**SEVENTH FRAMEWORK PROGRAMME**
**THEME – ICT**
**[Information and Communication Technologies]**

SEVENTH FRAMEWORK PROGRAMME

| | |
|---|---|
| **Contract Number:** | 223854 |
| **Project Title:** | Hierarchical and Distributed Model Predictive Control of Large-Scale Systems |
| **Project Acronym:** | HD-MPC |

HD–MPC

| | |
|---|---|
| **Deliverable Number:** | D8.3.1 |
| **Deliverable Type:** | Report |
| **Contractual Date of Delivery:** | September 1, 2011 |
| **Actual Date of Delivery:** | **August 31, 2011** |
| **Title of Deliverable:** | **Proceedings of the international HD-MPC workshop** |
| **Dissemination level:** | Public |
| **Workpackage contributing to the Deliverable:** | WP8 |
| **WP Leader:** | Moritz Diehl |
| **Partners:** | K.U.Leuven, TUD, EDF, POLIMI, RWTH, USE, UNC, UWM, SUPELEC, INOCSA |
| **Author(s):** | M. Diehl, R. Scattolini |

# Table of contents

## Project co-ordinator

*Name:* Bart De Schutter
*Address:* Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 Delft, The Netherlands
*Phone Number:* +31-15-2785113
*Fax Number:* +31-15-2786679
*E-mail:* b.deschutter@tudelft.nl

*Project web site:* `http://www.ict-hd-mpc.eu`

## Executive Summary

This deliverable contains the slides of the presentations given at the final HD-MPC Workshop which took place in Milan, Italy, on August 28, 2011 as a pre-congress workshop of the IFAC World Congress. The aim of this workshop was to present recent advances on hierarchical and distributed model predictive control, with the presentation of significant case studies

# Chapter 1

# General information on the workshop

### Title

Hierarchical and Distributed Model Predictive Control, Algorithms and Applications

### Organizers

Moritz Diehl (K.U.Leuven, Belgium) and Riccardo Scattolini (Politecnico di Milano, Italy)

### Date

Sunday, August 28, 2011

### Location

Milan, Italy

### Abstract

The workshop is aimed to present recent advances in the field of hierarchical and distributed control and estimation for large-scale complex networked systems. The main technique underlying all the proposed solutions is Model Predictive Control, in view of its flexibility in the definition of the control problem and of the possibility to include in the problem formulation state and control constraints.

Two mainstreams of recent research in the field will be covered. The first one refers to distributed optimization techniques for the solution of a centralized MPC problem. In this case, the goal is to decompose the optimization problem into a number of smaller and more easily tractable ones. In this framework, primal and dual approaches will be considered. The second approach relies on the solution of a number of local control problems with information exchange among them. In this case, the control algorithm itself, rather than its numerical solution, is distributed. Convergence properties of the methods can be achieved by resorting to robust MPC algorithms, where the uncertainties are related to the mutual influences among the subsystems. In the same way, it will be shown how to construct hierarchical control methods, where the hierarchical structure stems either from a structural decomposition of the system under control, or from its multi-level and multi time scale description.

A number of examples will be discussed to witness the potentialities of the methods. In particular, reference will be made to spatially distributed systems, such as irrigation channels and water networks. A complex application will deal with the control of a hydroelectric power valley, with five reservoirs,

three river reaches and a number of additional plants (ducts, turbines, generators, dams). The design of a hierarchical control scheme for Combined Cycle Power Plants will also be discussed, with particular emphasis to the problems related to the start-up phase, where particular attention must be posed to the thermal and mechanical stresses of the components, which strongly affect the life time of the plant.

## Agenda

- M. Diehl, R. Scattolini
  K.U.Leuven, Belgium and Politecnico di Milano, Italy
  Opening
- J. Rawlings
  University of Wisconsin, USA
  An overview of distributed MPC
- M. Diehl, A. Kozma, C. Savorgnan
  K.U.Leuven, Belgium
  Hierarchical and distributed optimization algorithms
- M. Farina, B. Picasso, R. Scattolini
  Politecnico di Milano, Italy
  Design of hierarchical and distributed MPC control systems with robustness tools
- J.M. Maestre, D. Limón, D. Muñoz de la Peña
  University of Seville, Spain
  Distributed MPC based on game theory
- W. Marquardt, H. Scheu
  RWTH Aachen, Germany
  Distributed model predictive control by primal decomposition
- B. De Schutter
  Delft University of Technology, The Netherlands
  Hierarchical MPC with applications in transportation and infrastructure networks
- D. Faille, F. Davelaar
  EDF, France
  Hierarchical and distributed control of a hydro power valley
- A. Tica, H. Guéguen, D. Dumur, D. Faille, F. Davelaar
  Supélec and EDF, France
  Application to start-up of combined-cycle power plant
- L. Sánchez, M.A. Ridao
  INOCSA and University of Seville, Spain
  Distributed control of irrigation canals
- B. De Schutter
  Delft University of Technology, The Netherlands
  Closing

# Chapter 2

# Slides of the presentations

## 2.1 Opening (M. Diehl, R. Scattolini)

## Hierarchical and Distributed Model Predictive Control, Algorithms and Applications

Organizers: *M. Diehl, R. Scattolini*
*Sunday, August 28, 2011*

MILANO 2011 IFAC

HD-MPC

IFAC 2011

## FP7 ICT Project HD-MPC, 2008-2011

2

Hierarchical and Distributed Model Predictive Control – Riccardo Scattolini, Moritz Diehl HD-MPC

## Agenda - morning

3

9:00 *M. Diehl, R. Scattolini*
KU Leuven, Politecnico di Milano
**Opening**

9:15 *J. Rawlings*
University of Winsconsin
**An overview of distributed MPC**

10:00 *M. Diehl, A. Kozma, C. Savorgnan*
KU Leuven
**Hierarchical and Distributed Optimization Algorithms**

10:30 Break

11:00 *M. Farina, B. Picasso, R. Scattolini*
Politecnico di Milano
**Design of hierarchical and distributed MPC control systems with robustness tools**

11:30 *D. Muñoz de la Peña, J.M. Maestre and D. Limón*
University of Seville
**Distributed MPC based on game theory**

12:00 *M. Diehl*
KU Leuven
**Interactive Session on Methods of Hierarchical and Distributed MPC**

Hierarchical and Distributed Model Predictive Control, Algorithms and Applications HD-MPC

## Agenda - afternoon

4

14:00 *W. Marquardt, H. Scheu*
RWTH, Aachen
**Distributed model predictive control by primal decomposition**

14:30 *B. De Schutter*
TUDelft
**Hierachical MPC with applications in transportation and infrastructure networks**

15:00 *D. Faille, F. Davelaar*
EDF
**Hierarchical and distributed control of a hydro power valley**

15:30 Break

16:00 *A.Tica, H Guéguen, D. Dumur, D. Faille, F Davelaar*
Supelec and EDF
**Application to start-up of Combined-Cycle Power Plant**

16:30 *L. Sanchez, M. A. Ridao*
INOCSA and University of Seville
**Distributed Control of Irrigation Canals**

17:00 *B. De Schutter*
TUDelft
**Closing**

Hierarchical and Distributed Model Predictive Control, Algorithms and Applications HD-MPC

## 2.2 An overview of distributed MPC (J. Rawlings)

## An overview of distributed model predictive control (MPC)

James B. Rawlings

Department of Chemical and Biological Engineering

THE UNIVERSITY WISCONSIN MADISON

August 28, 2011
IFAC Workshop: Hierarchical and Distributed Model Predictive Control, Algorithms and Applications
Milano, Italy

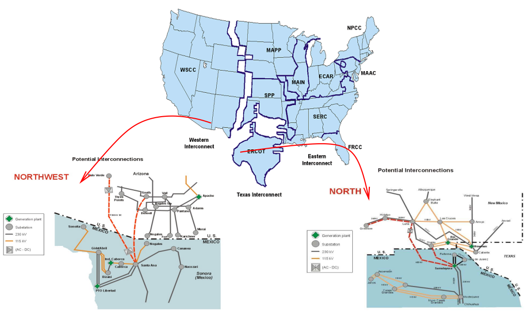


## Outline



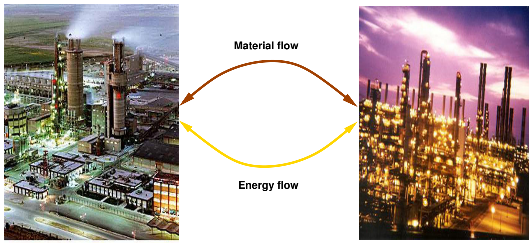


## Electrical power distribution






## Chemical plant integration






## MPC at the large scale

### Decentralized Control

- Most large-scale systems consist of networks of interconnected/interacting subsystems
  - Chemical plants, electrical power grids, water distribution networks, . . .
- Traditional approach: Decentralized control
  - Wealth of literature from the early 1970's on improved decentralized control [a]
  - Well known that poor performance may result if the interconnections are not negligible

[a](Sandell Jr. et al., 1978; Šiljak, 1991; Lunze, 1992)



## MPC at the large scale

### Centralized Control

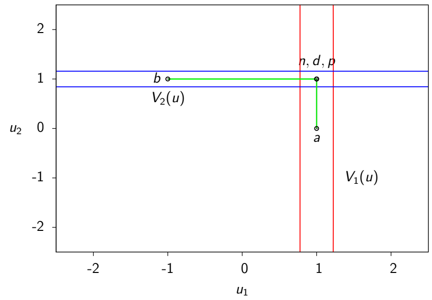
- Steady increase in available computing power has provided the opportunity for centralized control
- Coordinated control: Distributed optimization to achieve fast solution of centralized control (Necoara et al., 2008; Cheng et al., 2007)
- Most practitioners view centralized control of large, networked systems as impractical and unrealistic
- A divide and conquer strategy is essential for control of large, networked systems (Ho, 2005)
- Centralized control: A benchmark for comparing and assessing distributed controllers

## Nomenclature: consider two interacting units

| | |
|---|---|
| Objective functions | $V_1(u_1, u_2), \; V_2(u_1, u_2)$ |
| and | $V(u_1, u_2) = w_1 V_1(u_1, u_2) + w_2 V_2(u_1, u_2)$ |
| decision variables for units | $u_1 \in \Omega_1, \quad u_2 \in \Omega_2$ |
| Decentralized Control | $\min_{u_1 \in \Omega_1} \widetilde{V}_1(u_1) \quad \min_{u_2 \in \Omega_2} \widetilde{V}_2(u_2)$ |
| Noncooperative Control | $\min_{u_1 \in \Omega_1} V_1(u_1, u_2) \quad \min_{u_2 \in \Omega_2} V_2(u_1, u_2)$ |
| (Nash equilibrium) | |
| Cooperative Control | $\min_{u_1 \in \Omega_1} V(u_1, u_2) \quad \min_{u_2 \in \Omega_2} V(u_1, u_2)$ |
| (Pareto optimal) | |
| Centralized Control | $\min_{u_1, u_2 \in \Omega_1 \times \Omega_2} V(u_1, u_2)$ |
| (Pareto optimal) | |

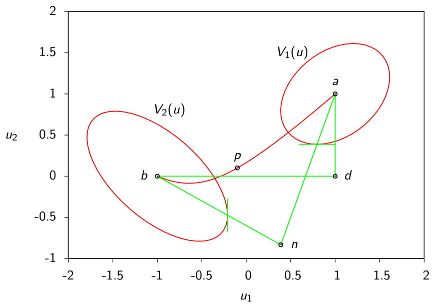


## Noninteracting systems



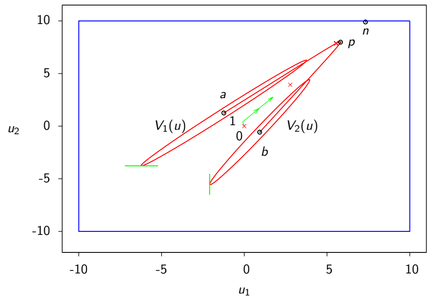



## Moderately interacting systems



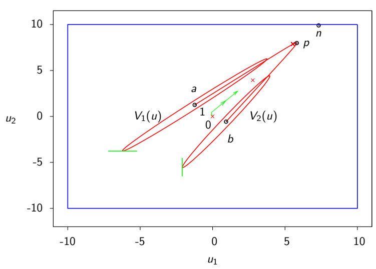



## Geometry of cooperative vs. noncooperative MPC






## Plantwide suboptimal MPC




- Early termination of optimization gives suboptimal plantwide feedback
- Use suboptimal MPC theory to prove stability



## Plantwide suboptimal MPC

Consider closed-loop system augmented with input trajectory

$$\begin{pmatrix} x^+ \\ \mathbf{u}^+ \end{pmatrix} = \begin{pmatrix} Ax + Bu \\ g(x, \mathbf{u}) \end{pmatrix}$$

- Function $g(\cdot)$ returns suboptimal choice
- Stability of augmented system is established by Lyapunov function

$$a\,|(x, \mathbf{u})|^2 \leq V(x, \mathbf{u}) \leq b\,|(x, \mathbf{u})|^2$$
$$V(x^+, \mathbf{u}^+) - V(x, \mathbf{u}) \leq -c\,|(x, u)|^2$$

- Adding constraint establishes closed-loop stability of the origin for all $\mathbf{u}$[1]

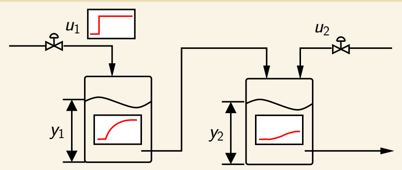
$$|\mathbf{u}| \leq d\,|x| \quad x \in \mathbb{B}_r, r > 0$$

- Cooperative optimization satisfies these properties for plantwide objective function $V(x, \mathbf{u})$

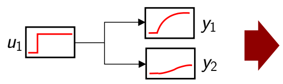
[1](Rawlings and Mayne, 2009, pp.418-420)

## Modeling

### Plantwide step response




- Interaction models found by decentralized identification[2]




$$x_{11}^+ = A_{11}x_{11} + B_{11}u_1$$
$$x_{21}^+ = A_{21}x_{21} + B_{21}u_1$$

[2] Gudi and Rawlings (2006)



## Modeling

Consider the linearized physical model

$$x^+ = Ax + B_1u_1 + B_2u_2 \qquad y_1 = C_1x, \quad y_2 = C_2x$$

- Kalman canonical form of the triple $(A, B_j, C_i)$

$$\begin{bmatrix} z_{ij}^{oc} \\ z_{ij}^{\bar{o}c} \\ z_{ij}^{o\bar{c}} \\ z_{ij}^{\bar{o}\bar{c}} \end{bmatrix}^+ = \begin{bmatrix} A_{ij}^{oc} & 0 & A_{ij}^{oc\bar{c}} & 0 \\ A_{ij}^{\bar{o}oc} & A_{ij}^{\bar{o}c} & A_{ij}^{\bar{o}co\bar{c}} & A_{ij}^{\bar{o}c\bar{c}} \\ 0 & 0 & A_{ij}^{o\bar{c}} & 0 \\ 0 & 0 & A_{ij}^{\bar{o}\bar{c}o} & A_{ij}^{\bar{o}\bar{c}} \end{bmatrix} \begin{bmatrix} z_{ij}^{oc} \\ z_{ij}^{\bar{o}c} \\ z_{ij}^{o\bar{c}} \\ z_{ij}^{\bar{o}\bar{c}} \end{bmatrix} + \begin{bmatrix} B_{ij}^{oc} \\ B_{ij}^{\bar{o}c} \\ 0 \\ 0 \end{bmatrix} u_j$$

$$y_{ij} = \begin{bmatrix} C_{ij}^{oc} & 0 & C_{ij}^{o\bar{c}} & 0 \end{bmatrix} \begin{bmatrix} z_{ij}^{oc} \\ z_{ij}^{\bar{o}c} \\ z_{ij}^{o\bar{c}} \\ z_{ij}^{\bar{o}\bar{c}} \end{bmatrix} \qquad y_i = \sum_j y_{ij}$$

- Interaction models

$$A_{ij} \leftarrow A_{ij}^{oc} \quad B_{ij} \leftarrow B_{ij}^{oc} \quad C_{ij} \leftarrow C_{ij}^{oc} \quad x_{ij} \leftarrow z_{ij}^{oc}$$



## Unstable modes

For unstable systems, we zero the unstable modes with terminal constraints.

- For subsystem 1

$$S_{11}^{u\,\prime} x_{11}(N) = 0 \qquad S_{21}^{u\,\prime} x_{21}(N) = 0$$

- To ensure terminal constraint feasibility for all $x$, we require $(\underline{A}_1, \underline{B}_1)$ stabilizable

$$\underline{A}_1 = \begin{bmatrix} A_{11} & \\ & A_{21} \end{bmatrix} \qquad \underline{B}_1 = \begin{bmatrix} B_{11} \\ B_{21} \end{bmatrix}$$

- For output feedback, we require $(A_1, C_1)$ detectable

$$A_1 = \begin{bmatrix} A_{11} & \\ & A_{12} \end{bmatrix} \qquad C_1 = \begin{bmatrix} C_{11} & C_{12} \end{bmatrix}$$

- Similar requirements for other subsystem



## Output feedback

Consider augmented system perturbed by stable estimator

$$\begin{pmatrix} \hat{x}^+ \\ \mathbf{u}^+ \\ e^+ \end{pmatrix} = \begin{pmatrix} A\hat{x} + B\mathbf{u} + Le \\ g(\hat{x}, \mathbf{u}, e) \\ A_L e \end{pmatrix}$$

- Stable estimator error implies Lyapunov function

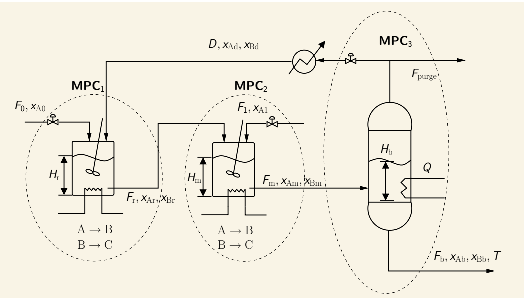
$$\bar{a}\,|e| \leq J(e) \leq \bar{b}\,|e|$$
$$J(e^+) - J(e) \leq -\bar{c}\,|e|$$

- Stability of perturbed system established by Lyapunov function

$$W(\hat{x}, \mathbf{u}, e) = V(\hat{x}, \mathbf{u}) + J(e)$$

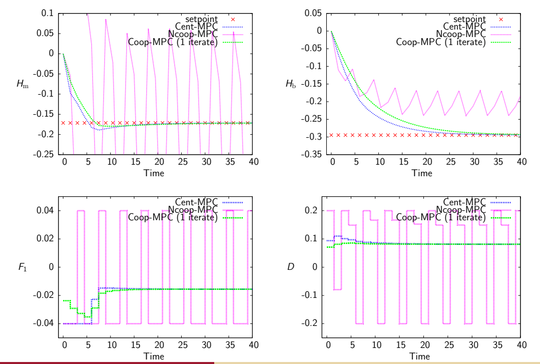


## Two reactors with separation and recycle






## Two reactors with separation and recycle



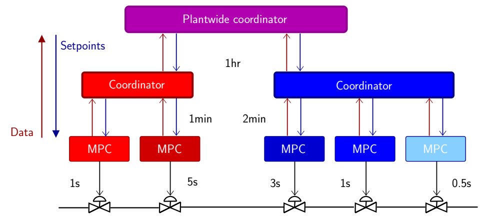
## Two reactors with separation and recycle

Performance comparison

| | Cost ($\times 10^{-2}$) | Performance loss |
|---|---|---|
| Centralized MPC | 1.75 | 0 |
| Decentralized MPC | $\infty$ | $\infty$ |
| Noncooperative MPC | $\infty$ | $\infty$ |
| Cooperative MPC (1 iterate) | 2.2 | 25.7% |
| Cooperative MPC (10 iterates) | 1.84 | 5% |



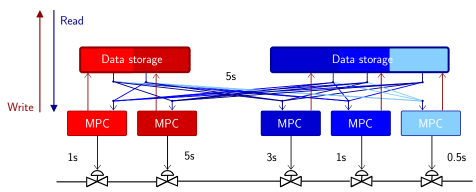
## Traditional hierarchical MPC




- Multiple dynamical time scales in plant
- Data and setpoints are exchanged on chosen scale
- Optimization performed at each layer



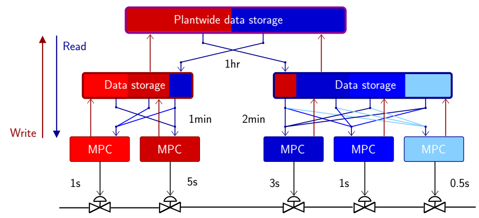
## Cooperative MPC data exchange




- All data exchanged plantwide
- Data exchange at each controller execution



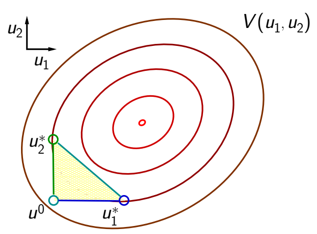
## Cooperative hierarchical MPC




- Optimization at MPC layer only
- Only subset of data exchanged plantwide
- Data exchanged at chosen time scale



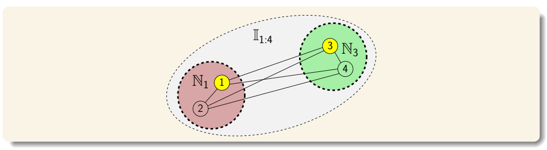
## Motivating the hierarchical optimization




- Any point in the triangle decreases the cost of $V$



## Hierarchical optimization




Consider the optimization

$$\min_u V(u_1, u_2, u_3, u_4)$$

We group the variables into two neighborhoods

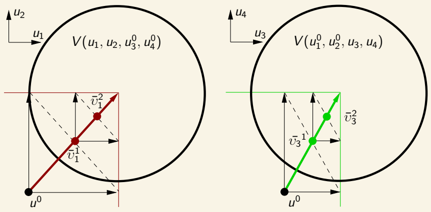
- $\mathbb{N}_1 = \{1, 2\}$ and $\mathbb{N}_2 = \{3, 4\}$

We solve the optimization in a distributed fashion

- suboptimizations utilize the latest iterate only from variables in their neighborhood

## Hierarchical optimization

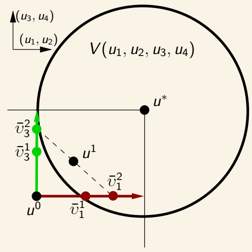
### Suboptimizations






## Hierarchical optimization

### Overall





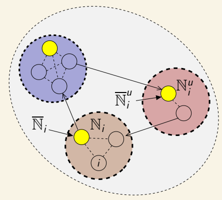

## Two reactors with separation and recycle

### Performance comparison

| | Cost | Performance loss |
|---|---|---|
| Centralized | 0.95 | - |
| Cooperative (1 iterate) | 1.60 | 68% |
| $N_s = 1$ | 1.633 | 71% |
| $N_s = 2$ | 1.646 | 73% |
| $N_s = 5$ | 1.661 | 75% |
| $N_s = 10$ | 1.669 | 76% |
| $N_s = 25$ | 1.670 | 76% |
| $N_s = 50$ | 1.670 | 76% |



## Reducing communication



We define a leader in each neighborhood and a graph between the leaders



## Reducing communication

We define the state propagation in the following way

$$x_i(k) = \bar{A}_{ii}^k x_i(0) + \sum_{\tau=0}^{k-1} \sum_{j \in \mathbb{N}_i} \bar{A}_{ii}^{k-\tau-1} \bar{B}_{ij} u_j(\tau) + \sum_{\tau=0}^{k-1} \sum_{l \in \mathbb{L}} \sum_{s \in \mathbb{I}_{1:M} \setminus l} \bar{A}_{is}^{[k-\tau-1]} \bar{A}_{sl} \alpha_l(\tau)$$

such that

$$\alpha_i^+ = \bar{A}_{ii}\alpha_i + \sum_{j \in \mathbb{N}_i} \bar{B}_{ij} u_j$$

- $\alpha$ is defined only for the leaders
- Computation requires only information from within the neighborhood and from other leaders



## Nonlinear Distributed MPC

We assume the model is of the form

$$\frac{dx_1}{dt} = f_1(x_1, x_2, u_1, u_2)$$
$$y_1 = C_1 x_1$$
$$\frac{dx_2}{dt} = f_2(x_1, x_2, u_1, u_2)$$
$$y_2 = C_2 x_2$$

Given these physical system models of the subsystems, the overall plant model is

$$\frac{dx}{dt} = f(x, u)$$
$$y = Cx$$

in which

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \quad C = \begin{bmatrix} C_1 & \\ & C_2 \end{bmatrix}$$

## Nonconvexity

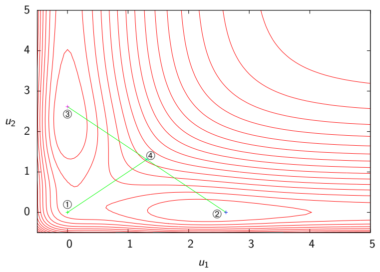



Figure: Cost contours for a two-player, nonconvex game; cost *increases* for the convex combination of the two players' optimal points.



## Requirements for distributed, nonlinear control

- Must handle nonconvex objectives
- Two criteria in design:
  1. the optimizers should *not* rely on a central coordinator
  2. the exchange of information between the subsystems and the iteration of the subsystem optimizations should be able to terminate before convergence without compromising closed-loop properties.



## Distributed nonconvex optimization

- Consider the optimization

$$\min_u V(u) \quad \text{s.t.} \quad u \in \mathbb{U}$$

- We require approximate solutions to the following suboptimizations at iterate $p \geq 0$ for all $i \in \mathbb{I}_{1:M}$

$$\overline{u}_i^p = \arg\min_{u_i \in \mathbb{U}_i} V(u_i, u_{-i}^p)$$

in which $u_{-i} = (u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_M)$.
- Define the step $\upsilon_i^p = \overline{u}_i^p - u_i^p$.



## Algorithm

- To choose the stepsize $\alpha_i^p$, each suboptimizer initializes the stepsize[3] with $\overline{\alpha}_i$

$$V(u^p) - V(u_i^p + \alpha_i^p \upsilon_i^p, u_{-i}^p) \geq -\sigma \alpha_i^p \nabla_i V(u^p)' \upsilon_i^p$$

in which $\sigma \in (0, 1)$.
- After all suboptimizers finish the backtracking process, they exchange steps. Each suboptimizer forms a candidate step

$$u_i^{p+1} = u_i^p + w_i \alpha_i^p \upsilon_i^p \quad \forall i \in \mathbb{I}_{1:M}$$

[3] Armijo rule: (Bertsekas, 1999, p.230)



## Algorithm

- Check the following inequality, which tests if $V(u^p)$ is convex-like

$$V(u^{p+1}) \leq \sum_{i \in \mathbb{I}_{1:M}} w_i V(u_i^p + \alpha_i^p \upsilon_i^p, u_{-i}^p) \qquad (1)$$

in which $\sum_{i \in \mathbb{I}_{1:M}} w_i = 1$ and $w_i > 0$ for all $i \in \mathbb{I}_{1:M}$.
- If the condition above is not satisfied, then we find the direction with the worst cost improvement

$$i_{\max} = \arg\max_i \{V(u_i^p + \alpha_i^p \upsilon_i^p, u_{-i}^p)\}$$

and eliminate this direction by setting $w_{i_{\max}}$ to zero and repartitioning the remaining $w_i$ so that they sum to 1.
- *At worst, condition* (1) *is satisfied with one direction only.*



## Distributed nonconvex optimization — Properties

**Lemma (Feasibility)**

*Given a feasible initial condition, the iterates $u^p$ are feasible for all $p \geq 0$.*

**Lemma (Objective decrease)**

*The objective function decreases at every iterate, that is, $V(u^{p+1}) \leq V(u^p)$.*

**Lemma (Convergence)**

*Every accumulation point of the sequence $\{u^p\}$ is stationary.*

## Distributed nonconvex optimization

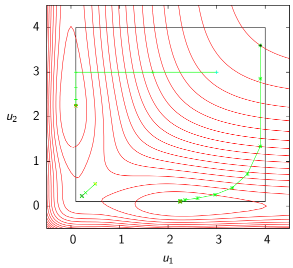



Figure: Nonconvex function optimized with Distributed nonconvex optimization algorithm



## A nonlinear example

- Consider the unstable nonlinear system

$$x_1^+ = x_1^2 + x_2 + u_1^3 + u_2$$
$$x_2^+ = x_1 + x_2^2 + u_1 + u_2^3$$

with initial condition $(x_1, x_2) = (3, -3)$.

