Numerical NMPC for systems with state dependent jumps and discrete actuators

Moritz Diehl

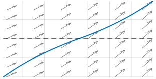
Systems Control and Optimization Laboratory
Department of Microsystems Engineering and Department of Mathematics
University of Freiburg, Germany

based on joint work with
Armin Nurkanovic, Sebastian Albrecht, Jonas Hall, Florian Messerer, Gianluca Frison, Benjamin Stickan, Sebastian Sager, Clemens Zeile, Angelika Altmann-Dieses and **Adrian Bürger** .

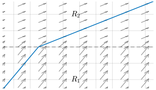
IFAC NMPC 2021 Conference
July 13, 2021

Non-Smooth Dynamics (NSD) - Informal Classification

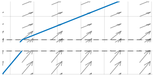
Regard ordinary differential equation (ODE) with **non-smooth** right hand side (RHS).
Distinguish three cases:



NSD1: non-differentiable RHS, e.g. $\dot{x} = 1 + |x|$



NSD2: state dependent ("internal") switch of RHS, e.g. $\dot{x} = 2 - \mathrm{sign}(x)$
(similar but different: external switch by discrete actuator)

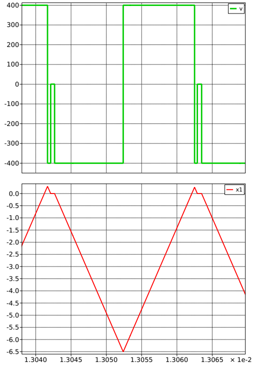


NSD3: state dependent jump, e.g. $x(t_+) = 3 + x(t_-)$

Switched NMPC for Electric DC-AC Power Converter (NSD2)

PhD work by Benjamin Stickan (Fraunhofer ISE) and Gianluca Frison

- NMPC aim: follow sinusoidal reference, react fast to grid failures
- 3 states, **1 binary input, 1 state dependent switch due to diodes (in blanking time)**
- sampling time: 25 microseconds, ARM A53@1.1GHz, horizon $N = 2$
- switching integrator, 3 RK4 and 4 Euler steps, generated as C code via CasADi
- hand tailored SQP real-time iteration, on track to be applied on industrial photovoltaic power converter (in DyConPV project).



Overview

- Optimization with Complementarity Constraints: Embracing the Nonconvex
- Finite Elements with Switch Detection (FESD)
- Time Freezing for State Dependent Jumps
- Three Step Decomposition for Discrete Actuators

NMPC needs to solve Nonlinear Programs (NLP)

Continuous Time NMPC Problem

$$\begin{aligned} \min_{x(\cdot),u(\cdot)} \quad & \int_0^T L(x,u)\mathrm{d}t + E(x(T)) \\ \text{s.t.} \quad & x(0) = \bar{x}_0 \\ & \dot{x}(t) = f(x(t),u(t)) \\ & 0 \geq h(x(t),u(t)),\ t \in [0,T] \\ & 0 \geq r(x(T)) \end{aligned}$$

Assume smooth convex L, E, h, r.
Nonlinear f makes problem nonconvex.
Direct methods discretize, then optimize.
E.g. collocation or multiple shooting.

Discretized NMPC Problem (an NLP)

$$\begin{aligned} \min_{x,z,u} \quad & \textstyle\sum_{k=0}^{N-1} \Phi_L(x_k, z_k, u_k) + E(x_N) \\ \text{s.t.} \quad & x_0 = \bar{x}_0 \\ & x_{k+1} = \Phi_f^{\text{dif}}(x_k, z_k, u_k) \\ & 0 = \Phi_f^{\text{alg}}(x_k, z_k, u_k) \\ & 0 \geq \Phi_h(x_k, z_k, u_k),\ k = 0, \dots, N-1 \\ & 0 \geq r(x_N) \end{aligned}$$

Again, smooth convex Φ_L, E, Φ_h, r.
Variables $x = (x_0, \dots)$ and $z = (z_0, \dots)$ and $u = (u_0, \dots, u_{N-1})$ can be summarized in vector $w \in \mathbb{R}^{n_w}$.

Nonlinear Programs (NLP) with Convex Structure

Newton-type methods generate a sequence $w_0, w_1, w_2, \ldots$ by linearizing and solving convex subproblems. E.g., sequential convex programming (SCP) linearizes nonconvex constraints.

Summarized NLP

$$\min_{w \in \mathbb{R}^{n_w}} \; J(w) \quad \text{s.t.} \quad 0 = F(w), \quad 0 \geq H(w)$$

Still assume smooth convex J, H.
Nonlinear F makes problem nonconvex.

SCP subproblem at linearization point w_i

$$w_{i+1} \in \arg \min_{w \in \mathbb{R}^{n_w}} \; J(w) \quad \text{s.t.} \quad 0 = F_{\mathrm{L}}(w; w_i), \quad 0 \geq H(w)$$

First order Taylor series:
$F_{\mathrm{L}}(w; w_i) := F(w_i) + \frac{\partial F}{\partial w}(w_i)(w - w_i)$

Works extremely well for mildly nonlinear F, also in microsecond NMPC [cf. Zanelli 2021, Lekic 2020, Hausberger 2020]

But what if there is significant nonconvex structure in the NLP ?

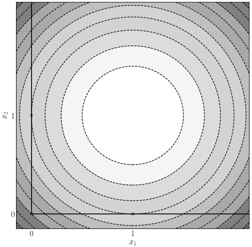
Mathematical Programs with Complementarity Constraints (MPCC)

NLP with additional constraints of complementarity type: $x \perp y \Leftrightarrow x^\top y = 0$

MPCC

$$\min_{w \in \mathbb{R}^{n_w}} \; J(w)$$
$$\text{s.t.} \quad 0 = F(w)$$
$$0 \geq H(w)$$
$$0 \leq Lw \perp Rw \geq 0$$

Convex J, H and smooth F.
Fixed matrices L, R.



Toy MPCC example:

$$\min_{w \in \mathbb{R}^2} \; (w_1 - 1)^2 + (w_2 - 1)^2$$
$$\text{s.t.} \;\; 0 \leq w_1 \perp w_2 \geq 0$$

Two local minimizers.
One local maximizer
(without constraint qualification)

Due to complementarity constraints, MPCC are nonsmooth and nonconvex.

MPCC Solution by Penalty Method

The penalty MPCC method [cf. Ferris 1999, Ralph&Wright 2004] generates sequence $w_0^*, w_1^*, w_2^*, \ldots$ by solving NLP with increasing weights $0 = \rho_0 < \rho_1 < \rho_2 < \ldots$, and NLP warm-starting.

MPCC

$$\begin{aligned} \min_{w \in \mathbb{R}^{n_w}} \quad & J(w) \\ \text{s.t.} \quad & 0 = F(w) \\ & 0 \geq H(w) \\ & 0 \leq Lw, \quad Rw \geq 0 \\ & 0 = \phi(w) \end{aligned}$$

with nonlinear nonconvex scalar

$$\phi(w) := (Lw)^\top Rw$$

Penalty subproblem for weight ρ_j

$$\begin{aligned} w_j^* \in \arg \min_{w \in \mathbb{R}^{n_w}} \quad & J(w) + \rho_j \, \phi(w) \\ \text{s.t.} \quad & 0 = F(w) \\ & 0 \geq H(w) \\ & 0 \leq Lw, \quad Rw \geq 0 \end{aligned}$$

Objective contribution $\rho_j \phi(w)$ is nonconvex.
Need good NLP solver (SCP, SQP, Interior Point, ...)
Crucial: start NLP solver at previous solution w_{j-1}^*.

One can often find "good" local minima with the penalty method.

Algorithms for MPCC: Two Examples

MPCC often exhibit structure that can be exploited by tailored solvers. We give two examples.

Generic Penalty Loop:

Penalty subproblem = NLP

$$\min_{w \in \mathbb{R}^{n_w}} \; J(w) + \rho_j \, \phi(w)$$
$$\text{s.t.} \quad 0 = F(w)$$
$$0 \geq H(w)$$
$$0 \leq Lw, \quad Rw \geq 0$$

For generic nonlinear MPCC, sequence of penalty NLPs can be solved e.g. by open-source solver IPOPT [Wächter and Biegler 2006].

