Lecture 6: Finite Elements with Switch Detection for Filippov Systems

Moritz Diehl and Armin Nurkanović

Systems Control and Optimization Laboratory (syscop)
Summer School on Direct Methods for Optimal Control of Nonsmooth Systems
September 11-15, 2023

universität freiburg

Outline of the lecture

How to discretize optimal control problems subject to Filippov systems?

In direct optimal control, we first discretize, and then solve a finite-dimensional nonlinear program.

Original optimal control problem in continuous time

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

Assume smooth (convex) L, E, h, r
Nonsmooth dynamics make problem nonconvex.

How to discretize optimal control problems subject to Filippov systems?

In direct optimal control, we first discretize, and then solve a finite-dimensional nonlinear program.

Optimal control problem with Stewart's formulation

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

Assume smooth (convex) L, E, h, r
Nonsmooth dynamics make problem nonconvex.

How to discretize optimal control problems subject to Filippov systems?

In direct optimal control, we first discretize, and then solve a finite-dimensional nonlinear program.

Optimal control problem with Stewart's formulation

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

Assume smooth (convex) L, E, h, r
Nonsmooth dynamics make problem nonconvex.

Goal: discretized optimal control problem (an NLP)

$$\begin{aligned}
\min_{s,z,u} \quad & \sum_{k=0}^{N-1} \Phi_L(s_k,z_k,u_k) + E(s_N) \\
\text{s.t.} \quad & s_0 = \bar{x}_0 \\
& s_{k+1} = \Phi_f(s_k,z_k,u_k) \\
& 0 = \Phi_{\mathrm{int}}(s_k,z_k,u_k) \\
& 0 \geq h(s_k,u_k),\ k = 0,\ldots,N-1 \\
& 0 \geq r(s_N)
\end{aligned}$$

Variables $s = (s_0,\ldots)$, $z = (z_0,\ldots)$ and $u = (u_0,\ldots,u_{N-1})$
Nonsmooth Φ_{int}

What happens if we use time stepping methods in direct optimal control?

Direct optimal control with a time stepping IRK discretization

Tutorial example inspired by [Stewart & Anitescu, 2010]

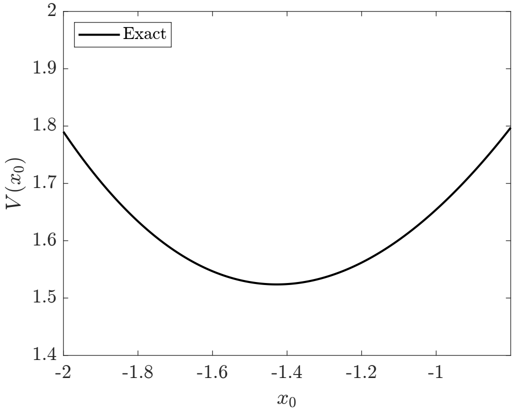
Continuous-time OCP

$$\min_{x(\cdot)\in\mathcal{C}^0([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$
$$\text{s.t.} \quad \dot{x}(t) = 2 - \mathrm{sign}(x(t)), \quad t \in [0,2]$$

Free initial value $x(0)$ is the effective degree of freedom.

Equivalent reduced problem

$$\min_{x_0\in\mathbb{R}} V(x_0)$$



- Denote by $V(x_0)$ the nonsmooth objective value for the unique feasible trajectory starting at $x(0) = x_0$.

Direct optimal control with a time stepping IRK discretization

Tutorial example inspired by [Stewart & Anitescu, 2010]

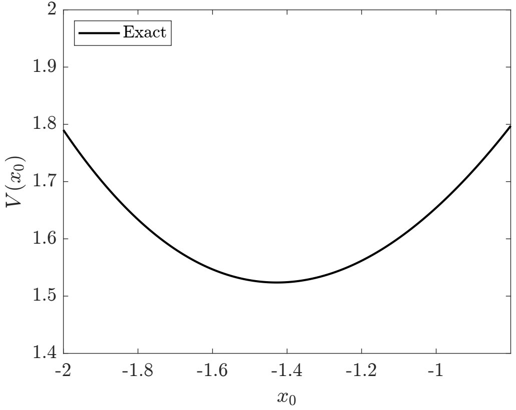
Continuous-time OCP

$$\min_{x(\cdot),\lambda(\cdot),s(\cdot)} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$
$$\text{s.t.} \quad \dot{x}(t) = 2 - s(t)$$
$$0 \leq \lambda(t) - x(t) \perp 1 + s(t) \geq 0$$
$$0 \leq \lambda(t) \perp 1 - s(t) \geq 0, \ t \in [0, 2]$$

Free initial value $x(0)$ is the effective degree of freedom.