- For this example, we use the stage cost

$$\ell_1(x_1, u_1) = \frac{1}{2}(x_1' Q_1 x_1 + u_1' R_1 u_1)$$
$$\ell_2(x_2, u_2) = \frac{1}{2}(x_2' Q_2 x_2 + u_2' R_2 u_2)$$

- For the simulation we choose the parameters

$$Q = I \quad R = I \quad N = 2 \quad \bar{p} = 3 \quad \mathbb{U}_i = [-2.5, 2.5] \quad \forall i \in \mathbb{I}_{1:2}$$



## Distributed nonlinear cooperative control

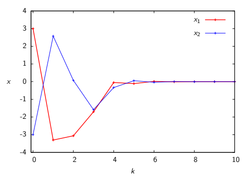


Figure: State trajectory ($\bar{p} = 3$)

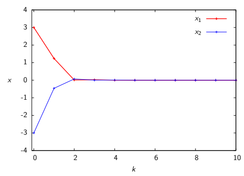


Figure: Centralized state trajectory ($\bar{p} = 10$)



## Distributed nonlinear cooperative control

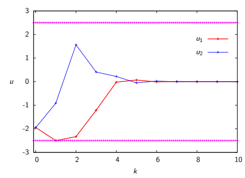


Figure: Input trajectory ($\bar{p} = 3$)

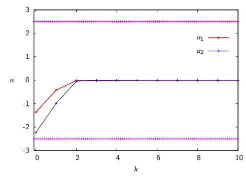


Figure: Centralized input trajectory ($\bar{p} = 10$)



## Distributed nonlinear cooperative control

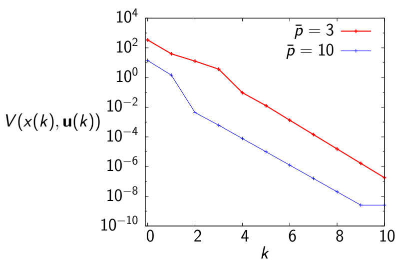



Figure: Open-loop cost to go versus time on the closed-loop trajectory for different numbers of iterations.



## Distributed nonlinear cooperative control

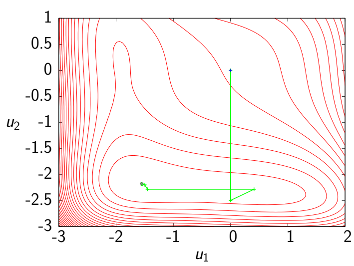



Figure: Contours of $V$ with $N = 1$ for $k = 0$ with $(x_1(0), x_2(0)) = (3, -3)$. Iterations of the subsystem controllers with initial condition $(u_1^0, u_2^0) = (0, 0)$.

## Why study robustness of *suboptimal* MPC?

- *Cooperative*, distributed MPC is a special case of *suboptimal* MPC. Anything we establish about suboptimal MPC can be applied to cooperative, distributed MPC (and optimal MPC!)
- Suboptimal MPC has an interesting feature: a nonunique, point-to-set control law $u \in \kappa_N(x)$.
- *Optimal* solution of nonconvex

$$\mathbb{P}_N(x): \qquad \min_{\mathbf{u} \in \mathcal{U}_N} V_N(x, \mathbf{u})$$

cannot be computed online for *any* nonlinear model. Practitioners implement only suboptimal MPC.

- We should know something about its inherent robustness properties.[4]

[4] Pannocchia et al. (2011)



## For suboptimal MPC; again, the basic MPC setup

- The system model

$$x^+ = f(x, u) \tag{2}$$

- State and input constraints

$$x(k) \in \mathbb{X}, \qquad u(k) \in \mathbb{U} \qquad \text{for all } k \in \mathbb{I}_{\geq 0}$$

- Terminal constraint (and penalty)

$$\phi(N; x, \mathbf{u}) \in \mathbb{X}_f \subseteq \mathbb{X}$$



## Cost function and control problem

- For any state $x \in \mathbb{R}^n$ and input sequence $\mathbf{u} \in \mathbb{U}^N$, we define

$$V_N(x, \mathbf{u}) = \sum_{k=0}^{N-1} \ell(\phi(k; x, \mathbf{u}), u(k)) + V_f(\phi(N; x, \mathbf{u}))$$

- $\ell(x, u)$ is the stage cost; $V_f(x(N))$ is the terminal cost
- Consider the finite horizon optimal control problem

$$\mathbb{P}_N(x): \qquad \min_{\mathbf{u} \in \mathcal{U}_N} V_N(x, \mathbf{u})$$



## Suboptimal MPC

- Rather then solving $\mathbb{P}_N(x)$ *exactly*, we consider using any (unspecified) suboptimal algorithm having the following properties.
- Let $\mathbf{u} \in \mathcal{U}_N(x)$ denote the (suboptimal) control sequence for the initial state $x$, and let $\tilde{\mathbf{u}}$ denote a *warm start* for the successor initial state $x^+ = f(x, u(0; x))$, obtained from $(x, \mathbf{u})$ by

$$\tilde{\mathbf{u}} := \{u(1; x), u(2; x), \ldots, u(N-1; x), u_+\} \tag{3}$$

- $u_+ \in \mathbb{U}$ is any input that satisfies the invariance condition in the terminal region



## Suboptimal MPC

- The warm start satisfies $\tilde{\mathbf{u}} \in \mathcal{U}_N(x^+)$.
- The suboptimal input sequence for any given $x^+ \in \mathcal{X}_N$ is defined as *any* $\mathbf{u}^+ \in \mathbb{U}^N$ that satisfies:

$$\mathbf{u}^+ \in \mathcal{U}_N(x^+) \tag{4a}$$
$$V_N(x^+, \mathbf{u}^+) \leq V_N(x^+, \tilde{\mathbf{u}}) \tag{4b}$$
$$V_N(x^+, \mathbf{u}^+) \leq V_f(x^+) \qquad \text{when } x^+ \in r\mathbb{B} \tag{4c}$$

in which $r$ is a positive scalar sufficiently small that $r\mathbb{B} \subseteq \mathbb{X}_f$.

- Notice that constraint (4c) is required to hold only if $x^+ \in r\mathbb{B}$, and it implies that $|\mathbf{u}^+| \to 0$ as $|x^+| \to 0$.
- Condition (4b) ensures that the computed suboptimal cost is no larger than that of the warm start.



## Inherent robustness of the suboptimal controller

- Consider a process disturbance $d$, $x^+ = f(x, \kappa(x)) + d$
- A measurement disturbance $x_m = x + e$
- Nominal controller with disturbance

$$x^+ \in f(x, \kappa_N(x_m)) + d$$
$$x^+ \in f(x, \kappa_N(x + e)) + d$$
$$x^+ \in F_{ed}(x) \tag{5}$$

Robust stability; is the system $x^+ \in F_{ed}(x)$ input-to-state stable considering $(d, e)$ as the input.

## Robust exponential stability of suboptimal MPC

### Definition (SRES)

The origin of the closed-loop system (5) is *strongly robustly exponentially stable* (SRES) on a compact set $\mathcal{C} \subset \mathcal{X}_N$, $0 \in \text{int}(\mathcal{C})$, if there exist scalars $b > 0$ and $0 < \lambda < 1$ such that the following property holds: Given any $\epsilon > 0$, there exists $\delta > 0$ such that for all sequences $\{d(k)\}$ and $\{e(k)\}$ satisfying

$$|d(k)| \leq \delta \text{ and } |e(k)| \leq \delta \quad \text{for all } k \in \mathbb{I}_{\geq 0},$$

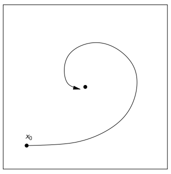
and all $x \in \mathcal{C}$, we have that

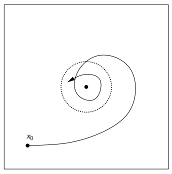
$$x_m(k) = x(k) + e(k) \in \mathcal{X}_N, \; x(k) \in \mathcal{X}_N, \; \text{for all } k \in \mathbb{I}_{\geq 0}, \tag{6a}$$

$$|\phi_{ed}(k; x)| \leq b\lambda^k |x| + \epsilon, \; \text{for all } k \in \mathbb{I}_{\geq 0}. \tag{6b}$$



## Behavior with and without disturbances





### Nominal System

$$x^+ = f(x, u)$$
$$u = \kappa_N(x)$$

### System with Disturbance

$$x^+ = f(x, u) + d$$
$$u = \kappa_N(x + e)$$

$d$ is the process disturbance
$e$ is the measurement disturbance



## Main results

### Theorem (SRES of suboptimal MPC (Pannocchia et al., 2011))

*Under standard MPC assumptions, the origin of the perturbed closed-loop system*

$$x^+ \in F_{ed}(x)$$

*is SRES on* $\mathcal{C}_\rho$.

This result applies also to distributed, cooperative MPC.
See also Pannocchia talk on Wednesday, 14:30, WEB07.4.



## Conclusions

### Cooperative MPC theory maturing[a]

[a]Stewart et al. (2010); Maestre et al. (2011)

- Avoids coordination layer
- Satisfies hard input constraints
- Provides nominal stability for plants with even strongly interacting subsystems
- Retains closed-loop stability for early iteration termination
- Converges with iteration to Pareto optimal (centralized) control
- Remains stable under perturbations



## Future directions

### Lots to do!

- Applications in which players *compete* as well as cooperate
- Framework(s) for decomposing large-scale systems
- Modeling versus performance tradeoffs poorly understood
- Unstable systems and coupled constraints difficult to handle (supply chain)
- Distributed state estimation has received less attention than control (Farina et al., 2010a,b)
- Applications exposing limitations of current approaches (De Schutter and Scattolini, 2011; Tarau et al., 2011; Baskar et al., 2011)



## Further reading I

L. D. Baskar, B. De Schutter, J. Hellendoorn, and Z. Papp. Traffic control and intelligent vehicle highway systems: a survey. *IET Intelligent Transport Systems*, 5 (1):38–52, MAR 2011.

D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, second edition, 1999.

R. Cheng, J. F. Forbes, and W. S. Yip. Price-driven coordination method for solving plant-wide MPC problems. *J. Proc. Cont.*, 17(5):429–438, 2007.

B. De Schutter and R. Scattolini. Introduction to the special issue on hierarchical and distributed model predictive control. *J. Proc. Cont.*, 21(5, SI):683–684, JUN 2011.

M. Farina, G. Ferrari-Trecate, and R. Scattolini. Moving-horizon partition-based state estimation of large-scale systems. *Automatica*, 46(5):910–918, 2010a.

M. Farina, G. Ferrari-Trecate, and R. Scattolini. Distributed moving horizon estimation for linear constrained systems. *IEEE Trans. Auto. Cont.*, 55(11):2462–2475, 2010b.

R. D. Gudi and J. B. Rawlings. Identification for Decentralized Model Predictive Control. *AIChE J.*, 52(6):2198–2210, 2006.

Y.-C. Ho. On Centralized Optimal Control. *IEEE Trans. Auto. Cont.*, 50(4):537–538, 2005.

## Further reading II


J. Lunze. *Feedback Control of Large Scale Systems*. Prentice-Hall, London, U.K., 1992.

J. M. Maestre, D. Muñoz de la Peña, and E. F. Camacho. Distributed model predictive control based on a cooperative game. *Optimal Cont. Appl. Meth.*, 32(2):153–176, 2011.

I. Necoara, D. Doan, and J. Suykens. Application of the proximal center decomposition method to distributed model predictive control. In *Proceedings of the IEEE Conference on Decision and Control*, Cancun, Mexico, December 9-11 2008.

G. Pannocchia, J. B. Rawlings, and S. J. Wright. Conditions under which suboptimal nonlinear MPC is inherently robust. *Sys. Cont. Let.*, 60:747–755, 2011.

J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, Madison, WI, 2009. 576 pages, ISBN 978-0-9759377-0-9.

N. R. Sandell Jr., P. Varaiya, M. Athans, and M. Safonov. Survey of decentralized control methods for large scale systems. *IEEE Trans. Auto. Cont.*, 23(2):108–128, 1978.

D. D. Šiljak. *Decentralized Control of Complex Systems*. Academic Press, London, 1991. ISBN 0-12-643430-1.

B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia. Cooperative distributed model predictive control. *Sys. Cont. Let.*, 59:460–469, 2010.




## Further reading III


A. N. Tarau, B. De Schutter, and J. Hellendoorn. Predictive route control for automated baggage handling systems using mixed-integer linear programming. *Transportation Research Part C-Emerging Technologies*, 19(3):424–439, JUN 2011.

## 2.3 Hierarchical and distributed optimization algorithms (M. Diehl, A. Kozma, C. Savorgnan)

## Hierarchical and Distributed Optimization Methods

Moritz Diehl, Attila Kozma, Carlo Savorgnan

Optimization in Engineering Center OPTEC
and Electrical Engineering Department ESAT
K.U. Leuven, Belgium

*IFAC WC Milano, August 28, 2011*

HD-MPC

## Overview

- Motivation for Centralized Computation
- Distributed Multiple Shooting Framework
- Adjoint Based SCP Methods, from Hierarchical to Distributed
- Software

## Motivation for Hierarchical and Distributed MPC

Large-scale systems in engineering
- composed of multiple subsystems
- complex nonlinear dynamics and
- mutual influences

E.g. river networks, chemical production sites, airflow in buildings.

**How to compute optimal controls e.g. for transients?**


## Two Central Observations on distributed MPC

(1) For cooperative model predictive control, we ideally want to solve **one large centralized MPC problem**.

Reasons for distributed setup:
- Robustness and easier reconfigurability
- Distribution of data and model maintenance
- Parallel computations (ideally, solution time independent of size)
- Hope that less communication is needed than in centralized setting

(2) Most distributed MPC methods work iteratively and focus on parallelizing each iteration. But even if solution time for each iteration is independent of size, the convergence speed mostly deteriorates with size of the problem (usually linear or sublinear rates).
**Distributed computation and communication time might be much higher than for one centralized solution, i.e. many processors together working very hard can be slower than one single one!**

## (Interlude: Large Scale QP algorithms)

Decomposition by Lagrangian dual function

$$\min_{\underline{x}_1,\ldots,\underline{x}_N} \sum_{i=1}^{N} \frac{1}{2}\underline{x}_i^T Q_i \underline{x}_i + \underline{c}_i^T \underline{x}_i \quad \text{s.t.} \quad H_i\underline{x}_i \leq \underline{d}_i \quad i=1,\ldots,n, \quad \sum_{i=1}^{N} A_i \underline{x}_i = \underline{b}$$

- Convex separable QP
- Coupling lin. equality

$$\max_{\underline{\lambda}} \sum_{i=1}^{N} \underbrace{\begin{pmatrix} \min_{\underline{x}_i} & \left(\frac{1}{2}\underline{x}_i^T Q_i \underline{x}_i + \left(\underline{c}_i^T + \underline{\lambda}^T A_i\right)\underline{x}_i - \underline{\lambda}^T \frac{\underline{b}}{N}\right) \\ \text{s.t.} & H_i \underline{x}_i \leq \underline{d}_i \end{pmatrix}}_{P_i(\underline{\lambda})}$$

- Two-level problem
- Low-level: parametric QPs (online act. set strat.)
- High-level: unconstr. problem with gradient avail. (fast gradient method)

## (Runtime Comparison in Our Initial Work)

Solve large distributed quadratic program with 100 subsystems on 100 CPUs, using different dual decomposition methods:

**Wall clock:**

| $\delta$ | Nesterov | Gradient |
|---|---|---|
| $10^{-3}$ | 0:55 | 02:58 |
| $10^{-4}$ | 1:55 | 03:59 |
| $10^{-5}$ | 2:52 | 04:56 |
| $10^{-6}$ | 3:29 | 05:52 |

Same problem takes **0:03 seconds on a single CPU** when solved with a sparse IP method (OOQP from S. Wright).

Problem of all gradient methods: no second order information, slow linear convergence. Better parallelize IP solver!

## Can simulation efficiently be parallelized ?

**Assumption: simulators for individual subsystems exist**

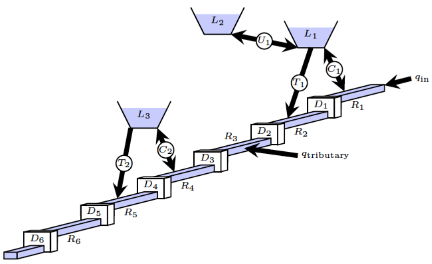
- use their own adaptive numerical integration schemes
- based on possibly different modelling languages
- can provide derivatives in forward and reverse mode (not yet standard, but provided e.g. by SUNDIALS, DASPK, DAESOL-II, ACADO Integrators, ...)

## Example: Hydro Power Valley (HPV) Benchmark

River reaches connected by dams and hydro power units.
NMPC control aims:

- strictly respect level constraints
- match total power demand
- keep levels as constant as possible

## HPV consists of 8 coupled subsystems



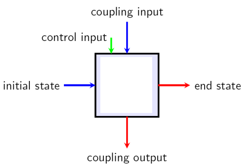

## Hydro Power Valley (HPV)

Water flow in reaches modeled by Saint Venant PDE:

$$\begin{cases} \frac{\partial Q(z,t)}{\partial z} + w\frac{\partial H(z,t)}{\partial t} = 0 \\ \frac{1}{gw}\frac{\partial}{\partial t}\left(\frac{Q(z,t)}{H(z,t)}\right) + \frac{1}{2gw^2}\frac{\partial}{\partial z}\left(\frac{Q^2(t,z)}{H^2(t,z)}\right) + \frac{\partial H(t,z)}{\partial z} + I_f(z) - I_0 = 0 \end{cases}$$

Transform PDE into ODE by spatial discretization.

## The "Simulation Box" (e.g. one reach of HPV)




I

## Centralized Optimal Control

$$\min_{\substack{x,u,z,\\ y,e}} \int_0^T \ell(e(t))dt + \sum_{i=1}^{M} \int_0^T \ell^i(x^i(t), u^i(t), z^i(t))dt$$

s.t.
$$\dot{x}^i(t) = f^i(x^i(t), u^i(t), z^i(t))$$
$$y^i(t) = g^i(x^i(t), u^i(t), z^i(t))$$
$$x^i(0) = \bar{x}_0^i$$
$$z^i(t) = \textstyle\sum_{j=1}^M A_{ij} y^j(t)$$
$$e(t) = r(t) + \textstyle\sum_{i=1}^M B^i y^i(t)$$
$$p^i(x^i(t), u^i(t)) \geq 0, \quad q(e(t)) \geq 0 \quad t \in [0, T]$$

**Key idea**: The signals $z^i(t)$, $y^i(t)$ and $e(t)$ can be represented as a linear combination of orthogonal polynomials.

## Distributed Multiple Shooting yields sparse NLP

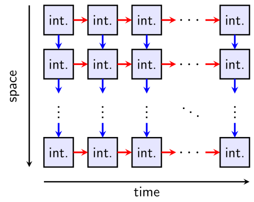
$$\min_{\substack{u_n^i, x_n^i, \mathbf{z}_n^i,\\ \mathbf{y}_n^i, \mathbf{e}_n}} \sum_{n=0}^{N-1} \left( L_n(\mathbf{e}_n) + \sum_{i=1}^{M} L_n^i(x_n^i, u_n^i, \mathbf{z}_n^i) \right)$$

s.t.

$$x_{n+1}^i = F_n^i(x_n^i, u_n^i, \mathbf{z}_n^i) \quad n = 0, \dots, N-1$$
$$\mathbf{y}_n^i = G_n^i(x_n^i, u_n^i, \mathbf{z}_n^i) \quad n = 0, \dots, N-1$$
$$x_0^i = \bar{x}_0^i$$
$$\mathbf{z}_n^i = \sum_{i=1}^{M} A_{ij} \mathbf{y}_n^j$$
$$\mathbf{e}_n = \mathbf{r}_n + \sum_{i=1}^{M} B_{ij} \mathbf{y}_n^j$$
$$p^i(x_n^i, u_n^i) \geq 0, \qquad Q_n(\mathbf{e}_n) \geq 0$$

## Distributed Multiple Shooting

Multiple Shooting [Bock and Plitt 1984], but in time AND SPACE




- discretized subsystem connections (polynomials)
- gaps between subsystems
- any complex topology

## Large Scale Nonlinear Program (NLP)

Each simulation box $x_i = \phi_i(X_i, u_i)$ also evaluates an objective $f_i(X_i, u_i)$ and inequality constraints $g_i(X_i, u_i)$.

$x_i$ = output of each simulation box.
$X_i$ = Input, lin. combination of other outputs

$$\begin{aligned} \text{minimize}_{x,u} \quad & \sum_{i=1}^{N} f_i(X_i, u_i) \\ \text{subject to} \quad & \phi_i(X_i, u_i) - x_i = 0, \\ & g_i(X_i, u_i) \leq 0, \quad i = 1, \dots, N. \end{aligned}$$

**Note: coupling constraints only feasible in solution!**
***Simultaneous* method for simulation and optimization.**

## Sequential Convex Programming (SCP)

Assuming $f_i, g_i$ convex and known to central optimizer, can linearize simulation boxes at linearization points $\bar{X}_i, \bar{u}_i$.

$$\begin{aligned} \text{minimize}_{x,u} \quad & \sum_{i=1}^{N} f_i(X_i, u_i) \\ \text{subject to} \quad & \phi_i(\bar{X}_i, \bar{u}_i) + \frac{\partial \phi_i(\bar{X}_i, \bar{u}_i)}{\partial(X, u)} \begin{bmatrix} X_i - \bar{X}_i \\ u_i - \bar{u}_i \end{bmatrix} - x_i = 0 \\ & g_i(X_i, u_i) \leq 0, \ i \in [1, N]. \end{aligned}$$

Iteratively solving linearized convex problems for obtaining the next linearization point yields a generalization of SQP, **Sequential Convex Programming (SCP)**. Can prove linear convergence towards local minima [Necoara et al, CDC, 2009], [T. D. Quoc and MD, BFG, 2010].

## Adjoint based SCP Method

Approximate $\frac{\partial \phi_i(\bar{X}_i, \bar{u}_i)}{\partial(X, u)}$ by cheaper $A_i$. Add gradient correction to objective.

$$\begin{aligned} \text{minimize}_{x,u} \quad & \sum_{i=1}^{N} f_i(X_i, u_i) + [X_i^T | u_i^T] \frac{\partial \phi_i(\bar{X}_i, \bar{u}_i)}{\partial(X, u)}^T \bar{\lambda}_i \\ \text{subject to} \quad & \phi_i(\bar{X}_i, \bar{u}_i) + A_i \begin{bmatrix} X_i - \bar{X}_i \\ u_i - \bar{u}_i \end{bmatrix} - x_i = 0, \\ & g_i(X_i, u_i) \leq 0, \ i \in [1, N]. \end{aligned}$$

Solution $x^*, u^*$ and equality multipliers $\delta^*$ yield next linearization point $\bar{x}^+, \bar{u}^+$ and multiplier guess, $\bar{\lambda}^+ = \bar{\lambda} + \delta^*$.
Linear convergence proven [D., Walther, Bock, Kostina, OMS, 2009], [Quoc et al. 2010].

## Why are inexact derivatives interesting ?

- derivative $\frac{\partial \phi_i(\bar{X}_i, \bar{u}_i)}{\partial(X, u)}$ is a large dense matrix, expensive to compute
- often, only few strongly coupling variables $X_i^A$ in $X_i = (X_i^A, X_i^B)$, so can cheaply approximate derivative:

$$\left[ \frac{\partial \phi_i}{\partial X_i^A} \ \middle| \ \frac{\partial \phi_i}{\partial X_i^B} \ \middle| \ \frac{\partial \phi_i}{\partial u_i} \right] \approx \left[ \frac{\partial \phi_i}{\partial X_i^A} \ \middle| \ 0 \ \middle| \ \frac{\partial \phi_i}{\partial u_i} \right] =: A_i$$

- evaluate gradient correction $\frac{\partial \phi_i(\bar{X}_i, \bar{u}_i)}{\partial(X, u)}^T \bar{\lambda}_i$ by reverse differentiation, only 4 times more expensive than simulation $\phi_i(X_i, u_i)$. One single **extended simulation box** call.
- Less communication: variables $x^B$ and multipliers $\lambda^B$ only passed between child and parent nodes. Central optimizer works with **aggregate model** in $x^A$ and $u$ only.

Variant: left part of $A_i$ = 0, get completely distributed convex subproblems!

## Adjoint SCP: both Hierarchical and Distributed

- Exact SCP: all coordination work done by central agent who solves convex subproblems. 100% hierarchical.
- Adjoint based SCP with partially zero derivative matrices $A_i$ : only most influential variables coordinated by central agent, fine interactions are exchanged locally.
- Adjoint based SCP with completely zero derivative matrices $A_i$ : all information is exchanged locally, convex problem decomposes, no central agent necessary: 100% distributed.

Trade-off: convergence speed vs distributed solution

## Overview

- Motivation for Centralized Computation
- Distributed Multiple Shooting Framework
- Adjoint Based SCP Methods, from Hierarchical to Distributed
- **Software**

## ACADO Toolkit for Nonlinear MPC

with Joachim Ferreau and Boris Houska

## Software for Nonlinear MPC: ACADO Toolkit

- ACADO = Automatic Control and Dynamic Optimization
- Open source (LGPL): www.acadotoolkit.org
- User interface close to mathematical syntax
- Self containedness: only need C++ compiler
- Focus on small but fast applications

## Problem Classes in ACADO

- Optimal Control of Dynamic Systems (ODE/DAE)

$$\underset{y(\cdot),u(\cdot),p,T}{\text{minimize}} \quad \int_0^T L(\tau, y(\tau), u(\tau), p)\,\mathrm{d}\tau + M(y(T), p)$$

subject to:

$$\begin{aligned} \forall t \in [0,T]: \quad 0 &= f(t, \dot{y}(t), y(t), u(t), p) \\ 0 &= r(y(0), y(T), p) \\ \forall t \in [0,T]: \quad 0 &\geq s(t, y(t), u(t), p) \end{aligned}$$

- Nonlinear Model Predictive Control
- Parameter Estimation and Optimum Experimental Design
- Robust Optimization
- **Automatic Code Generation for fast MPC applications**

## Example for Code Generation ("Tiny" Scale)

```
DifferentialState    p,v,phi,omega;
Control              a;

Matrix Q = eye( 4 );
Matrix R = eye( 1 );

DifferentialEquation f;
f « dot(p)     == v;
f « dot(v)     == a;
f « dot(phi)   == omega;
f « dot(omega) == -g*sin(phi)
                  -a*cos(phi)-b*omega;

OCP ocp( 0.0, 5.0 ,10 );
ocp.minimizeLSQ( Q, R );

ocp.subjectTo( f );
ocp.subjectTo( -0.2 <= a <=  0.2 );

OptimizationAlgorithm algorithm(ocp);
algorithm.solve();
```

Algorithm: Gauss Newton Real-Time Iterations

1 control input
10 control intervals
4 states

| | CPU time | Percentage |
|---|---|---|
| Integration & sensitivities | 34 $\mu$s | 63 % |
| Condensing | 11 $\mu$s | 20 % |
| QP solution (with qpOASES) | 5 $\mu$s | 9 % |
| Remaining operations | $<5$ $\mu$s | $<8$ % |
| One complete real-time iteration | 54 $\mu$s | 100 % |

NMPC with 200 kHz possible !

## Modelica and Automatic Derivatives with CasADi

with **Joel Andersson**

## CasADi

- CasADi
  - "Computer Algebra System for Automatic Differentiation"
  - Free (LGPL) open-source symbolic tool (www.casadi.org)
  - **Extends the NLP approach for OCP to shooting methods**
  - "Write a state-of-the-art multiple shooting code in 50 lines"

CasADi

## CasADi – NLP approach for Shooting Methods

### Components of CasADi

- A computer algebra system for algebraic modeling
- Efficient, general implementation of AD
  - AD on **sparse, matrix-valued** computational graphs
  - Forward/adjoint mode
  - Generate new graphs for Jacobians/Hessians
- Efficient virtual machine for function/derivative evaluation
- Front-ends to C++, Python and Octave
- Smart interfaces to numerical codes, e.g.:
  - NLP solvers: IPOPT, KNITRO, (SNOPT, LiftOpt)
  - DAE integrators: Cvodes, Idas, GSL
    - Automatic generation of Jacobian information (for BDF)
    - Automatic formulation of sensitivity equations (fwd/adj)
- Symbolic model import from Modelica (via Jmodelica.org)

## CasADi/CVODES for Sensitivies of HPV subsystem

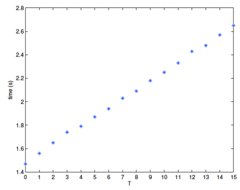


Fig. 4. Time required to integrate and linearize a subsystem dynamics for an time interval of 30 minutes using $S = 15$.

For full problem:

- Total: 48 time intervals, 8 subsystems = 384 simulation boxes
- Sensitivity Integration of full system on full horizon would take 1630 sec
- Compare this to 1.5 up to 2.7 sec per simulation box.