Used for many results in this talk.

Solver LCQP for Linear Complementarity QP:

Penalty subproblem = nonconvex quadratic program

$$\min_{w \in \mathbb{R}^{n_w}} \; \frac{1}{2} w^\top Q \, w + c^\top w + \frac{\rho_j}{2} w^T (L^\top R + R^\top L) \, w$$
$$\text{s.t.} \quad 0 = Aw - b, \; 0 \geq Cw - d, \; 0 \leq Lw, \; Rw \geq 0$$

Solve by exact line-search SCP started at $w_{j,0} := w^*_{j-1}$:

SCP subproblem = convex quadratic program (QP)

$$\min_{w \in \mathbb{R}^{n_w}} \; \frac{1}{2} w^\top Q \, w + (c + \rho_j \nabla \phi(w_{ji}))^\top w$$
$$\text{s.t.} \quad 0 = Aw - b, \; 0 \geq Cw - d, \; 0 \leq Lw, \; Rw \geq 0$$

Solve by hot-started qpOASES [Ferreau 2014].

Benchmarking LCQP on Test Example

LCQP code developed and tested by Jonas Hall

Example from discretization of non-smooth optimal control problem [Stewart & Anitescu 2010].

Continuous OCP with NSD2 System

$$\min_{x(\cdot)} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \mathrm{sign}(x(t)), \quad t \in [0, 2]$$

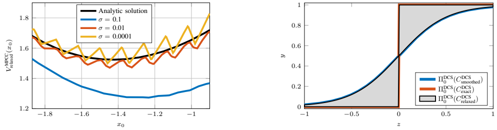
Use implicit Euler with step $h = 2/N$:

Linear Complementarity QP

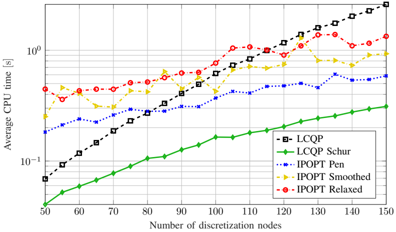
$$\min_{x,y,z} \sum_{k=0}^{N} E_k(x_k)$$

$$\begin{aligned} \text{s.t.} \quad & x_k = x_{k-1} + h(3 - 2y_k) \\ & 0 \leq x_k + z_k \perp 1 - y_k \geq 0 \\ & 0 \leq z_k \perp y_k \geq 0, \quad k = 1, \dots, N \end{aligned}$$

Visualization of relaxation for different $\sigma \sim 1/\rho$



Benchmark different MPCC solvers, vary N:



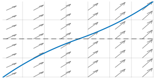
LCQP-Schur about 2x faster than IPOPT-Pen

Overview

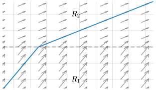
- Optimization with Complementarity Constraints: Embracing the Nonconvex
- **Finite Elements with Switch Detection (FESD)**
- Time Freezing for State Dependent Jumps
- Three Step Decomposition for Discrete Actuators

Non-Smooth Dynamics (NSD) - Informal Classification

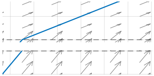
Regard ordinary differential equation (ODE) with **non-smooth** right hand side (RHS).
Distinguish three cases:



NSD1: non-differentiable RHS, e.g. $\dot{x} = 1 + |x|$



NSD2: state dependent ("internal") switch of RHS, e.g. $\dot{x} = 2 - \mathrm{sign}(x)$
(similar but different: external switch by discrete actuator)



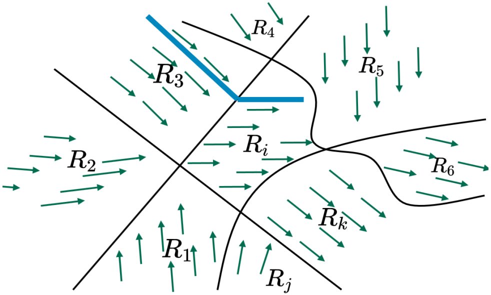
NSD3: state dependent jump, e.g. $x(t_+) = 3 + x(t_-)$

NSD2 Systems - State Dependent Switches

Regard **discontinuous** right hand side, piecewise smooth on disjoint open regions $R_i \subset \mathbb{R}^{n_x}$

Discontinuous ODE (NSD2)

$$\dot{x} = f_i(x,u), \text{ if } x \in R_i, \quad i \in \{1,\dots,m\}$$



Numerical aims:

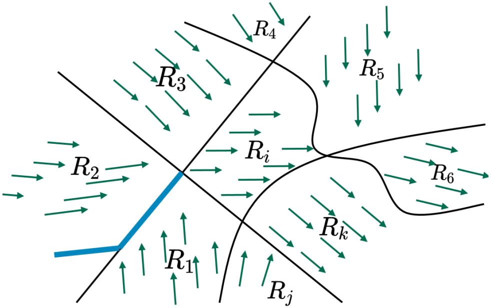
1. exactly detect switching times
2. obtain exact sensitivities across regions

NSD2 Systems - State Dependent Switches

Regard **discontinuous** right hand side, piecewise smooth on disjoint open regions $R_i \subset \mathbb{R}^{n_x}$

Discontinuous ODE (NSD2)

$$\dot{x} = f_i(x,u), \text{ if } x \in R_i, \quad i \in \{1, \ldots, m\}$$



Numerical aims:

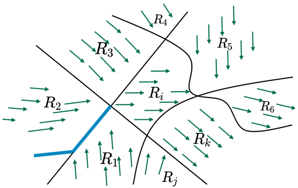
1. exactly detect switching times
2. obtain exact sensitivities across regions
3. appropriately treat evolution on boundaries (sliding mode $\rightarrow$ Filippov convexification)

Filippov Convexification

Dynamics not yet well-defined on region boundaries ∂R_i. Idea by A.F. Filippov (1923-2006): replace ODE by differential inclusion, using convex combination of neighboring vector fields.

Filippov Differential Inclusion

$$\dot{x} \in F_{\mathrm{F}}(x,u) := \left\{ \sum_{i=1}^{m} f_i(x,u)\,\theta_i \;\middle|\; \begin{array}{l} \sum_{i=1}^{m} \theta_i = 1, \\ \theta_i \geq 0, \quad i = 1, \ldots m, \\ \theta_i = 0, \quad \text{if } x \notin \overline{R_i} \end{array} \right\}$$

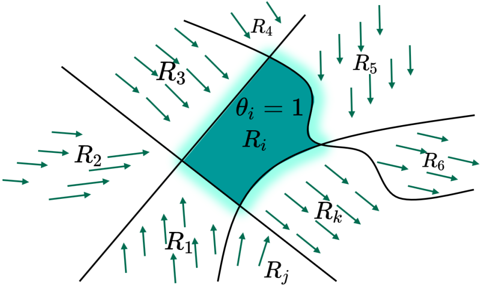


- for interior points $x \in R_i$ nothing changes: $F_{\mathrm{F}}(x,u) = \{f_i(x,u)\}$
- Provides meaningful generalization on region boundaries.
  E.g. on $\overline{R_1} \cap \overline{R_2}$ both θ_1 and θ_2 can be nonzero

How to compute convex multipliers θ?

