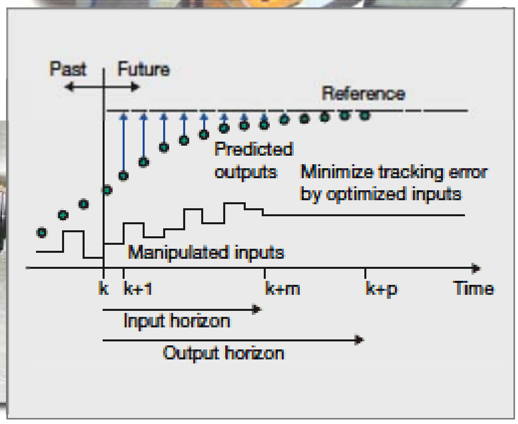
University of British Columbia
June 27, 2011

Nonlinear Model Predictive Control:
The Past, Present and Future

Frank Allgöwer

Institute for Systems Theory and Automatic Control

University of Stuttgart, Germany



Analytical vs. Numerics-based Control Methods

Analytical controllers

- Control law is synthesized off-line
- Control action is analytical function of present and past measurements

Numerics-based controllers

- Controller not implemented as an analytical function of past and present measurements
- On-line numerical computations take, at least, partly role of off-line controller design

A numerics-based control technique:

Model Predictive Control (MPC)

Formulation of Control Problem

$$\begin{aligned} \dot{x} &= f(x,u), \quad x(0) = x_0 \\ y &= h(x) \end{aligned}$$

Find **stabilizing** control strategy that

- minimizes **objective functional**

$$J = \int_t^{\infty} F(x(\tau), u(\tau)) d\tau$$

- satisfies **constraints**

$$\begin{aligned} u(\tau) &\in \mathcal{U} \\ x(\tau) &\in \mathcal{X} \end{aligned}$$

Different Solution Strategies
Open-loop vs. Closed-Loop Optimal Control

Closed-loop optimal control:

Feedback $u = k(x)$

s.t. closed-loop trajectories satisfy optimality conditions

- Feedback present!
- suited for uncertainty, disturbances, unstable systems, ...

- Finding closed solution hardly possible

Open-loop optimal control:

Input trajectory $u = u(t; x_0)$

solving optimization problem

- Computation often feasible

- No feedback!
- Unstable systems?
- Uncertain systems?
- No reaction to disturbances

A Possible Solution - Model Predictive Control

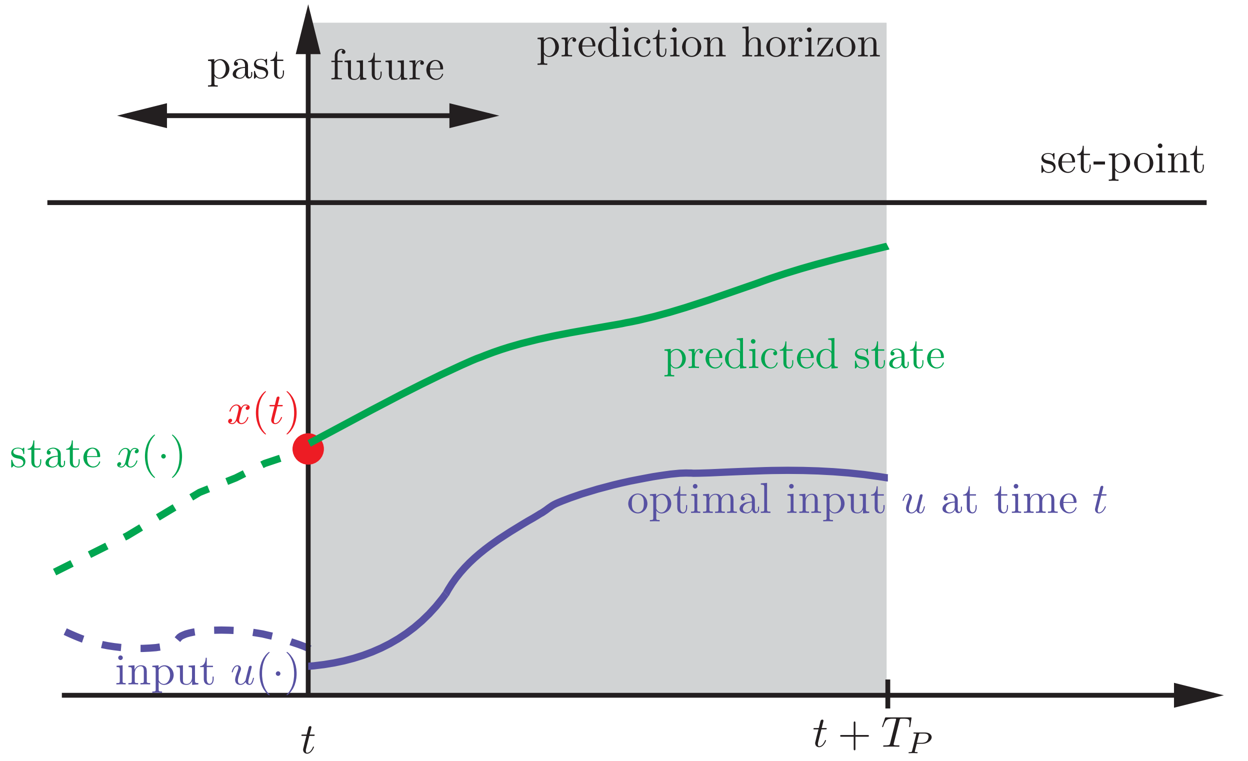
MPC= repeated open-loop optimal control
in feedback fashion

- Solve open-loop optimization problem all δ sampling instances

$$\min_{u(\cdot)} J(u(\cdot); x(t)) = \int_t^{T_P} F(x(\tau), u(\tau)) d\tau$$

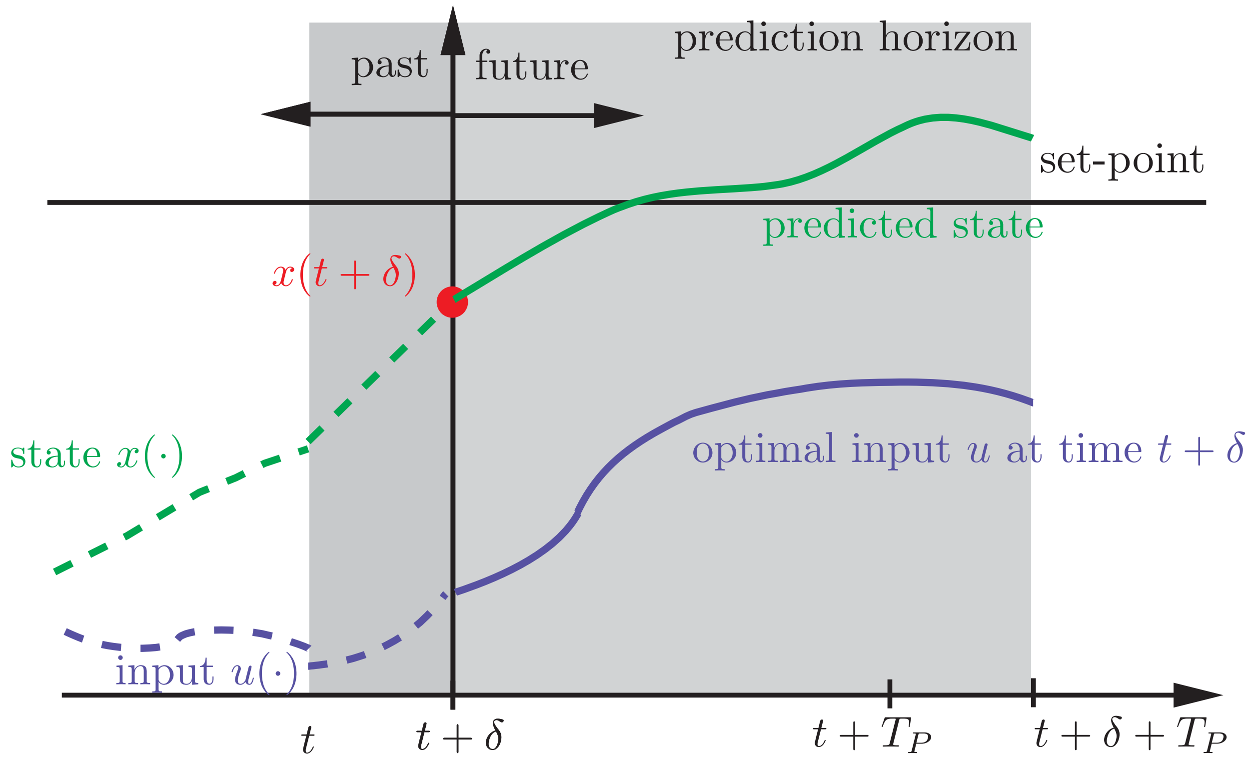
- Apply optimal open-loop input for one sampling interval $\tau \in [t, t+\delta]$ only
- Usually use finite prediction horizon T_p because of computational feasibility

MPC



past
future
prediction horizon
set-point
predicted state
$x(t)$
state $x(\cdot)$
optimal input u at time t
input $u(\cdot)$
t
$t + T_P$

MPC



past
future
prediction horizon
set-point
predicted state
$x(t+\delta)$
state $x(\cdot)$
optimal input u at time $t + \delta$
input $u(\cdot)$
t
$t + \delta$
$t + T_P$
$t + \delta + T_P$

A Brief History of MPC

Idea is rather old:

> "One technique for obtaining a **feedback controller synthesis from knowledge of open-loop controllers** is to measure the current control process state and then compute vary rapidly for this the open-loop control function. The first portion of this function is then used during a short time interval, after which a new value of the function is computed for this measurement, The process is then repeated."