0 = only forward w.r.t. controls and adjoint, fully distributed
15 = all forward derivatives, full space exact SCP

## Summary: Large Nonlinear MPC

- In cooperative MPC we want to solve centralized optimization problems, and centralized algorithms might be more efficient in both time and communication than distributed ones
- Distributed Multiple Shooting (DMS) is a way to parallelize simulation and sensitivity generation
- Adjoint based SCP Algorithms for DMS allow many variants between fully hierarchical and fully distributed algorithms

Software (LGPL):

- ACADO Toolkit and code generation allow fast nonlinear MPC for small problems (e.g. 200 kHz for 4 states)
- CasADi allows one to easily couple integrators and optimizers and setup e.g. distributed multiple shooting
  → Talk Attila Kozma, Monday, 11:20, room Vito

HD–MPC

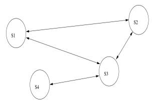
## Appendix

## Large-scale separable convex optimization (T.Quoc)

- Problem Statement

$$(\text{CP}) \begin{cases} \min_{x_1,\dots,x_M} f(x) := \sum_{i=1}^{M} f_i(x_i, \\ \text{s.t. } \sum_{i=1}^{M} A_i x_i = b, \\ \quad x_i \in X_i, \; i = 1, \dots, M. \end{cases}$$

- $f_i : \mathbb{R}^{n_i} \to \mathbb{R}$ - convex, possible nonsmooth
- $X_i \subset \mathbb{R}^{n_i}$ – closed convex, bounded




- Examples
  - Large-scale LPs, QPs.
  - Optimization in networks, graph theory.
  - Multi-stage stochastic convex optimization.
  - Distributed MPC, etc.
- Aim:
  - Design distributed algorithms to solve (CP)

## Main idea and algorithms

- Main idea: Combine three techniques
  - Lagrangian dual decomposition

$$d(y) = \sum_{i=1}^{M} d_i(y)$$

  - Smoothing technique via prox-functions

$$f_{\beta_2}(x) := \max_{y \in Y} \{\phi(x) + (Ax - b)^T y - \beta_2 p_Y(y)\}$$
$$d_{\beta_1}(y) := \min_{x \in X} \{f(x) + y^T(Ax - b) + \beta_1 p_X(x)\}$$

  - Excessive gap condition *[Nesterov2005]*

$$f_{\beta_2}(\bar{x}) \le d_{\beta_1}(\bar{y})$$

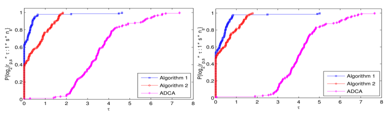
- Optimality and feasibility gaps

$$0 \le \phi(\bar{x}) - d(\bar{y}) \le \beta_1 D_o \quad \text{and} \quad ||A\bar{x} - b|| \le \beta_2 D_f.$$

- Algorithm: two variants – primal update and switching update
  - Generate a sequence $\{(\bar{x}, \bar{y})\}$ such that it maintains the excessive gap condition, while controls $\beta_1$ and $\beta_2$ to zero.

## Advantages and performance

- Advantages
  - Convergence rate **O(1/k)**
  - Fast (compared to dual-fast gradient method *[Necoara2008]*, subgradient, augmented Lagrangian)
  - Numerical robustness
  - Highly distributed
- Numerical test: Large scale separable QP problems (dense)




Compare three difference algorithms: primal update, switching update, dual-fast gradient for solving random QPs (left – iterations, right – CPU time)

## Coupling between subsystems

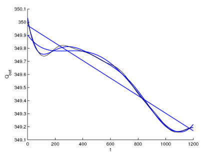
Dynamics of subsystems are coupled via in-/output profiles of "coupling variables". Infinite dimensional coupling.

## Coupling between subsystems

Dynamics of subsystems are coupled via in-/output profiles of "coupling variables". Infinite dimensional coupling.

Can approximate coupling profile by orthogonal polynomials:

$$\int_a^b P_i(t) P_j(t) dt = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{otherwise} \end{cases}$$



Approximation of typical output water discharge profile (black) by polynomials of degree 1, 4 and 7.

## CasADi Code Example: Single Shooting in 30 lines

```
from casadi import *

# Declare variables (use simple, efficient DAG)
t = SX("t") # time
x=SX("x"); y=SX("y"); u=SX("u"); L=SX("cost")

# ODE right hand side function
f = [(1 - y*y)*x - y + u, x, x*x + y*y + u*u]
rhs = SXFunction([[t],[x,y,L],[u]],[f])

# Create an integrator (CVodes)
I = CVodesIntegrator(rhs)
I.setOption("abstol",1e-10) # abs. tolerance
I.setOption("reltol",1e-10) # rel. tolerance
I.setOption("steps_per_checkpoint",100)
I.init()

# All controls (use complex, general DAG)
NU = 20; U = MX("U",NU)

# The initial state (x=0, y=1, L=0)
X  = MX([0,1,0])

# Time horizon
T0 = MX(0);  TF = MX(20.0/NU)

# State derivative, algebraic state (not used)
XP = MX(); Z = MX()

# Build up a graph of integrator calls
for k in range(NU):
  [X,XP,Z] = I.call([T0,TF,X,U[k],XP,Z])

# Objective function: L(T)
F = MXFunction([U],[X[2]])

# Terminal constraints: 0<=[x(T);y(T)]<=0
G = MXFunction([U],[X[0:2]])

# Create NLP solver
solver = IpoptSolver(F,G)
solver.setOption("tol",1e-5)
solver.setOption("hessian_approximation", \
                "limited-memory")
solver.setOption("max_iter",1000)
solver.init()

# Set bounds and initial guess
solver.setInput(NU*[-0.75], NLP_LBX)
solver.setInput(NU*[1.0],NLP_UBX)
solver.setInput(NU*[0.0],NLP_X_INIT)
solver.setInput([0,0],NLP_LBG)
solver.setInput([0,0],NLP_UBG)

# Solve the problem
solver.solve()
```

## 2.4 Design of hierarchical and distributed MPC control systems with robustness tools (M. Farina, B. Picasso, R. Scattolini)

Milano, August 28th, 2011

## Design of hierarchical and distributed MPC control systems with robustness tools

Marcello Farina, Bruno Picasso, Riccardo Scattolini
DEI-Politecnico di Milano

HD-MPC SEVENTH FRAMEWORK PROGRAMME

Pre-Congress Workshop - IFAC 2011 Milano

## Outline 2

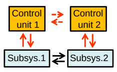
1. Introduction
2. Hierarchical MPC systems
   - Basic architecture
   - Extensions (performance & reconfigurability)
   - Conclusions
3. Distributed MPC systems
   - A "tube-based", non cooperative DMPC algorithm
   - Conclusions
4. Concluding remark

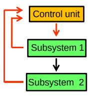
Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Motivations 3

Motivations for distributed / hierarchical control:

- Reduce the computational load
- Reduce the communication load
- Improve the robustness with respect to failures
  - in the transmission of information
  - in the central control unit
- Improve the modularity and the flexibility of the system
- Consider different goals at different time scales (Real-Time Optimization)
- Synchronize subsystems working at different time scales

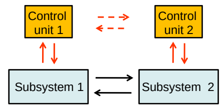


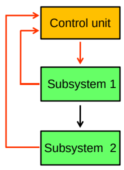




There has hence been a long time interest for decentralized / distributed **[Siljak '78… '91]** and hierarchical control **[Mesarovic '70, Findeisen '80, …]** for **large-scale** and **complex systems**.
Recent contributions include: **[Engell '07, Tatjewski '08** and **Scattolini '09 -** *"An overview on distributed and hierarchical MPC"*]).

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## The proposed approach 4




In both distributed and hierarchical structures, there are two possible approaches to the control synthesis allowing to deal with the interacting subsystems:

1. Game theory
2. **Robust control**

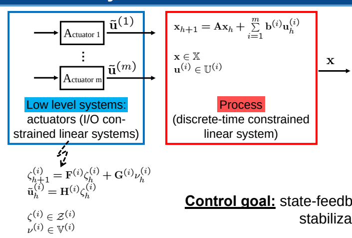
Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Outline 5

1. Introduction
2. Hierarchical MPC systems
   - Basic architecture
   - Extensions (performance & reconfigurability)
   - Conclusions
3. Distributed MPC systems
   - A "tube-based", non cooperative DMPC algorithm
   - Conclusions
4. Concluding remark

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

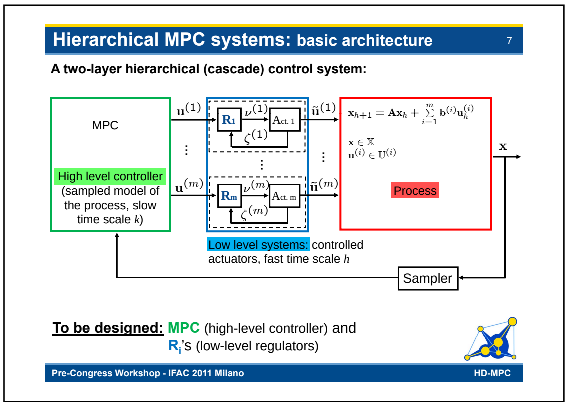
## Hierarchical MPC systems: basic architecture 6

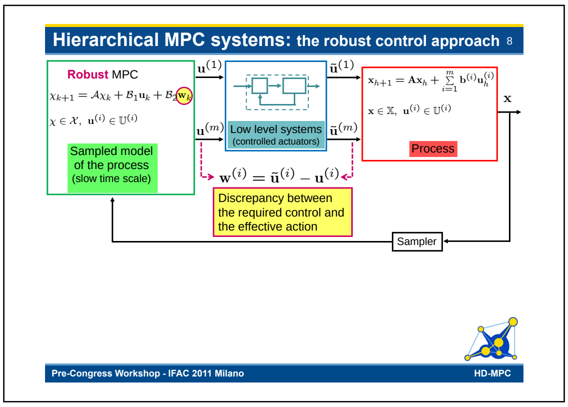


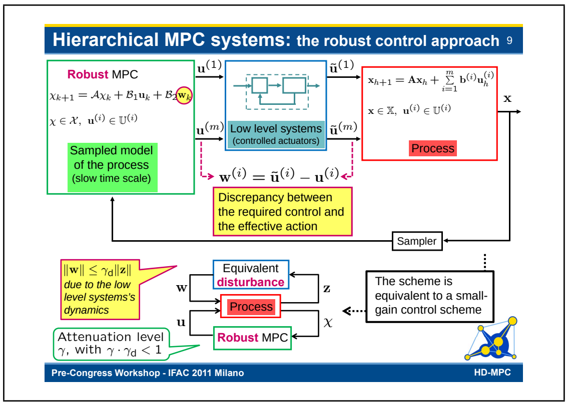

**Control goal:** state-feedback stabilization

Typical structure in many control applications:
- Process control **[Skogestadt '00]**
- Automotive **[Brahma et al. '00]**
- Production planning **[Golenko-Ginzburg et al. '93]**

HD-MPC




## High level robust MPC design 10

Two possible scenarios:

1. The high level unit can *simulate* the low level actuators
   $\Rightarrow$ the disturbance $\mathbf{w}$ is predicted ($\gamma_d$ is locally available)

2. The disturbance $\mathbf{w}$ is not predictable by the high level unit but $\gamma_d$ is globally available

**Main result**

In both cases, a **robust MPC** controller is designed so that:

- The high level controller is robustly stabilizing in the slow time scale $k$ ;
- Convergence to the equilibrium for the overall control system is guaranteed in the fast time scale $h$ .

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## High level robust MPC design 11

Scenario 1 ($\mathbf{w}$ is predictable):

$$\min_{\mathcal{F}} J(\chi, \mathcal{F}, N_p)$$

subject to the dynamics, the constraints + a suitable auxiliary law

where

Scenario 2 (only $\gamma_d$ is known):

$$\min_{\mathcal{F}} \max_{\mathcal{D}} J(\chi, \mathcal{F}, \mathcal{D}, N_p)$$

subject to the dynamics, the constraints + a suitable auxiliary law

where

$$J = \sum_{j=0}^{N_p-1}(\|\mathbf{z}_{k+j}\|^2_{Q_z} - \gamma^2\|\mathbf{w}_{k+j}\|^2_{Q_w}) + V_f(\chi_{k+N_p})$$

and (Scenario 1)

$\mathcal{F} = [\, \mathbf{u}_k \;\; \mathbf{u}_{k+1} \;\; \cdots \;\; \mathbf{u}_{k+N_c-1} \,]$

is a sequence of control **values**

and (Scenario 2)

$\mathcal{D} = [\, \mathbf{w}_k \;\; \mathbf{w}_{k+1} \;\; \cdots \;\; \mathbf{w}_{k+N_p-1} \,]$

is a sequence of disturbance values

$\mathcal{F} = [\, \mathbf{u}_k \;\; \mathbf{u}_{k+1}(\cdot) \;\; \cdots \;\; \mathbf{u}_{k+N_c-1}(\cdot) \,]$

is a sequence of control **policies**

**Features:** the high and low level designs are only partially decoupled but a global $\gamma_d$ is not needed and the optimization is less demanding

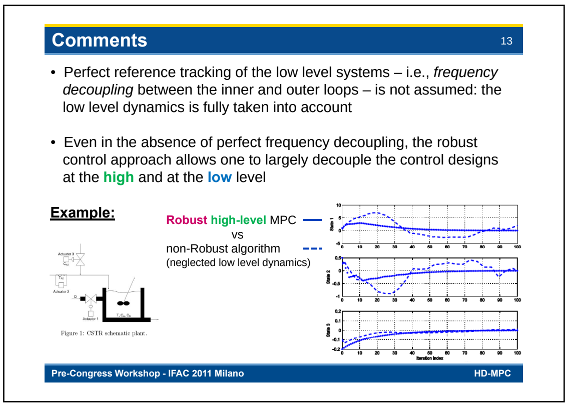
**Features:** the high and low level designs are decoupled but $\gamma_d$ and minmax optimization are needed

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

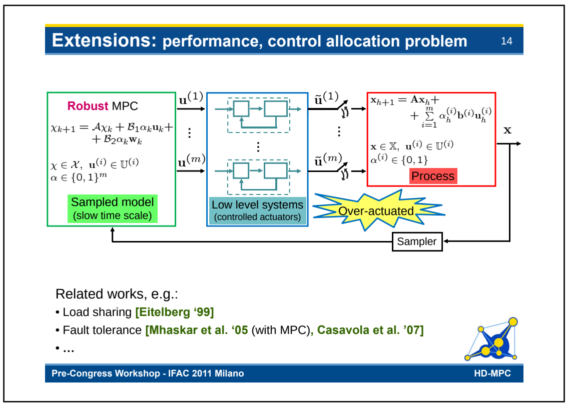
## Comments 12

- Perfect reference tracking of the low level systems – i.e., *frequency decoupling* between the inner and outer loops – is not assumed: the low level dynamics is fully taken into account
- Even in the absence of perfect frequency decoupling, the robust control approach allows one to largely decouple the control designs at the **high** and at the **low** level

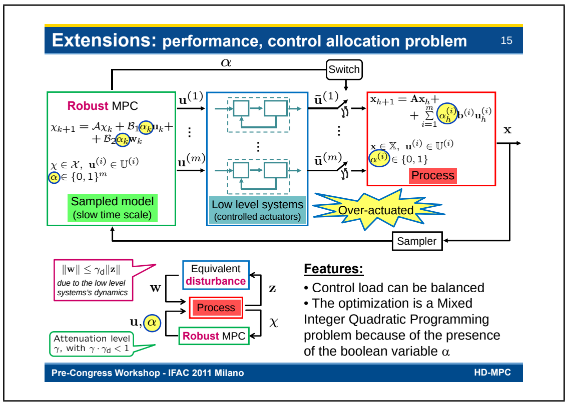
Pre-Congress Workshop - IFAC 2011 Milano HD-MPC



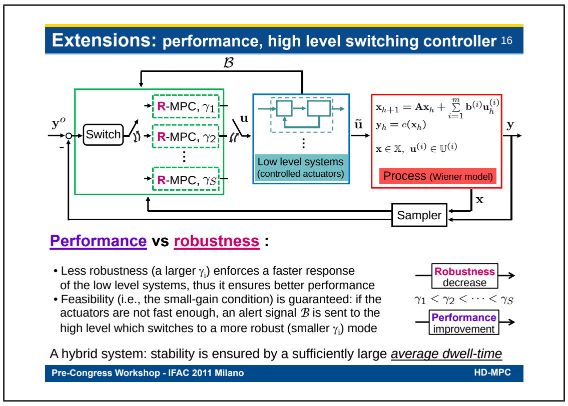
Comments 13
• Perfect reference tracking of the low level systems – i.e., frequency decoupling between the inner and outer loops – is not assumed: the low level dynamics is fully taken into account
• Even in the absence of perfect frequency decoupling, the robust control approach allows one to largely decouple the control designs at the high and at the low level
Example:
Robust high-level MPC
vs
non-Robust algorithm
(neglected low level dynamics)
Figure 1: CSTR schematic plant.
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC

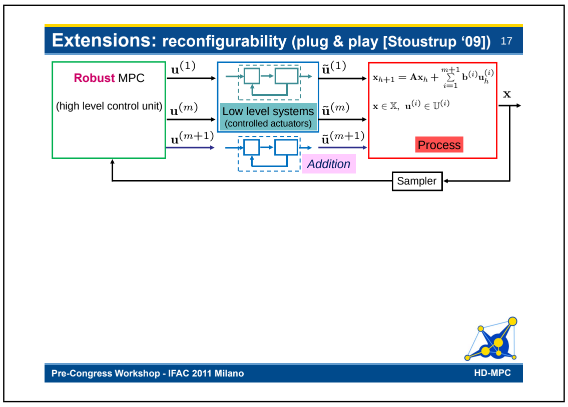
Extensions: performance, control allocation problem 14
Robust MPC
$\chi_{k+1} = \mathcal{A}\chi_k + \mathcal{B}_1\alpha_k u_k + \mathcal{B}_2\alpha_k w_k$
$\chi \in \mathcal{X}, \ u^{(i)} \in \mathbb{U}^{(i)}$
$\alpha \in \{0,1\}^m$
Sampled model
(slow time scale)
$u^{(1)}$
$u^{(m)}$
Low level systems
(controlled actuators)
$\tilde{u}^{(1)}$
$\tilde{u}^{(m)}$
$x_{h+1} = Ax_h + \sum_{i=1}^{m} \alpha_h^{(i)} b^{(i)} u_h^{(i)}$
$x \in \mathbb{X}, \ u^{(i)} \in \mathbb{U}^{(i)}$
$\alpha^{(i)} \in \{0,1\}$
Process
Over-actuated
x
Sampler
Related works, e.g.:
• Load sharing [Eitelberg '99]
• Fault tolerance [Mhaskar et al. '05 (with MPC), Casavola et al. '07]
• ...
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC

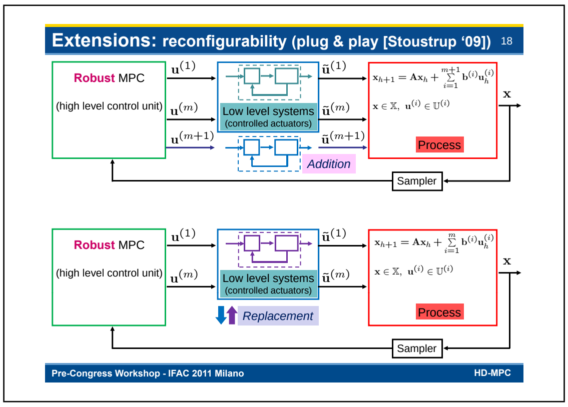
Extensions: performance, control allocation problem 15
α
Switch
Robust MPC
$\chi_{k+1} = \mathcal{A}\chi_k + \mathcal{B}_1\alpha_k u_k + \mathcal{B}_2\alpha_k w_k$
$\chi \in \mathcal{X}, \ u^{(i)} \in \mathbb{U}^{(i)}$
$\alpha \in \{0,1\}^m$
Sampled model
(slow time scale)
Low level systems
(controlled actuators)
$x_{h+1} = Ax_h + \sum_{i=1}^{m} \alpha_h^{(i)} b^{(i)} u_h^{(i)}$
$x \in \mathbb{X}, \ u^{(i)} \in \mathbb{U}^{(i)}$
$\alpha^{(i)} \in \{0,1\}$
Process
Over-actuated
Sampler
$\|w\| \le \gamma_d \|z\|$
due to the low level systems's dynamics
Equivalent disturbance
Process
Robust MPC
Attenuation level $\gamma$, with $\gamma \cdot \gamma_d < 1$
Features:
• Control load can be balanced
• The optimization is a Mixed Integer Quadratic Programming problem because of the presence of the boolean variable α
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC

Extensions: performance, high level switching controller 16
$\mathcal{B}$
$y^o$
Switch
R-MPC, $\gamma_1$
R-MPC, $\gamma_2$
R-MPC, $\gamma_S$
Low level systems
(controlled actuators)
$x_{h+1} = Ax_h + \sum_{i=1}^{m} b^{(i)} u_h^{(i)}$
$y_h = c(x_h)$
$x \in \mathbb{X}, \ u^{(i)} \in \mathbb{U}^{(i)}$
Process (Wiener model)
Sampler
Performance vs robustness :
• Less robustness (a larger $\gamma_i$) enforces a faster response of the low level systems, thus it ensures better performance
• Feasibility (i.e., the small-gain condition) is guaranteed: if the actuators are not fast enough, an alert signal $\mathcal{B}$ is sent to the high level which switches to a more robust (smaller $\gamma_i$) mode
Robustness decrease
$\gamma_1 < \gamma_2 < \cdots < \gamma_S$
Performance improvement
A hybrid system: stability is ensured by a sufficiently large average dwell-time
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC

Extensions: reconfigurability (plug & play [Stoustrup '09]) 17
Robust MPC
(high level control unit)
$u^{(1)}$
$u^{(m)}$
$u^{(m+1)}$
Low level systems
(controlled actuators)
$\tilde{u}^{(1)}$
$\tilde{u}^{(m)}$
$\tilde{u}^{(m+1)}$
$x_{h+1} = Ax_h + \sum_{i=1}^{m+1} b^{(i)} u_h^{(i)}$
$x \in \mathbb{X}, \ u^{(i)} \in \mathbb{U}^{(i)}$
Process
Addition
Sampler
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC

Extensions: reconfigurability (plug & play [Stoustrup '09]) 18
Robust MPC
(high level control unit)
Low level systems
(controlled actuators)
$x_{h+1} = Ax_h + \sum_{i=1}^{m+1} b^{(i)} u_h^{(i)}$
$x \in \mathbb{X}, \ u^{(i)} \in \mathbb{U}^{(i)}$
Process
Addition
Sampler
Robust MPC
(high level control unit)
Low level systems
(controlled actuators)
$x_{h+1} = Ax_h + \sum_{i=1}^{m} b^{(i)} u_h^{(i)}$
$x \in \mathbb{X}, \ u^{(i)} \in \mathbb{U}^{(i)}$
Process
Replacement
Sampler
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC

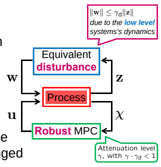
## Extensions: reconfigurability (plug & play) 19

Should one completely re-design the high level control unit ?

In the MPC approach reconfigurability is achieved if the auxiliary law can be kept unchanged

**Main idea:**

- The gain $\gamma_d$ is an abstraction of the low level system
- Different low level configurations characterized by the same (or similar) $\gamma_d$ can be considered
- Actuators can be substituted/added provided that $\gamma_d$ does not change. Otherwise a new "attenuation constraint" is added to the MPC problem
- In both cases (actuator substitution and addition) the auxiliary control law can be left (essentially) unchanged




Thus, reconfigurability properties are achieved !

**Remark:** the resulting control system switches among different stable configurations. Stability is preserved if proper *dwell-time* is guaranteed

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

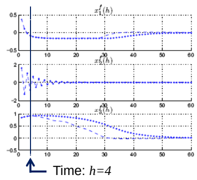
## Example 20

**Process** (basic configuration):

$$x^f(h+1) = \begin{bmatrix}0.4&0&0\\0&-0.8&0\\0&0&1.1\end{bmatrix} x^f(h) + \begin{bmatrix}1\\0\\1\end{bmatrix} u_1^f(h) + \begin{bmatrix}3\\1\\0\end{bmatrix} u_2^f(h).$$

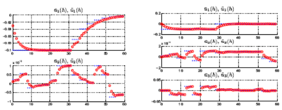
**Actuators** (low level system gain): $\gamma_d = 0.161$.

***Actuator addition:*** At time $h = 4$, a new actuator is added and $\gamma_d = 0.963 > 0.161$ (the supplementary "attenuation constraint" is needed in MPC).




State trajectories: basic configuration (dots) and with the added actuator (dashed line).

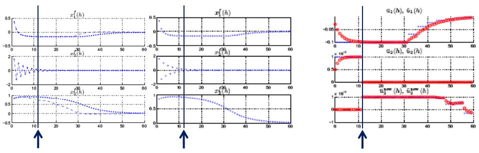
Control reference vs effective control action



Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Example 21

***Actuator replacement:*** At time $h = 12$, the second actuator is replaced with one guaranteeing a better attenuation level ($\gamma_d^{new} = 0.118 < 0.161$).




State trajectories:

basic configuration, and added actuator

with the replaced actuator

Control reference vs effective control action

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Conclusions 22

**A robust MPC approach has been presented for the design of two-layer hierarchical control systems**

- For constrained **linear discrete-time** systems
- The robust control approach allows to :
  - largely **decouple** the design at the two levels
  - to abstract subsystems with their *gain* and thus to obtain versatility resulting in numerous extensions (**reconfigurability**, **control allocation** problems, switching control for **performance** improvements)
- **Convergence results** have been established

**Papers:**

- B. Picasso, D. De Vito, R. Scattolini, P. Colaneri. An MPC approach to the design of two layer hierarchical control systems. *Automatica, Vol.46(5), pp. 823-831*, 2010.
- B. Picasso, C. Romani, R. Scattolini. Tracking control of Wiener models with hierarchical and switching MPC. *Submitted.*
- D. De Vito, B. Picasso, R. Scattolini. On the design of reconfigurable two layer hierarchical control systems with MPC. In *Proceedings of the American Control Conference, Baltimore, pp. 4704-4712*, 2010.