Assume sets R_i given by [cf. Stewart, 1990]

$$R_i = \left\{ x \in \mathbb{R}^n \,\middle|\, g_i(x) < \min_{j \neq i} g_j(x) \right\}$$



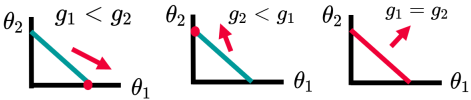
Linear program (LP) Representation

$$\dot{x} = \sum_{i=1}^{m} f_i(x,u)\,\theta_i^* \quad \text{with}$$

$$\theta^* \in \arg\min_{\theta \in \mathbb{R}^m} \quad \sum_{i=1}^{m} g_i(x)\,\theta_i$$

$$\text{s.t.} \quad \sum_{i=1}^{m} \theta_i = 1$$

$$\theta \geq 0.$$



From Filippov to dynamic complementarity systems

Using the KKT conditions of the parametric LP

LP representation

$$\dot{x} = F(x,u)\,\theta^*$$

$$\text{with} \quad \theta^* \in \underset{\theta\in\mathbb{R}^m}{\operatorname{argmin}} \quad g(x)^\top\theta \quad \text{s.t.} \quad 0 \leq \theta, \quad 1 = e^\top\theta$$

where

$$F(x,u) := [f_1(x,u),\dots,f_m(x,u)] \in \mathbb{R}^{n_x\times m}$$
$$g(x) := [g_1(x),\dots,g_m(x)]^\top \in \mathbb{R}^m$$
$$e := [1,1,\dots,1]^\top \in \mathbb{R}^m$$

Express equivalently by optimality conditions:

Dynamic Complementarity System (DCS)

$$\dot{x} = F(x,u)\,\theta \tag{1a}$$
$$0 = g(x) - \lambda - e\mu \tag{1b}$$
$$0 \leq \theta \perp \lambda \geq 0 \tag{1c}$$
$$1 = e^\top\theta \tag{1d}$$

- $\mu \in \mathbb{R}$ and $\lambda \in \mathbb{R}^m$ are Lagrange multipliers
- (1c) $\Leftrightarrow \min\{\theta,\lambda\} = 0 \in \mathbb{R}^m$
- Together, (1b), (1c), (1d) determine the $(2m+1)$ variables θ, λ, μ uniquely

Conventional Discretization by Implicit Runge Kutta (IRK) Method

Continuous time DCS

$$
\begin{aligned}
x(0) &= \bar{x}_0, \\
\dot{x}(t) &= v(t) \\
v(t) &= F(x(t), u(t))\,\theta(t) \\
0 &= g(x(t)) - \lambda(t) - e\mu(t) \\
0 &\le \theta(t) \perp \lambda(t) \ge 0 \\
1 &= e^\top \theta(t), \quad t \in [0, T]
\end{aligned}
$$

Discrete time IRK-DCS equation

$$
\begin{aligned}
x_{0,0} &= \bar{x}_0, \quad x_{k+1,0} = x_{k,0} + h \textstyle\sum_{n=1}^{s} b_n v_{k,n} \\
x_{k,j} &= x_{k,0} + h \textstyle\sum_{n=1}^{s} a_{jn} v_{k,n} \\
v_{k,j} &= F(x_{k,j}, u_{k,j})\,\theta_{k,j} \\
0 &= g(x_{k,j}) - \lambda_{k,j} - e\mu_{k,j} \\
0 &\le \theta_{k,j} \perp \lambda_{k,j} \ge 0 \\
1 &= e^\top \theta_{k,j}, \quad j = 1, \dots, s, \quad k = 0, \dots, N-1
\end{aligned}
$$

Notation: $x_{k,r} \in \mathbb{R}^{n_x}, \theta_{k,r} \in \mathbb{R}^m$ etc. with:

- $k \in \{0, 1, \dots, N\}$ - index of integration step; step length $h := T/N$
- $j, n \in \{0, 1, \dots, s\}$ - index of intermediate IRK stage / collocation point
- a_{jn} and b_n - Butcher tableau entries of Implicit Runge Kutta method

Conventional Collocation - Illustrative Example

Regard example with $x \in \mathbb{R}^2$ and constants $a, k, c > 0$:

$$\dot{x} = \begin{cases} f_1(x), & x_1 > 0, \\ f_2(x), & x_1 < 0. \end{cases}$$

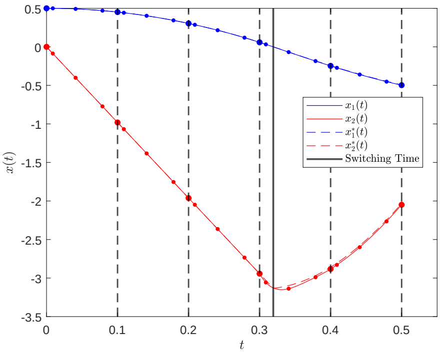
$$f_1(x) = \begin{pmatrix} x_2 \\ -a \end{pmatrix}, \quad f_2(x) = \begin{pmatrix} x_2 \\ -kx_1 - cx_2 \end{pmatrix}$$

$$g_1(x) = -x_1,$$
$$g_2(x) = x_1$$

$$\bar{x}_0 = [0.5\,,\ 0]^\top$$

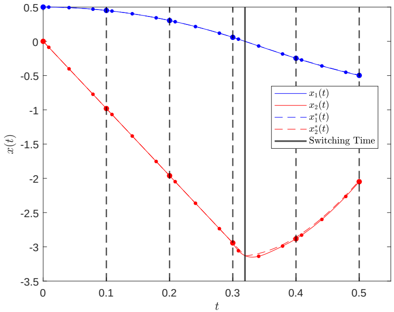
.

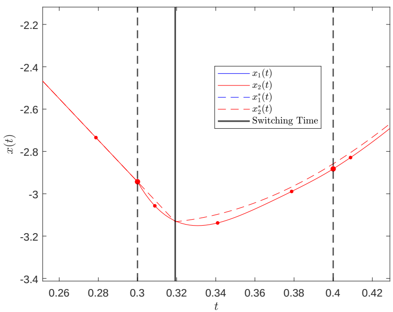
Solve with IRK Radau IIA method of order 7, $s = 4,\ N = 5,\ T = 0.5,\ h = 0.1.$



Conventional Collocation - Illustrative Example

Zoom in





High integration accuracy of 7th order IRK method is lost in fourth time step.
Reason: we try to approximate a non-smooth function by a (smooth) polynomial.

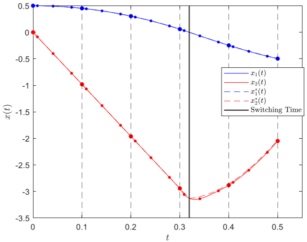
Question: could we ensure that switches happen only at element boundaries?
→ Finite Elements with Switch Detection (FESD)

Finite Elements with Switch Detection (FESD)

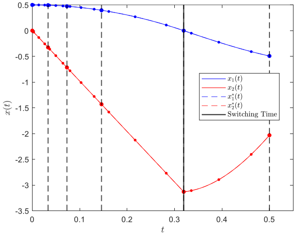
PhD work of A. Nurkanovic, to be submitted

FESD is a novel DCS discretization method based on three ideas:

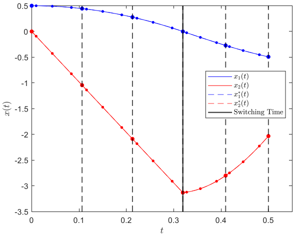
- make stepsizes h_k free, ensure $\sum_{k=0}^{N-1} h_k = T$ [cf. Baumrucker & Biegler 2009]
- allow switches only at element boundaries, enforce via *cross-complementarities*
- remove spurious degrees of freedom via *step equilibration*



conventional discretization



variable stepsizes and cross-complementarities



FESD discretization with step equilibration

Conventional DCS and FESD discretization without step equilibration

Conventional discretization

$$
\begin{aligned}
x_{0,0} &= \bar{x}_0, \quad h = T/N \\
x_{k+1,0} &= x_{k,0} + h \textstyle\sum_{n=1}^{s} b_n v_{k,n} \\
x_{k,j} &= x_{k,0} + h \textstyle\sum_{n=1}^{s} a_{jn} v_{k,n} \\
v_{k,j} &= F(x_{k,j}, u_{k,j})\, \theta_{k,j} \\
0 &= g(x_{k,j}) - \lambda_{k,j} - e\mu_{k,j} \\
0 &\leq \theta_{k,j} \perp \lambda_{k,j} \geq 0 \\
1 &= e^\top \theta_{k,j}
\end{aligned}
$$

for $j = 1, \ldots, s$

and $k = 0, \ldots, N-1$

FESD discretization without step equilibration

$$
\begin{aligned}
x_{0,0} &= \bar{x}_0, \; \textstyle\sum_{k=0}^{N-1} h_k = T \\
x_{k+1,0} &= x_{k,0} + h_k \textstyle\sum_{n=1}^{s} b_n v_{k,n} \\
x_{k,j} &= x_{k,0} + h_k \textstyle\sum_{n=1}^{s} a_{jn} v_{k,n} \\
v_{k,j} &= F(x_{k,j}, u_{k,j})\, \theta_{k,j} \\
0 &= g(x_{k,j'}) - \lambda_{k,j'} - e\mu_{k,j'} \\
0 &\leq \theta_{k,j} \perp \lambda_{k,j'} \geq 0 \quad \text{(cross-complementarities)} \\
1 &= e^\top \theta_{k,j}
\end{aligned}
$$

for $j = 1, \ldots, s$ and $k = 0, \ldots, N-1$

and $j' = 0, 1, \ldots, s$

- N extra variables $(h_0, \ldots, h_{N-1})$ restricted by one extra equality
- additional multipliers $\lambda_{k,0}, \mu_{k,0}$ are uniquely determined