Equivalent reduced problem

$$\min_{x_0 \in \mathbb{R}} V(x_0)$$



- Denote by $V(x_0)$ the nonsmooth objective value for the unique feasible trajectory starting at $x(0) = x_0$.

Direct optimal control with a time stepping IRK discretization

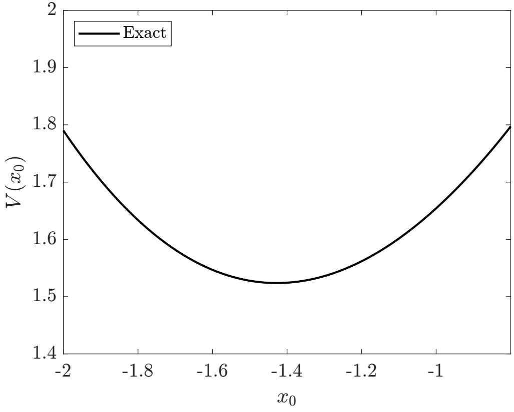
Tutorial example inspired by [Stewart & Anitescu, 2010]

Continuous-time OCP

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

$$\begin{aligned}\text{s.t.} \quad & \dot{x}(t) = 2 - s(t) \\ & 0 \leq \lambda(t) - x(t) \perp 1 + s(t) \geq 0 \\ & 0 \leq \lambda(t) \perp 1 - s(t) \geq 0, \ t \in [0, 2]\end{aligned}$$

- discretize the DCS with fixed step size IRK methods
- e.g., midpoint rule, Gauss-Legendre IRK with $n_{\mathrm{s}} = 1$, accuracy $O(h^2)$



Locally quadratic objective.

Direct optimal control with a time stepping IRK discretization

Tutorial example inspired by [Stewart & Anitescu, 2010]

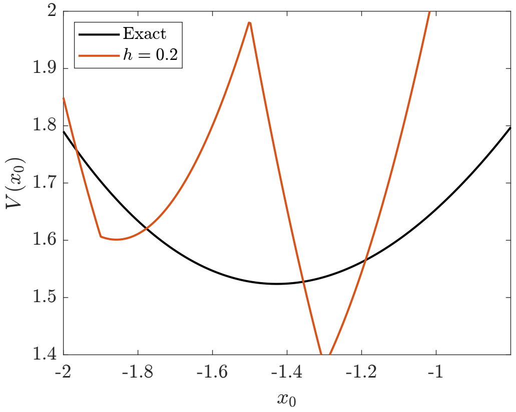
Discrete-time OCP

$$\min_{\mathbf{x},\mathbf{z}} \quad \sum_{n=0}^{N-1} \ell_n(x_n) + (x_N - 5/3)^2$$

$$\text{s.t.} \quad x_{n+1} = \phi_f(x_n, z_n)$$

$$0 = \phi_{\text{int}}(x_n, z_n), \;\; n = 0, \dots N-1$$

- discretize the DCS with fixed step size IRK methods
- e.g., midpoint rule, Gauss-Legendre IRK with $n_{\text{s}} = 1$, accuracy $O(h^2)$
- step size $h = 0.2$, i.e., $N = 10$ integration steps



Many artificial local minima and wrong derivatives.