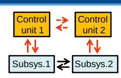
Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Outline 23

1. Introduction
2. Hierarchical MPC systems
   - Basic architecture
   - Extensions (performance & reconfigurability)
   - Conclusions
3. **Distributed MPC systems**
   - **A "tube-based", non cooperative DMPC algorithm**
   - **Conclusions**
4. Concluding remark

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

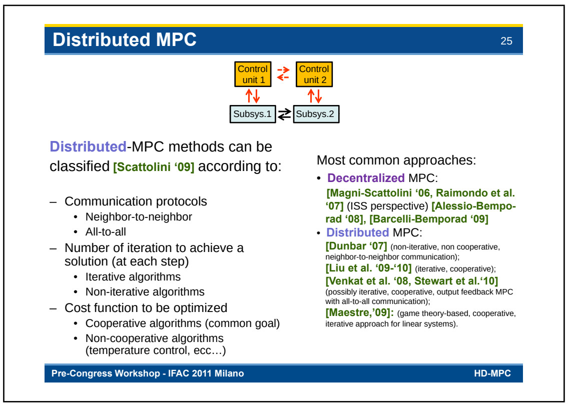
## Distributed MPC 24

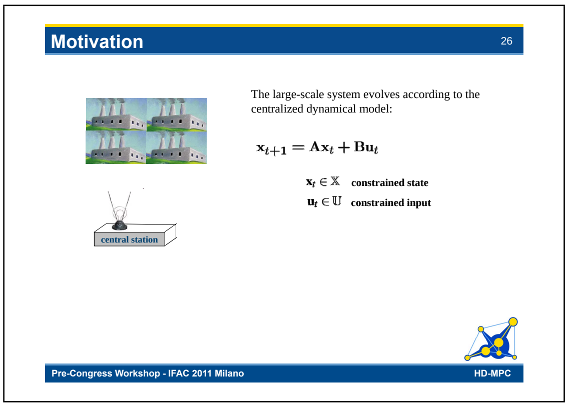


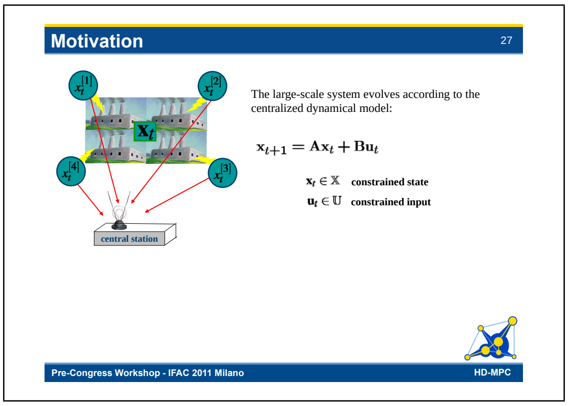

**Distributed**-MPC methods can be classified **[Scattolini '09]** according to:

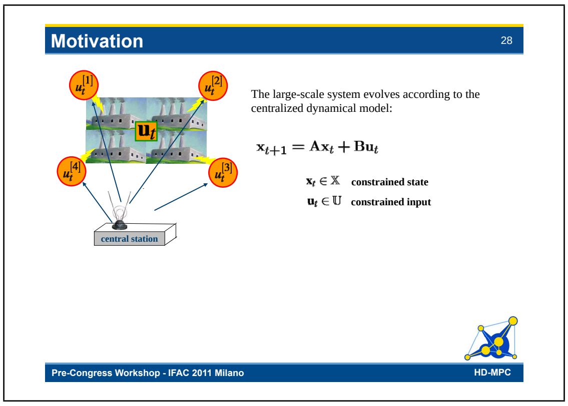
- Communication protocols
  - Neighbor-to-neighbor
  - All-to-all
- Number of iteration to achieve a solution (at each step)
  - Iterative algorithms
  - Non-iterative algorithms
- Cost function to be optimized
  - Cooperative algorithms (common goal)
  - Non-cooperative algorithms (temperature control, ecc…)

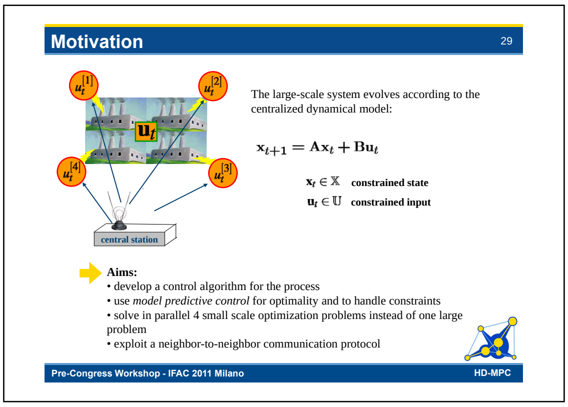
Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

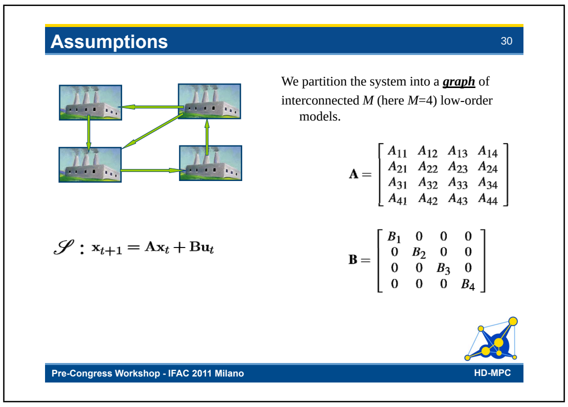

Distributed MPC
25
Control unit 1
Control unit 2
Subsys.1
Subsys.2
Distributed-MPC methods can be classified [Scattolini '09] according to:
– Communication protocols
• Neighbor-to-neighbor
• All-to-all
– Number of iteration to achieve a solution (at each step)
• Iterative algorithms
• Non-iterative algorithms
– Cost function to be optimized
• Cooperative algorithms (common goal)
• Non-cooperative algorithms (temperature control, ecc…)
Most common approaches:
• Decentralized MPC:
[Magni-Scattolini '06, Raimondo et al. '07] (ISS perspective) [Alessio-Bemporad '08], [Barcelli-Bemporad '09]
• Distributed MPC:
[Dunbar '07] (non-iterative, non cooperative, neighbor-to-neighbor communication);
[Liu et al. '09-'10] (iterative, cooperative);
[Venkat et al. '08, Stewart et al.'10] (possibly iterative, cooperative, output feedback MPC with all-to-all communication);
[Maestre,'09]: (game theory-based, cooperative, iterative approach for linear systems).
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC


Motivation
26
central station
The large-scale system evolves according to the centralized dynamical model:
$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$
$\mathbf{x}_t \in \mathbb{X}$ constrained state
$\mathbf{u}_t \in \mathbb{U}$ constrained input
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC


Motivation
27
$x_t^{[1]}$
$x_t^{[2]}$
$\mathbf{x}_t$
$x_t^{[4]}$
$x_t^{[3]}$
central station
The large-scale system evolves according to the centralized dynamical model:
$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$
$\mathbf{x}_t \in \mathbb{X}$ constrained state
$\mathbf{u}_t \in \mathbb{U}$ constrained input
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC

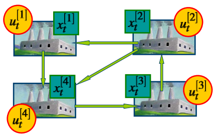

Motivation
28
$u_t^{[1]}$
$u_t^{[2]}$
$\mathbf{u}_t$
$u_t^{[4]}$
$u_t^{[3]}$
central station
The large-scale system evolves according to the centralized dynamical model:
$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$
$\mathbf{x}_t \in \mathbb{X}$ constrained state
$\mathbf{u}_t \in \mathbb{U}$ constrained input
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC


Motivation
29
$u_t^{[1]}$
$u_t^{[2]}$
$\mathbf{u}_t$
$u_t^{[4]}$
$u_t^{[3]}$
central station
The large-scale system evolves according to the centralized dynamical model:
$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$
$\mathbf{x}_t \in \mathbb{X}$ constrained state
$\mathbf{u}_t \in \mathbb{U}$ constrained input
Aims:
• develop a control algorithm for the process
• use model predictive control for optimality and to handle constraints
• solve in parallel 4 small scale optimization problems instead of one large problem
• exploit a neighbor-to-neighbor communication protocol
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC


Assumptions
30
We partition the system into a graph of interconnected M (here M=4) low-order models.
$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix}$
$\mathscr{S} : \mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$
$\mathbf{B} = \begin{bmatrix} B_1 & 0 & 0 & 0 \\ 0 & B_2 & 0 & 0 \\ 0 & 0 & B_3 & 0 \\ 0 & 0 & 0 & B_4 \end{bmatrix}$
Pre-Congress Workshop - IFAC 2011 Milano
HD-MPC

## Assumptions






We partition the system into a ***graph*** of interconnected $M$ (here $M$=4) low-order models.

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & A_{22} & A_{23} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & A_{33} & A_{34} \\ A_{41} & A_{42} & \mathbf{0} & A_{44} \end{bmatrix}$$

$$\mathscr{S} : \mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$$

$$\mathbf{B} = \begin{bmatrix} B_1 & 0 & 0 & 0 \\ 0 & B_2 & 0 & 0 \\ 0 & 0 & B_3 & 0 \\ 0 & 0 & 0 & B_4 \end{bmatrix}$$

$$\mathscr{S}_i : x_{t+1}^{[i]} = A_{ii}x_t^{[i]} + B_i u_t^{[i]} + \sum_{j\neq i} A_{ij}x_t^{[j]}$$



## DPC: the robust control approach



Large-scale system: $\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$

Graph of interconnected $M$ low-order subsystems:

$$x_{t+1}^{[i]} = A_{ii}x_t^{[i]} + B_i u_t^{[i]} + \sum_{j\neq i} A_{ij}x_t^{[j]}$$

$x_t^{[i]} \in \mathbb{X}_i$ local state constraints

$u_t^{[i]} \in \mathbb{U}_i$ local input constraints

$h_s^{[i]}(x_t^{[i]}, \mathbf{x}_t) \leq 0$



## DPC: the robust control approach



Large-scale system: $\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$

Graph of interconnected $M$ low-order subsystems:

$$x_{t+1}^{[i]} = A_{ii}x_t^{[i]} + B_i u_t^{[i]} + \sum_{j\neq i} A_{ij}x_t^{[j]}$$

$x_t^{[i]} \in \mathbb{X}_i$ local state constraints

$u_t^{[i]} \in \mathbb{U}_i$ local input constraints

$h_s^{[i]}(x_t^{[i]}, \mathbf{x}_t) \leq 0$

Each subsystem $i$

- has a reference trajectory $\tilde{x}_t^{[i]}$ and guarantees that $x_t^{[i]} - \tilde{x}_t^{[i]} \in \mathscr{E}_i$
- transmits, at each time, the nominal trajectory $\tilde{x}_t^{[i]}$ to its neghbors

$$x_{t+1}^{[i]} = A_{ii}x_t^{[i]} + B_i u_t^{[i]} + \sum_{j\neq i} A_{ij}\tilde{x}_t^{[j]} + \sum_{j\neq i} A_{ij}\overbrace{(x_t^{[j]} - \tilde{x}_t^{[j]})}^{\in \mathscr{E}_j}$$



## DPC: the robust control approach



Large-scale system: $\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t$

Graph of interconnected $M$ low-order subsystems:

$$x_{t+1}^{[i]} = A_{ii}x_t^{[i]} + B_i u_t^{[i]} + \sum_{j\neq i} A_{ij}x_t^{[j]}$$

$x_t^{[i]} \in \mathbb{X}_i$ local state constraints

$u_t^{[i]} \in \mathbb{U}_i$ local input constraints

$h_s^{[i]}(x_t^{[i]}, \mathbf{x}_t) \leq 0$

Each subsystem $i$

- has a reference trajectory $\tilde{x}_t^{[i]}$ and guarantees that $x_t^{[i]} - \tilde{x}_t^{[i]} \in \mathscr{E}_i$
- transmits, at each time, the nominal trajectory $\tilde{x}_t^{[i]}$ to its neghbors

$$x_{t+1}^{[i]} = A_{ii}x_t^{[i]} + B_i u_t^{[i]} + \sum_{j\neq i} A_{ij}\tilde{x}_t^{[j]} + \sum_{j\neq i} A_{ij}\overbrace{(x_t^{[j]} - \tilde{x}_t^{[j]})}^{\in \mathscr{E}_j}$$

$$= A_{ii}x_t^{[i]} + B_i u_t^{[i]} + \sum_{j\neq i} A_{ij}\tilde{x}_t^{[j]} + w_t^{[i]}$$

$w_t^{[i]} \in \mathbb{W}_i = \bigoplus_{j\neq i} A_{ij}\mathscr{E}_j$

constrained disturbance



## DPC: the robust control approach



**DPC** relies on the solution of $M$ robust MPC problems ($i$-DPC) with the tube-based approach presented in **[Mayne, Seron, Raković, *Automatica*, 2005]**

$i$-th "perturbed" model:

$$x_{t+1}^{[i]} = A_{ii}x_t^{[i]} + B_i u_t^{[i]} + \sum_{j\neq i} A_{ij}\tilde{x}_t^{[j]} + w_t^{[i]}$$

$i$-th nominal model:

$$\hat{x}_{t+1}^{[i]} = A_{ii}\hat{x}_t^{[i]} + B_i \hat{u}_t^{[i]} + \sum_{j\neq i} A_{ij}\tilde{x}_t^{[j]}$$

Assign $u_t^{[i]} = \hat{u}_t^{[i]} + K_i^{aux}(x_t^{[i]} - \hat{x}_t^{[i]})$

Define $z_t^{[i]} = x_t^{[i]} - \hat{x}_t^{[i]}$

⇒ $z_{t+1}^{[i]} = (A_{ii} + B_i K_i^{aux}) z_t^{[i]} + w_t^{[i]}$ $\quad w_t^{[i]} \in \mathbb{W}_i$

If $(A_{ii}+B_i K^{aux}_i)$ is as. stable, there exists a RPI (robust positively invariant) set $Z_i$ for all $i$. Therefore

$x_t^{[i]} - \hat{x}_t^{[i]} \in Z_i$ ⇒ $x_k^{[i]} - \hat{x}_k^{[i]} \in Z_i$ for all $k \geq t$



## DPC: the robust control approach



**MAIN UNDERLYING IDEA**

Guarantee that

$\hat{x}_{t+k}^{[i]} - \tilde{x}_{t+k}^{[i]} \in E_i, k = 0, \ldots, N-1$

$x_t^{[i]} - \hat{x}_t^{[i]} \in Z_i$

Guaranteed by suitable constraints in the optimization problem

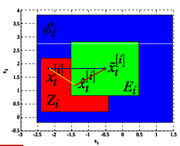
where $E_i \oplus Z_i \subseteq \mathscr{E}_i$

⇒ At time $t$:

$$x_t^{[i]} - \tilde{x}_t^{[i]} = (x_t^{[i]} - \hat{x}_t^{[i]}) + (\hat{x}_t^{[i]} - \tilde{x}_t^{[i]}) \in \mathscr{E}_i$$

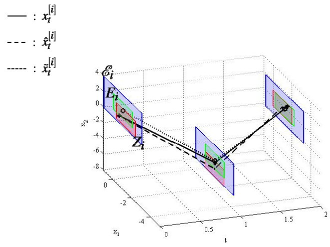
$$w_t^{[i]} = \sum_{j\neq i} A_{ij}(x_t^{[j]} - \tilde{x}_t^{[j]}) \in \bigoplus_{j\neq i} A_{ij}\mathscr{E}_j = \mathbb{W}_i$$

⇒ $x_{t+1}^{[i]} - \hat{x}_{t+1}^{[i]} \in Z_i$




By induction: ⇒ $x_{t+k}^{[i]} - \hat{x}_{t+k}^{[i]} \in Z_i, k = 1, \ldots, N$

## DPC: the robust control approach 37




Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Online phase 38

- Solve $M$ tube-based robust MPC problems ($i$-DPC), with dynamic constraints:

$$\hat{x}_{t+1}^{[i]} = A_{ii}\hat{x}_t^{[i]} + B_i\hat{u}_t^{[i]} + \sum_{j\neq i} A_{ij}\tilde{x}_t^{[j]}$$

- Coupling variables are the reference trajectories $\tilde{x}_k^{[i]}$ (known in all the prediction horizon $k=t,\dots,t+N-1$)
- Further constraint on the solution of the $i$-DPC:

$$\hat{x}_{t+k}^{[i]} - \tilde{x}_{t+k}^{[i]} \in E_i,\ k=0,\dots,N-1$$

$$x_t^{[i]} - \hat{x}_t^{[i]} \in Z_i$$

- Solution: $\hat{x}_{t/t}^{[i]}, \{\hat{u}_{k/t}^{[i]}\}_{k=t}^{t+N-1}$

➡ input to the real system: $u_t^{[i]} = \hat{u}_{t/t}^{[i]} + K_i^{aux}(x_t^{[i]} - \hat{x}_{t/t}^{[i]})$

➡ reference trajectory update: $\tilde{x}_{t+N}^{[i]} = \hat{x}_{t+N/t}^{[i]}$

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Online phase 39

**The optimization problem at time $t$**

Given - the ref. trajectory of $i$: $\tilde{x}_k^{[i]}, k=t,\dots,t+N-1$

- the ref. trajectories of its neighbors: $\tilde{x}_k^{[j]}, k=t,\dots,t+N-1$

$$\min_{\hat{x}_t^{[i]},\{\hat{u}_k^{[i]}\}_{k=t}^{t+N-1}} \sum_{k=t}^{t+N-1} l_i(\hat{x}_k^{[i]},\hat{u}_k^{[i]}) + V_i^F(\hat{x}_{t+N}^{[i]})$$

subject to

$$\hat{x}_{t+1}^{[i]} = A_{ii}\hat{x}_t^{[i]} + B_i\hat{u}_t^{[i]} + \sum_{j\neq i} A_{ij}\tilde{x}_t^{[j]}$$

$$x_t^{[i]} - \hat{x}_t^{[i]} \in Z_i$$

$$\hat{x}_k^{[i]} - \tilde{x}_k^{[i]} \in E_i \qquad k=t,\dots,t+N-1$$

| | | | | |
|---|---|---|---|---|
| $x_k^{[i]} \in \mathbb{X}_i$ | ➡ | $\hat{x}_k^{[i]} \in \hat{\mathbb{X}}_i$ | $\hat{\mathbb{X}}_i \oplus Z_i \subseteq \mathbb{X}_i$ | local state constraint |
| $u_k^{[i]} \in \mathbb{U}_i$ | ➡ | $\hat{u}_k^{[i]} \in \hat{\mathbb{U}}_i$ | $\hat{\mathbb{U}}_i \oplus KZ_i \subseteq \mathbb{U}_i$ | input constraint |
| $h_s^{[i]}(x_k^{[i]},\mathbf{x}_k) \leq 0$ | ➡ | $\hat{h}_s^{[i]}(\hat{x}_k^{[i]},\tilde{\mathbf{x}}_k) \leq 0$ | | coupled state constraint |
| $x_{t+N}^{[i]} \in \mathbb{X}_i^F$ | ➡ | $\hat{x}_{t+N}^{[i]} \in \hat{\mathbb{X}}_i^F$ | $\hat{\mathbb{X}}_i^F \oplus Z_i \subseteq \mathbb{X}_i^F$ | terminal constraint |

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

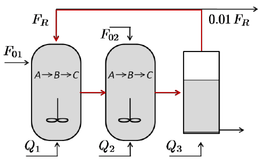
## Offline phase 40

1. Assign suitable decentralized stabilizing auxiliary control law.
2. Define suitable $i$-DPC optimization problem cost functions.
3. Define the sets $\mathcal{E}_i$, $E_i$, $Z_i$.
4. Initialize the reference trajectory and the set a suitable value for the prediction horizon $N$.

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Example 41

**Example: Chemical plant – reactor/separator process [Liu et al. 2010 ]**




The model is developed under the assumption of hydraulic equilibrium

States for each subsystem:

- $x_{Ai}$ : Concentration of compound $A$
- $x_{Bi}$ : Concentration of compound $B$
- $T_i$ : Temperature of subsystem $i$

Inputs for each subsystem:

- $Q_i$ :Heat

We use the linearized model around a given equilibrium point

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

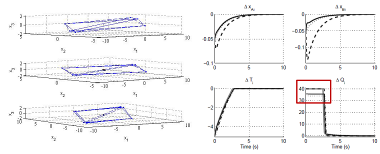
## Example 42

**Example: Chemical plant – reactor/separator process**

We study the response of linearized model to a perturbation of magnitude

$$\begin{bmatrix} \Delta x_{Ai} \\ \Delta x_{Bi} \\ \Delta T_i \end{bmatrix} = \begin{bmatrix} -0.05 \\ -0.05 \\ -5 \end{bmatrix}$$

Input constraints: $0 \leq Q_i \leq 50$ ➡ $-10 \leq \Delta Q_i \leq 40$




Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Conclusions

43

**A distributed predictive control algorithm has been presented**

- For **linear discrete-time** systems
- A **large scale control problem** has been subdivided into *M* low order, almost independent subproblems
- **Non cooperative algorithm**: each subsystem minimizes a local cost function
- **Neighbor-to-neighbor transmission** is required: low transmission burden
- Only **local knowledge on the systems** dynamics is required
- The algorithm is highly **scalable**: transmission, memory and computational loads do not grow.
- **Constraints** on state and input variables (local and global) can be handled
- **Convergence results** can be established

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Conclusions

44

**Advances:**

- Efficient algorithms for the initialization of DPC
- Output feedback DPC
- Extension for coping with non input-decoupled systems (*B* is not block diagonal)

**Wide area of application of DPC:**

- Independent systems with coupled constraints (e.g., transportation network)
- Cascade systems (e.g., simplified model of an HPV)
- Chemical plants with relevant couplings and feedbacks

**Future developments:**

- Explore applications in a plug-and-play architecture
- DPC for tracking

**Papers:**


- M. Farina, R. Scattolini. Distributed non-cooperative MPC with neighbor-to-neighbor communication. *Proceedings of the IFAC World Conference, 2011*.
- M. Farina, R. Scattolini. Distributed predictive control: a non-cooperative algorithm with neighbor-to-neighbor communication for linear systems. Submitted.
- M. Farina, R. Scattolini. An output feedback distributed predictive control algorithm. To appear in *Proceedings of the IEEE Conference on Decision and Control 2011*.

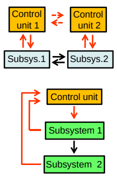

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## Concluding remark

45

Motivations for distributed / hierarchical control:

- Reduce the computational load
- Reduce the communication load
- Improve the robustness with respect to failures
  - in the transmission of information
  - in the central control unit
- Improve the modularity and the flexibility of the system
- Consider different goals at different time scales (Real-Time Optimization)
- Synchronize subsystems working at different time scales




**Both for distributed and hierarchical control systems, robust control turns out to be a suitable tool to deal with the main issues concerned with large-scale and complex systems.**

Pre-Congress Workshop - IFAC 2011 Milano HD-MPC

## 2.5 Distributed MPC based on game theory (J.M. Maestre, D. Limón, D. Muñoz de la Peña)

## Distributed Model Predictive Control Based on Game Theory

DISA

Departamento de Ingeniería de Sistemas y Automática
Universidad de Sevilla

J.M. Maestre, D. Limón and
D. Muñoz de la Peña

Pre-Congress Workshop - IFAC 2011
Hierarchical and Distributed Model Predictive Control, Algorithms and Applications
Milano, August 28, 2011

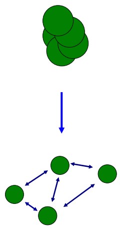
## Outline

- Introduction
- DMPC scheme for Two Agents
- DMPC scheme for Multiple Agents
- Conclusions and Further Research



## Introduction

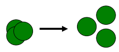
- Standard centralized control systems
  - Single controller
  - Flawless communication
- Implementation problems
  - System-wide model
  - Computation time
  - Large scale systems
    - Transportation networks
  - Communication constraints
  - Concerns about privacy
    - Supply chains
- Distributed control
  - Multiple controllers/agents
  - Communication
  - Partial system knowledge

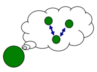


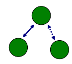


## Introduction

- Many control schemes have been proposed with differences on:
  - System decomposition
    - Systems coupled through the inputs
    - 2 and N subsystems
  - Information available
    - Local model and measurements
  - Communicational constraints
    - Agent to agent communication
    - Low communicational burden







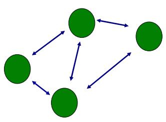


## Introduction

- Game theory

*"Game theory is a mathematical field that studies the process of interactive decision making, that is, situations in which there are several entities, namely players or agents, whose individual decisions determine jointly the final outcome."*

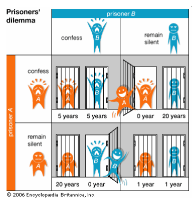
Cooperative games
Bargaining processes
Coalitions





## Introduction

- Cooperative games
  - Prisioners' dilemma

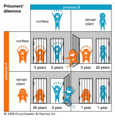



What would ***you*** do?

Cooperative game theory assume all the players *"cooperate"*

## Introduction

- Cooperative games
  - Prisioners' dilemma



Prisioner A point of view

| | B remains silent | B confesses |
|---|---|---|
| A remains silent | 1 year | 20 year |
| A confesses | goes free | 5 years |

Prisioner B point of view

| | B remains silent | B confesses |
|---|---|---|
| A remains silent | 1 year | goes free |
| A confesses | 20 years | 5 years |

"Greater good"

| | B remains silent | B confesses |
|---|---|---|
| A remains silent | **2 years** | 20 year |
| A confesses | 20 years | 10 years |

(global cost function)
(global knowledge)
(communicate)



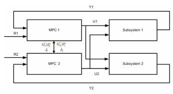
## DMPC scheme for two agents

- Assumptions
  - There is no coupling between the states of the agents, only in the actuation
  - Input and state constraints
  - Each agent has local information about the state and model
  - Agents optimize according to a local cost function

$$x_1(t+1) = A_1 x_1(t) + B_{11}u_1(t) + B_{12}u_2(t)$$
$$x_2(t+1) = A_2 x_2(t) + B_{21}u_1(t) + B_{22}u_2(t)$$

$$x_i \in \mathcal{X}_i, u_i \in \mathcal{U}_i, \ i = 1, 2$$

$$J_1(x_1, U_1, U_2) = \sum_{k=0}^{N-1} L_1(x_{1,k}, u_{1,k}) + F_1(x_{1,N})$$
$$J_2(x_2, U_2, U_1) = \sum_{k=0}^{N-1} L_2(x_{2,k}, u_{2,k}) + F_2(x_{2,N})$$

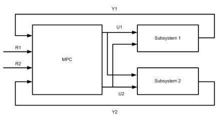




## DMPC scheme for two agents

- Centralized MPC

$$\min_{U_1,U_2} \ J(U_1, U_2, x_1, x_2) = J_1 + J_2$$



Definition of the global cost function ("greater good")



## DMPC scheme for two agents

- Algorithm
  - Each agent receives its state info $x_i$
  - Each agent evaluates $U^s$ shifting the last decided input trajectory $U^d$

$$U_1^d(t) = \begin{bmatrix} u_{1,0}^d \\ u_{1,1}^d \\ \vdots \\ u_{1,N-1}^d \end{bmatrix}, \ U_2^d(t) = \begin{bmatrix} u_{2,0}^d \\ u_{2,1}^d \\ \vdots \\ u_{2,N-1}^d \end{bmatrix} \qquad U_1^s(t) = \begin{bmatrix} u_{1,1}^d \\ u_{1,2}^d \\ \vdots \\ u_{1,N-1}^d \\ K_1 x_{1,N} \end{bmatrix}, \ U_2^s(t) = \begin{bmatrix} u_{2,1}^d \\ u_{2,2}^d \\ \vdots \\ u_{2,N-1}^d \\ K_2 x_{2,N} \end{bmatrix}$$

  - Each agent calculates its optimal control action assuming the other actuates according to the last agreed trajectory $U^s$

$$U_i^* = \min_{U_i} \ J_i(U_i, U_{nei}^s, x_i)$$
$$x_i(k+1) = A_i x_i(k) + B_{ii} u_i(k) + B_{i,nei} u_{nei}(k)$$
$$x_{i,0} = x_i$$
$$x_{i,k} \in X_i$$
$$x_{i,N} \in \Omega_i$$
$$u_{i,k} \in U_i$$



## DMPC scheme for two agents

- Algorithm
  - Then it calculates the wished action for the neighbor, assuming agent *i* will play the action calculated before

$$U_{nei}^w = \min_{U_{nei}} \ J_i(U_i^*, U_{nei}, x_i)$$
$$x_i(k+1) = A_i x_i(k) + B_{ii} u_i(k) + B_{i,nei} u_{nei}(k)$$
$$x_{i,0} = x_i$$
$$x_{i,k} \in X_i$$
$$x_{i,N} \in \Omega_i$$
$$u_{nei,k} \in U_{nei}$$

Each agent has computed two different trajectories using their local model and measurements



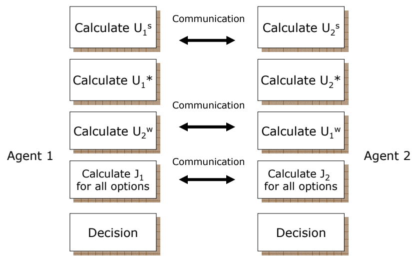
## DMPC scheme for two agents

- Algorithm
  - Agents communicate again and build a cooperative game corresponding to the following team problem

| $A1/A2$ | $U_2^s$ | $U_2^*$ | $U_2^w$ |
|---|---|---|---|
| $U_1^s$ Stable option | $J_1(x_1(t), U_1^s(t), U_2^s(t)) + J_2(x_2(t), U_1^s(t), U_2^s(t))$ | $J_1(x_1(t), U_1^s(t), U_2^*(t)) + J_2(x_2(t), U_1^s(t), U_2^*(t))$ | $J_1(x_1(t), U_1^s(t), U_2^w(t)) + J_2(x_2(t), U_1^s(t), U_2^w(t))$ |
| $U_1^*$ Selfish option | $J_1(x_1(t), U_1^*(t), U_2^s(t)) + J_2(x_2(t), U_1^*(t), U_2^s(t))$ | $J_1(x_1(t), U_1^*(t), U_2^*(t)) + J_2(x_2(t), U_1^*(t), U_2^*(t))$ | $J_1(x_1(t), U_1^*(t), U_2^w(t)) + J_2(x_2(t), U_1^*(t), U_2^w(t))$ |
| $U_1^w$ Altruist option | $J_1(x_1(t), U_1^w(t), U_2^s(t)) + J_2(x_2(t), U_1^w(t), U_2^s(t))$ | $J_1(x_1(t), U_1^w(t), U_2^*(t)) + J_2(x_2(t), U_1^w(t), U_2^*(t))$ | $J_1(x_1(t), U_1^w(t), U_2^w(t)) + J_2(x_2(t), U_1^w(t), U_2^w(t))$ |

  - Agents implement the first global minimum they find
  - The algorithm is repeated the next sampling time

## DMPC scheme for two agents






## DMPC scheme for two agents

- Application to a supply chain (MIT beer problem)
  - States
    - Stock
    - Unfulfilled order of stock
    - Backlog of unfulfilled orders
  - Manipulated variable
    - Orders
- Simulation scenarios
  - 4 different scenarios
- Comparison
  - Centralized MPC
  - Iterative MPC (based on information broadcast)

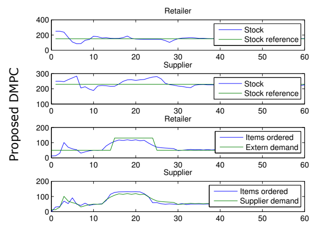


## DMPC scheme for two agents

| | $J$ | $T_{sim}$ |
|---|---|---|
| **Centralized** | 3.6179e+006 | 1.4187 |
| **DMPC** | 4.9827e+006 | 0.6246 |
| **Iter1** | 2.1866e+007 | 0.379715 |
| **Iter2** | 5.6999e+006 | 0.488593 |
| **Iter5** | 5.8449e+006 | 1.2611 |
| **Iter10** | 4.1679e+006 | 1.3750 |