Conventional DCS and FESD discretization with step equilibration

Conventional discretization

$$
\begin{aligned}
x_{0,0} &= \bar{x}_0, \quad h = T/N \\
x_{k+1,0} &= x_{k,0} + h \textstyle\sum_{n=1}^{s} b_n v_{k,n} \\
x_{k,j} &= x_{k,0} + h \textstyle\sum_{n=1}^{s} a_{jn} v_{k,n} \\
v_{k,j} &= F(x_{k,j}, u_{k,j})\, \theta_{k,j} \\
0 &= g(x_{k,j}) - \lambda_{k,j} - e\mu_{k,j} \\
0 &\leq \theta_{k,j} \perp \lambda_{k,j} \geq 0 \\
1 &= e^\top \theta_{k,j}
\end{aligned}
$$

for $j = 1, \dots, s$

and $k = 0, \dots, N-1$

FESD discretization with step equilibration

$$
\begin{aligned}
x_{0,0} &= \bar{x}_0, \; \textstyle\sum_{k=0}^{N-1} h_k = T \\
x_{k+1,0} &= x_{k,0} + h_k \textstyle\sum_{n=1}^{s} b_n v_{k,n} \\
x_{k,j} &= x_{k,0} + h_k \textstyle\sum_{n=1}^{s} a_{jn} v_{k,n} \\
v_{k,j} &= F(x_{k,j}, u_{k,j})\, \theta_{k,j} \\
0 &= g(x_{k,j'}) - \lambda_{k,j'} - e\mu_{k,j'} \\
0 &\leq \theta_{k,j} \perp \lambda_{k,j'} \geq 0 \quad \text{(cross-complementarities)} \\
1 &= e^\top \theta_{k,j} \\
0 &= \nu(\theta_{k'}, \theta_{k'+1}, \lambda_{k'}, \lambda_{k'+1}) \cdot (h_{k'} - h_{k'+1})^2
\end{aligned}
$$

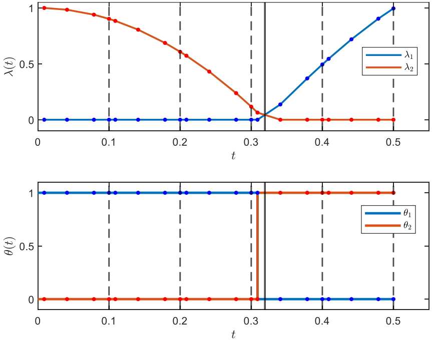
for $j = 1, \dots, s$ and $k = 0, \dots, N-1$

and $j' = 0, 1, \dots, s$ and $k' = 0, \dots, N-2$

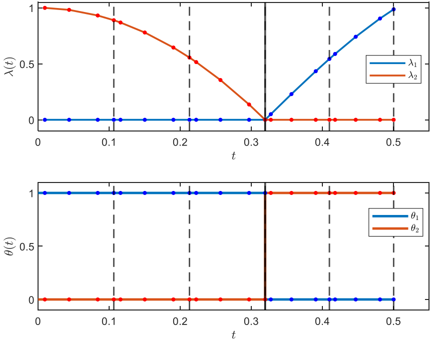
- N extra FESD variables $(h_0, \dots, h_{N-1})$ now locally uniquely determined by N constraints
- "Nurkanovic's indicator function" $\nu(\theta_{k'}, \theta_{k'+1}, \lambda_{k'}, \lambda_{k'+1})$ only zero if a switch occurs

Multipliers in Conventional and FESD Discretization

Conventional Collocation:



FESD Discretization:



FESD's cross-complementarities exploit the fact that the multiplier $\lambda_i(t)$ is continuous in time.
On boundary, $\lambda_i(t_k)$ must be zero if $\theta_i(t) > 0$ for any $t \in [t_{k-1}, t_{k+1}]$ on the adjacent intervals.

Numerical simulation example: unstable switched oscillator

Regard an unstable non-smooth oscillator

$$\dot{x}(t) = \begin{cases} A_1 x, & c(x) < 0, \\ A_2 x, & c(x) > 0, \end{cases}$$

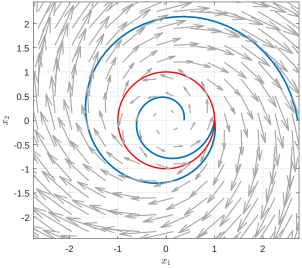
with

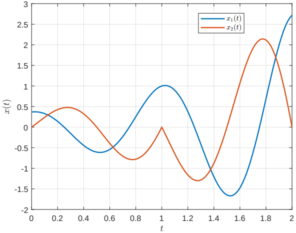
$$A_1 = \begin{bmatrix} 1 & \omega \\ -\omega & 1 \end{bmatrix}, \; A_2 = \begin{bmatrix} 1 & -\omega \\ \omega & 1 \end{bmatrix},$$

$$c(x) = x_1^2 + x_2^2 - 1, \; \omega = 2\pi, \; x(0) = [e^{-1} \; 0]^\top$$

For $t \in [0, 2]$, we have

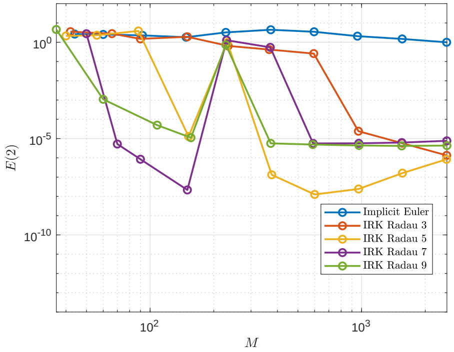
$$x(2) = [e \; 0]^\top.$$



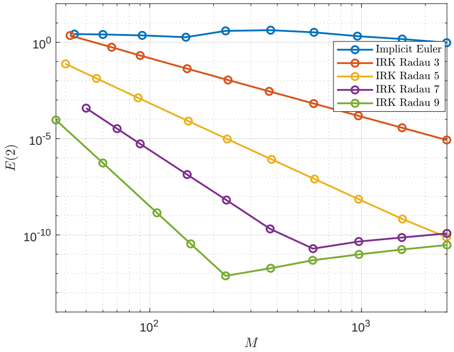


FESD recovers high integration order for switched systems

Conventional Collocation:



FESD Discretization:



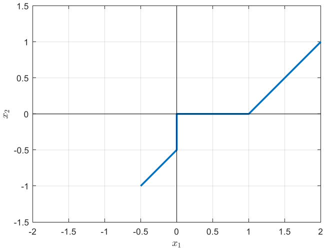
Integration error $E(T)$ at time $T = 2$ vs. total number $M = s\,N$ of collocation points, for different Radau IIA methods.

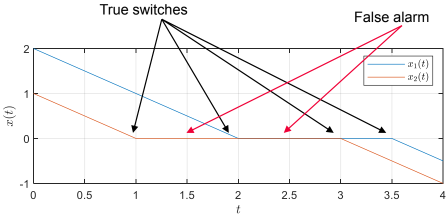
FESD discretization delivers versatile MPCC formulation with high integration order

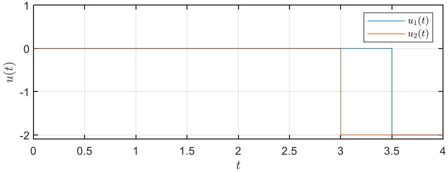
Optimal Control Example: Solution Trajectory with 3 Sliding Modes

Regard the following OCP

$$
\begin{aligned}
\min_{x(.),u(.)} \quad & \int_0^4 u(t)^\top u(t) \mathrm{d}t \\
\text{s.t.} \quad & x(0) = (2,1), \\
& \dot{x}(t) \in -\mathrm{sign}(x(t)) + u(t), \; t \in [0,4], \\
& -2e_2 \leq u(t) \leq 2e_2, \; t \in [0,4], \\
& x(4) = (-1,-0.5).
\end{aligned}
$$

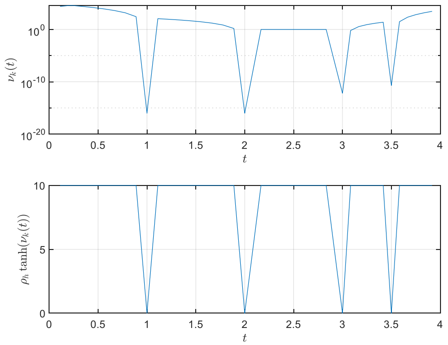




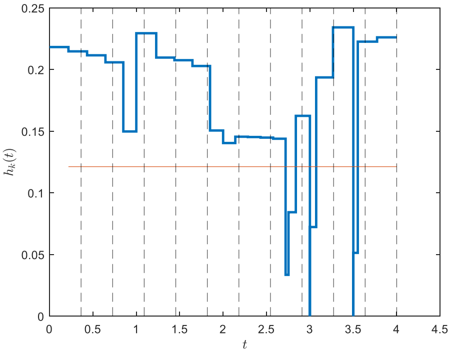


Numerical Solution without Equilibration

Nurkanovic's indicator function over time:



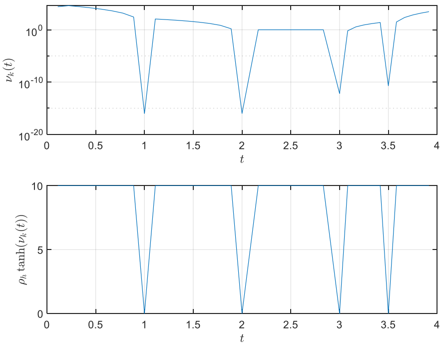
Step size over time:



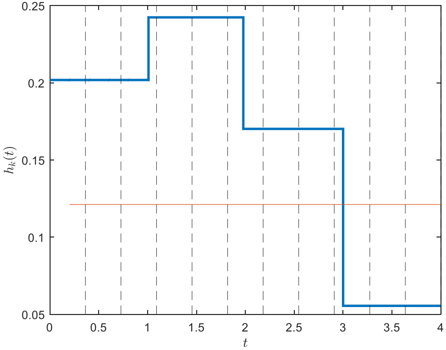
Optimizer varies step size randomly, potentially playing with integration errors.

Numerical Solution with Equilibration

Nurkanovic's indicator function over time:



Step size over time:



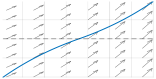
Equidistant grid on each "switching stage". Jumps exactly at switching times.

Overview

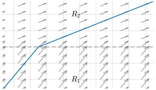
- Optimization with Complementarity Constraints: Embracing the Nonconvex
- Finite Elements with Switch Detection (FESD)
- **Time Freezing for State Dependent Jumps**
- Three Step Decomposition for Discrete Actuators

Non-Smooth Dynamics (NSD) - Informal Classification

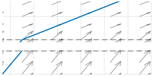
Regard ordinary differential equation (ODE) with **non-smooth** right hand side (RHS).
Distinguish three cases:



NSD1: non-differentiable RHS, e.g. $\dot{x} = 1 + |x|$



NSD2: state dependent ("internal") switch of RHS, e.g. $\dot{x} = 2 - \mathrm{sign}(x)$
(similar but different: external switch by discrete actuator)



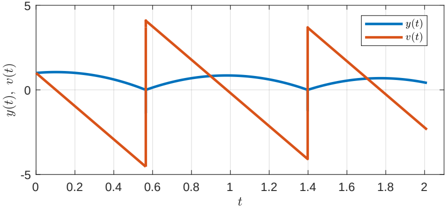
NSD3: state dependent jump, e.g. $x(t_+) = 3 + x(t_-)$

NSD3 State Jump Example: Bouncing Ball

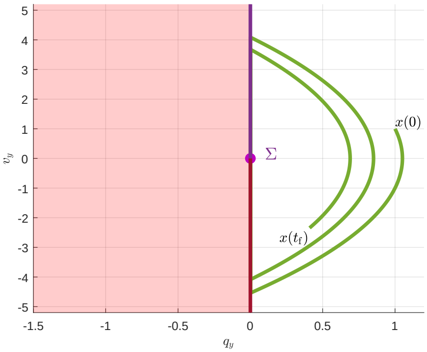
Bouncing ball with state $x = (y, v)$:

$$\begin{aligned} m\dot{v} &= -mg, && \text{if } y > 0 \\ v(t^+) &= -0.9\, v(t^-), && \text{if } y(t-) = 0 \text{ and } v(t^-) < 0 \end{aligned}$$

Time plot of bouncing ball trajectory:



Phase plot of bouncing ball trajectory:



Question: could we transform NSD3 systems into (easier) NSD2 systems?

Three ideas:

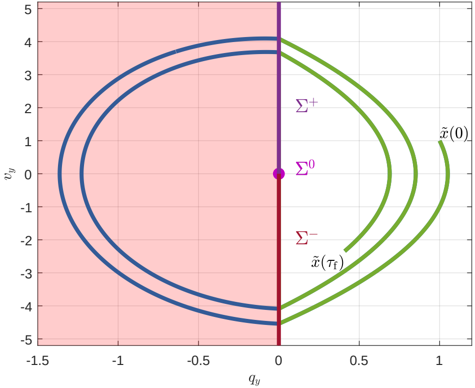
1. mimic state jump by **auxiliary dynamic system** $\dot{x} = \varphi(x)$ on prohibited region
2. introduce a **clock state** $t(\tau)$ that stops counting when the auxiliary system is active
3. adapt speed of time, $\frac{\mathrm{d}t}{\mathrm{d}\tau} = s$ with $s \geq 1$, and **impose terminal constraint** $t(T) = T$

The time-freezing reformulation

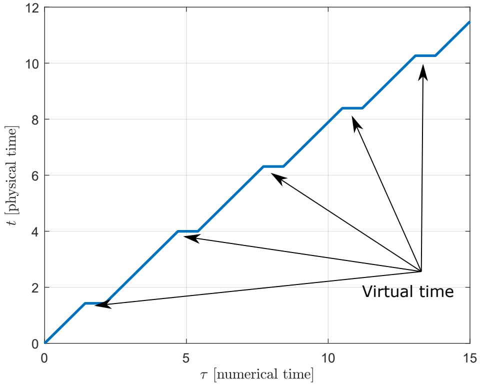
Augmented state $(x, t) \in \mathbb{R}^{n+1}$ evolves in **numerical time** τ. Augmented system is nonsmooth, of NSD2 type:

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\begin{bmatrix} x \\ t \end{bmatrix} = \begin{cases} s\begin{bmatrix} f(x) \\ 1 \end{bmatrix}, & \text{if } c(x) \geq 0 \\ \begin{bmatrix} \varphi(x) \\ 0 \end{bmatrix}, & \text{if } c(x) < 0 \end{cases}$$

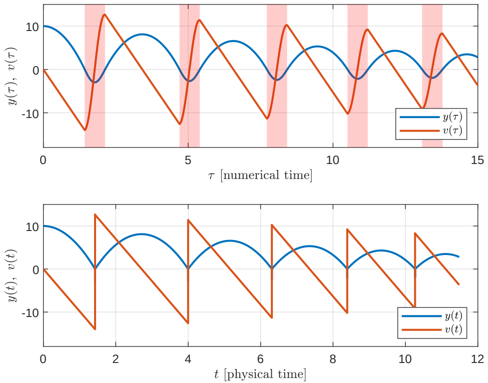
- During normal times, system and clock state evolve with adapted speed $s \geq 1$.
- Auxiliary system $\frac{\mathrm{d}x}{\mathrm{d}\tau} = \varphi(x)$ mimics state jump while time is frozen, $\frac{\mathrm{d}t}{\mathrm{d}\tau} = 0$.