Direct optimal control with a time stepping IRK discretization

Tutorial example inspired by [Stewart & Anitescu, 2010]

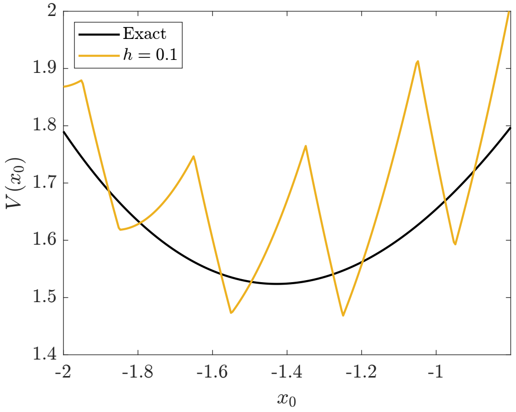
Discrete-time OCP

$$\min_{\mathbf{x},\mathbf{z}} \quad \sum_{n=0}^{N-1} \ell_n(x_n) + (x_N - 5/3)^2$$

$$\text{s.t.} \quad x_{n+1} = \phi_f(x_n, z_n)$$

$$0 = \phi_{\text{int}}(x_n, z_n), \;\; n = 0, \ldots N-1$$

- discretize the DCS with fixed step size IRK methods
- e.g., midpoint rule, Gauss-Legendre IRK with $n_{\text{s}} = 1$, accuracy $O(h^2)$
- step size $h = 0.1$, i.e., $N = 20$ integration steps



Many artificial local minima and wrong derivatives.

Direct optimal control with a time stepping IRK discretization

Tutorial example inspired by [Stewart & Anitescu, 2010]

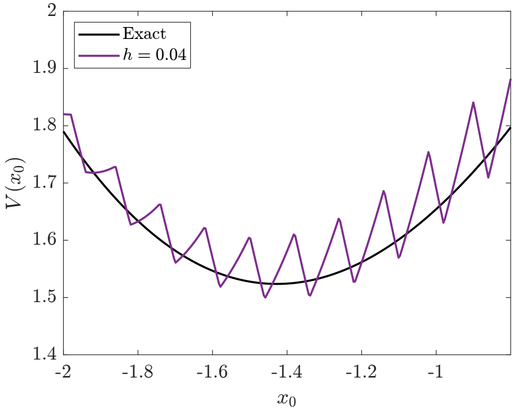
Discrete-time OCP

$$\min_{\mathbf{x},\mathbf{z}} \quad \sum_{n=0}^{N-1} \ell_n(x_n) + (x_N - 5/3)^2$$

$$\text{s.t.} \quad x_{n+1} = \phi_f(x_n, z_n)$$

$$0 = \phi_{\text{int}}(x_n, z_n), \; n = 0, \dots N-1$$

- discretize the DCS with fixed step size IRK methods
- e.g., midpoint rule, Gauss-Legendre IRK with $n_{\text{s}} = 1$, accuracy $O(h^2)$
- step size $h = 0.04$, i.e., $N = 50$ integration steps



Many artificial local minima and wrong derivatives.

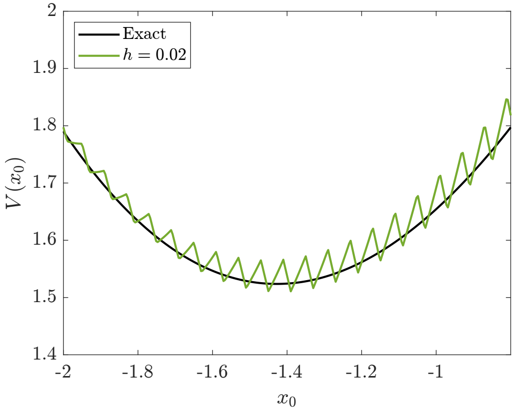
Discrete-time OCP

$$\min_{\mathbf{x},\mathbf{z}} \quad \sum_{n=0}^{N-1} \ell_n(x_n) + (x_N - 5/3)^2$$

$$\text{s.t.} \quad x_{n+1} = \phi_f(x_n, z_n)$$

$$0 = \phi_{\text{int}}(x_n, z_n), \; n = 0, \dots N-1$$

- discretize the DCS with fixed step size IRK methods
- e.g., midpoint rule, Gauss-Legendre IRK with $n_{\mathrm{s}} = 1$, accuracy $O(h^2)$
- step size $h = 0.02$, i.e., $N = 100$ integration steps



Many artificial local minima and wrong derivatives.