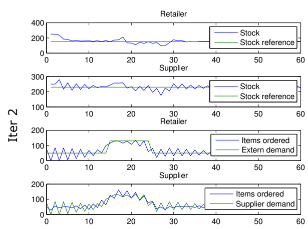


## DMPC scheme for two agents






## DMPC scheme for two agents

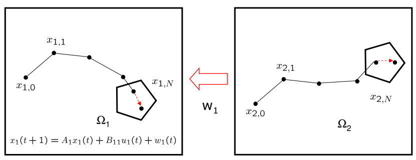





## DMPC scheme for two agents

$$U_i^* = \min_{U_i} \; J_i(U_i, U_{nei}^s, x_i)$$
$$x_i(k+1) = A_i x_i(k) + B_{ii} u_i(k) + B_{i,nei} u_{nei}(k)$$
$$x_{i,0} = x_i$$
$$x_{i,k} \in X_i$$
$$x_{i,N} \in \Omega_i$$
$$u_{i,k} \in U_i$$

Terminal region/constraint approach
Robust design approach
*Decentralized* properties (subspace $x_i$)



## DMPC scheme for two agents

- Stability theorem
  - Terminal cost / Local controller $J_1(x_1, U_1, U_2) = \sum_{k=0}^{N-1} L_1(x_{1,k}, u_{1,k}) + F_1(x_{1,N})$
    - Conditions for each subsystem and also for the overall system

$$F_1((A_1 + B_{11}K_1)x_1 + B_{12}K_2x_2) - F_1(x_1) + L_1(x_1, K_1x_1) \leq 0,\ \forall x_2 \in \Omega_2$$

$$F_2((A_2 + B_{22}K_2)x_2 + B_{21}K_1x_1) - F_1(x_2) + L_1(x_2, K_2x_2) \leq 0,\ \forall x_1 \in \Omega_1$$

$$F((A + BK)x) - F(x) + L(x, Kx) \leq 0,\ \forall x \in \Omega = \Omega_1 \times \Omega_1$$

  - Recursive feasibility

$$x_1 \in \Omega_1 \to (A_1 + B_{11}K_1)x_1 + B_{12}K_2x_2 \in \Omega_1,\ \forall x_2 \in \Omega_2$$
$$x_2 \in \Omega_2 \to (A_2 + B_{22}K_2)x_2 + B_{21}K_1x_1 \in \Omega_2,\ \forall x_1 \in \Omega_1$$
$$K_1x_1 \in U_1,\ \forall x_1 \in \Omega_1$$
$$K_2x_2 \in U_2,\ \forall x_2 \in \Omega_2$$
$$\Omega_1 \in X_1$$
$$\Omega_2 \in X_2$$

Note: Local controllers only depend on local state measurements

DMPC based on game theory 19/48

## DMPC scheme for two agents

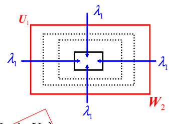
Design procedure based on robust positive invariance

$$\Omega(A, B, D, K, \mathcal{X}, \mathcal{U}, \mathcal{W})$$

$$x \in \Omega \to (A + BK)x + Dw \in \Omega,\ \forall w \in \mathcal{W}$$

$$Kx \in \mathcal{U}$$

$$\Omega \in \mathcal{X}$$




$$\Omega_1(\lambda_1, \lambda_2) = \Omega(A_1, B_{11}, B_{12}, X_1, K_1, \lambda_1 U_1, \lambda_2 U_2)$$
$$\Omega_2(\lambda_1, \lambda_2) = \Omega(A_2, B_{22}, B_{21}, X_2, K_2, \lambda_2 U_2, \lambda_1 U_1)$$

Convex optimization problem

$$\max_{\lambda_1 \in (0,1], \lambda_2 \in (0,1]} f(\Omega_1(\lambda_1, \lambda_2) \times \Omega_2(\lambda_1, \lambda_2))$$

DMPC based on game theory 20/48

## DMPC scheme for two agents

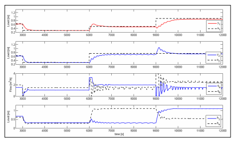
- Local state and model knowledge
- Cooperative solution based on a strategic team problem
- Two/three communications steps
  - Input trajectories
  - Cost function values
- In order to design a stabilizing controller the centralized model is needed
  - And an initial feasible solution!
- Approximate design procedure of jointly invariant sets
  - Parameterization of the input constraints

DMPC based on game theory 21/48

## DMPC scheme for two agents

- HD-MPC four-tank benchmark

"A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark". I. Alvarado, D. Limon, D. Muñoz de la Peña, J.M. Maestre, M.A. Ridao, H. Scheu, W. Marquardt, R.R Negenborn, B. De Schutter, F. Valencia and J. Espinosa. Journal of Process Control, 21:5, June 2011, 800-815, Special Issue on Hierarchical and Distributed Model Predictive Control.



DMPC based on game theory 22/48

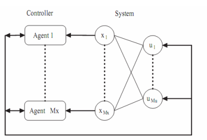
## DMPC scheme for multiple agents

- Assumptions
  - There is no coupling between the states of the agents, only in the actuation
  - Each agent has local information about the state and knows how it is affected by the different inputs
  - Input and state constraints
  - Inputs are not assigned to agents

$$x_i(t+1) = A_i x_i(t) + \textstyle\sum_{j \in n_i} B_{ij} u_j(t)$$

$$x_i \in \mathcal{X}_i,\ i = 1, \ldots, M_x$$

$$u_j \in \mathcal{U}_j,\ j = 1, \ldots, M_u$$




DMPC based on game theory 23/48

## DMPC scheme for multiple agents

Agents optimize according to a local cost function

$$J_i(x_i, \{U_j\}_{j \in n_i}) = \sum_{k=0}^{N-1} L_i(x_{i,k}, \{u_{j,k}\}_{j \in n_i}) + F_i(x_{i,N})$$

$$L_i(x_i, \{u_j\}_{j \in n_i}) = x_i^T Q_i x_i + \textstyle\sum_{j \in n_i} u_j^T R_{ij} u_j$$

$$F_i(x_i) = x_i^T P_i x_i$$

Control objective $\sum_{i=1}^{M_x} J_i(x_i(t), \{U_j(t)\}_{j \in n_i})$

GLOBAL PERFORMANCE INDEX

The different agents must reach an agreement on the value of the shared inputs

DMPC based on game theory 24/48

## DMPC scheme for multiple agents

- Proposed DMPC scheme
  - Subsystems coupled through the inputs
  - Each agent has only partial information of the system
  - Low communicational requirements
  - Cooperative solution
    - Cooperative algorithm from a game theory point of view
  - Guaranteed closed-loop stability properties
- Direct extension of the previous algorithm is not possible because of the combinatorial explosion
  - N agents with q proposals lead to $q^N$ options!
- Negotiation based scheme

DMPC based on game theory 25/48

## DMPC scheme for multiple agents

- Algorithm
  - Each agent receives its state info $x_i$
  - The agents communicate (if needed) to evaluate the initial value of the input trajectories $U_s$ (shifted inputs) from the latest decided input

$$U_i^d(t-1) = \begin{bmatrix} u_{j,0}^d \\ u_{j,1}^d \\ \vdots \\ u_{j,N-1}^d \end{bmatrix} \qquad U_j^s(t) = \begin{bmatrix} u_{j,1}^d \\ u_{j,2}^d \\ \vdots \\ u_{j,N-1}^d \\ \sum_{p \in m_j} K_{jp} x_{p,N} \end{bmatrix}$$

  - A number of proposals are made by a set of agents
    - A proposal consists of a future trajectory for a subset of inputs
    - A proposal is accepted if and only if it improves the costs for all the agents affected by that control action
    - After a predefined number of proposals are made, the latest agreed input trajectory is applied

DMPC based on game theory 26/48

## DMPC scheme for multiple agents

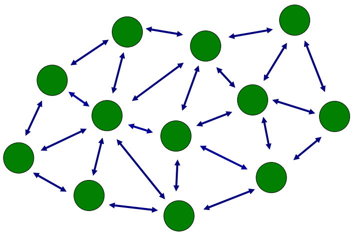
- Algorithm
  - In order to make a proposal, each agent calculates the optimal control action for a (sub)set of inputs that affect its dynamics

$$\begin{aligned} \{U_j^p(t)\}_{j \in n_p} = \ & \arg \min_{\{U_j\}_{j \in n_p}} J_p(x_p, \{U_j\}_{j \in n_p}) \\ & s.t. \\ & x_{p,k+1} = A_p x_{p,k} + \textstyle\sum_{j \in n_p} B_{pj} u_{j,k} \\ & x_{p,0} = x_i(t) \\ & x_{p,k} \in \mathcal{X}_p, \ k = 0, \dots N \\ & u_{j,k} \in \mathcal{U}_j, \ k = 0, \dots N-1, \ \forall j \in n_p \\ & x_{p,N} \in \Omega_p \\ & U_j = U_j^s(t), \ \forall j \notin P_p \end{aligned}$$

Different communication protocols: Round robin, asynchronous...

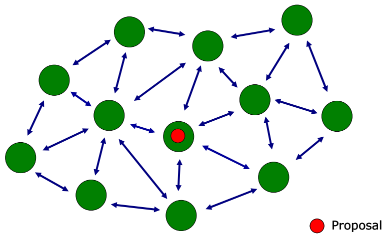
DMPC based on game theory 27/48

## DMPC scheme for multiple agents



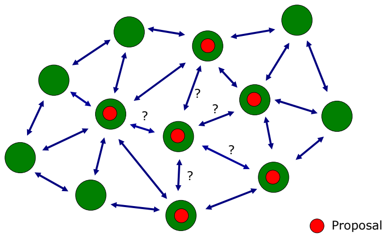
DMPC based on game theory 28/48

## DMPC scheme for multiple agents



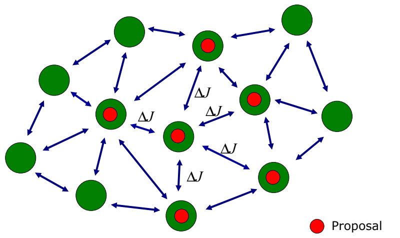

DMPC based on game theory 29/48

## DMPC scheme for multiple agents



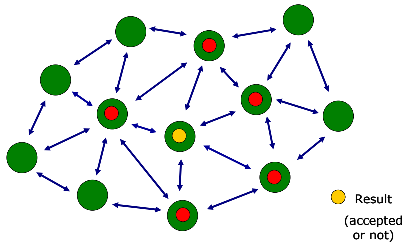

DMPC based on game theory 30/48

## DMPC scheme for multiple agents



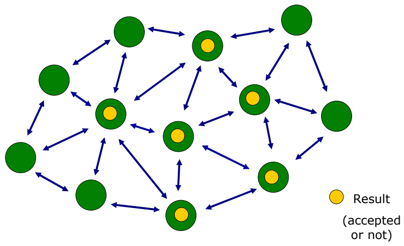

DMPC based on game theory 31/48

## DMPC scheme for multiple agents




DMPC based on game theory 32/48

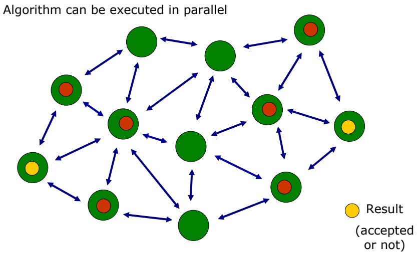
## DMPC scheme for multiple agents




DMPC based on game theory 33/48

## DMPC scheme for multiple agents

Algorithm can be executed in parallel




DMPC based on game theory 34/48

## DMPC scheme for multiple agents

- Stability theorem
  - Terminal cost / Local controllers
    - Stabilizing linear controller (centralized or decentralized)

$$\sum_{i=1}^{M_x} F_i(A_i x_i + \sum_{j\in n_i} B_{ij} \sum_{p\in m_j} K_{jp}x_p) - F_i(x_i) + L_i(x_i, \{\sum_{p\in m_j} K_{jp}x_p\}_{j\in n_i}) \le 0$$

  - Recursive feasibility

If $x_i \in \Omega_i$ for all i then

$$A_i x_i + \sum_{j\in n_i} B_{ij} \sum_{p\in m_j} K_{jp}x_p \in \Omega_i$$
$$\sum_{p\in m_j} K_{jp}x_p \in U_j$$
$$\Omega_i \in X_i$$

  - Jointly invariant set: robust stability w.r.t. neighbors
    - Standard LMI design techniques
    - Centralized model is needed

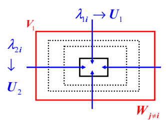
DMPC based on game theory 35/48

## DMPC scheme for multiple agents

$$x_i(t+1) = A_i x_i(t) + B_i v_i(t) + E_i w_i(t)$$

$$B_i v_i(t) = \sum_{j\in n_i} B_{ij} K_{ji} x_i$$
$$K_{ji} x_i \in \lambda_{ji} \mathcal{U}_j$$
$$\sum_{i\in m_j} \lambda_{ji} \le 1$$

$$E_i w_i(t) = \sum_{j\in n_i} B_{ij} \sum_{p\in m_j - \{i\}} K_{jp} x_p$$




$$V_i(\Lambda) = \lambda_{1i}\mathcal{U}_1 \times \lambda_{2i}\mathcal{U}_2 \times \ldots \times \lambda_{M_u i}\mathcal{U}_{M_u}$$
$$W_i(\Lambda) = (\sum_{p\in m_1 - \{i\}} \lambda_{1p})\mathcal{U}_1 \times (\sum_{p\in m_2 - \{i\}} \lambda_{2p})\mathcal{U}_2 \times \ldots \times (\sum_{p\in m_{M_u} - \{i\}} \lambda_{M_u p}\mathcal{U}_{M_u})$$

Convex optimization problem

$$\max_{\lambda_{ji}} f(\Omega_1 \times \Omega_2 .. \times \Omega_{M_x})$$
$$\Omega_i = \Omega(A_i, B_i, E_i, \mathcal{X}_i, K_i, \mathcal{V}_i(\Lambda), \mathcal{W}_i(\Lambda))$$
$$\lambda_{ji} \in (0,1),\ \forall j, i$$
$$\sum_{i\in m_j} \lambda_{ji} \le 1,\ \forall i$$

DMPC based on game theory 36/48

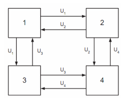
## DMPC scheme for multiple agents

- Local state and model knowledge
- Cooperative solution based on negotiation
- Multiple communications with neighbors
  - Input trajectories
  - Cost function values
- Parallel implementation
- In order to design a stabilizing controller the centralized model is needed
  - And an initial feasible solution!
- Approximate design procedure of jointly invariant sets
  - Parameterization of the input constraints



## DMPC scheme for multiple agents

- Example
  - Four coupled systems




$$A_1 = \begin{bmatrix} 1 & 0.8 \\ 0 & 0.7 \end{bmatrix}, B_{11} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, B_{12} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix}, B_{13} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 1 & 0.6 \\ 0 & 0.7 \end{bmatrix}, B_{21} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix}, B_{22} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, B_{24} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0.9 \\ 0 & 0.8 \end{bmatrix}, B_{31} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix}, B_{33} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, B_{34} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 1 & 0.8 \\ 0 & 0.5 \end{bmatrix}, B_{42} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix}, B_{43} = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix}, B_{44} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

  - Bounds on states and inputs

$$|x_1|_\infty \le 1,\ |x_2|_\infty \le 2,\ |x_3|_\infty \le 1,\ |x_4|_\infty \le 2$$
$$|u_1|_\infty \le 1,\ |u_2|_\infty \le 1,\ |u_3|_\infty \le 1,\ |u_4|_\infty \le 1$$

  - Cost functions

$$J_i(x_i, \{U_j\}_{j\in n_i}) = \sum_{k=0}^{N-1} L_i(x_{i,k}, \{u_{j,k}\}_{j\in n_i}) + F_i(x_{i,N})$$
$$L_i(x_i, \{u_j\}_{j\in n_i}) = x_i^t Q_i x_i + \sum_{j\in n_i} u_j^t R_{ij} u_j$$
$$F_i(x_i) = x_i^T P_i x_i$$

$$Q_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, R_{ij} = 10$$

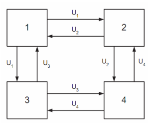
Ki, Pi have to be designed



## DMPC scheme for multiple agents

LMI design:

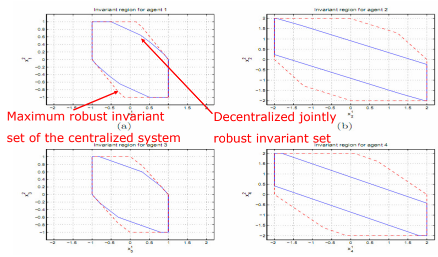
$$K = \begin{bmatrix} -0.2732 & -0.5935 & -0.0065 & -0.0112 & -0.0055 & -0.0151 & 0 & 0 \\ -0.0122 & -0.0263 & -0.2891 & -0.5024 & 0 & 0 & -0.0138 & -0.0216 \\ -0.0025 & -0.0052 & 0 & 0 & -0.2463 & -0.6878 & -0.0027 & -0.0042 \\ 0 & 0 & -0.0118 & -0.0204 & -0.0100 & -0.0276 & -0.3081 & -0.4845 \end{bmatrix}$$




$$P = \begin{bmatrix} 4.9218 & 5.7645 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5.7645 & 11.3025 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5.6585 & 5.4284 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5.4284 & 8.8248 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4.4536 & 5.8125 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5.8125 & 13.7428 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5.6128 & 5.8004 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5.8004 & 8.9564 \end{bmatrix}$$

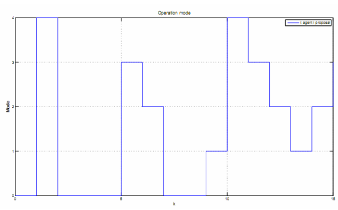


## DMPC scheme for multiple agents



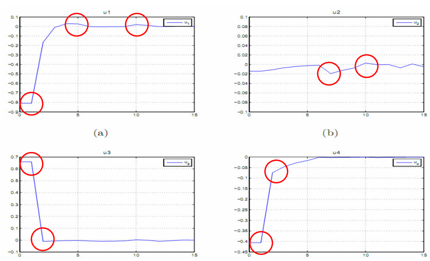



## DMPC scheme for multiple agents





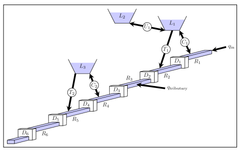

## DMPC scheme for multiple agents



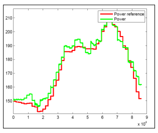
## DMPC scheme for multiple agents

- Control benchmark of a Hydro Power Plant

Nonlinear system
Power reference tracking
Profit maximization



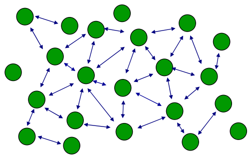
DMPC with 8 agents
Linear model
Two time scales (state decoupling)



DMPC based on game theory 43/48

## Food for thought…

Questions
-Do all the links have to be enabled all the time?
-How to divide profits/costs between the agents?
-Which are the most relevant agents/links?

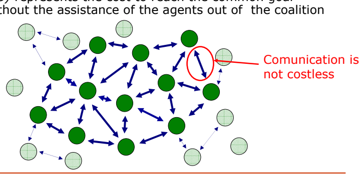


DMPC based on game theory 44/48

## Food for thought…

- A cooperative game is defined by…
  - A set of agents *N={1,2,…,n}*
    - *Separated into coalitions S*
  - A *characteristic* function *v* that assigns a value to each of the possible $2^n$ coalitions
    - *v(S)* represents the cost to reach the common goal without the assistance of the agents out of the coalition

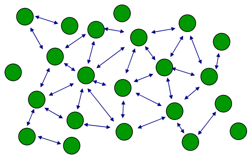
V(s)




DMPC based on game theory 45/48

## Food for thought…

An application of Cooperative Game Theory to Distributed Control. J. M. Maestre, D. Muñoz de la Peña, A. Jiménez Losada, E. Algaba Durán, E. F. Camacho. Proceedings of the 18th IFAC World Congress.



DMPC based on game theory 46/48

## Related publications

- Distributed model predictive control based on a cooperative game. J. M. Maestre, D. Muñoz de la Peña, E. F. Camacho. Optimal Control Applications and Methods, 32:2, March/April 2011, 153–176.
- Distributed model predictive control based on agent negotiation, J.M. Maestrea, D. Muñoz de la Peña, E.F. Camacho and T. Alamo. Journal of Process Control, 21:5, June 2011, 685-697.
- A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark. I. Alvarado, D. Limon, D. Muñoz de la Peña, J.M. Maestre, M.A. Ridao, H. Scheu, W. Marquardt, R.R Negenborn, B. De Schutter, F. Valencia and J. Espinosa. Journal of Process Control, 21:5, June 2011, 800-815.
- An application of Cooperative Game Theory to Distributed Control. J. M. Maestre, D. Muñoz de la Peña, A. Jiménez Losada, E. Algaba Durán, E. F. Camacho. Proceedings of the 18th IFAC World Congress.

DMPC based on game theory 47/48

## The end

Thanks for your attention!

Questions, suggestions, comments…

DMPC based on game theory 48/48

## 2.6 Distributed model predictive control by primal decomposition (W. Marquardt, H. Scheu)

AACHENER VERFAHRENSTECHNIK

# Distributed Model Predictive Control by Primal Decomposition

Wolfgang Marquardt & Holger Scheu

IFAC World Congress, 2011, Milan
Pre-Congress Workshop:
"Hierarchical and Distributed Model-Predictive Control"

RWTH

## Motivation and Background

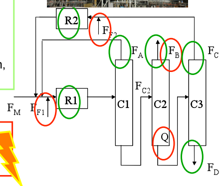
- **Chemical & energy process plants**
  - large-scale, structured
  - nonlinear, stiff
- **Process control and operations**
  - industrial state of the art
    - decentralized (PID) control & supervisory control
    - linear (centralized) MPC using step response or state space models from plant tests)
  - (selected) research activities
    - nonlinear centralized MPC and RHE using first principles models
    - dynamic real-time optimization (DRTO)
    - hierarchical or/and decentralized optimal control (MPC, DRTO) and matching nonlinear data/model reconciliation & state estimation



## Industrial Case Study (1)

**Large-scale industrial process (Shell):**
- How should decentralized control scheme be designed for a range of operating conditions and transitions in between?
- How fast can plant be moved from operating point A to B?
- 2 reactors, 3 distillation columns
- rigorous model including base layer control system: 14.000 DAEs
- 4 controls & 6 path constraints for transition, long time horizon >> 24 hrs

**Optimal transition control:**
- complexity estimate (single shooting): NLP with 100 Mio embedded DAEs






## Industrial Case Study (2)

Computational results: adaptive discretization and parallelization

Discretization of control 3
- Initial guess: 25 parameters
- Adaptive parameterization at final solution: 129 parameters
- Equivalent non-adaptive parameterization: 3072 parameters

→ 95% (or 41 million) equations eliminated by adaptive refinement!

- Calculation time per sensitivity integration: ~ 7500 sec
- Total computation times (adaptive, serial): > 1 month
- Total computation times (adaptive, parallel, 8 CPUs): ~ 1 week

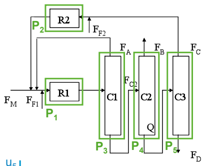
**Optimal solution (offline) successful!**
**Savings of 50 k€ per transition!**
**Is dynamic real-time optimization feasible?**

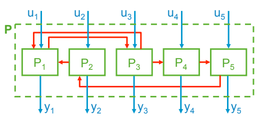
(Hartwich, Marquardt, 2010)



## Control of Process Plants (1)

- Process plants can naturally be decomposed into subsystems $P_i$
  - interconnecting variables: flows, i.e. rate, conc., temp., etc.
  - local inputs: flow rates, etc.
  - local outputs: measurements and interconnecting flows

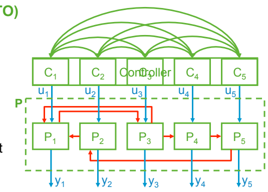





## Control of Process Plants (2)

- **Centralized MPC (or DRTO)**
  - optimal and stable
  - large-scale problem
- **Decentralized MPC (or DRTO)**
  - small-scale problems
  - optimality and stability not guaranteed
- **Distributed MPC (or DRTO)**
  - small-scale problems
  - optimality and stability can be guaranteed (if properly set-up)
  - communication required




(Scattolini, 2009)

## Classic Approach – Dual Decomposition (1)

- Consider the convex NLP

$$\min_p \sum_{i=1}^N \Phi_i(p_i), \qquad p' = [p'_1, ..., p'_N],$$
$$\text{s.t.} \sum_{i=1}^N c_i(p_1) \geq 0.$$

- decomposed into subproblems, with primal problems

$$\min_{p_i} \underbrace{\Phi_i(p_i) - \lambda\, c_1(p_i)}_{=L_i(p_i,\lambda)}, \ \forall i \in 1, ..., N$$

- and dual problem

$$\max_\lambda h(\lambda) \overset{def}{=} \min_p \sum_{i=1}^N L_i(p_i, \lambda)$$

→ **iterate to convergence**

(Lasdon, 1970)

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 7

## Classic Approach – Dual Decomposition (2)

- Primal problems

$$\min_{p_i} \underbrace{\Phi_i(p_i) - \lambda\, c_1(p_i)}_{=L_i(p_i,\lambda)}, \ \forall i \in 1, ..., N$$

  - cost functions and constraint functions are additive
  - straight forward implementation

- Dual problem

$$\max_\lambda h(\lambda) \overset{def}{=} \min_p \sum_{i=1}^N L_i(p_i, \lambda)$$

  - **main challenge** for the solution in dual decomposition
  - **normally requires many iterations**

  - convergence can be proven under convexity assumptions

(Lasdon, 1970)

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 8

## Sensitivity-Driven Decomposition (1)

- Consider a more general NLP:

$$\min_p \sum_{i=1}^N \Phi_i(p),$$
$$\text{s.t. } c_i(p) \geq 0, \quad \forall i$$

neither constraints nor objective functions of subsystems are additive!

(Scheu and Marquardt 2011a)

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 9

## Sensitivity-Driven Decomposition (2)

- Consider a more general NLP:

$$\min_p \sum_{i=1}^N \Phi_i(p),$$
$$\text{s.t. } c_i(p) \geq 0, \quad \forall i$$

- Parallel iterative solution using decomposed subproblems

$$\min_{p_i} \Phi_i^*(p)$$
$$\text{s.t. } c_i(p) \geq 0,$$

with the strictly convex objective functions

$$\Phi_i^* = \Phi_i(p) + \left[ \sum_{\substack{j=1 \\ j\neq i}}^N \left.\frac{d\Phi_j}{dp_i}\right|_{p^{[k]}}^T - \lambda_j^{[k]T} \left.\frac{dc_j}{dp_i}\right|_{p^{[k]}} \right] (p_i - p_i^{[k]})$$

iterations

(Scheu and Marquardt 2011a)

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 10

## Why Might this Decomposition Work?

- Let us look at the NCO for the (centralized) NLP

$$\frac{\partial L}{\partial p} = \sum_{i=1}^N \left( \frac{\partial \Phi_i}{\partial p} - \lambda_i \frac{\partial c_i}{\partial p} \right) = 0,$$

Condition depends only on first order sensitivities

$$c_i(p) \geq 0, \ \forall i,$$
$$\lambda_i \geq 0, \ \forall i,$$
$$\lambda_i c_i(p) = 0, \ \forall i.$$

Directly guaranteed by the subproblems

- Proof of optimality requires comparison of the NCO for the centralized problem and the decomposed problem.

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 11

## Theorem on Optimality

- Assumptions on centralized NLP:
  - cost functions $\Phi_i$ are strictly convex
  - constraint functions $c_i$ are concave

$$\min_p \sum_{i=1}^N \Phi_i(p),$$
$$\text{s.t. } c_i(p) \geq 0, \quad \forall i$$

- Further assumptions
  - $p^*$ solves the centralized NLP and satisfies LICQ
  - distributed algorithm converges and its minimizer satisfies the LICQ

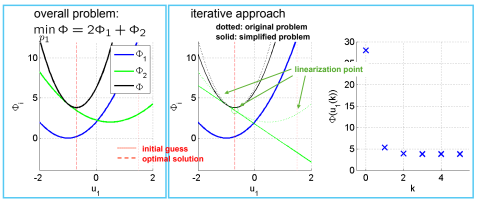
- Then, the minimizer $p^{[k]}$, $k \to \infty$, of the distributed problem and the minimizer $p^*$ of the centralized problem are the same, i.e. $\lim_{k\to\infty} p^{[k]} = p^*$.

(Scheu and Marquardt 2011a)

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 12

## Graphical Interpretation

$$\Phi_i^* = \Phi_i + \left[ \sum_{\substack{j=1\\ j\neq i}}^{N} \frac{\mathrm{d}\Phi_j}{\mathrm{d}p_i}\bigg|_{p^{[k]}} \right] (p_i - p_i^{[k]})$$






## Linear Continuous-Time Systems (1)

- Finite-horizon linear continuous-time optimal control problem:

$$\min_{x,u} \frac{1}{2}\int_{t_0}^{t_f} \left(\|x(t)\|_Q^2 + \|u(t)\|_R^2\right) dt,$$
$$\text{s.t. } \dot{x}(t) = Ax(t) + Bu(t), \quad t \in (t_0, t_f],$$
$$x(t_0) = x_0,$$
$$x \in X, \quad u \in U$$

- Transcribe into QP

$$\min_p \sum_{i=1}^{N} \frac{1}{2} p' H_i p + f_i' p$$
$$\text{s.t. } 0 \leq A_i p + b_i, \quad \forall i$$

(Scheu and Marquardt 2011a)



## Sketch of Transcription

1. Discretize the input variables
$$u_{i,j}(t) = \sum_l p_{i,j,l}\ \phi_l(t)$$
2. Solve the state variables $x(k)$ for the input parameters $p$ and the initial condition $x_0$ in discrete time, i.e.
$$x(k) = \mathbf{T}\,p + \mathbf{S}\,x_0$$
3. Transform continuous-time cost function into discrete cost function (Pannocchia et al. 2010)
$$\int_{t_0}^{t_f} x^T \mathcal{Q}_i x + u^T \mathcal{R}_i u d\tau = \sum_{\eta=0}^{\gamma-1} x(\eta)^T \mathcal{Q}^0(\eta) x(\eta) + p(\eta)^T \mathcal{R}^0(\eta) p(\eta) + 2x(\eta)^T \mathcal{S}^0(\eta) p(\eta)$$
4. Substitute $x(k)$ in the discrete cost function



## Linear Continuous-Time Systems (2)

- Transcribe into QP

$$\min_p \sum_{i=1}^{N} \frac{1}{2} p' H_i p + f_i' p$$
$$\text{s.t. } 0 \leq A_i p + b_i, \quad \forall i$$

$$x(t_0) = x_0,$$
$$x \in X, \quad u \in U$$

- Apply sensitivity-driven decomposition and coordination:

$$\min_{p_i} \Phi_i^* \stackrel{def}{=} \frac{1}{2} \tilde{p}_i^{[k]\ T} \boldsymbol{H}^i \tilde{p}_i^{[k]} + \tilde{p}_i^{[k]\ T} \boldsymbol{f}^i$$
$$+ \left[ \sum_{\substack{j=1\\ j\neq i}}^{N} \left( \left[\boldsymbol{H}_{i1}^j \ \ \ldots \ \ \boldsymbol{H}_{iN}^j\right] p^{[k]} + \boldsymbol{f}_i^j - \boldsymbol{A}_i^j \lambda_j^{[k]} \right) \right]^T (p_i - p_i^{[k]}),$$
$$\text{s.t. } c_i(\tilde{p}_i^{[k]}) = \boldsymbol{A}^{i\ T} \tilde{p}_i^{[k]} + \boldsymbol{b}^i \geq 0$$

(Scheu and Marquardt 2011a)



## Convergence Analysis

- Algorithm defines a fixed point iteration method, analysis based on the KKT NCO

$$\begin{bmatrix} p^{[k+1]} \\ \lambda^{[k+1]} \end{bmatrix} = - \begin{bmatrix} \boldsymbol{H}_{\text{diag}} & -\boldsymbol{A}_{\text{diag}} \\ -\boldsymbol{A}_{\text{diag}}^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{j=1}^N \boldsymbol{H}^j & -\boldsymbol{A} \\ -\boldsymbol{A}^T & 0 \end{bmatrix} \begin{bmatrix} p^{[k]} \\ \lambda^{[k]} \end{bmatrix} + \begin{bmatrix} p^{[k]} \\ \lambda^{[k]} \end{bmatrix}$$
$$- \begin{bmatrix} \boldsymbol{H}_{\text{diag}} & -\boldsymbol{A}_{\text{diag}} \\ -\boldsymbol{A}_{\text{diag}}^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{j=1}^N \boldsymbol{f}^j \\ -\boldsymbol{b} \end{bmatrix}$$

- Small-gain theorem can be applied, convergence for

$$L = \left\| \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} - \begin{bmatrix} \boldsymbol{H}_{\text{diag}} & -\boldsymbol{A}_{\text{diag}} \\ -\boldsymbol{A}_{\text{diag}}^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{j=1}^N \boldsymbol{H}^j & -\boldsymbol{A} \\ -\boldsymbol{A}^T & 0 \end{bmatrix} \bigg|_{\mathcal{A}} \right\| < 1$$

(Scheu and Marquardt 2011a)



## Enforce Convergence

- Further modification of the cost function

$$\Phi_i^{+} = \Phi_i^* + \frac{1}{2}(p_i - p_i^{[k]})' \Omega_i (p_i - p_i^{[k]})$$

- → constant L does also depend on **Ω_i**:

$$L = \left\| \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} - \begin{bmatrix} \boldsymbol{H}^{\Omega}_{\text{diag}} & -\boldsymbol{A}_{\text{diag}} \\ -\boldsymbol{A}_{\text{diag}}^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{j=1}^N \boldsymbol{H}^j & -\boldsymbol{A} \\ -\boldsymbol{A}^T & 0 \end{bmatrix} \bigg|_{\mathcal{A}} \right\| < 1$$

- gradient-free optimization (Wegstein, 1958; Westerberg et al., 1979)
- generalization of proximal minimization algorithm (Rockafellar 1976; Censor 1992)

## Sensitivity-Driven Distributed MPC (S-DMPC)

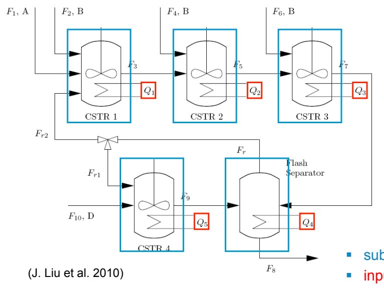
In closed loop, do on each horizon:

1. Measure or estimate the current system state.
2. Transcribe the optimal control problem into QP.
3. Select
   - initial parameters $p^{[0]}(h)$ and
   - initial Lagrange multipliers $\lambda^{[0]}(h)$.
   - Warm start based on preceding horizon.
4. Apply the distributed QP algorithm described before.
5. Apply the calculated optimal control inputs $u_{i,j}(t) = \sum_l p_{i,j,l}\ \phi_l(t)$ to the plant.

**coooperative, iterative, optimal on convergence, neighbor-to-neighbor communcation**



## Illustrative Example – Alkylation of Benzene




(J. Liu et al. 2010)

- subsystems
- inputs



## Sketch of Mathematical Model

**For each subsystem:**

- Mass balances for each species and energy balance

$$\begin{bmatrix} \frac{\mathrm{d}\,c_{Ai}}{\mathrm{d}t} \\ \frac{\mathrm{d}\,c_{Bi}}{\mathrm{d}t} \\ \frac{\mathrm{d}\,c_{Ci}}{\mathrm{d}t} \\ \frac{\mathrm{d}\,c_{Di}}{\mathrm{d}t} \\ \frac{\mathrm{d}\,T_i}{\mathrm{d}t} \end{bmatrix} = f_i(\ldots)$$

"Medium-scale" DAE system:
- 25 differential equations
- ~100 algebraic equations

**For stirred tank reactors:**

- nonlinear reaction kinetics

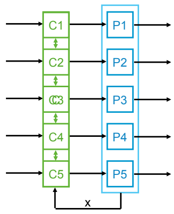
**For flash separator:**

- nonlinear phase equilibrium and physical property models



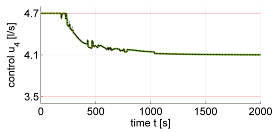
## Sketch of Controller Design

- Nonlinear process model
- Full state feedback
- Linear controller, based on linearization of nonlinear model
  - centralized
  - distributed
- no further disturbances, but plant-model mismatch
- set-point tracking

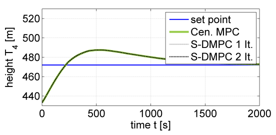





## Results




S-DMPC provides the same controller performance as a centralized MPC




Solve 5 small QP in parallel instead of 1 large QP

→ faster computation possible

(Scheu and Marquardt, 2011a)



## Linear Discrete-Time Systems

- Finite horizon discrete-time linear optimal control problem:

$$\min_{x,u} \frac{1}{2} \sum_{k=k'}^{k'+K-1} \left(\|x(k)\|_Q^2 + \|u(k)\|_R^2\right) + \|x(h)\|_P^2,$$
$$\text{s.t. } x(k+1) = Ax(k) + Bu(k), \quad k = k', \ldots, k'+K-1,$$
$$x(k') = x_{k'},$$
$$x(k) \in X, u(k) \in U$$

- Write as QP

$$\min_p \sum_{i=1}^{N} \frac{1}{2} p' H_i p + f_i' p$$
$$\text{s.t. } 0 \le A_i p + b_i, \quad \forall i,$$
$$0 = A_i^{eq} p + b_i^{eq}, \quad \forall i$$

- Apply sensitivity-driven coordination

(Scheu & Marquardt 2011b)

## Continuous-time vs. discrete-time

| Continuous-time | Discrete-time |
|---|---|
| also possible for higher order input representations | only piecewise constant inputs |
| non-uniform control-grid possible | uniform control-grid |
| system couplings are solved during transcription | system couplings are included in equality-constraints |
| couplings could also be included in finite number of equality-constraints | couplings could also be solved by transcription |
| most natural for nonlinear case | difficult to extent to nonlinear cases |



## Case Study

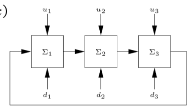
- Discrete-time linear system with unknown disturbances

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

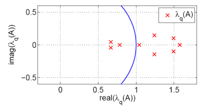
- where

$$A = \begin{bmatrix} A_{11} & 0 & A_0 \\ A_0 & A_{22} & 0 \\ 0 & A_0 & A_{33} \end{bmatrix}$$

$$B = D = \begin{bmatrix} B_0 & 0 & 0 \\ 0 & B_0 & 0 \\ 0 & 0 & B_0 \end{bmatrix}$$



- 9 differential state variables
- 3 scalar inputs
- 3 scalar disturbances
- unstable system dynamics:



(Scheu and Marquardt 2011b)



## MPC Setup

- Centralized MPC – 1 monolithic controller with full system knowledge, **large QP**
- Decentralized MPC – 3 independent controllers, **small QP**
- Dual Decomposition – 3 low layer controller, 1 coordinator, **small QP**
- S-DMPC – 3 cooperative controllers, **small QPs**

- Disturbances

$$d_1(k) = \begin{cases} 0.1, & \text{for } 75 \le k \le 150 \\ 0, & \text{else} \end{cases},$$

$$d_2(k) = \begin{cases} 0.1, & \text{for } 225 \le k \le 300 \\ 0, & \text{else} \end{cases},$$

$$d_3(k) = \begin{cases} 0.1, & \text{for } 375 \le k \le 450 \\ 0, & \text{else} \end{cases}$$



## MPC Setup (cont.)

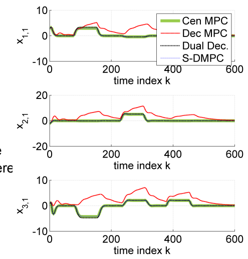
- no terminal cost
- long prediction and control horizon (K = 50)
- solved using Matlab standard QP solver `quadprog` with standard settings
- J = 30 iterations required for dual decomposition approach for convergence
- J = 1 and J = 2 iterations for S-DMPC → low communication and computing requirements



## Closed-loop Trajectories

**State trajectories:**

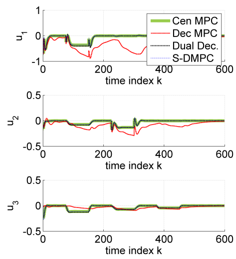
- Decentralized MPC
  - bad disturbance compensation
  - almost unstable control
- Dual Decomposition
  - achieves good performance
  - requires many iterations (here 30)
- S-DMPC
  - only one iteration
  - almost matches the centralized control






## Closed-loop trajectories

**Input trajectories**



## Controller Performance

- Absolute performance (quadratic performance index)

$$\Phi_{\text{abs}} = \sum_{k=0}^{H-1} \left( \|x(k)\|_Q^2 + \|u(k)\|_R^2 \right)$$

- Relative performance (Centralized controller is reference)

$$\Phi_{\text{rel}} = \frac{\Phi_{\text{abs}} - \Phi_{\text{abs,ref}}}{\Phi_{\text{abs,ref}}}$$

- Simulation results

| Method | It. | $\Phi_{\text{abs}}$ | $\Phi_{\text{rel}}$ [%] |
|---|---|---|---|
| Cen. MPC | – | 1.94e4 | – |
| Dec. MPC | – | 1.34e5 | 589 |
| Dual Dec. | 30 | 2.30e4 | 18.6 |
| S-DMPC | 1 | 1.95e4 | 0.5 |
| S-DMPC | 2 | 1.94e4 | 0 |

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 31

## Computing Time

- Comparison of average computing time for the methods considered

| Method | It. | $\bar{t}$ [s] |
|---|---|---|
| Cen. MPC | – | 0.112 |
| Dec. MPC | – | $3 \times 0.026$ |
| Dual Dec. | 30 | $3 \times 0.922$ |
| S-DMPC | 1 | $3 \times 0.030$ |
| S-DMPC | 2 | $3 \times 0.059$ |

- Computing time can be reduced, in particular with multiple CPU cores

- Dual decomposition is not competitive

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 32

## Conclusions & Future Work

**Conclusions**

- S-DMPC: a new method for distributed optimal control
  - inherits properties of centralized optimal control problem
  - S-DMPC provides optimal performance
- S-DMPC enables distributed computing
  - size of QP to be solved reduced
  - computing time can be reduced

**Future work**

- guaranteed stability (e.g. infinite horizon, terminal constraint, …)
- output feedback
- convergence (adaptation of QP via Wegstein extension)
- nonlinear systems
- Efficient implementation and integration into dynamic real-time optimization platform of AVT.PT

AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 33

## References (1)


- Hartwich, A. and Marquardt, W (2010). Dynamic optimization of the load change of a large-scale chemical plant by adaptive single shooting. *Computer & Chemical Engineering*, 34, 1873–1889.
- Lasdon, L.S. (1970). *Optimization Theory for Large Systems*. Macmillan Series for Operations Research.
- Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control - a review. *Journal of Process Control*, 19, 723–731.
- Scheu, H. and Marquardt, W. (2011a). Sensitivity-based coordination in distributed model predictive control. *Journal of Process Control*, 21, 800–815.
- Scheu, H. and Marquardt, W. (2011b). Distributed model-predictive control driven by simultaneous derivation of prices and resources. *Proc. of 18th IFAC World Congress.*
- Liu, J., Chen, X., Muñoz de la Peña, D. and Christofides P.D. (2010), Sequential and iterative architectures for distributed model predictive control of nonlinear process systems. *AIChE Journal* 56 (8) 2137–2149.
- Pannocchia, G., Rawlings, J., Mayne, D., Marquardt, W. (2010). On computing solutions to the continuous time constrained linear quadratic regulator. *IEEE Transactions on Automatic Control*, 55 (9) 2192–2198.


AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 34

## References (2)


- Wegstein, J.H. (1958). Accelerating convergence of iterative processes. *Communications of the ACM* 1 (6) 9–13.
- Westerberg, A.W., Hutchison, H.P., Motard, R.L., Winter, P(1979). *Process Flowsheeting*. Cambridge University Press.
- Rockafellar, R.T. (1976). Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization* 14, 877–898.
- Censor, Y. (1992). Proximal minimization algorithm with D-functions. *Journal of Optimization Theory and Applications* 73 (3) 451–464.


AACHENER VERFAHRENSTECHNIK IFAC World Congress 2011, Milan - HD-MPC Workshop 35

## 2.7 Hierarchical MPC with applications in transportation and infrastructure networks (B. De Schutter)

Bart De Schutter
Delft Center for Systems and Control

## Hierarchical MPC with applications in transportation and infrastructure networks

Milan, Italy, August 28, 2011

HD–MPC

SEVENTH FRAMEWORK PROGRAMME

---

### Outline







---

### HD-MPC for large-scale systems



**Challenges in control of large-scale networks**:

- Large-scale networks
- Distributed vs centralized control
- Optimality ↔ computational efficiency/tractability
- Global ↔ local
- Scalability
- Communication requirements (bandwidth)
- Robustness against failures

→ multi-level multi-agent approach



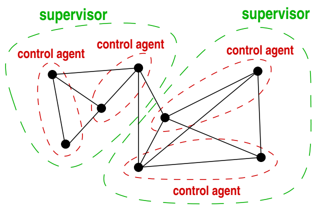
---

### HD-MPC for large-scale systems



**Multi-level multi-agent control**

- Multi-level control with intelligent control agents & coordination
- Time-based and space-based separation into layers





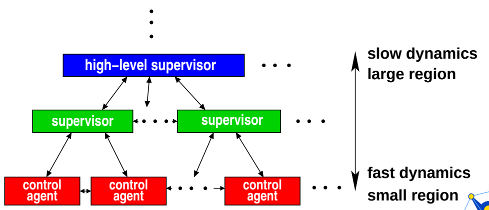

---

### HD-MPC for large-scale systems



**Multi-level multi-agent control**

- Multi-level control with intelligent control agents & coordination
- Time-based and space-based separation into layers






---

### HD-MPC for large-scale systems



**Multi-level control framework**

- Lowest level:
  - local control agents
  - "fast" control
  - small region
  - operational control
- Higher levels:
  - supervisors
  - "slower" control
  - larger regions
  - operational, tactical, strategic control
- Multi-level, multi-objective control structure
- *Coordination at and across all levels*
- Combine with model predictive control (MPC)

## HD-MPC for large-scale systems 6/52

**Major problem for MPC in practice**: Required computation time for large-scale systems

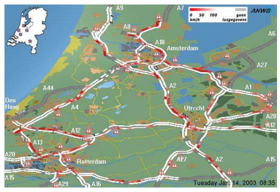
- Use distributed and/or hierarchical control approach
- Choice of the prediction model: accuracy versus computational complexity
- Right optimization approach
  - parallel and/or distributed optimization
  - approximate original MPC optimization problem by another optimization problem that can be solved efficiently
- Include application-specific knowledge



## Traffic management and automated highway systems 7/52

**Need for traffic control**

Traffic jams & congestion
→ cause time losses, extra costs, more incidents
have negative impact on economy, environment, society





## Traffic management and automated highway systems 8/52

Several ways to reduce traffic jams and to improve traffic performance:

- New infrastructure, missing links
- Pricing
- Modal shift
- Better use of available capacity through intelligent traffic control



## Traffic management and automated highway systems 9/52

**Intelligent traffic control**

Next generation traffic control and management system

- Use in-car telematics (navigation, telecommunication, information, ...) systems
- Vehicle-vehicle + vehicle-roadside communication
- Use intelligent vehicles (IVs)
  - control system senses environment using sensors
  - enhances either performance of driver or vehicle itself
    - assisting (advisory/warning)
    - taking partial or complete control (full automation)
- Two variants of traffic management using IVs:
  - cooperative vehicle-infrastructure systems (CVIS): drivers are still in charge of their vehicles
  - Automated Highway Systems (AHS): autonomous vehicles organized in platoons



## Traffic management and automated highway systems 10/52

**Automated highway systems (AHS)**

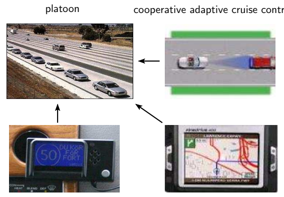
- Platoons of intelligent, autonomous vehicles
- Small inter-vehicle distance inside distances + high speeds
  → higher throughput
- Larger inter-platoon distance for safety
- Problems:
  - transition
  - psychological & legal aspects
  - → long-term, trucks



## Traffic management and automated highway systems 11/52

**Automated highway systems (AHS)**

- Integrate various in-vehicle and roadside-based traffic control measures that support platoons of fully autonomous IVs

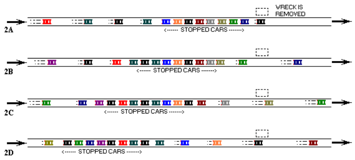



- Goal: improved traffic performance (safety, throughput, environment, ...) + constraints (robustness, reliability, ...)

## Traffic management and automated highway systems 12/52

**Additional advantage of platoons: No capacity drop**

- Capacity drop for human drivers: If traffic flow breaks down, then afterwards outflow from congested area is less than previous higher flow




- Reason: Human drivers tend to accelerate more slowly when they are coming out of congestion
- This effect plays less or even not with autonomous vehicles

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## MPC for traffic control 13/52

**Traffic flow models**

Two main classes:

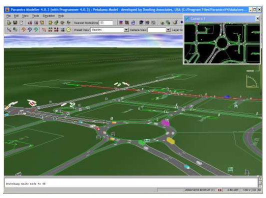
- Microscopic models → individual vehicles
- Macroscopic models → aggregated variables

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## MPC for traffic control 14/52

**Microscopic traffic flow models**

- Consider individual vehicles
- Car following + lane changing + overtaking models
- Different driver classes (with different parameters settings)
- Simulation rather time-consuming for large networks
  → less suited as prediction model for MPC
  → better suited as simulation/validation model



Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## MPC for traffic control 15/52

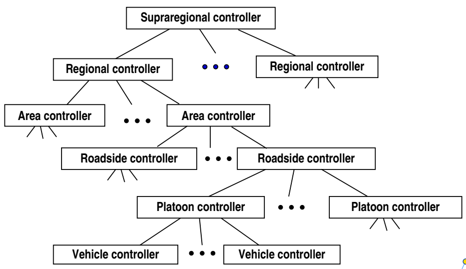
**Macroscopic traffic flow models**

- Work with aggregated variables (average speed, density, flow)
- Examples:
  - fluid-like models: Lighthill-Whitham-Richards (LWR), Payne, METANET, ...
  - gas-kinetic models: Helbing model, ...
- Trade-off between computational speed versus accuracy
  → well suited as prediction model for MPC
  → less suited as simulation/validation model
- In this presentation we use macroscopic models *for automated highway systems* as prediction model for MPC

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS 16/52

**A multi-level multi-scale HD-MPC approach for AHS** → hierarchical multi-layer control approach (∼ California PATH)




Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS 17/52

| Controller | Unit | Control | Time scale |
|---|---|---|---|
| Vehicle | vehicle | throttle, brake, steering | $\ll$ s |
| Platoon | vehicles | distances & speeds, trajectories | $<$ s |
| Roadside | platoons | lanes & speeds, split & merge | s–min |
| Area | flows of platoons | routing | $>$ min |
| Regional | flows | routing | $>$ 15–30 min |

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS (18/52)

**Control strategies**

- Vehicle controllers: (adaptive) PID + logic (for safety)
- Platoon controllers: rule-based control, hybrid control
- Roadside, area, regional controllers: MPC

$$\min_{u(k),\dots,u(k+N_c-1)} J(k)$$
s.t. model of system
operational constraints

$\longrightarrow$ medium-sized problems due to temporal & spatial division
$\longrightarrow$ still tractable

- Coordination (top-down) via performance criterion $J$ or constraints

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Roadside control (19/52)

**Roadside controllers**

- Control highway or stretch of highway
- Measurements: position, speed, lanes of platoon leaders
- Control inputs: platoon speeds, lane allocations, on-ramp release times
- Objectives:
  - track speed and splitting rate profiles imposed by area controllers
  - minimize total time spent (TTS) in network and queues, ...
- Constraints: min. headway, min. and max. speeds

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Roadside control (20/52)

**MPC for roadside controllers**

- Model: "big-car" model
platoon = vehicle with speed-dependent length

$$L_{\text{platoon},p}(k) = (n_p - 1)S_0 + \sum_{i=1}^{n_p-1} T_{\text{gap},i} v_{n_p}(k) + \sum_{i=1}^{n_p} L_i$$

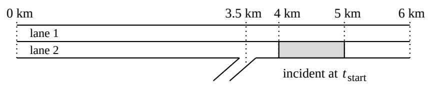
with $S_0$ minimum safe distance at zero speed and $T_{\text{gap},i}$ the desired time gap
- Nonlinear optimization problem:
min (TTS links + TTS queues)
subject to nonlinear model
operational constraints
- Optimization: mixed-integer nonlinear programming
Simplify by bi-level approach in which first lane allocation is determined (via heuristics, optimized, slower rate, ...)

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Roadside control (21/52)

**Case study – Problem statement**

Two-lane highway with an incident causing traffic




Scenario:

- Demand: 2500 veh/h (mainstream) and 350 veh/h (on-ramp)
- Incident at 4-5 km, start of simulation (10 minutes)
- Queues at start: empty
- Simulation period: 10 min, controller sampling time: 1 min
- Simulation sampling time: 1 s

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Roadside control (22/52)

**Case study – Cases**

Cases considered:

- Uncontrolled human drivers
- Controlled human drivers (current situation)
- Platoon approach – our approach

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Roadside control (23/52)

**Case study – Results**

| Case | TTS (veh·h) | Relative improvement (%) |
|---|---|---|
| Uncontrolled | 71.80 | 0 % |
| Controlled (human drivers) | 63.38 | 10.96 % |
| Controlled (platoons) | 57.75 | 18.86 % |

Reduced TTS $\rightarrow$ decreased travel times, increased trips, ...

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Area control


**Area controllers**

- Route guidance + provide set-points for roadside controllers
- Traffic network is represented by graph with nodes and links
- Due to computational complexity, optimal route choice control done via flows on links
- Optimal route guidance: nonlinear *integer* optimization with high computational requirements → intractable



## Multi-level multi-scale HD-MPC for AHS– Area control


**Area controllers (contd.)**

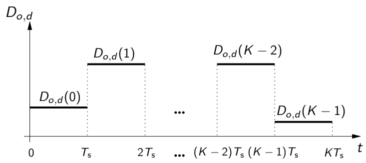
- Fast approaches based on
  - Mixed-Integer *Linear* Programming (MILP)
    - transform nonlinear problem into system of linear equations using binary variables
    - can be solved efficiently using branch-and-bound; several efficient commercial and freeware solvers available
  - macroscopic METANET-like traffic flow model
    - for humans, splitting rates are determined by traffic assignment
    - in AHS, splitting rates considered as controllable input
    - will result in non-convex *real-valued* optimization



## Multi-level multi-scale HD-MPC for AHS– Area control


**MILP approach – General set-up**

- Only consider flows and queue lengths
- Each link has maximal allowed capacity constraint
- Piecewise constant time-varying demand - $[kT_s, (k+1)T_s)$ for $k = 0, \dots, K-1$ with $K$ (simulation horizon)




- Main goal: assign optimal flows $x_{l,o,d}(k)$



## Multi-level multi-scale HD-MPC for AHS– Area control


**MILP approach – Model**

- Inflow at origin:

$$\sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k) \leqslant D_{o,d}(k) + \frac{q_{o,d}(k)}{T_s} \quad \text{for each } d \in \mathcal{D}$$

- Outflow from origin to destination:

$$F_{o,d}^{\text{out}}(k) = \sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k)$$

- Assume constant delay $\kappa$ between beginning and end of link
- Queue behavior at origin: Total demand − outflow
- More specifically, $D_{o,d}(k) - F_{o,d}^{\text{out}}(k)$ in time interval $[kT_s, (k+1)T_s)$

$$q_{o,d}(k+1) = \max\left(0,\, q_{o,d}(k) + (D_{o,d}(k) - F_{o,d}^{\text{out}}(k))T_s\right)$$



## Multi-level multi-scale HD-MPC for AHS– Area control


**MILP approach – Equivalences**

P1: $[f(x) \leqslant 0] \iff [\delta = 1]$ is true if and only if

$$\begin{cases} f(x) \leqslant M(1-\delta) \\ f(x) \geqslant \epsilon + (m - \epsilon)\delta \end{cases}$$

P2: $y = \delta f(x)$ is equivalent to $\begin{cases} y \leqslant M\delta \\ y \geqslant m\delta \\ y \leqslant f(x) - m(1-\delta) \\ y \geqslant f(x) - M(1-\delta) \end{cases}$

- $f$ function with upper and lower bounds $M$ and $m$
- $\delta$ is a binary variable
- $y$ is a real-valued scalar variable
- $\epsilon$ is a small tolerance (machine precision)

→ transform max equations into MILP equations



## Multi-level multi-scale HD-MPC for AHS– Area control


**MILP approach – Transforming the queue model**

$$q_{o,d}(k+1) = \max\left(0,\, q_{o,d}(k) + (D_{o,d}(k) - F_{o,d}^{\text{out}}(k))T_s\right)$$

Define

$$[\,\delta_{o,d}(k) = 1\,] \iff [\, q_{o,d}(k) + (D_{o,d}(k) - F_{o,d}^{\text{out}}(k))T_s \geqslant 0\,]$$

Can be transformed into MILP equations using equivalence P1

$$q_{o,d}(k+1) = \delta_{o,d}(k)\big(\underbrace{q_{o,d}(k) + (D_{o,d}(k) - F_{o,d}^{\text{out}}(k))T_s}_{f \text{ (linear)}}\big)$$

$$= z_{o,d}(k)$$

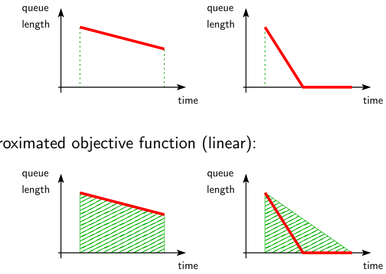
Product between $\delta_{o,d}(k)$ and $f$ can be transformed into system of MILP equations using equivalence P2

Queue model → system of MILP equations

**MILP approach – Objective function for queues**

Original objective function: time spent in queues (linear/quadratic):

Approximated objective function (linear):








**MILP approach – Objective Functions**

- Time spent in links:

$$J_{\text{links}} = \sum_{k=0}^{K_{\text{end}}-1} \sum_{(o,d)\in\mathcal{O}\times\mathcal{D}} \sum_{l\in L_{o,d}} x_{l,o,d}(k)\kappa_l T_s^2$$

- Time spent in queues:

$$J_{\text{queue}} = \sum_{k=0}^{K_{\text{end}}-1} \sum_{(o,d)\in\mathcal{O}\times\mathcal{D}} \frac{1}{2}(q_{o,d}(k) + q_{o,d}(k+1))T_s$$





**MILP approach – Overall area control problem**

Nonlinear optimization problem:

min (TTS links + TTS queues)
subject to
nonlinear model
operational constraints

MILP optimization problem:

min (TTS links + $\widehat{\text{TTS}}$ queues)
subject to
MILP model
operational constraints





**MILP approach – Case study**

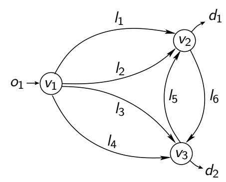



Figure: Set-up of case study network





**MILP approach – Case study – Set-up**

- Dynamic demand case with queues only at origins of network

| Period (min) | 0–10 | 10–30 | 30–40 | 40–60 |
|---|---|---|---|---|
| $D_{o_1,d_1}$ (veh/h) | 5000 | 8000 | 2500 | 0 |
| $D_{o_1,d_2}$ (veh/h) | 1000 | 2000 | 1000 | 0 |

- Scenario:
  - simulation period: 60 min, sampling time: 1 min
  - capacities: $C_1$=1900 veh/h, $C_2$=2000 veh/h, $C_3$=1800 veh/h, $C_4$=1600 veh/h, $C_5$=1000 veh/h, and $C_6$=1000 veh/h
  - delay factor: $\kappa_1$=10, $\kappa_2$=9, $\kappa_3$=6, $\kappa_4$=7, $\kappa_5$=2, and $\kappa_6$=2





**MILP approach – Case study – Cases**

Cases considered

- Case A: no control
- Case B: controlled using the MILP solution
- Case C: controlled using the exact solution

## Multi-level multi-scale HD-MPC for AHS– Area control 36/52

**MILP approach – Case study – Results**

| Case | $TTS_{tot}$ (veh.h) | improvement | CPU time (s) |
|---|---|---|---|
| No control | 1434 | 0 % | – |
| MILP | 1081 | 24.6 % | 0.27 |
| SQP (5 initial points) | 1067 | 25.6 % | 90.0 |
| SQP (50 initial points) | 1064 | 25.8 % | 983 |
| SQP (with MILP solution as initial point) | 1064 | 25.8 % | 1.29 |

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Area control 37/52

**MILP approach – Case study – Analysis**

- Uncontrolled case: only direct/short routes are used. Length of origin queue increases with time
- Controlled cases: flows assigned to both short and long routes

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 38/52

**Regional controllers**

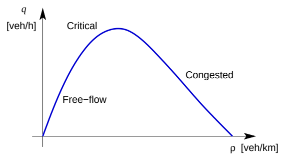
- Control collection of areas
- Determine optimal flows of platoons between areas
- Model: aggregate model – AHS variant of the Macroscopic Fundamental Diagram (MFD)
- Optimization: Nonlinear non-convex programming problem
  Will be approximated using mixed-integer linear programming

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 39/52

**Macroscopic Fundamental Diagram (MFD)**

- Introduced by Geroliminis and Daganzo
- Describes relation between space-mean flow and density in neighborhood-sized sections of cities (up to 10 $km^2$)
- Macroscopic fundamental diagram is independent of the demand
- Outflow of area is proportional to space-mean flow within area




Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 40/52

**Macroscopic Fundamental Diagram for AHS**

- Adopt modified version of MFD for AHS
- Shape of MFD will be sharper and maximal flow will be much higher than in MFD for human drivers
- Represent AHS network by graph
  - links correspond to areas, with inflow $q_{\mathrm{in},a}(k)$, outflow $q_{\mathrm{out},a}(k)$, and density $\rho_a(k)$
  - nodes correspond to connections between areas, external origins (with inflow $q_{\mathrm{orig},o}(k)$), or external exits (with outflow $q_{\mathrm{exit},e}(k)$)

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 41/52

**Model for regional controllers**

- Network MFD for AHS results in static description of form

$$q_{\mathrm{out},a}(k) = \mathcal{M}_a(\rho_a(k))$$

- Evolution of densities inside each area is described using simple conservation equation:

$$\rho_a(k+1) = \rho_a(k) + \frac{T}{L_a}(q_{\mathrm{in},a}(k) - q_{\mathrm{out},a}(k))$$

  with $T$ sample time step system and $L_a$ measure for total length of highways and roads in area $a$
- For every node $\nu$ balance between inflows and outflows:

$$\sum_{a\in\mathcal{I}_\nu} q_{\mathrm{out},a}(k) + \sum_{o\in\mathcal{I}_{\mathrm{orig},\nu}} q_{\mathrm{orig},o}(k) = \sum_{a\in\mathcal{O}_\nu} q_{\mathrm{in},a}(k) + \sum_{e\in\mathcal{O}_{\mathrm{exit},\nu}} q_{\mathrm{exit},e}(k)$$

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 42/52

**MPC for regional controllers**

- Try to keep density in each region below critical density $\rho_{\text{crit},a}$:

$$J_{\text{pen}}(k) = \sum_{j=1}^{N_p} \sum_{a} \left[ \max(0, \rho_a(k+j) - \rho_{\text{crit},a}) \right]^2$$

- Also minimize total time spent (TTS) by all vehicles in region:

$$J_{\text{TTS}}(k) = \sum_{j=1}^{N_p} \sum_{a} L_a \rho_a(k+j) T$$

- Total objective function:

$$J(k) = J_{\text{pen}}(k) + \gamma J_{\text{TTS}}(k)$$

- Constraints on maximal flows from one area to another,. . .
- Results in nonlinear, non-convex optimization problem

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 43/52

**Mixed integer linear programming (MILP) – Two properties**

- Given function $f$ with lower bound $m$ and upper bound $M$
- **Property 1**:
  $[f(x) \leq 0] \Leftrightarrow [\delta = 1]$ is equivalent to

$$\begin{cases} f(x) \leq M(1-\delta) \\ f(x) \geq \varepsilon + (m - \varepsilon)\delta \end{cases}$$

- **Property 2**:
  $y = \delta f(x)$ with $\delta \in \{0, 1\}$ is equivalent to

$$\begin{cases} y \leq M\delta \\ y \geq m\delta \\ y \leq f(x) - m(1-\delta) \\ y \geq f(x) - M(1-\delta) \end{cases}$$

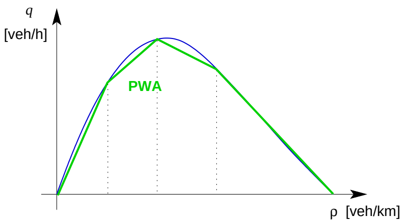
Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 44/52

**Transformation into MILP problem**

- Approximate MFD by Piece-Wise Affine (PWA) function

$$q_{\text{out},a}(k) = \alpha_{a,i}\rho_a(k) + \beta_{a,i} \quad \text{if } \rho_a(k) \in [\rho_{a,i}, \rho_{a,i+1}]$$




Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 44/52

**Transformation into MILP problem**

- Approximate MFD by Piece-Wise Affine (PWA) function

$$q_{\text{out},a}(k) = \alpha_{a,i}\rho_a(k) + \beta_{a,i} \quad \text{if } \rho_a(k) \in [\rho_{a,i}, \rho_{a,i+1}]$$

- Introduce binary variables $\delta_{a,i}(k)$ such that

$$\delta_{a,i}(k) = 1 \quad \text{if and only if} \quad \rho_{a,i} \leq \rho_a(k) \leq \rho_{a,i+1}$$

  Can be transformed into MILP equations using Property 1
- Now we have

$$q_{\text{out},a}(k) = \sum_{i=1}^{N_a} (\alpha_{a,i}\rho_a(k) + \beta_{a,i})\delta_{a,i}(k)$$

- Introduce real-valued auxiliary variables $y_{a,i}(k) = \rho_a(k)\delta_{a,i}(k)$
  Can be transformed into MILP equations using Property 2

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 45/52

**Transformation into MILP problem**

- Results in

$$q_{\text{out},a}(k) = \sum_{i=1}^{N_a} \alpha_{a,i} y_{a,i}(k) + \beta_{a,i}\delta_{a,i}(k)$$

- If we combine all equations and inequalities, we obtain a system of mixed-integer linear inequalities

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Multi-level multi-scale HD-MPC for AHS– Regional control 46/52

**Transformation into MILP problem**

- Recall

$$J_{\text{pen}}(k) = \sum_{j} \sum_{a} \left[ \max(0, \rho_a(k+j) - \rho_{\text{crit},a}) \right]^2 \quad \rightarrow \text{not linear}$$

$$J_{\text{TTS}}(k) = \sum_{j} \sum_{a} L_a \rho_a(k+j) T \quad \rightarrow \text{linear!}$$

- Removing square in $J_{\text{pen}}(k)$ results in PWA objective function
  Can be transformed in MILP equations using Properties 1 & 2
- Hence, we get MILP problem
- Solution of MILP problem can be directly applied or it can be used as good initial starting point for original nonlinear, non-convex MPC optimization problem

Hierarchical MPC for transportation networks — Bart De Schutter HD-MPC

## Related work 

**Related work: Traffic management using MPC**

- More viable option on short term: roadside intelligence
  → traffic control center + current infrastructure
- Use conventional control measures: variable speed limits, ramp metering, traffic signals, lane closures, shoulder lane openings, tidal flow, . . .
- Also include "soft" control measures: dynamic route information, travel time information, . . .



## Related work: Traffic management using MPC 

**Ongoing research**

- Address complexity issues for large-scale systems
  - simplified models for urban traffic networks
  - parametrized MPC
- Alternative objective functions + related models
  - emissions: CO, $NO_x$, $CO_2$, HC, . . .
  - fuel consumption



## Related work: Traffic management using MPC 

**Cooperative Vehicle Infrastructure Systems**

- Intermediate step between current system and AHS



## Related work: Traffic management using MPC 

**Other applications**

- Electricity networks
- Water networks
- Railway networks
- Logistic systems



## Conclusions and future work 

**Conclusions**

- Hierarchical control framework for automated highway systems (AHS)
- Focus on roadside, area, and regional controllers
- In general: nonlinear, non-convex mixed-integer optimization problems
- Reduce complexity of problem by selecting appropriate models and making approximations
- Results by bi-level, mixed-integer linear programming, or nonlinear, non-convex real-valued optimization problems

**Future work**

- extensive integrated case study & assessment
- further development of HD-MPC approaches
- further improvements in efficiency and performance



## Conclusions and future work 

**Main issues and topics in HD-MPC for transportation and infrastructure networks**

- How to obtain tractable prediction models?
- What is the best division into subnetworks?
- Selection of static/dynamic region boundaries?
- How to determine subgoals so as to optimize overall goal?
- How can existing approaches be extended to hybrid systems?
- How can the computation/iteration time be reduced further? (algorithms, properties, approximations, reductions, . . . )

## 2.8 Application to start-up of combined-cycle power plant (A. Tica, H. Guéguen, D. Dumur, D. Faille, F. Davelaar)

IFAC World Congress - Milano 2011
Workshop on Hierarchical and Distributed MPC: Algorithms and Applications

## Start-Up of Combined Cycle Power Plants

A. Tica, **H Guéguen**, D. Dumur - SUPELEC
D. Faille, F. Davelaar - EDF

HD–MPC SEVENTH FRAMEWORK PROGRAMME

SUPELEC

### Introduction 2/23

**General context**

- Diffusion of Combined Cycle Power Plants (CCPP)
  - Efficiency
  - Lower pollutant emissions
- Production to consumption fit
  - Partial load (ancillary services)
  - Frequent start-up and shut-down
- Flexibility Improvement
  - Reduction of start-up and shut-down time
  - Avoidance of start-up failure
  - Minimization of life-time consumption

CCPP are complex plants with numerous systems and sub-systems

Start-Up of Combined Cycle Power Plants — H. Guéguen HD-MPC

### Introduction 3/23

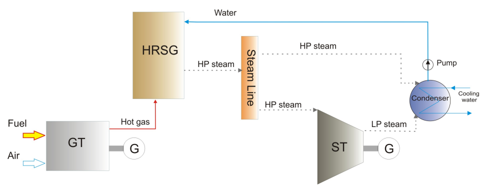
**Objectives**

- How can MPC control help to reduce start-up time while saving life-time consumption?
- How can Distribution and Hierarchy help to design and implement control?
- How can design models (Modelica) of the plant be used for operationnal phases?

Start-Up of Combined Cycle Power Plants — H. Guéguen HD-MPC

### Combined Cycle Power Plants 4/23

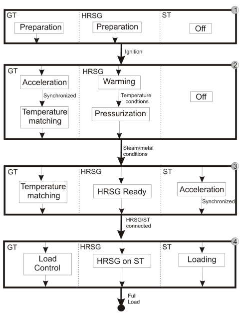
**Schematic view**




Start-Up of Combined Cycle Power Plants — H. Guéguen HD-MPC

### Combined Cycle Power Plants 5/23

**Start-up procedure**




Start-Up of Combined Cycle Power Plants — H. Guéguen HD-MPC

### Combined Cycle Power Plants 6/23

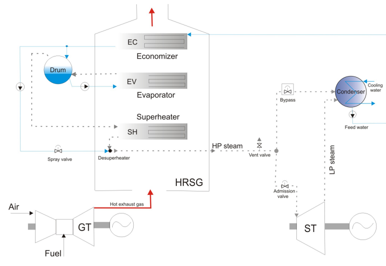
**Modelica model**
**Politecnico di Milano (F Casella)**

- 1 1 1 CCPP with 3 levels of pressure
- ThermoPower Library
- can be used from low load to high load
- Simplified model:
  - gas turbine
  - low pressure components
- Stress model of critical components
  - high and intermediate pressure superheated steam headers
  - high and intermediate pressure steam turbine rotor

Start-Up of Combined Cycle Power Plants — H. Guéguen HD-MPC

## Case Study


### 1-1-1 CCPP with 1 level of pressure






## Case Study


### Increasing Load Phase

Initial state
- Turbine generators are connected to the grid
- Gas turbine load is around 15%
- Pressure is around 60 e5 Pa
- Steam is admitted in steam turbine (Bypass valve is closed)

Aim: full load
- GT load near 100%
- admission valve: fully open



## Case Study


### Local control
- Gas turbine control
- Feed water flow by drum level control loop
- *Steam turbine admission valve by control wrt gas turbine load*
  - when GT load $\leq$ 50%: open-loop control of pressure
  - when GT load $>$ 50%: fully open

### Control
- Minimize start-up time
- Constraints on stress level
- Control variables: GTload, *admission valve*



## Case Study


### Modelica Model
- ThermoPower library
- Stress model (ASME)
  - Header stress: combination of mechanical and thermal stress
  - Rotor stress: thermal stress
- Complexity
  - $\approx$2400 equations
  - 42 state variables
  - simulation time for increasing load phase: 12.4s (PC with 2 GHz CPU)



## Overview


1 pressure level CCPP

ThermoPower Modelica Model



## Overview


1 pressure level CCPP

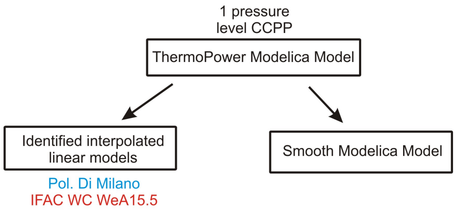
ThermoPower Modelica Model

↓

Identified interpolated linear models

Pol. Di Milano
IFAC WC WeA15.5

## Overview




## Smooth Model


- Aim:
  - elimination of discontinuities
    - *if* clauses
    - piecewise affine functions
    - steam/water tables
  - reduction of simulation time
- Constraint: keep the structure of the ThermoPower model
  - corresponding components
  - new media functions for steam and water



## Smooth Model


- simplification/specialization of some components (e.g. reverse flow elimination)
- discontinuities approximations



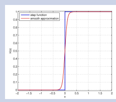
## Smooth Model


- simplification/specialization of some components (e.g. reverse flow elimination)
- discontinuities approximations

**Discontinuity approximation**

$$\forall x \in R \; H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases}$$

$$\forall x \in R \; H_k(x) = \frac{1}{1 + e^{-kx}}$$

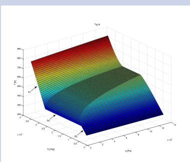


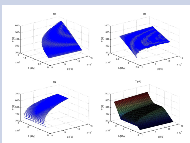


## Smooth Model


**Steam/Water functions approximation**

piecewise polynomial approximations of the Modelica.Media functions
e.g. $T = f(P, h)$





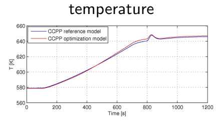


## Smooth Model

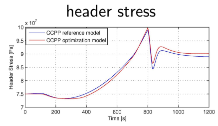

Complexity

- ≈2000 equations
- 42 state variables
- simulation time for increasing load phase: 1.4s (PC with 2 GHz CPU)

temperature




header stress



## Overview


1 pressure
level CCPP

ThermoPower Modelica Model

Identified interpolated
linear models

Pol. Di Milano
IFAC WC WeA15.5

Smooth Modelica Model



## Overview


1 pressure
level CCPP

ThermoPower Modelica Model

Identified interpolated
linear models

Pol. Di Milano
IFAC WC WeA15.5

Smooth Modelica Model

Modelica
optimization tools



## Overview


1 pressure
level CCPP

ThermoPower Modelica Model

Identified interpolated
linear models

Pol. Di Milano
IFAC WC WeA15.5

Smooth Modelica Model

Black Box
optimization

Modelica
optimization tools



## BB optimization


**Profile optimization.**

- Choice of parametrized profiles: $L(t) = L_p(t,q)$, e.g.

$$L_{2H}(t,q) = L_m + (L_i - L_m)\frac{t^h}{t^h + k^h} + (L_M - L_i)\frac{t^p}{t^p + r^p}$$

- Optimization problem

$$min_{t_f,q}(J), \ J = \int_{t_f}^{t_0} dt$$

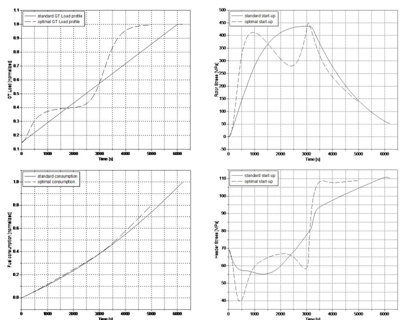
subject to the constraints

$$\dot{x} = f(x, L_p(t,q))$$
$$L_p(t_f,q) \geq L_M - \epsilon_1$$
$$\|f(x(t_f), L_p(t_f,q))\| \leq \epsilon_2$$
$$h(x(t)) \leq 0$$



## BB optimization


Example: $L_{2H}(t,q)$ gains wrt ramp: time 20%; consumption 20%





## BB optimization


- Gas turbine load
  - 2 hills functions; start-up time: 4790s (-20%)
  - spline functions (3); start-up time: 4530s (-25%)
- Gas turbine load and steam turbine admission valve
  - spline functions (2); start-up time: 4440s (-26%)

## MPC



- Control variable: gas turbine load
- Every computation time ($T_C$)
  - profile computation for the next $N.T_C$
  - Lagrange polynomials (N)
  - minimization of $J = \int_{t_0}^{t_0+N.T_c} \|L_{L_N}(t,q) - L_0(L(t_0))\|^2 \, dt$



## MPC



- Control variable: gas turbine load
- Every computation time ($T_C$)
  - profile computation for the next $N.T_C$
  - Lagrange polynomials (N)
  - minimization of $J = \int_{t_0}^{t_0+N.T_c} \|L_{L_N}(t,q) - L_0(L(t_0))\|^2 \, dt$

**Start-up time: 3400s (-43%) [$T_C = 60s$, $N = 5$]**



## Hierarchy and Distribution



**Hierarchical MPC control**

- Robustness of control
  - Introduction of variations into the model?
  - Simulation on sets?



## Hierarchy and Distribution



**Hierarchical MPC control**

- Robustness of control
  - Introduction of variations into the model?
  - Simulation on sets?

**Distributed control**

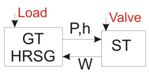
- Gradient based methods?
- Robustness?
- Range of admissible input signals



## Hierarchy and Distribution



Example

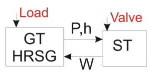


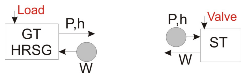



## Hierarchy and Distribution



Example



## Conclusion

23/23

- Smooth Modelica model for a 1-1-1 1 pressure level CCPP
  - new components / media consistent with ThermoPower
  - systematic design of the optimisation model
- Start-up profile optimization
  - reduction of start-up time
  - importance of profile functions

Start-Up of Combined Cycle Power Plants — H. Guéguen HD-MPC

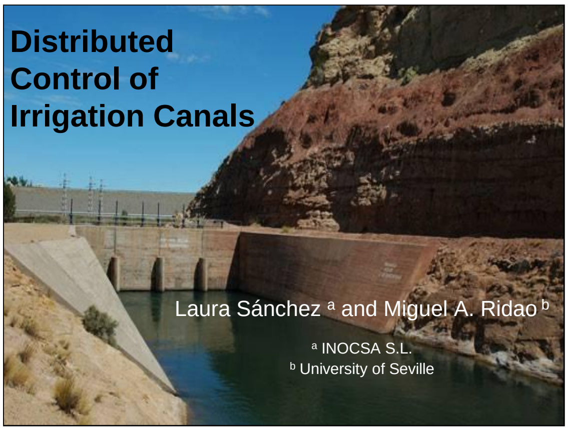
## Conclusion

23/23

- Smooth Modelica model for a 1-1-1 1 pressure level CCPP
  - new components / media consistent with ThermoPower
  - systematic design of the optimisation model
- Start-up profile optimization
  - reduction of start-up time
  - importance of profile functions
- Such approach for such plants is still challenging
  - optimization tools / model development
  - simulation tools: admissible state and feasible trajectories
  - distributed approaches: steam interactions

Start-Up of Combined Cycle Power Plants — H. Guéguen HD-MPC

## 2.9 Distributed control of irrigation canals (L. Sánchez, M.A. Ridao)



# Distributed Control of Irrigation Canals

Laura Sánchez [a] and Miguel A. Ridao [b]

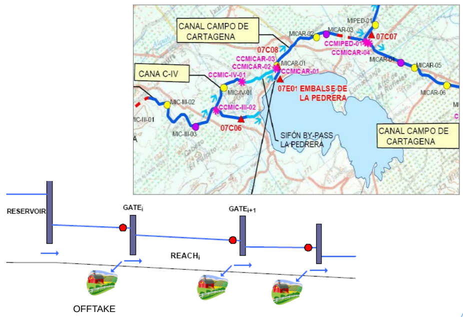
[a] INOCSA S.L.
[b] University of Seville

## Outline

- Irrigation Canal System
  - Main Elements
  - Operation of an Irrigation Canal
- Models
- Control of Irrigation Canals

HD-MPC

## Irrigation Canal Scheme




HD-MPC

## Control structures - Gates


Taintor Gate


Sluice Gates


Two Taintor Gates with side weirs


Side weirs

HD-MPC

## Canal elements


Gravity offtake


Wasteweirs


Syphon


Canal Head

HD-MPC

## Canal Operation Concepts

- Supply oriented operation
  - Upstream water supply source or inflow determines the canal system flow schedule
  - Used when the inflow is fixed by a different organization than the canal manager
- Demand oriented operation
  - Downstream water demand (offtakes) determines the canal system flow schedule
  - The inflow is determined by the canal manager accordingly with the demand

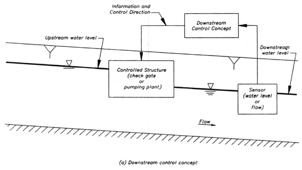
HD-MPC

## Control objectives

- **Main objective**: guarantee flows requested by users. It is necessary to maintain the level of the canal over the off-take gate.
- **Controlled Variables**:
  - levels upstream or downstream the gates.
  - flows through gates, mainly at the head of the canal and secondary canals.
  - Water volume
- **Manipulated variables:**
  - Gate opening
  - flow is considered as a manipulated variable to control levels when a two level controller is used.
- **Disturbances:**
  - Off-takes flows: measured, aggregate values or predicted
  - Rainfall: Measured or predicted
- **Contraints:**
  - Maximum and minimum levels along the canal
  - Maximum and minimum flows
  - Operating levels on reservoir at the tail of the canal

HD-MPC

## Control Concepts – Downstream Control

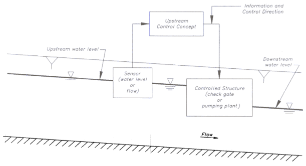



(a) Downstream control concept

- Control structure adjustments (gates) are based upon information from downstream (usually levels)
- Downstream control transfers the downstream offtake demand to the upstream water supply source (flow at the head)
- Compatible with demand oriented operation
- Impossible with supply oriented operation

HD-MPC

## Control Concepts – Upstream Control




- Control structure adjustments (gates) are based upon information from upstream (usually levels)
- Upstream control transfers the upstream water supply (or inflow) downstream to points of diversion or to the end of the canal
- Compatible with supply oriented operation
- Inefficient with demand oriented operation

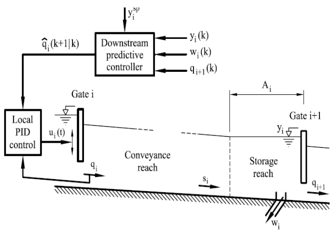
HD-MPC

## Irrigation Canal Control – General Ideas

- Controlled variables: Water level, water volume or discharge (most common, level)
- Two global strategies:
  - Directly manipulate gate opening in order to control levels
  - Two level control
    - Compute required gate discharges in order to control water levels (discharges as manipulated variable)
    - Manipulate gate openings to obtain the requested gate discharges
      - Local Controller (Cascade control)
      - Inverting the gate discharge equation

HD-MPC

## Irrigation Canal Control – General Ideas




Example of a two level downstream controller. The first level is a predictive controller and the lower level controller is a PID

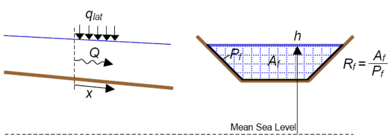
HD-MPC

## Outline

- Irrigation Canal System
- Models
  - Saint-Venant equations
  - Models of control structures
  - Control models
- Control of Irrigation Canals

HD-MPC

## Irrigation Canal Model - Reaches



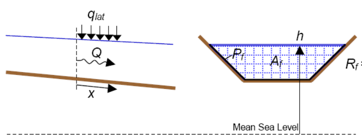

$$\frac{\partial Q}{\partial x} + \frac{\partial A_f}{\partial t} = q_{lat}$$ Mass Balance

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2}{A_f}\right) + g \cdot A_f \frac{\partial h}{\partial x} + \frac{g \cdot Q|Q|}{C^2 \cdot R_f \cdot A_f} = 0$$ Momentum Balance

**Partial Differential Saint-Venant Equations**

HD-MPC

## Irrigation Canal Model - Reaches



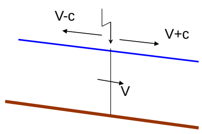

Inertia — Convective acceleration — Gravitational Force — Friction Force

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2}{A_f}\right) + g \cdot A_f \frac{\partial h}{\partial x} + \frac{g \cdot Q|Q|}{C^2 \cdot R_f \cdot A_f} = 0$$ Momentum Balance

**Partial Differential Saint-Venant Equations**

HD-MPC

## Saint_Venant Equations – Water Movement




A disturbance, created in a reach, results in two wave movements., one wave travels with velocity V+ c and one travels with velocity V - c.

$$c = \sqrt{\frac{gA_f}{B}}$$

B Top width of wetted cross section

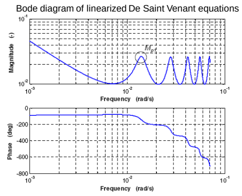
$A_f$ Wetted cross section surface

- Flow Regimes
  - If c>V, **subcritical flow**, a change in flow results in two waves in opposite directions
  - If c=V, **critical flow**, a change in flow results in only one wave travelling downstream
  - If c<V, **supercritical flow**, a change in flow results in two waves travelling downstream
- Subcritical flow is presented in most real irrigation canal

HD-MPC

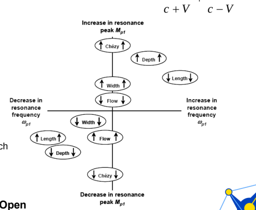
## Saint_Venant Equations – Water Movement

Bode diagram of linearized De Saint Venant equations




When a wave arrives at a boundary (a control structure), part of the wave is reflected. If a wave is initiated from one of the boundaries, it returns after a period

$$T_r = \frac{L}{c+V} + \frac{L}{c-V} \qquad \omega = \frac{2\pi}{\frac{L}{c+V} + \frac{L}{c-V}}$$




Influence of changes in parameter values of reach dimension on basic frequency

**P.J van Overloop, "Model Predictive Control on Open Water Systems". 2006**

HD-MPC

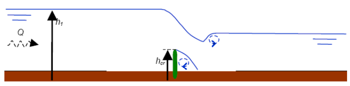
## Structure Models – Overshot Gates

Many theoretical or empirical formulas have been proposed, for example:

$$Q = C_d L \sqrt{\frac{2}{3} g}\,(h_1 - h_{cr})^{3/2}$$

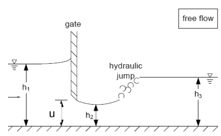
$L$ : With of gate

$C_d$ : Discharge coeficient



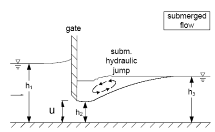

HD-MPC

## Structure Models – Undershot gates




$$Q = C_d \cdot L \cdot u \sqrt{2gh_1}$$

$u$ : Gate opening




$$Q = C_d \cdot L \cdot u \sqrt{2g(h_1 - h_3)}$$

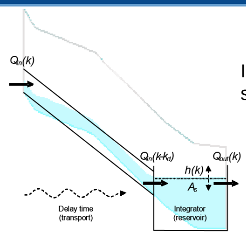
HD-MPC

## Simplified models for control

- Some approaches in bibliography
  - Based on mathematical models
    - Integrator-delay model (Shuurmans, TU Delft)
    - Linearization of Saint-Venant equations (Litrico and Fromion, Cemegraph)
  - Identification models
    - Weyer et al. (University of Melbourne)
    - Rivas Perez (*Havana Polytechnic University)*
    - *Rodellar, Sepulveda (Universidad Politécnica de Cataluña)*

HD-MPC

## Simplified models for control – ID Model




Integrator-delay simplified model

$$A_s(h(k+1)-h(k)) = T_d(Q_{in}(k-k_d)+q_{in}(k)-Q_{out}(k)-q_{out}(k))$$

$T_d$ *Sampling time*

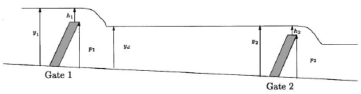
$q_{in}(k)$ *Lateral input : rain fall ,...*

$q_{out}(k)$ *Offtakes*

Schuurmans, J. (1997), 'Control of water levels in open channels', Ph.D.-dissertation TUDelft

HD-MPC

## Identification Models I




First and third order non-linear and linear models for a reach with overshot gates

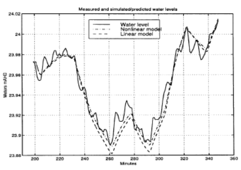
Non-linear model

$$y_2(t+1) = y_2(t) + c_1 h_1^{3/2}(t-\tau) + c_2(y_2(t) - p_2(t))^{3/2}$$

Linear model

$$y_2(t+1) = y_2(t) + c_1 h_1(t-\tau) + c_2(y_2(t) - p_2(t))$$

Parameters: $c_1$ $c_2$ $\tau$




E. Weyer. System identification of an open water channel. Control Engineering Practice 9, 2001

HD-MPC

## Identification Models II

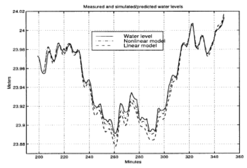
Third order models for the wave dynamics

$$\dddot{y}(t) + a_1\ddot{y}(t) + a_2(t)\dot{y}(t) = Inflow - Outflow$$

Non-linear model

$$y_2(t+1) = c_1 h_1^{3/2}(t-\tau) + c_3 h_1^{3/2}(t-\tau-1) + c_5 h_1^{3/2}(t-\tau-2)$$
$$+ c_2(y_2(t) - p_2(t))^{3/2} + c_4(y_2(t-1) - p_2(t-1))^{3/2} +$$
$$c_6(y_2(t-2) - p_2(t-2))^{3/2} + y_2(t) +$$
$$(1-a_1)(y_2(t) - 2y_2(t-1) + y_2(t-2)) +$$
$$(1-a_2)(y_2(t) - 2y_2(t-1))$$

Parameters: $c_1$ $c_2$ $c_3$ $c_4$ $c_5$ $c_6$ $a_1$ $a_2$ $\tau$




**Conclusions:**

The models can be used for accurate simulation of the water levels at least 7.5 h ahead of time

(2) The models are valid under both high and low flow conditions

E. Weyer. System identification of an open water channel. Control Engineering Practice 9, 2001

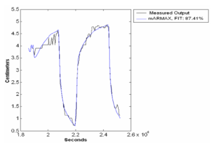
HD-MPC

## Identification Models III

Second order Model

Rivas Perez et al. *System identification for control of a main irrigation canal pool*. Proceedings of the 17th World Congress

$$T_1 T_2 \frac{d^2 y_1(t)}{dt^2} + (T_1 + T_2)\frac{dy_1(t)}{dt} + y(t) = Ku_1(t-\tau)$$




ARX higher models and Laguerre Models

Sepulveda, Instrumentation, model identification and control of an experimental irrigation canal. PhD Dissertation. Universidad Politécnica de Cataluña.

HD-MPC

## Outline

- Irrigation Canal System
- Models
- Control of Irrigation Canals
  - Decentralized control
  - Distributed control

HD-MPC

## Irrigation Canal Control – Common solutions

- Most of the implemented techniques are based on local PI
  - EL-FLO: A PI controller with a filter applied to downstream control.
  - P+PR: A PI applied to upstream control.
  - BIVAL: The controlled variable used both upstream and downstream measures (volume control)

$$y = \alpha y_{up} + (1-\alpha) y_{dwn}$$

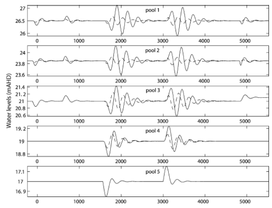
  - AVIS: P controller for radial gates (upstream control)
  - AMIL:P controller for radial gates (downstream control)
  - PIR: PI+ Smith predictor

Malaterre et al., "Classification of Canal Control Algorithms", ASCE Journal of Irrigation and Drainage Engineering. Jan./Feb. 1998, Vol. 124, .

HD-MPC

## Decentralized control

- The most used solution in practice consist of a PI compensator and a filter
  - The compensator need at least one pole in s=0 to achieve zero steady-state waterlevel error for step load disturbances
  - Several PI tuning rules based on ID model: Schuurmans, Litrico…
  - The low pass filter diminish the controller sensibility to wave resonance
  - A typical problem is the level error amplification upstream (Cantoni, et al. 2007)

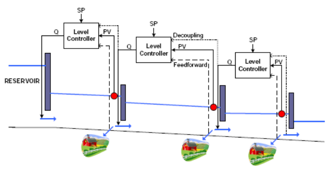



HD-MPC

## Decentralized Control: Decoupling and Feedforward

- Decoupling: Feedforward control considering the flow at the next gate ($u_{i+1}$) as a disturbance
  - This flow is always measured (or computed) – no additional cost
  - Diminish the interrelationship among coupled variables – reduction of the amplification error problem
- Feedforward – offtake discharges
  - Not always available a reliable measure.

$$u_i(s) = C_i(s)e_i(s) + F_d(s)u_{i+1}(s) + F_{ff}(s)d_i(s)$$




HD-MPC

## MPC approaches

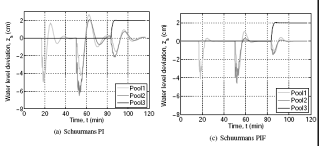
- Decentralized
  - "Predictive Control Applied to ASCE Canal 2". K. Akouz et al. IEEE International Conference on Systems, Man, and Cybernetics. (1998).
  - "Decentralized Predictive Controller for Delivery Canals".S. Sawadogo et al. IEEE International Conference on Systems, Man, and Cybernetics, volume 4.(1998).
  - *"A Simulink-Based Scheme for Simulation of Irrigation Canal Control Systems"*. J. A. Mantecón et al.. SIMULATION (2002)
  - "Predictive control method for decentralized operation of irrigation canals". M. Gómez et al. Applied Mathematical Modelling 26 (2002)
- Centralized
  - "Multivariable predictive control of irrigation canals. Design and evaluation on a 2-pool model". P.O. Malaterre. International Workshop on the Regulation of Irrigation Canals: State of the Art of Research and Applications (1997).
  - "Instrumentation, model identification and control of an experimental irrigation canal". C.A. Sepulveda. PhD. Thesis. (1997)
  - Model Predictive Control on Open Water Systems. P.J. Overloop. PhD. Thesis. (2006)
  - "Predictive Control with constraints of a multi-pool irrigation canal prototype". O. Begovich. Latin American Applied Research, 37 (2007)
  - "Adaptive and non-adaptive model predictive control of an irrigation canal" J.M. Lemos et al. Networks and heterogeneous media. Volume 4, Number 2, (2009).

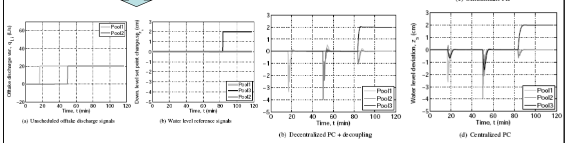
HD-MPC

## Some comparative results

A three reaches canal:

- T=1000 s: Offtake increment in Pool1.
- T= 3000 s: Offtake increment in Pool2.
- T= 5000 s: Water level set point increment in Pool3.



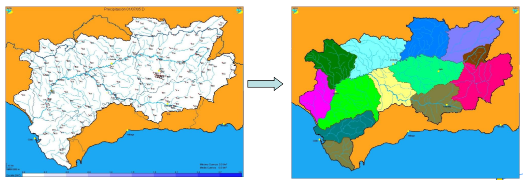




Sepulveda, Instrumentation, model identification and control of an experimental irrigation canal. PhD Dissertation. Universidad Politécnica de Cataluña.

HD-MPC

## Why distributed Control?

- Coordination between sub-systems is needed, i.e. the avoidance of upstream disturbance amplification in canals consisting of canal reaches in series
- The number of reaches and gates can be high (near one hundred in the Postrasvase Tajo-Segura): computational limitations for a Centralized MPC
- Different section of the canal can be managed by different Control Centers and even by different organizations.

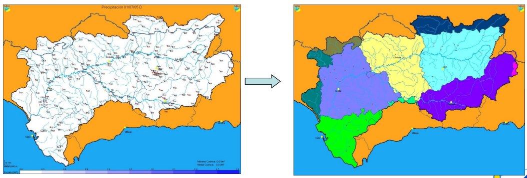


DECISIONS TAKEN IN ONE HYDROGRAPHICAL AREA CAN INFLUENCE OTHER CLOSE AREAS

HD-MPC

## Why distributed Control?

- Coordination between sub-systems is needed, i.e. the avoidance of upstream disturbance amplification in canals consisting of canal reaches in series
- The number of reaches and gates can be high (near one hundred in the Postrasvase Tajo-Segura): computational limitations for a Centralized MPC
- Different section of the canal can be managed by different Control Centers and even by different organizations.



DISTRIBUTED CONTROL CAN TACKLE WITH POLITICAL DECISIONS TAKEN IN DIFFERENT PROVINCES

HD-MPC

## Distributed approaches to Irrigation Canal

- **Decentralized predictive controller for delivery canals**
  S. Sawadogo, R. M. Faye, P. O. Malaterre and F. Mora-Camino.
  Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics (San Diego, California), 1998
- **Optimal control of complex irrigation systems via decomposition -coordination and the use of augmented Lagrangian**
  H. El Fawal, D. Georges and G. Bornard
  Proceedings of the 1998 International Conference on Systems, Man, and Cybernetics (San Diego, California), 1998.
- **Decentralized adaptive control for a water distribution system.**
  G. Georges.
  Proceedings ofthe 3rd IEEE Conference on Control Applications (Glasgow, UK), 1999.
- **Cooperative Control of Water Volumes of Parallel Ponds Attached to An Open Channel Based on Information Consensus with Minimum Diversion Water Loss.**
  Christophe Tricaud and YangQuan Chen
  Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation, Harbin, China, 2007,
- **Distributed controller design for open water channels**
  Y. Li and M. Cantoni,
  *Proceedings of the 17th IFAC World Congress*, Korea, 2008.
- **Distributed Model Predictive Control of Irrigation Canals**
  R.R Negenborn, P.J. Overloop, T. Keviczky and B. De Shutter
  NETWORKS AND HETEROGENEOUS MEDIA Vol. 4-2, 2009.
- **Performance Analysis of Irrigation Channels with Distributed Control.**
  Yuping Li and Bart De Schutter.
  2010 IEEE International Conference on Control Applications. Yokohama, Japan, 2010
- **A hierarchical distributed model predictive control approach to irrigation canals: A risk mitigation perspective.**
  A. Zafra-Cabeza, J.M.Maestre, Miguel A.Ridao, E.F.Camacho and L. Sánchez
  Journal of Process Control - Special Issue on HD-MPC.2011

HD-MPC

## A serial distributed MPC

- Control strategy: Downstream control
  - Controlled variable: Downstream level
  - Manipulated variables: Flows at the gates (set-point provided to the local flow controllers)
- Subsystems: A gate and the downstream reach
- Each controller requires the current state of its subsystem and predictions of the values of interconnecting variables.
- The controllers perform several iterations consisting of local problem solving and communication with neighbors.
- Serial communication scheme: One agent after another performs computations
- Iterative method based on Lagrange Multipliers.

"DISTRIBUTED MODEL PREDICTIVE CONTROL OF IRRIGATION CANALS"
R.R Negenborn, P.J. Overloop, T. Keviczky and B. De Shutter
NETWORKS AND HETEROGENEOUS MEDIA Vol. 4-2, (2009)

HD-MPC

## A serial distributed MPC: Models

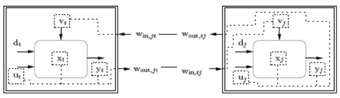
ID Model: $$h_i(k+1) = h_i(k) + \frac{T_c}{c_i} q_{in,i}(k - k_{d,i}) - \frac{T_c}{c_i} q_{out,i}(k) + \frac{T_c}{c_i} q_{ext,in,i}(k) - \frac{T_c}{c_i} q_{ext,out,i}(k)$$

State-Space Model:
$$x_i(k+1) = A_i x_i(k) + B_{1,i} u_i(k) + B_{2,i} d_i(k) + B_{3,i} v_i(k)$$
$$y_i(k) = C_i x_i(k)$$

where
$$x_i(k) = \begin{bmatrix} h_i(k) \\ q_{in,i}(k - k_{d,i}) \\ \ldots \\ q_{in,i}(k-1) \end{bmatrix} \quad d_i(k) = \begin{bmatrix} q_{ext,in,i}(k) \\ q_{ext,out,i}(k) \end{bmatrix}$$
$$u_i(k) = q_{in,i}(k) \quad v_i(k) = q_{out,i}(k) \quad y_i(k) = h_i(k)$$

HD-MPC

## A serial distributed MPC: Interconnecting variables




$$w_{in,i}(k) = v_i(k)$$
$$w_{out,i}(k) = K_i \left[ x_i^T(k) \quad u_i^T(k) \quad y_i^T(k) \right]$$

$K_i$ *is a* interconnecting output selection matrix

*and an interconnecting constraint:*
$$w_{in,ji}(k) = w_{out,ij}(k)$$
$$w_{out,ji}(k) = w_{in,ij}(k)$$

$$w_{in,j_{i,down}i}(k) = q_{out,i}(k)$$
$$w_{out,j_{i,up}i}(k) = q_{in,i}(k)$$

$j_{i,down}$ : index of the downstream canal reach of reach i

$j_{i,up}$ : index of the upstream canal reach of reach i

HD-MPC

## A serial distributed MPC: Control algorithm

- The controllers solve their control problems in the following serial iterative way:
  - Set the iteration counter and initialize the Lagrange multipliers arbitrarily.
  - One controller after another solves its optimization problem:
    $$\min \ J_{local,i} + \sum_j J^{(s)}_{inter,i}(w_{in,ji}(k), w_{out,ji}, \lambda))$$
  - Update the Lagrange Multipliers with the new values of the interconnecting variables
  - Send and receive the multipliers from the neighbor agent
  - Move on to the next iteration until a stopping condition is satisfied
- The controllers implement the actions until the beginning of the next control cycle

HD-MPC

## A serial distributed MPC: Control objective

- The deviations of water levels from provided set-points are minimized
- The changes in the set-points provided to the local flow controllers are minimized to reduce equipment wear

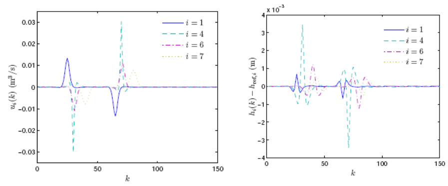
$$J_{local,i} = \sum_{l=0}^{N-1} p_{h,i}(h_i(k+1+l) - h_{ref,i})^2 + \sum_{l=0}^{N-1} p_{u,i}(u_i(k+l) - u_i(k+1+l))^2$$

$$J_{inter,i} = \begin{bmatrix} \tilde{\lambda}_{in,ji}(k) \\ -\tilde{\lambda}_{out,ij}(k) \end{bmatrix}^T \begin{bmatrix} \tilde{w}_{in,ji}(k) \\ \tilde{w}_{out,ji}(k) \end{bmatrix} + \frac{\gamma_c}{2} \left\| \begin{matrix} \tilde{w}_{in,prev,ij}(k) - \tilde{w}_{out,ji}(k) \\ \tilde{w}_{out,prev,ij}(k) - \tilde{w}_{in,ji}(k) \end{matrix} \right\|_2^2$$

HD-MPC

## A serial distributed MPC: Simulation Results

- 7 reaches canal
- The length of the canal is almost 10 km
- Maximum capacity of the head gate is 2.8m3/s
- Control cycle length: 240 s.
- Prediction horizon length; 31 (to take into account the total delay in the irrigation canal
- Scenario: a sudden increase of 0.1m3/s at control cycle k = 30 in the water offtake of canal reach 3 and a sudden decrease of 0.1m3/s at control cycle k = 70 in the same canal reach.



HD-MPC

## A HD-MPC approach based on risk management

- This approach shows how **risk management** can be applied to **optimize the Irrigation Canal operation** in order to consider **process uncertainties**.
- The proposed method, for the use of risk metrics, **forecasts the water level of reaches**, benefits and costs associated to IC.
- Formulation of a Hierarchical and Distributed MPC (HDMPC) to optimize the strategic plan (mitigation actions) that optimizes the operation of the IC.
  - Higher Level: MPC with a risk-based strategy
  - Lower Level: DMPC to optimize the operation (based on the DMPC based on game theory presented previously)

"A hierarchical distributed model predictive control approach to irrigation canals: A risk mitigation perspective"

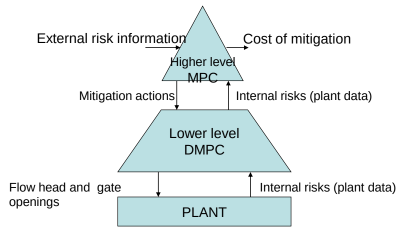
A. Zafra-Cabeza, J.M.Maestre, Miguel A.Ridao, E.F.Camacho and L. Sánchez

Journal of Process Control - Vol 21-5 - Special Issue on HD-MPC (June-2011)

HD-MPC

## HD-MPC and Risk Management

- **General structure**




HD-MPC

## Lower level: DMPC approach

- Downstream control, considering underflow gates and gate position as manipulated variable
- Each subsystem corresponds with a reach
- The Integrator delay model has been used for the reach movement and the flow through the gates as manipulated variables
- Each agent has only partial information of the system. Agents optimize according to a local cost function
- Low communicational requirements
- Cooperative solution: Cooperative algorithm from a game theory point of view. The different agents must reach an agreement on the value of the shared inputs

HD-MPC

## Lower level: DMPC approach

ID Model: $h_i(k+1) = h_i(k) + \frac{T_c}{c_i} q_{in,i}(k - k_{d,i}) - \frac{T_c}{c_i} q_{out,i}(k) + \frac{T_c}{c_i} q_{ext,in,i}(k) - \frac{T_c}{c_i} q_{ext,out,i}(k)$

State state model: $x_i(k+1) = A_i x_i(k) + \sum_{j \in n_i} B_{ij} u_j(k) + d_i(k)$

where: $u_1(k) = q_{in,i}(k)$, $u_2(k) = q_{out,i}(k)$

There is no coupling between the states of the agents (only coupled by the actuations)

Each agent has local information about the state and knows how it is affected by the different inputs

Inputs are not assigned to agents

HD-MPC

## Lower level: Cost functions

- **Agents optimize according to a local cost function**

$$J_i(x_i, \{U_j\}_{j\in n_i}) = \sum_{k=0}^{N-1} L_i(x_i(k), \{u_j(k)\}_{j\in n_i})$$

$$L_i(x_i, \{u_j\}_{j\in n_i}) = (x_i - \hat{h}_i(t))^T Q_i (x_i - \hat{h}_i(t)) + \sum_{j\in n_i} u_j^T S_{ij} u_j$$

- **Control objective: Global Performance Index**

$$\sum_{i=1}^{M_x} J_i(x_i(t), \{U_j(t)\}_{j\in n_i})$$

- **The different agents must reach an agreement on the value of the shared inputs**

HD-MPC

## Lower level: Algorithm

1. Each agent p measures its current state $x_p(t)$
2. Agents try to submit their proposals randomly. To this end, each agent asks the neighbors affected if they are free to evaluate a proposal.
3. In order to generate its proposal, each agent p minimizes $J_p$ solving the following optimization problem:

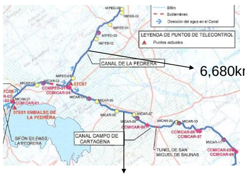
$$\begin{aligned}
\{U_j^p(t)\}_{j\in n_p} = \; & \arg \min_{\{U_j\}_{j\in n_p}} J_p(x_p, \{U_j\}_{j\in n_p}) \\
& s.t. \\
& x_{p,k+1} = A_p x_{p,k} + \textstyle\sum_{j\in n_p} B_{pj} u_{j,k} \\
& x_{p,0} = x_i(t) \\
& x_{p,k} \in \mathcal{X}_p, \; k = 0, \dots N \\
& u_{j,k} \in \mathcal{U}_j, \; k = 0, \dots N-1, \; \forall j \in n_p \\
& x_{p,N} \in \Omega_p \\
& U_j = U_j^s(t), \; \forall j \notin P_p
\end{aligned}$$

4. Each agent i affected by the proposal of agent p evaluates the predicted cost corresponding to the proposed solution. To do that, the agent calculates the difference between the cost of the new proposal and the cost of the current accepted proposal. The difference is sent back to agent p.
5. Once agent p receives the local cost increments from each neighbor, it can evaluate the impact of their proposal.
6. The algorithm returns to Step 1 until the maximum number of proposals has been made or the sampling time ends.
7. The first input of each optimal sequence is applied and the procedure is repeated the next sampling time from Step 1.

HD-MPC

## Lower level: Case study

Benchmark: postrasvase Tajo-Segura in the south-east of Spain



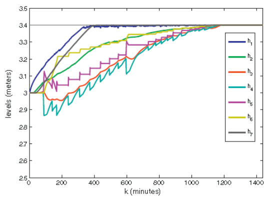

7 main gates
17 off-take gates
7 subsystem in DMPC

- Lower Level
  - Control water management in canals by satisfying demands
  - Controlled variables: downstream levels
  - Manipulated variables: flow at the head and the position of the gates
  - Sampling time: 1 minute
  - Nc=5
  - The prediction horizon for each reach is the control horizon plus the delay of the reach :Np(i)=Nc+Ki
  - 7 agents

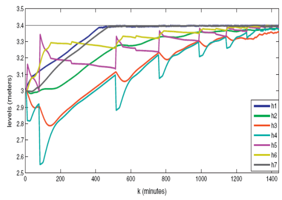
HD-MPC

## Case study: Lower level results

- Scenario: All reaches begin with a water level of 3.0 m and there is a change of set points for all the reaches to 3.40m (from higher level, day 150)




Nominal case. No disturbances




Measure disturbances

HD-MPC

## 2.10 Closing (B. De Schutter)

STREP Project 223854 HD-MPC

Bart De Schutter

## Closing of the HD-MPC workshop

Milan, Italy, August 28, 2011

HD–MPC SEVENTH FRAMEWORK PROGRAMME

## Extra information

2/7

- Website of HD-MPC project: `http://www.ict-hd-mpc.eu`
- Slides of this workshop: `http://www.ict-hd-mpc.eu/index.php?page=ifac_workshop` or via HD-MPC website → Events

Closing — Bart De Schutter HD-MPC

## Additional HD-MPC activities at IFAC World Congress

3/7

Two special sessions on HD-MPC organized by Bart De Schutter and Alfredo Núñez:

- MoA12 (10.00–12.00): Hierarchical and Distributed Model Predictive Control – I. Fundamentals
- MoB12 (13.30–15.30): Hierarchical and Distributed Model Predictive Control – II. Applications

- Room: Vito
- Presenters from HD-MPC and other FP7 EU projects (WIDE, HYCON2, Embocon, MoVeS)

Closing — Bart De Schutter HD-MPC

## Presentations in Session I – Fundamentals

4/7

- Feasible-cooperation distributed model predictive control scheme based on game theory
  by Valencia, Espinosa, De Schutter, Staňková
- A dual decomposition-based optimization method with guaranteed primal feasibility for hierarchical MPC problems
  by Doan, Keviczky, De Schutter
- Distributed model-predictive control driven by simultaneous derivation of prices and resources
  by Scheu and Marquardt
- Distributed non-cooperative MPC with neighbor-to-neighbor communication
  by Farina and Scattolini
- Adjoint-based distributed multiple shooting for large-scale systems
  by Savorgnan, Kozma, Andersson, Diehl
- Distributed model predictive control and estimation of large-scale multi-rate systems
  by Roshany-Yamchi, Negenborn, Cychowski, Connell, Delaney

Closing — Bart De Schutter HD-MPC

## Presentations in Session II – Applications

5/7

- Distributed MPC for multi-zone temperature regulation with coupled constraints
  by Moroşan, Bourdais, Dumur, Buisson
- Coordination of a multiple link HVDC system using local communications based distributed model predictive control
  by Mc Namara, Negenborn, De Schutter, Lightbody
- Hierarchical control with prioritized MPC for conflict resolution in air traffic control
  by Chaloulos, Hokayem, Lygeros
- Fixed-profile load scheduling for large-scale irrigation channels
  by Li and De Schutter
- Decentralised MPC based on a graph partitioning approach applied to the Barcelona drinking water network
  by Ocampo-Martinez, Puig, Bovo
- Cooperative distributed MPC for tracking
  by Ferramosca, Limón, Rawlings, Camacho

Closing — Bart De Schutter HD-MPC

## Special issue of the Journal of Process Control on HD-MPC

6/7

- Special Issue on Hierarchical and Distributed Model Predictive Control
- Journal of Process Control, Volume 21, Issue 5 (2011), pages 683–816
- Currently 2 papers in top-10 of most downloaded papers:
  - A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark
    by Alvarado et al.
  - Decentralized model predictive control of dynamically coupled linear systems
    by Alessio et al.


Closing — Bart De Schutter HD-MPC

## Call for papers: ADHS'12

7/7

- ADHS'12: 4th IFAC Conference on Analysis and Design of Hybrid Systems
- Eindhoven, The Netherlands, June 6–8, 2012
- Topics:
  - modeling, simulation, analysis, verification, and control of hybrid systems
  - applications in networked control systems, large-scale process industries, transportation systems, energy distribution networks, communication networks, etc.
- URL: `www.adhs12.org`
- Submission deadline: Nov. 15, 2011

Closing — Bart De Schutter HD-MPC