(Lee & Marcus, 1967)

A Brief History of MPC cont.

Early industrial MPC applications:

- Model Predictive Heuristic Control (IDCOM)
  Richalet et al. 1976 . . .
  Adersa
- Dynamic Matrix Control (DMC)
  Cutler & Ramaker 1979 . . .
  Shell Oil

Academic research:

- Few early theoretical investigations:
  Kleinmann 1970, Thomas 1975, Chen & Shaw 1982, . . .
- Predictive control theory:
  Keerthi & Gilbert 1988, Mayne & Michalska 1990, . . .

Linear MPC - Nonlinear MPC

Linear MPC

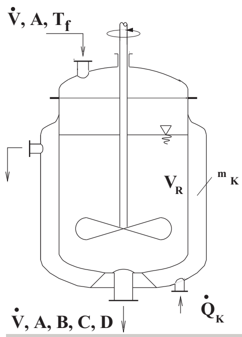
- Uses **linear model**: $\dot{x} = Ax + Bu$
- Quadratic cost function $F = x^T Rx + u^T Ru$
- Linear constraints $Hx + Gu < 0$
- $\Rightarrow$ Quadratic program

Nonlinear MPC (NMPC)

- **Nonlinear model:** $\dot{x} = f(x, u)$
- Cost function can be nonquadratic $F(x, u)$
- Nonlinear constraints $h(x, u) < 0$
- Nonlinear program

Application of NMPC to a Chemical Reactor

Cyclopentenol Production — Klatt/Engell/Kremling/Allgöwer 93



Van der Vusse reaction scheme:

$$\mathrm{A} \xrightarrow{k_1} \mathrm{B} \xrightarrow{k_2} \mathrm{C}$$
$$2\mathrm{A} \xrightarrow{k_3} \mathrm{D}$$

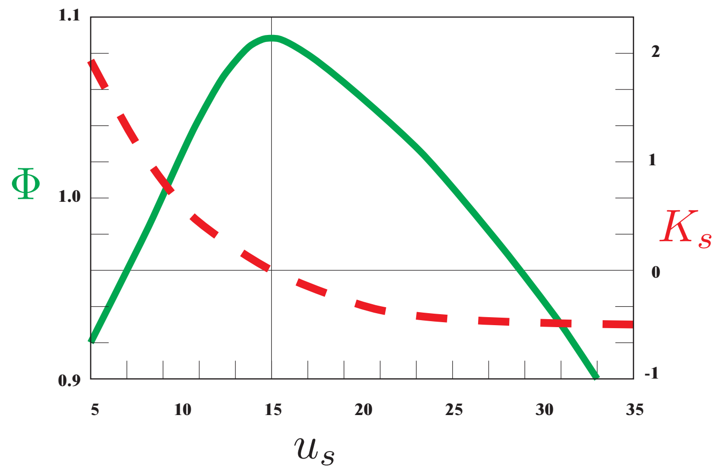
controlled variables:	concentration c_B
manipulated variables:	flow rate $\dot{V}$
disturbances:	feed temperature T_f, c_{A0}

$$\dot{c}_A = \dot{V}(c_{A0} - c_A) - k_1(\vartheta)c_A - k_3(\vartheta){c_A}^2,$$
$$\dot{c}_B = -\dot{V}c_B + k_1(\vartheta)\,c_A - k_2(\vartheta)c_B\,,$$
$$\dot{\vartheta} = \dot{V}(\vartheta_0 - \vartheta) - \frac{1}{\rho C_p}\Big(k_1(\vartheta)c_A\Delta H_{R_{AB}} + k_2(\vartheta)c_B\Delta H_{R_{BC}} + k_3(\vartheta){c_A}^2\Delta H_{R_{AD}}\Big) + \frac{k_w A_R}{\rho C_p V_R}(\vartheta_K - \vartheta)$$
$$\dot{\vartheta}_K = \frac{1}{m_K C_{PK}}\Big(\dot{Q}_K + k_w A_R(\vartheta - \vartheta_K)\Big)$$
$$k_i(\vartheta) = k_{i0}\cdot\exp\left(\frac{E_i}{\vartheta}\right),\ i = 1, 2, 3$$

Operation at Point of Optimal Yield

Yield: $\Phi_s = \left.\frac{c_{B_s}}{c_{A_0}}\right|_s$

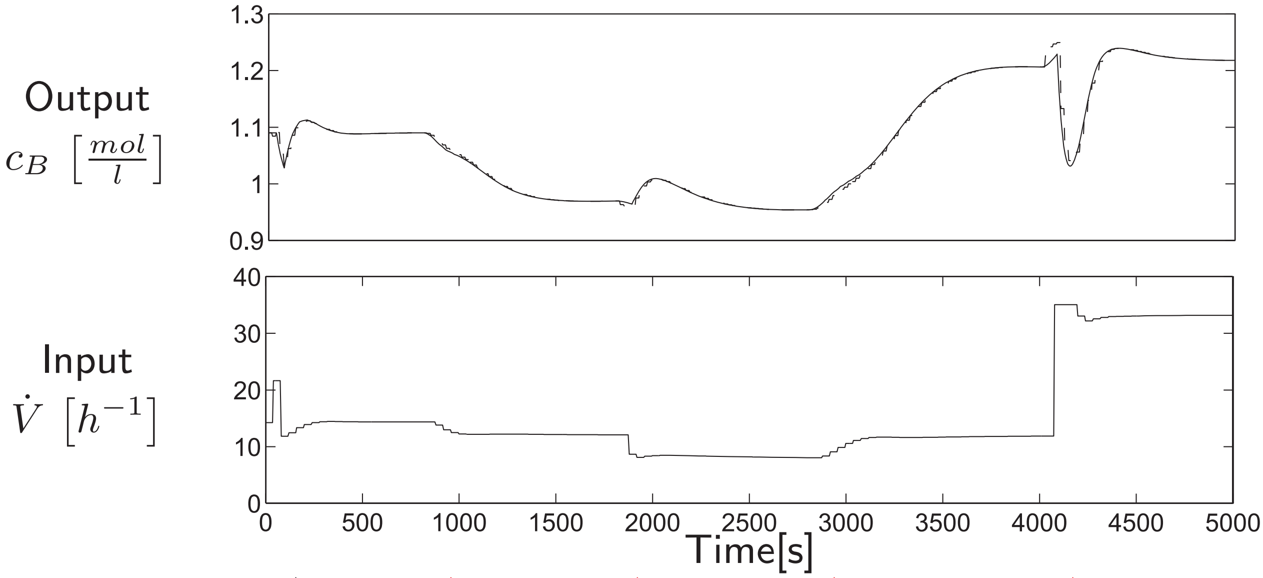
at optimum: $\left.\frac{\partial \Phi}{\partial u}\right|_{opt} = 0 \rightarrow \left.\frac{\partial y}{\partial u}\right|_{opt} = 0$
$\rightarrow K_{opt} = 0$



- gain changes sign at optimal operating point
  $\Rightarrow$ strong nonlinearity
  $\Rightarrow$ not integral controllable

Control for Optimal Yield

task: on-line optimization of yield $J = \int_t^{t+T_p} c_B^2(\tau)d\tau$



c_{A0} $\left[\frac{\text{mol}}{\text{l}}\right]$ = 5.1 ↑ 4.5 ↑ 5.7

T_f [^{o} C] = 104.9 ↑ 100 ↑ 115

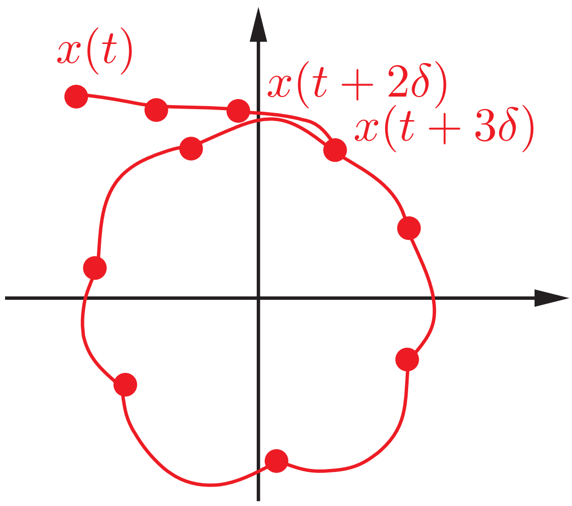
Issues in Nonlinear MPC

- **System theoretic formulation and investigation:**
  stability, performance, robustness, ...
- **Implementation issues:**
  efficient and reliable real-time optimization, modeling, ...

An Important Issue in (N)MPC Theory

Even in nominal case:
- no model plant mismatch
- no disturbances

Predicted open-loop trajectories
$\neq$
Closed-loop trajectories

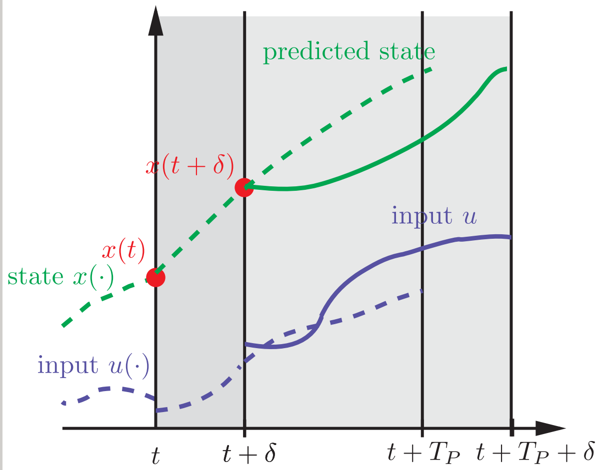


- **Performance?** Goal: $\min \int_0^\infty F(x(\tau), u(\tau))d\tau$. What is achieved by repeatedly minimizing $\int_t^{t+T_p} F(\mathbf{x}(\tau), u(\tau))d\tau$?
- **Stability?** Why should the closed-loop be stable?

Schemes that guarantee stability?

NMPC Theory in the 90s: Instantaneous State Feedback NMPC

Assumptions:
- On-line computation of optimal input trajectory $u^\star$ requires **no computation time** (instantaneous solution)
- Continuous application of optimal input $u = u^\star(x)$, i.e. $\delta \to 0$ (instantaneous implementation)



Unrealistic assumptions, however
(i) allows to develop system theoretic understanding
(ii) lays basis for development of methodological tools

key focus on stability issue

Instantaneous State Feedback NMPC: Guaranteed Stability and Performance

Idea:

Alter problem setup (cost function, constraints) such that repeated open-loop solution leads to **desired closed-loop properties** for **all nonlinear systems and all performance cost functions**

nominal stability: *[Keerthi&Gilbert '89], [Mayne&Michalska '90],...*

nominal stability and performance: *[Chen&Allgöwer '96], ...*

robust stability: *[Chen/Scherer/Allgöwer '97], [Magni/Nijmeijer/van der Schaft '01],...*

Instantaneous State Feedback NMPC: Guaranteed Stability and Performance

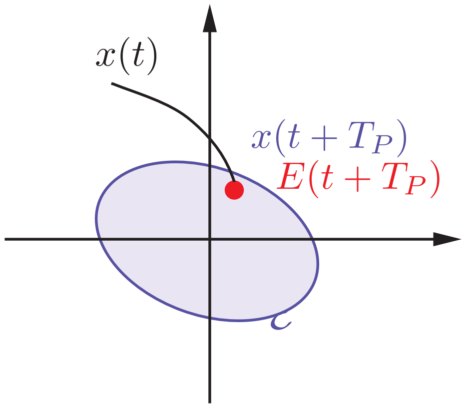
Add, suitably computed

- terminal region constraint $x(t+T_P) \in \mathcal{E}$
- terminal penalty term $E(x(t+T_p))$

$$\min_u J(x(t),u)$$

$$J(\cdot) = \int_t^{t+T_p} F(x(\tau),u(\tau))d\tau + E(x(t+T_p))$$

subject to:	$\dot{x} = f(x,u)$,	system dynamics
	$x(t)$ given	"state feedback"
	$u(\tau) \in \mathcal{U}$	input constraints
	$x(\tau) \in \mathcal{X}$	state constraints
	$x(t+T_p) \in \mathcal{E}$	terminal constraint



Additional terms computed such that $E(x(t+T_p))$ approximates **infinite horizon cost** in terminal region

Quasi-infinite horizon NMPC *[Chen&Allgöwer '96]*

Guaranteed Stability Result

[Chen&Allgöwer '96], [Mayne et al. '00], [Fontes '00]

$$\min_u J(x(t),u)$$

with: $J(\cdot) = \int_t^{t+T_p} F(x(\tau),u(\tau))d\tau + E(x(t+T_p))$

and: $x(t+T_p) \in \mathcal{E}$

Theorem (Nominal Stability): If

a) $E(\cdot)$ and $\mathcal{E}$ are determined s.t.:

$\forall x \in \mathcal{E}\ \exists u \in \mathcal{U}$ with $\frac{\partial E}{\partial x} f(x,u) + F(x,u) < 0$

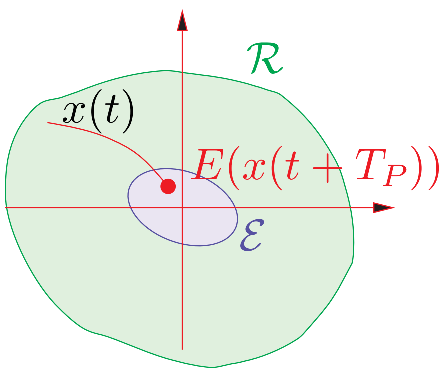
b) optimization feasible for $t = 0$

$\Downarrow$

Asymptotic Stability

Guaranteed Region of Attraction:

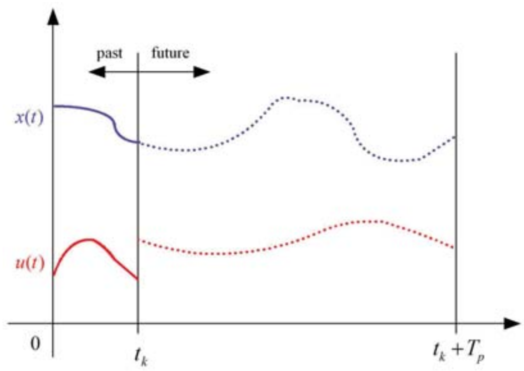
Set $\mathcal{R}$ of states satisfying b)



Robust MPC

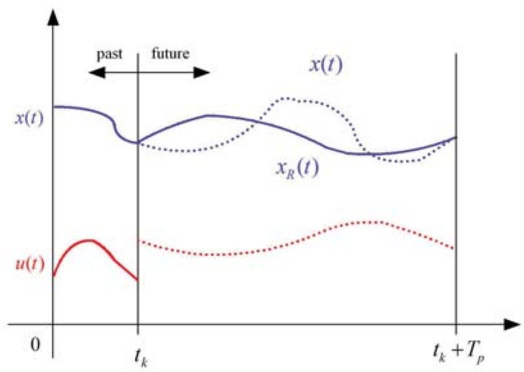
Nominal MPC

- Optimality
- Constraint satisfaction
- Guaranteed stability
- ...



Robust case

- Deteriorating performance
- Instability
- Infeasibility



Tube MPC of constrained systems

Considered control problem

$$\dot{x} = Ax + Bu + w \qquad (\star) \qquad \text{system dynamics}$$

subject to

$$x \in \mathcal{X}, \quad u \in \mathcal{U} \qquad \text{constraints}$$

$$w \in \mathcal{W} := \{w \mid \|w\| \leq w_{\max}\}, \qquad \text{bounded disturbances}$$

Objective

Design MPC controller that robustly stabilizes $(\star)$

Crucial aspects in robust MPC

- Robust constraint satisfaction
- Robust recursive feasibility
- Guaranteed robust stability
- Computational burden

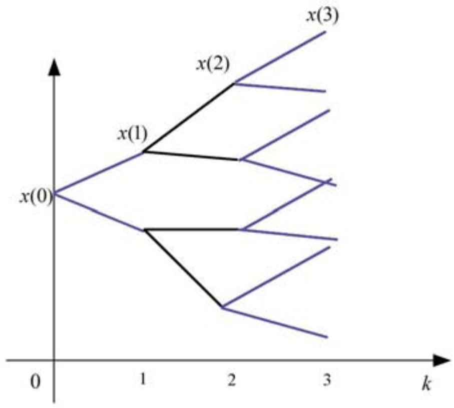
Min-max model predictive control

Optimization problem

$$\min_{u(\cdot)} \max_{w(\cdot)} J(x(0), u(\cdot), w(\cdot))$$

subject to

$$x^+ = Ax + Bu + w, \qquad x(0) = x_0,$$
$$x(k) \in \mathcal{X}, \quad u(k) \in \mathcal{U}, \quad k \in [0, N-1],$$
$$x(N) \in \Omega,$$



Key points

- Closed-loop prediction, i.e., $u = \kappa(x)$

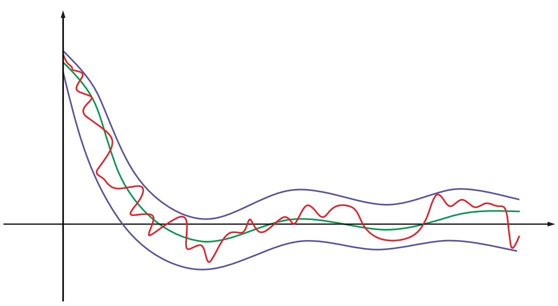
Disadvantages

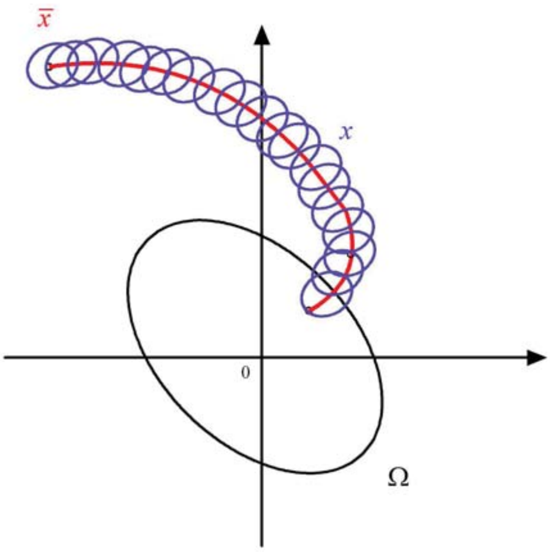
Computationally intractable

Tube MPC of constrained systems

Key idea

- Suggested control $u := \bar{u} + K(x - \bar{x})$
- Predict nominal trajectories
- Predict tube containing all admissible disturbed trajectories
- Solve **nominal** optimization problem with **tightened** constraints





Tube MPC of constrained systems

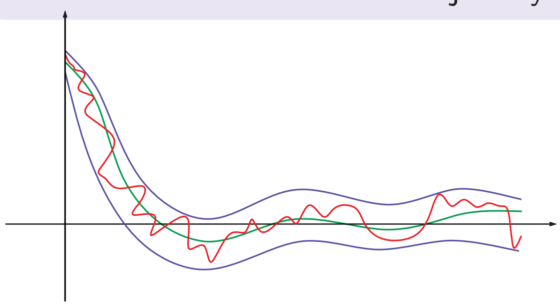
Define nominal system

$$\dot{\bar{x}} = A\bar{x} + B\bar{u}$$

$\bar{x}, \bar{u}$: nominal states/inputs
x, u: actual states/inputs

Nominal input $\bar{u}$

- Solves nominal optimal control problem
- Defines nominal state trajectory



Error system

$$v = x - \bar{x}$$
$$\dot{v} = (A + BK)v + w$$

Suggested control structure

$$u = \bar{u} + K(x - \bar{x})$$

Auxiliary control law $K(x - \bar{x})$

- Calculated **offline**
- Keeps error system in a **tube** along the nominal trajectory

Tube MPC of constrained systems

Considered optimization problem

$$\underset{\bar{x}(0),\bar{u}(\cdot)}{\text{minimize}} \quad J(\bar{x}(0), \bar{u}(\cdot))$$

subject to

$$\dot{\bar{x}} = A\bar{x} + B\bar{u}$$
$$x(0) - \bar{x}(0) \in \Sigma,$$
$$\bar{x}(\tau) \in \mathcal{X}_0,\ \bar{u}(\tau) \in \mathcal{U}_0, \qquad \tau \in [0, T_p)$$
$$\bar{x}(T_p) \in \Omega,$$

- $J(\bar{x}(0), \bar{u}(\cdot)) := \int_0^{T_p} \{\|\bar{x}(\tau)\|_Q^2 + \|\bar{u}(\tau)\|_R^2\} d\tau + E(\bar{x}(T_p))$
- Ω: robust invariant set of error system, $\mathcal{X}_0 := \mathcal{X} \ominus \Sigma$, $\mathcal{U}_0 := \mathcal{U} \ominus K\Sigma$,
- $E(\cdot)$ is terminal penalty, $\Omega \subseteq \mathcal{X} \ominus \Sigma$ is terminal region.

Tube and auxiliary control law are calculated offline

Tube MPC of constrained systems

Control structure

- Online: nominal input $\bar{u}$ minimizes nominal cost function
- Offline: auxiliary feedback law $K(x - \bar{x})$
- Applied input: $u = \bar{u} + K(x - \bar{x})$

Properties of tube MPC

- Recursive **feasibility**
- **Robustly asymptotically ultimately bounded**
- has been extended to nonlinear systems (Yu et al., 2010)

vs. min-max MPC

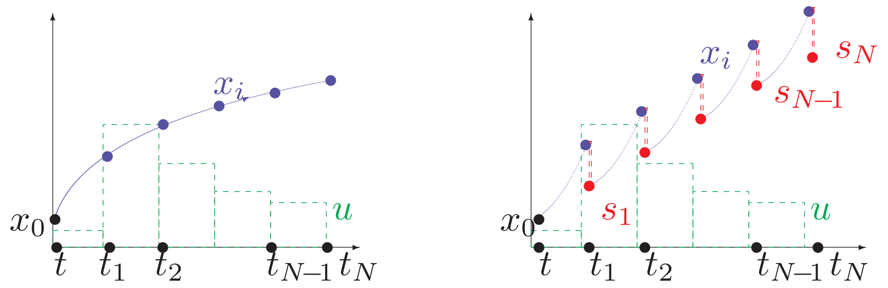
Minimization problem is solved online $\Longrightarrow$ **moderate online computational burden**

Issues in Nonlinear MPC

- **System theoretic formulation and investigation:**
  stability, performance, output feedback, robustness, ...
- **Implementation issues:**
  efficient and reliable real-time optimization, modeling, ...

Efficient Numerical Solution of the Open-Loop Optimal Control Problem

- Typically solved by **finite dimensional input parameterization**
- Can, for example be solved by: **direct multiple shooting method**



- Optimization problem shows special **sparse** structure
- Consecutive optimization problems are similar
  ⇒ use efficient **hot starting/embedding strategy**

State of the art solution strategy, specially tailored to NMPC
⇒ **allows real-time solution for realistically sized problems**
(based on MUSCOD II, *[Diehl et al.]*)

Approximate Solution: Real-time Iteration Scheme

Question:
Is it really necessary to solve optimal control problem "exactly"?

- Often sufficient to perform only **one** SQP **subiteration per sampling time** if:
  - special update is used
  - close to optimal solution
- Can be performed **fast**
- **Rigorous stability proof available**

[Diehl/Findeisen/Allgöwer/Bock/Schlöder '03]

Real-time iteration scheme
- Expands range of applicability (faster systems, ...)
- Utilizes inherent robustness of NMPC

OptCon Toolbox

- **OptCon is an efficient, user friendly package**
- **Freeware, based on HQP, TCL, (GNU license)**
- **Trivial installation: copy and it works!**
- **Good for *dummies* as well as experts**
- **Modular, easily extendable**
- **Simple Matlab, *fmincon* type usage**
- **Comes with OPC features, real-time simulator, for easy prototyping**
- **Several case studies have been developed: crane, pendulum, Klatt-Engell, MMA copolymerization, thin film deposition (180 ODEs), large scale FCCU (>2000 ODEs)**
- **Practical applications: BASF industrial batch reactor, four tank (IST laboratory)**

Distributed MPC



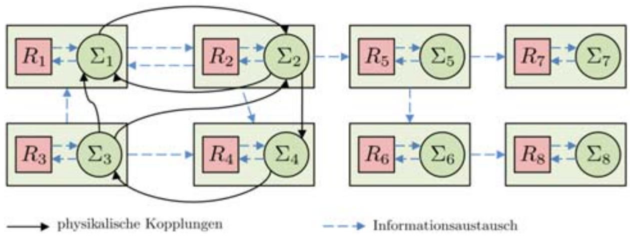
Motivation

- Several independent systems pursuing a cooperative task; e.g. formation flight, manufacturing robots
- Large-scale systems comprised of interconnected subsystems; e.g. chemical plant
- Centralized control not possible / not scalable

Goal

Distributed control with **limited information exchange** through **efficient use of predicted trajectories**.

Distributed MPC



Distributed MPC Setup

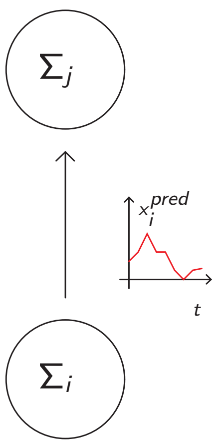
Several coupled subsystems Σ_i with local MPC-based controllers R_i

Different types of coupling between subsystems

Coupling through

- System dynamics (physical coupling), e.g. chemical plant
- Constraints, e.g. collision avoidance
- Cost function (common goal), e.g. consensus and synchronization (like formation flight)
- Communication network for information exchange

Distributed MPC



Solution strategy

At each sampling instant, in a sequential order, each system

- solves its local optimization problem, taking into account the latest plans $\tilde{\mathbf{x}}_{-i}$ of its neighbors,
- sends the optimal solution trajectory x_i^{pred} to its neighbors,
- applies the first part of the solution.

Main challenges

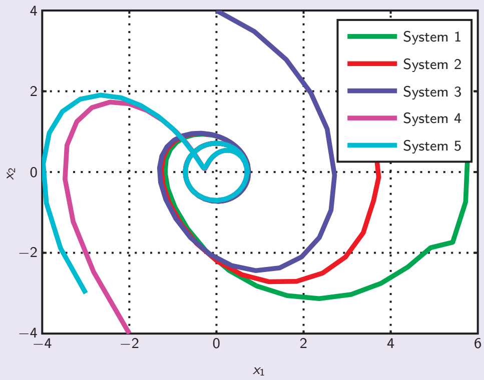
- How to define **terminal region**? $\rightarrow$ Use **time-varying distributed terminal regions** in order to satisfy centralized terminal constraint
- How to ensure decaying of optimal value functions? $\rightarrow$ Use **modified cost functionals** $\bar{J}_i(x_i, \tilde{\mathbf{x}}_{-i}, \mathbf{u}_i) := J_i(x_i, \tilde{\mathbf{x}}_{-i}, \mathbf{u}_i) + \sum_{\ell \in \mathcal{N}_i} J_{\ell i}(x_i, \tilde{\mathbf{x}}_\ell)$

Distributed MPC

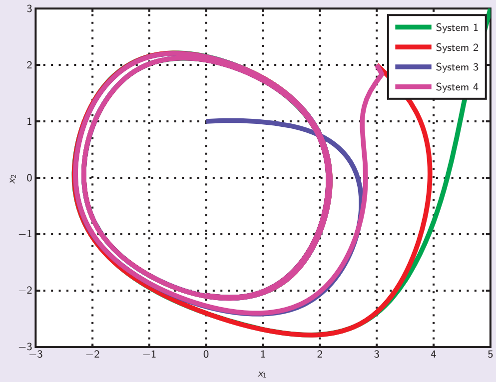
Properties

Both **recursive feasibility** and **convergence to cooperative goal** can be guaranteed for the proposed DMPC algorithm.

Examples



Synchronization of linear oscillators



Synchronization of nonlinear Van der Pol oscillators

Distributed MPC

Discussion

Features of distributed MPC:

- New field, high expectations, first promising results
- Exchange of predicted trajectories can efficiently be used to achieve desired cooperative goal
- Can be used for a variety of different practically relevant control tasks

Conclusions

- Wanted to demonstrate that on-line controller computation in real time may be an interesting alternative to analytical control.
- NMPC is by now well developed both from the control theoretic side as well as regarding practical aspects like computation.
- There are, however, still many open problems and the field is more active than ever.

My personal estimate is that in five years' time nonlinear techniques will be routinely applied in many industrial fields.