Direct optimal control with a time stepping IRK discretization

Tutorial example inspired by [Stewart & Anitescu, 2010]

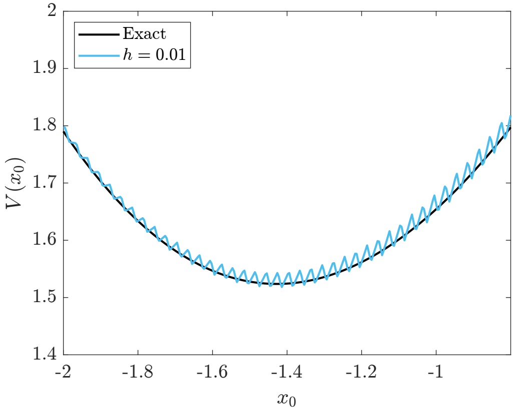
Discrete-time OCP

$$\min_{\mathbf{x},\mathbf{z}} \quad \sum_{n=0}^{N-1} \ell_n(x_n) + (x_N - 5/3)^2$$

$$\text{s.t.} \quad x_{n+1} = \phi_f(x_n, z_n)$$

$$0 = \phi_{\text{int}}(x_n, z_n), \;\; n = 0, \ldots N-1$$

- discretize the DCS with fixed step size IRK methods
- e.g., midpoint rule, Gauss-Legendre IRK with $n_\mathrm{s} = 1$, accuracy $O(h^2)$
- step size $h = 0.01$, i.e., $N = 200$ integration steps



Many artificial local minima and wrong derivatives.

Direct optimal control with a time stepping IRK discretization

Tutorial example inspired by [Stewart & Anitescu, 2010]

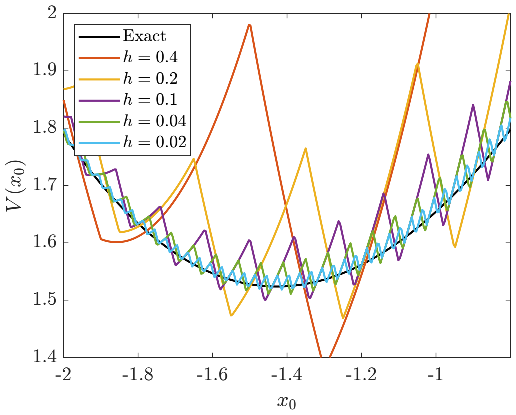
Discrete-time OCP

$$\min_{\mathbf{x},\mathbf{z}} \quad \sum_{n=0}^{N-1} \ell_n(x_n) + (x_N - 5/3)^2$$

$$\text{s.t.} \quad x_{n+1} = \phi_f(x_n, z_n)$$

$$0 = \phi_{\text{int}}(x_n, z_n), \;\; n = 0, \dots N-1$$

- discretize the DCS with fixed step size IRK methods
- e.g., midpoint rule, Gauss-Legendre IRK with $n_{\text{s}} = 1$, accuracy $O(h^2)$
- decreasing the step size might worsen the situation



Many artificial local minima and wrong derivatives.

What happens if we use smoothed models in direct optimal control?

Direct optimal control with a standard IRK discretization - smoothing

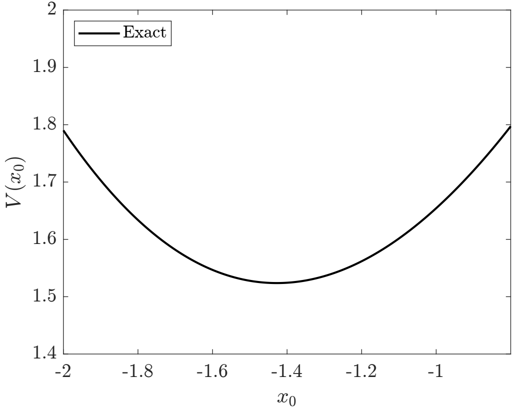
Tutorial example inspired by [Stewart & Anitescu, 2010]

Continuous-time OCP

$$\min_{x(\cdot)\in\mathcal{C}^0([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2)-5/3)^2$$

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

- midpoint rule, with $h = 0.05;\ N = 40$



Direct optimal control with a standard IRK discretization - smoothing

Tutorial example inspired by [Stewart & