z}, & k &= t+1, \ldots, t+P
\end{aligned}$$

with

$$J(x_t, \{\Delta u_k\}_{k=t}^{t+N-1}) = \sum_{k=t+1}^{t+P} (z_k - r_k)^T Q (z_k - r_k) + \sum_{k=t}^{t+N-1} \Delta u_k^T S \Delta u_k$$

Incentives for MPC

- Regularity, handling constraints transparently (replacing logic), "better control" (optimizing, MIMO), reduced maintenance, …
- Also directly economy?

- But MPC follows setpoints given by RTO – how does MPC improve economy?
  - RTO:

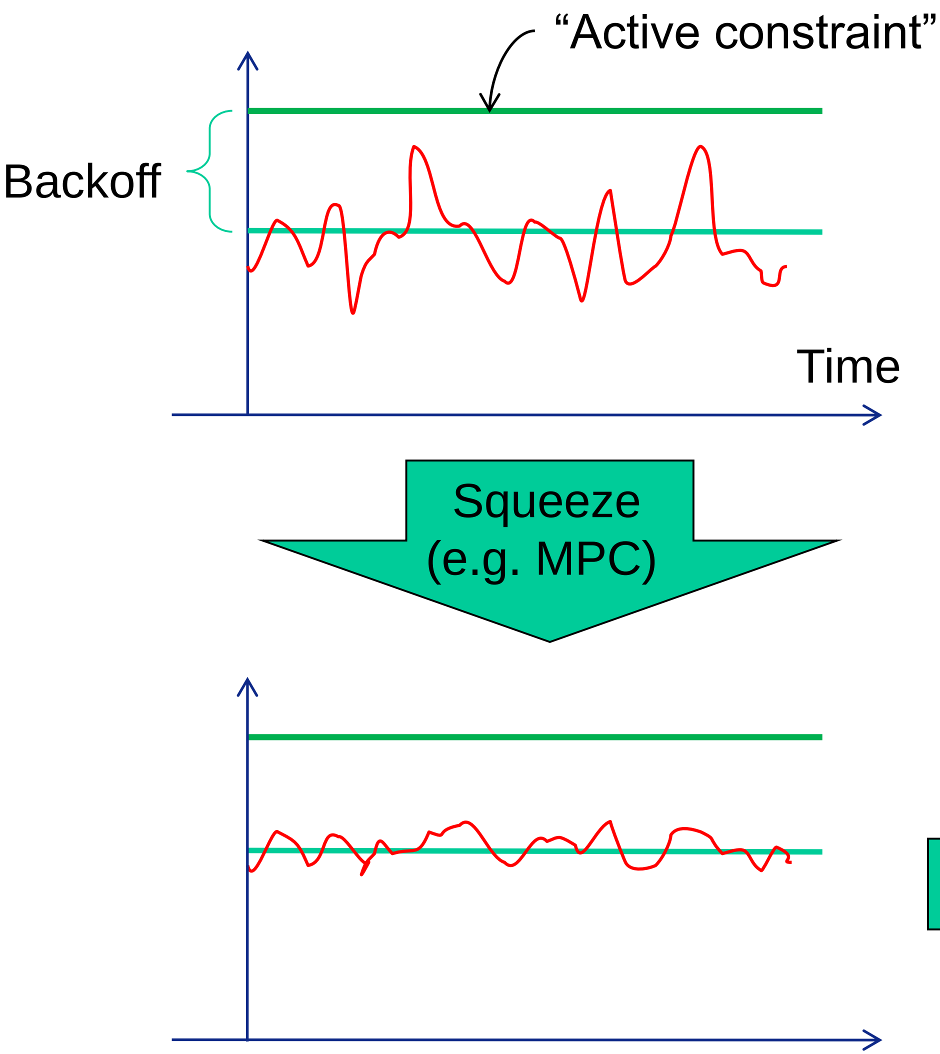
$$E(x,u,d) = \text{cost of feed} + \text{cost of energy} - \text{value of product}$$

  - MPC:

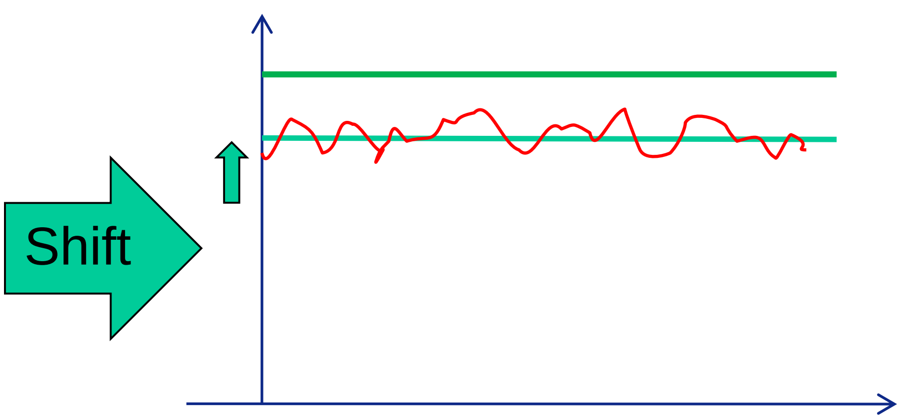
$$J(x_t, \{\Delta u_k\}_{k=t}^{t+N-1}) = \sum_{k=t+1}^{t+P} (z_k - r_k)^T Q (z_k - r_k) + \sum_{k=t}^{t+N-1} \Delta u_k^T S \Delta u_k$$

“Squeeze and shift”

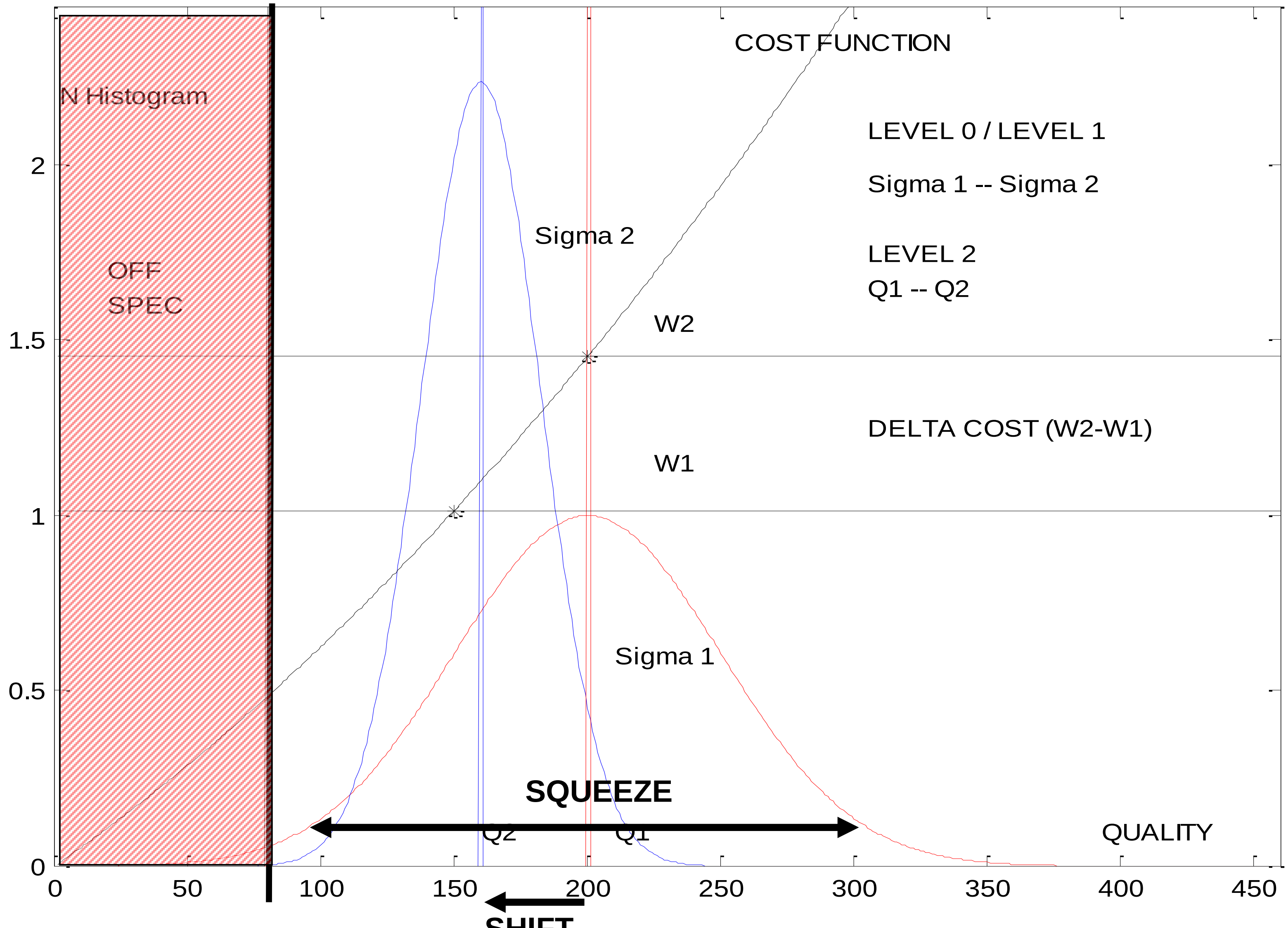
Traditional view of how MPC (or improved control) improves economy



Backoff = loss (lower throughput, off-spec product, ...)



Squeeze & shift



Economic optimization and MPC

(current research)

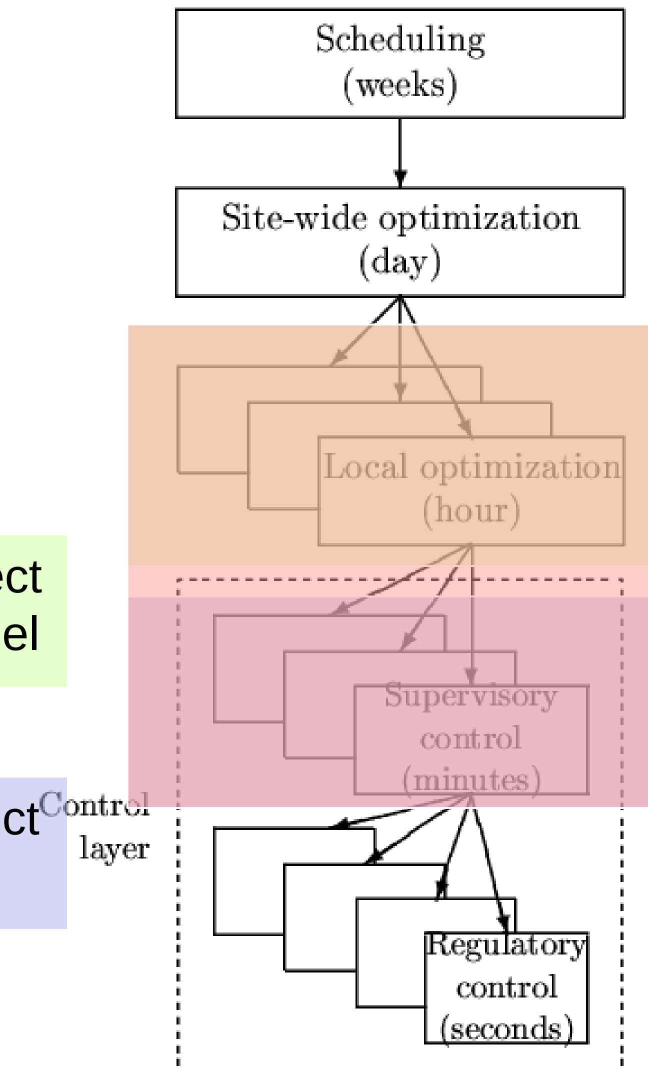
- “Squeeze and shift” – Indirect economic optimization by (N)MPC

- Can we use a more direct approach?
  - Dynamic RTO (= “**Economic MPC**”)

RTO: Minimize *E* subject to steady state model

D-RTO: Minimize *E* subject to dynamic model

MPC: Minimize *J* subject to dynamic model



RTO + NMPC = Economic MPC (D-RTO)

Opportunities with D-RTO

- Optimize performance also under transients/non-steady state (transients can be long!)
  - Disturbances
  - Grade changes
- Consistent models for performance optimization
- Consistent specification of process control objectives
- Optimization software becoming mature
- Leverage from industrial (N)MPC implementations

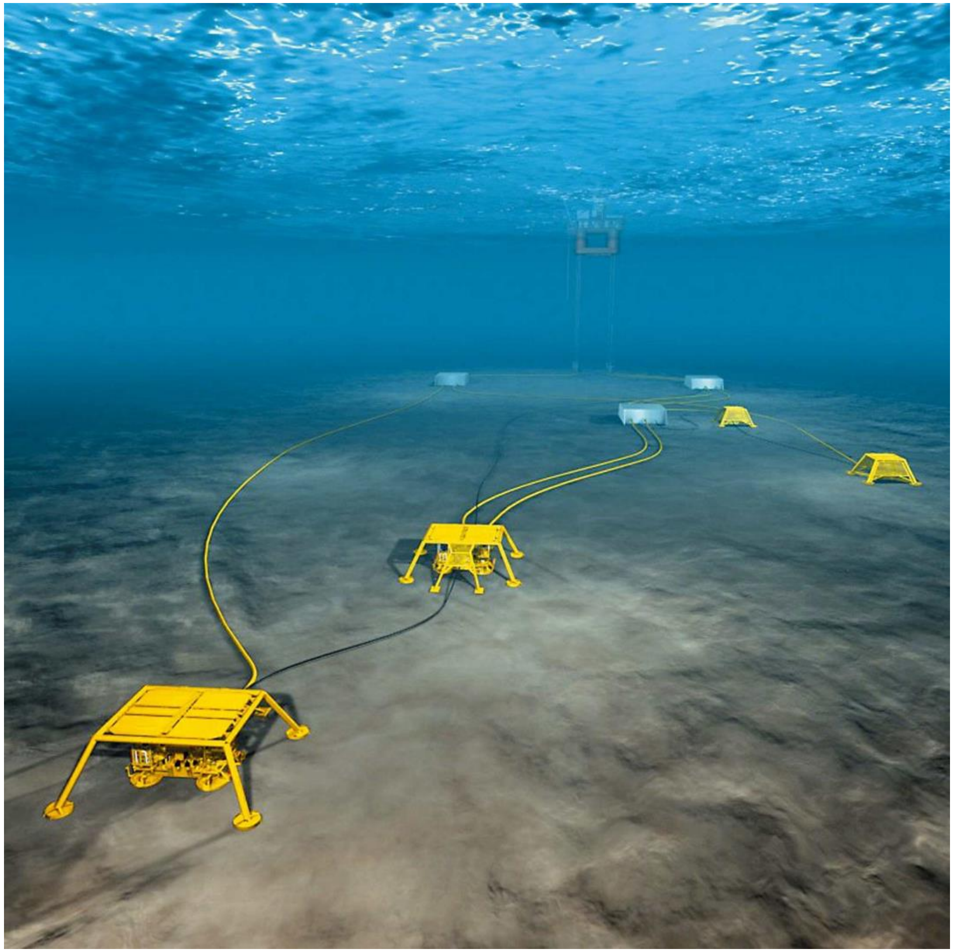
Challenges with D-RTO

- Managing complexity
- Large, dynamic, non-linear optimization problems have to be solved at high sample rates
- Developing theory&algorithms for tuning (control performance vs economy trade-off) and stability

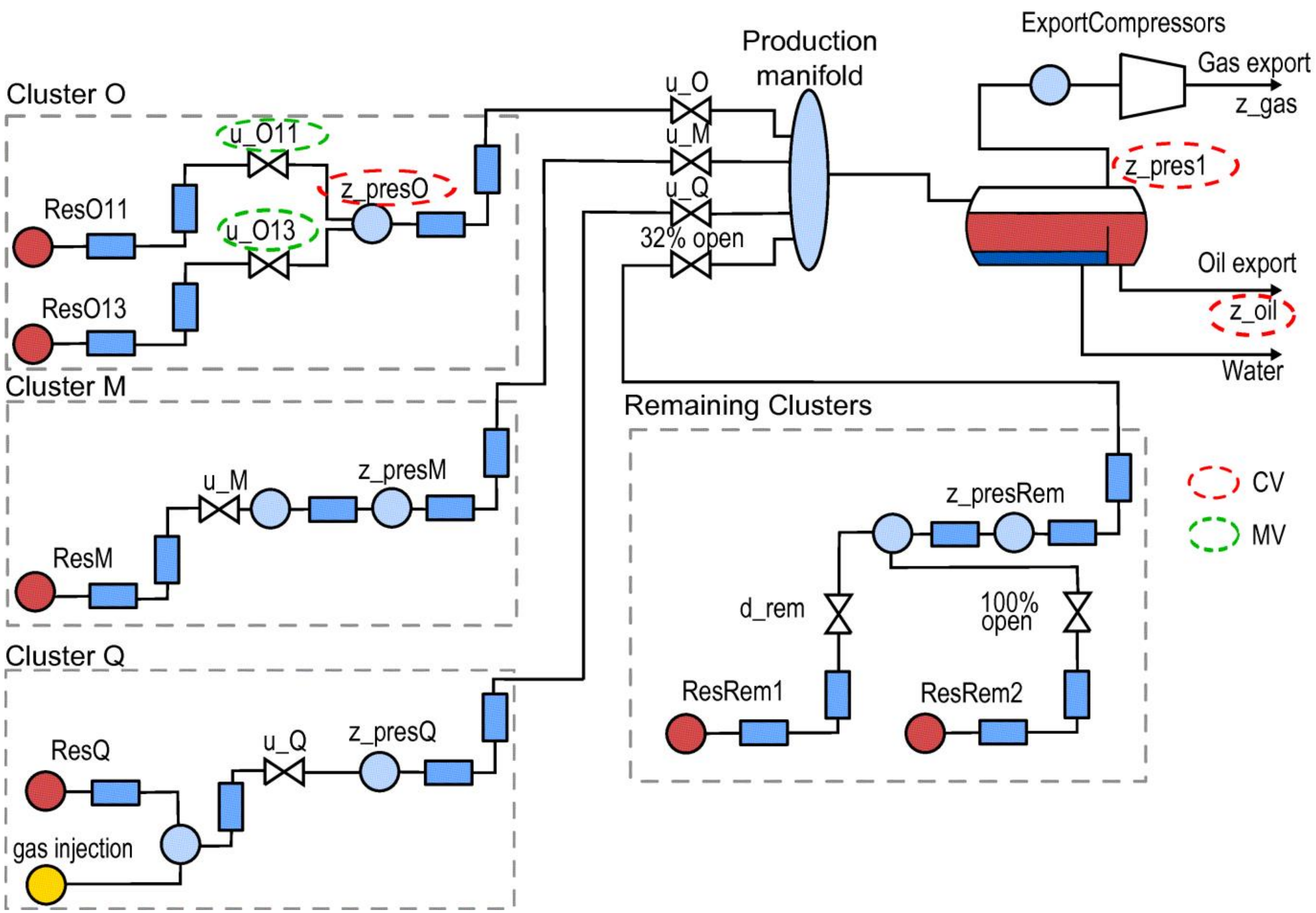
Simulation study: Offshore oil and gas production

Together with Anders Willersrud & Cybernetica AS

- Short-term production optimization: Maximize (e.g.) oil production, given a long-term recovery strategy
- "Adjust chokes of producing wells to obtain largest possible oil production"
- Wells have different gas-oil ratios (GOR)

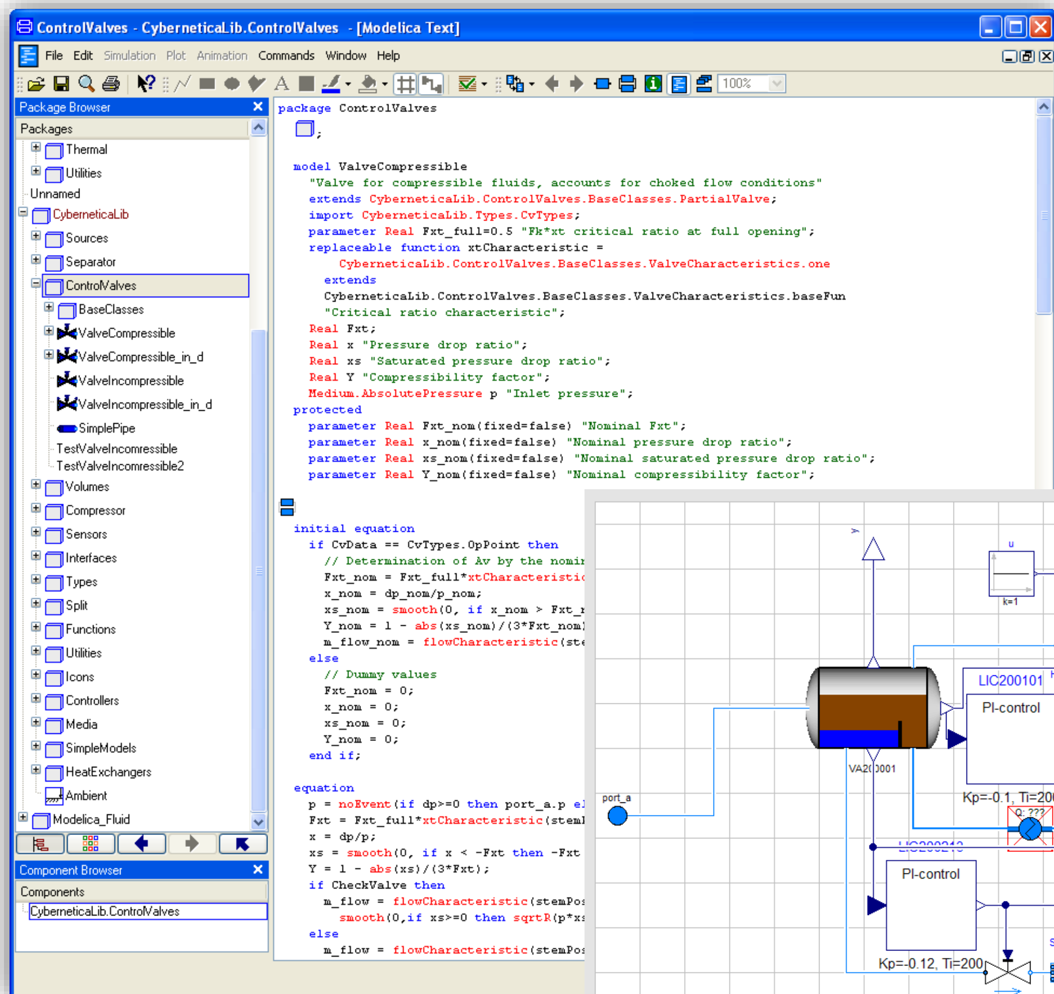


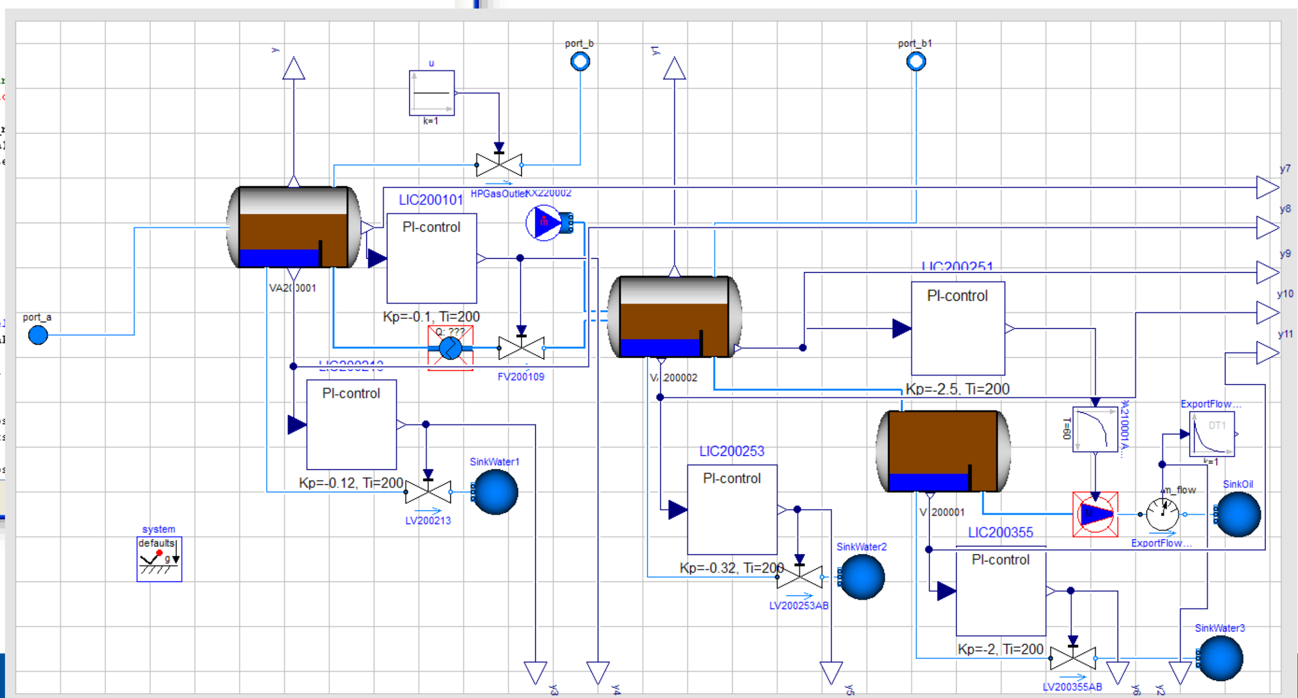
Simulation study



Modelling&Control

- Model developed in Modelica/Dymola,
  - 249 dynamic states
- NMPC tool:
  - Cybernetica CENIT with Dymola model
  - Using unreachable setpoint for z_{oil}





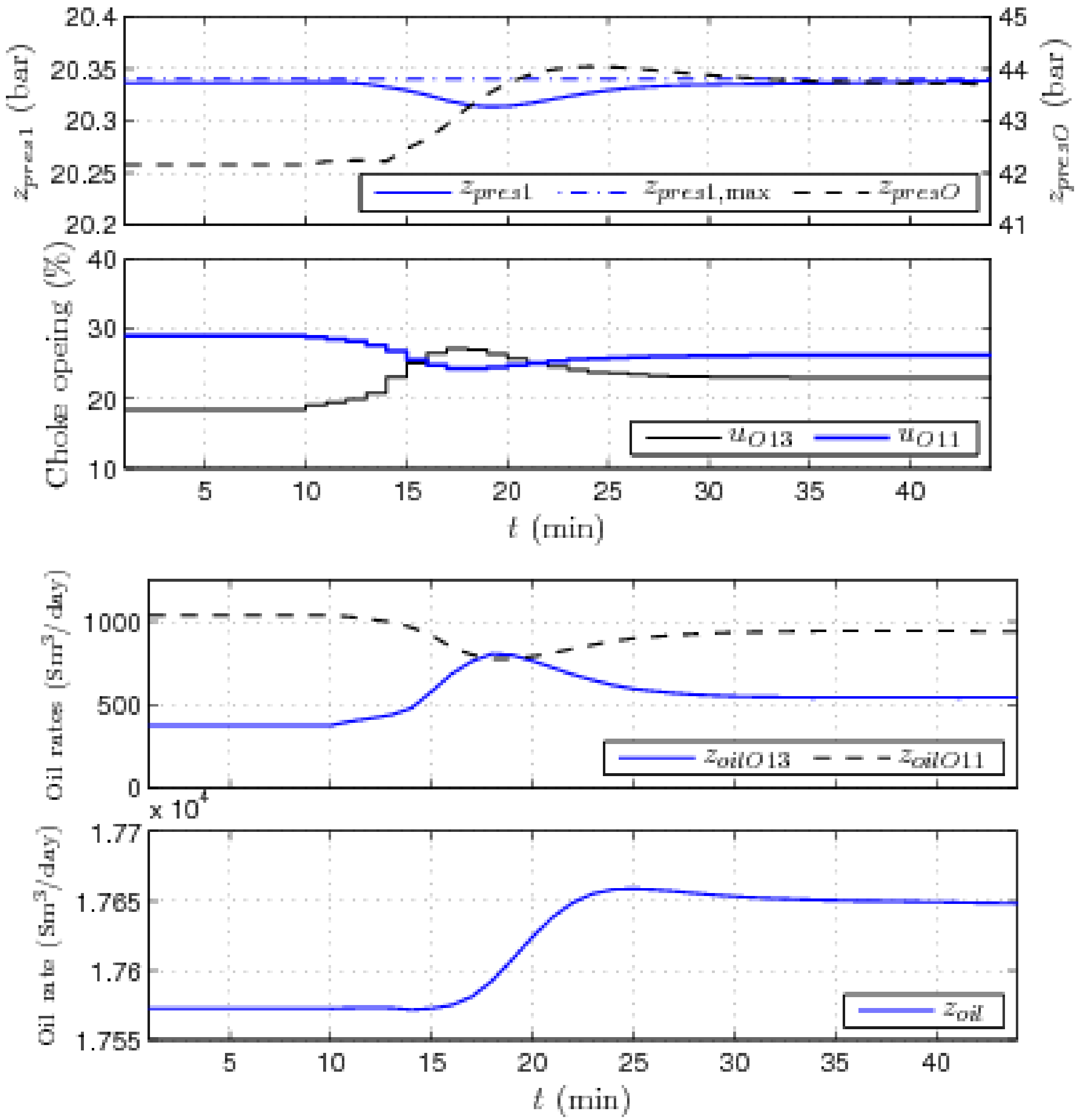
Production optimization

- We want to maximize oil production
  - Assumption: Bottleneck is gas processing capacity
  - Implication: Upper limit on inlet separator pressure is active constraint

- Due to different GOR in wells, it can be many combinations of choke openings that meet active constraint
  - Conventional solution: Use a “swing producer” to stay at active constraint

- We want to optimize online!

Results unreachable setpoints



- Initial state: Inlet separator constraint active
- NMPC start: 10 min.