Time-freezing for bouncing ball example



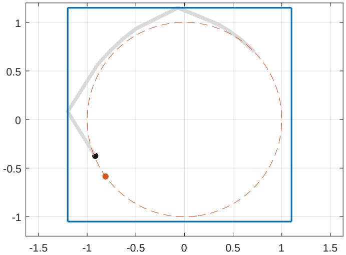
Evolution of physical time (clock state) during augmented system simulation ($s = 1$).



We can recover the true solution by plotting $x(\tau)$ vs. $t(\tau)$ and disregarding "frozen pieces".

A Tracking NMPC Example with Time-Freezing and FESD

Regard bouncing ball in two dimensions driven by bounded force: $\ddot{q} = u$



- augmented state $x = (q, \dot{q}, t) \in \mathbb{R}^5$
- $m = 9$ regions (8 with auxiliary dynamics for state jumps)

$$\min_{\substack{x(.),u(.),s(.),\\ \theta(.),\lambda(.),\mu(.)}} \int_0^T (q - q_{\text{ref}}(\tau))^\top (q - q_{\text{ref}}(\tau))\, s(\tau)\, \mathrm{d}\tau$$

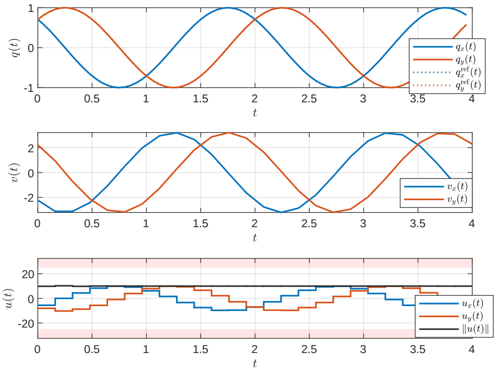
$$\begin{aligned}
\text{s.t.}\quad & x(0) = x_0, \quad t(T) = T, \\
& x'(\tau) = \sum_{i=1}^{m} \theta_i(\tau) f_i(x(\tau), u(\tau), s(\tau)), \\
& 0 = g(x(\tau)) - \lambda(\tau) - \mu(\tau) e, \\
& 0 \leq \lambda(\tau) \perp \theta(\tau) \geq 0, \\
& 1 = e^\top \theta(\tau), \\
& \|u(\tau)\|_2^2 \leq u_{\max}^2, \\
& 1 \leq s(\tau) \leq s_{\max}, \;\; \tau \in [0, T].
\end{aligned}$$

$$q_{\text{ref}}(\tau) = (R\cos(\omega\, t(\tau)), R\sin(\omega\, t(\tau))).$$

Results with slowly moving reference

For $\omega = \pi$, tracking is easy: no jumps occur in optimal solution.

- Regard time horizon of two periods
- $N = 25$ equidistant control intervals
- use FESD with $N_{\text{FESD}} = 3$ finite elements with Radau 3 on each control interval
- each FESD interval has one constant control u and one speed of time s
- MPCC solved via ℓ_∞ penalty reformulation and homotopy
- For homotopy convergence: in total 4 NLPs solved with `IPOPT` via `CasADi`



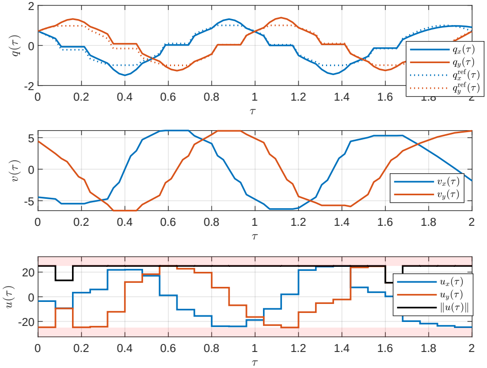
States and controls in physical time.

Results with slowly moving reference - Movie

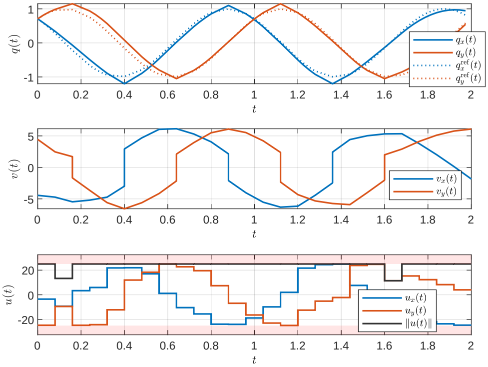
For $\omega = \pi$, tracking is easy: no jumps occur in optimal solution.

Results with fast reference

For $\omega = 2\pi$, tracking is only possible if ball bounces against walls.



States and controls in numerical time.



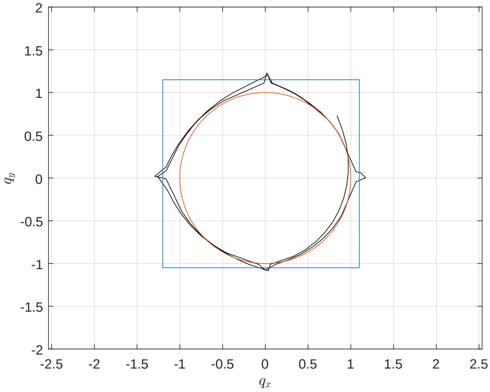
States and controls in physical time time.

Results with fast reference - Movie

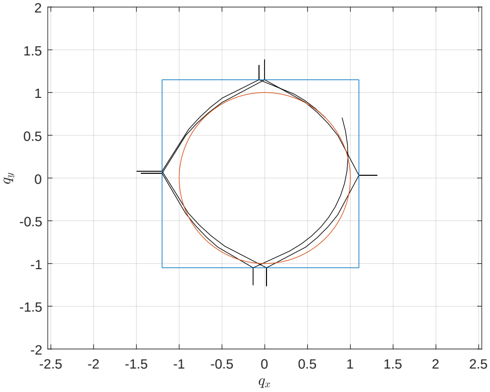
For $\omega = 2\pi$, tracking is only possible if ball bounces against walls.

Homotopy: first iteration vs converged solution

Geometric trajectory



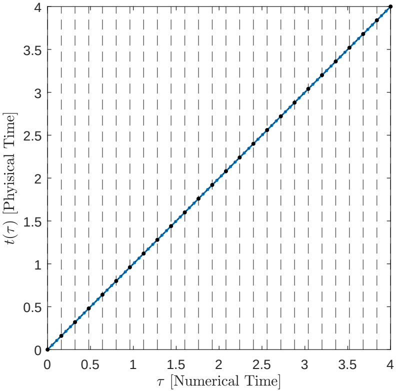
After the first homotopy iteration



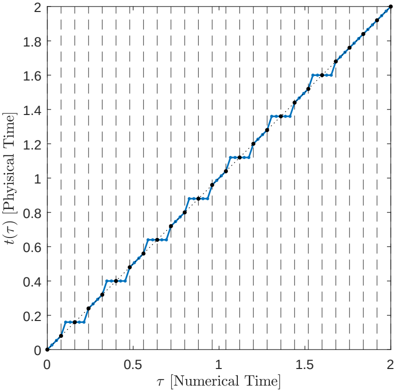
The solution trajectory after convergence

Physical vs. Numerical Time

for $\omega = \pi$



for $\omega = 2\pi$



Hopping Robot - move with minimal effort from start to end position

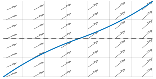
Homotopy initialized with start position everywhere. Optimizer finds creative soluton. Not with FESD yet.

Overview

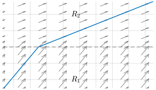
- Optimization with Complementarity Constraints: Embracing the Nonconvex
- Finite Elements with Switch Detection (FESD)
- Time Freezing for State Dependent Jumps
- **Three Step Decomposition for Discrete Actuators**

Non-Smooth Dynamics (NSD) - Informal Classification

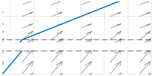
Regard ordinary differential equation (ODE) with **non-smooth** right hand side (RHS).
Distinguish three cases:



NSD1: non-differentiable RHS, e.g. $\dot{x} = 1 + |x|$



NSD2: state dependent ("internal") switch of RHS, e.g. $\dot{x} = 2 - \mathrm{sign}(x)$
(**similar but different: external switch by discrete actuator**)



NSD3: state dependent jump, e.g. $x(t_+) = 3 + x(t_-)$

Mixed Integer Optimal Control Problem with Binary Inputs $b(t)$

Formulated in outer convexified form. Can equivalently be formulated with complementarity constraints.

$$
\begin{aligned}
\underset{x(\cdot),u(\cdot),b(\cdot),s(\cdot)}{\text{minimize}} \quad & \int_0^T L\big(x,u,b,s\big)\,\mathrm{d}t + M\big(x(T)\big) && \text{(2a)}\\
\text{subject to} \quad & x(0) = \bar{x}_0 && \text{(2b)}\\
& \frac{\mathrm{d}x}{\mathrm{d}t} = \sum_{i=1}^{n_b} b_i \cdot f_i\big(x,u,c\big), \quad \sum_{i=1}^{n_b} b_i(t) = 1, && \text{(2c)}\\
& b_i(t) \in \{0,1\} \quad \Big[\Leftrightarrow 0 \le b_i(t) \perp (1 - b_i(t)) \ge 0 \Big] \quad \text{for } \ i = 1,\dots n_b, && \text{(2d)}\\
& -s + r_\mathrm{l} \le r\big(x,u,b,c\big) \le r_\mathrm{u} + s, \qquad \text{for } \ t \in \big[0,T\big] && \text{(2e)}\\
& (\ + \text{additional combinatorial constraints}) && \text{(2f)}
\end{aligned}
$$

$x(t)$: states, $u(t)$: continuous controls, $b(t)$: binary controls, $s(t)$: slack variables

$c(t)$: time-varying parameters, f: system dynamics, $r_\mathrm{l} \le r \le r_\mathrm{u}$: path constraints

Discretize to obtain MINLP. Global solution usually prohibitive (cf. Ruth Misener's plenary).

Three step decomposition for fast approximate MIOCP solution

Combinatorial Integral Approximation (CIA)[1]

1. Solve relaxed NLP with $b(t) \in [0,1]^{n_b}$ to obtain relaxed solution $b^*(t)$ for $t \in [0,T]$.
2. Solve minimum distance problem to find binary trajectory $b^{**}(\cdot)$ closest to $b^*(\cdot)$.
3. Solve an NLP where the binary controls are fixed to $b^{**}(t)$, to adjust $x(\cdot)$ and $u(\cdot)$.

Distance function in Step 2 is the "CIA distance" which measures the maximum of the integral of the difference of the trajectories. Fast tailored solvers for this special problem – an MILP – exist, e.g. in the python package pycombina [Bürger 2019].

[1] S. Sager, M. Jung, and C. Kirches: Combinatorial Integral Approximation, Mathematical Methods of Operations Research, vol. 73, no. 3, pp. 363-380, 2011.

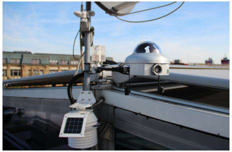
NMPC for a solar thermal test plant

at Karlsruhe University of Applied Sciences, with two discrete actuators

Control cabinet, cold storage, ACM, hot storage, pumps (cellar)

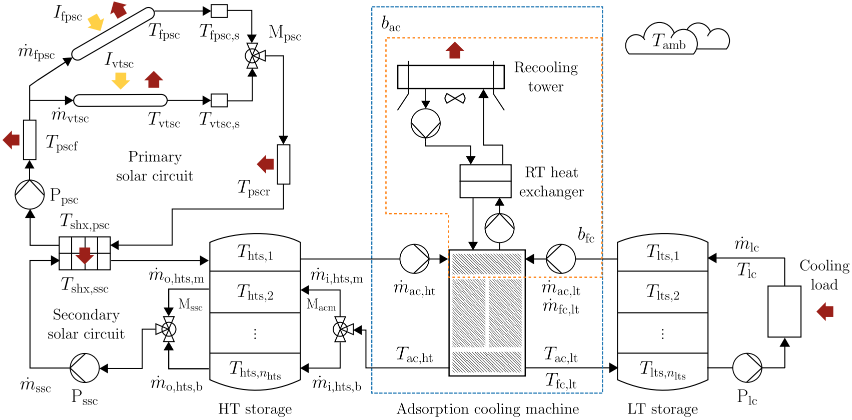
Plate collectors (roof)

Recooling unit (roof)



Ambient sensors (roof)

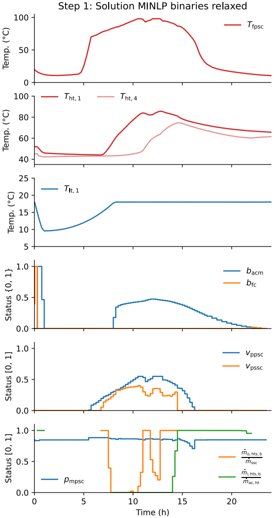
Vacuum tube collectors (roof)



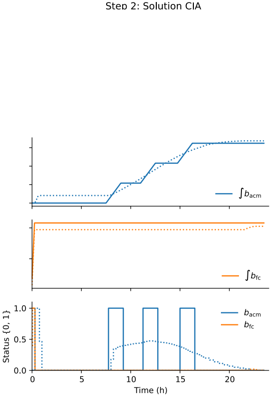
Nonlinear switched system ODE model with $n_x = 20$, $n_b = 2$, $n_u = 5$, and $n_c = 4$,

differentiable in all arguments within the domain of interest

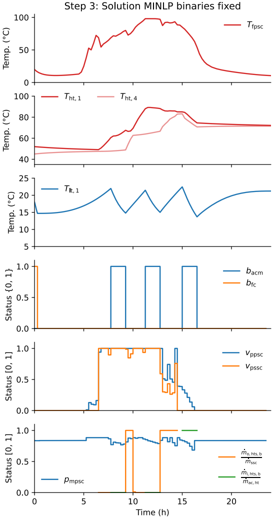
Numerical results: Three Step CIA Decomposition



(25 CPU sec)



(0.02 CPU sec)

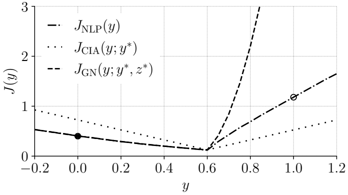


(18 CPU sec)

Alternative to CIA Decomposition: Gauss-Newton based MIQP

[Bürger et al., submitted to CDC 2021]

- Derive convex Gauss-Newton-type approximation of original MINLP from linearization at relaxed MINLP solution.
- Solution of resulting MIQP can yield improved integer solution in terms of objective and feasibility of the original MINLP.
- MIQP is equivalent to minimization of a distance function that is a first order accurate approximation of the true objective.



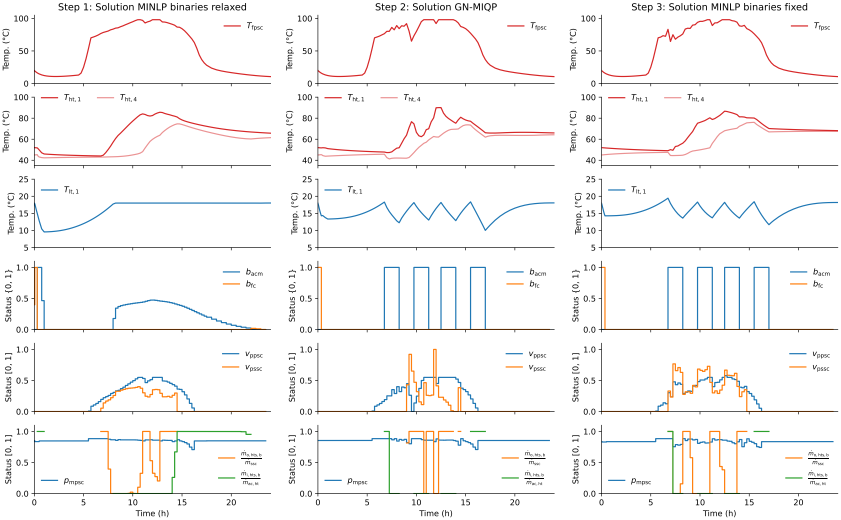
Original MINLP

$$\begin{aligned} \min_{y,z} \quad & \frac{1}{2}\|F_1(y,z)\|_2^2 + F_2(y,z) \\ \text{s.t.} \quad & G(y,z) = 0 \\ & H(y,z) \leq 0 \\ & y \in \mathbb{Z}^{n_y} \end{aligned}$$

GN-MIQP from linearization at (y^*, z^*)

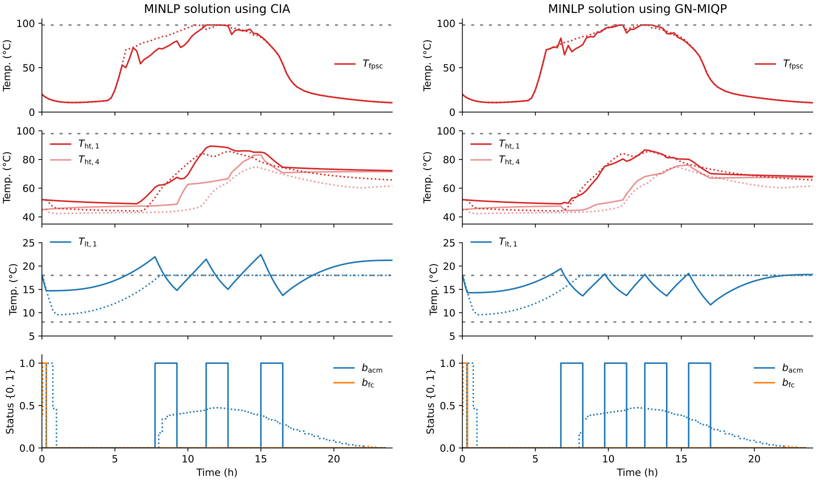
$$\begin{aligned} \min_{y,z} \quad & \frac{1}{2}\|F_{1,\mathrm{L}}(y,z;\bar{y},z^*)\|_2^2 + F_{2,\mathrm{L}}(y,z;y^*,z^*) \\ \text{s.t.} \quad & G_{\mathrm{L}}(y,z;y^*,z^*) = 0 \\ & H_{\mathrm{L}}(y,z;y^*,z^*) \leq 0 \\ & y \in \mathbb{Z}^{n_y} \end{aligned}$$

Numerical results: Three Step GN-MIQP Decomposition



(25 CPU sec) (130 CPU sec) (15 CPU sec)

Comparison of CIA and GN-MIQP Solution



(43 CPU sec) (170 CPU sec)

GN-MIQP delivers significant feasibility improvements, at the expense of increased computational cost.

Conclusions

- Mathematical Programs with Complementarity Constraints (MPCC) are a powerful tool to formulate and solve nonsmooth and nonconvex optimization problems.
- Finite Elements with Switch Detection (FESD) allow highly accurate simulation and optimal control for switched systems of level NSD2.
- Time-Freezing allows us to transform systems with state jumps of level NSD3 to the easier level NSD2 (which can be treated with FESD).
- NMPC with discrete actuators can efficiently be addressed by a three-step decomposition method. In Step 2, either a cheap MILP or a more accurate MIQP can be solved.

References 1 (own, most related to this talk)

- Survey of Sequential Convex Programming and Generalized Gauss-Newton Methods. F. Messerer, K. Baumgärtner and M. Diehl, ESAIM: Proc. Surv., accepted, 2022.
- A Sequential Convex Programming Approach to Solving Quadratic Programs and Optimal Control Problems with Linear Complementarity Constraints. J. Hall, A. Nurkanovic, F. Messerer and M. Diehl, in IEEE Cont. Sys. Lett., 2022.
- A time-freezing approach for numerical optimal control of nonsmooth differential equations with state jumps. A. Nurkanovic, T. Sartor, S. Albrecht, and M. Diehl, IEEE Cont. Sys. Lett., 2021.
- Design, implementation and simulation of an MPC algorithm for switched nonlinear systems under combinatorial constraints. A. Bürger, C. Zeile, A. Altmann-Dieses, S. Sager, M. Diehl. Journal of Process Control, 2019.
- A Gauss-Newton-based Decomposition Algorithm for Nonlinear Mixed-Integer Optimal Control Problems. A. Bürger, C. Zeile, A. Altmann-Dieses, S. Sager, M. Diehl., submitted to CDC, 2021.

References 2 (external, most related to this talk)

- A. F. Filippov, Differential equations with discontinuous right-hand side, Matematicheskii sbornik, 1960.
- B. Baumrucker and L. Biegler, MPEC strategies for optimization of a class of hybrid dynamic systems, Journal of Process Control, 2009.
- D. E. Stewart, A high accuracy method for solving ODEs with discontinuous right-hand side, Numerische Mathematik, 1990.
- A. Wächter and L. T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Mathematical Programming, 2006.
- S. Sager, M. Jung, C. Kirches. Combinatorial Integral Approximation, Mathematical Methods of Operations Research, 2011.

References 3 (own, related to this talk, software)

- pycombina: An Open-Source Tool for Solving Combinatorial Approximation Problems Arising in Mixed-Integer Optimal Control. A. Bürger, C. Zeile, M. Hahn, A. Altmann-Dieses, S. Sager, M. Diehl Proceedings of the IFAC World Congress, 2020).
- CasADi: a software framework for nonlinear optimization and optimal control. J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, Mathematical Programming Computation, 2019.
- qpOASES: A parametric active-set algorithm for quadratic programming, Mathematical Programming Computation. H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, vol. 6, no. 4, pp. 327363, 2014.
- acados: a modular open-source framework for fast embedded optimal control. R. Verschueren, G. Frison, D. Kouzoupis, J. Frey, N. van Duijkeren, A. Zanelli, B. Novoselnik, T. Albin, R. Quirynen, and M. Diehl, Math. Prog. Comp. con. acc., 2022.

References 4 (own, related to this talk)

- Limits of MPCC Formulations in Direct Optimal Control with Nonsmooth Differential Equations.
  A. Nurkanovic, S. Albrecht, and M. Diehl, European Control Conference (ECC), 2020.
- Continuous Control Set Nonlinear Model Predictive Control of Reluctance Synchronous Machines. A. Zanelli, J. Kullick, H. Eldeeb, G. Frison, C. Hackl, and M. Diehl, IEEE Trans. Cont. Sys. Tech., 2021.
- Experimental operation of a solar-driven climate system with thermal energy storages using mixed-integer nonlinear model predictive control.
  A. Bürger, D. Bull, P. Sawant, M. Bohlayer, A. Klotz, D. Beschtz, A. Altmann-Dieses, M. Braun, M. Diehl, Optimal Control Applications and Methods, 2021.
- A whole-year simulation study on nonlinear mixed-integer model predictive control for a thermal energy supply system with multi-use components. A. Bürger, M. Bohlayer, S. Hoffmann, A. Altmann-Dieses, M. Braun, M. Diehl, Applied Energy, 2020.

References 5 (external, related to this talk)

- D. E. Stewart and M. Anitescu, Optimal control of systems with discontinuous differential equations, Numerische Mathematik, 2010.
- M. Ferris and F. Tin-Loi, On the solution of a minimum weight elastoplastic problem involving displacement and complementarity constraints, Computer Methods in Applied Mechanics and Engineering, 1999.
- D. Ralph and S. J. Wright, Some properties of regularization and penalization schemes for MPECs, Optimization Methods and Software, 2004.
- S. Sager. Reformulations and algorithms for the optimization of switching decisions in nonlinear optimal control, Journal of Process Control, 2009.
- A. Lekic, B. Hermans, N. Jovicic, P. Patrinos, Microsecond nonlinear model predictive control for DC-DC converters, Int J Circ Theor Appl., 2020.
- T. Hausberger, A. Kugi, A. Eder, W. Kemmetmüller, High-speed nonlinear model predictive control of an interleaved switching DC/DC-converter, Cont. Eng. Pract., 2020.

Thank you very much for your attention!

Ďakujem!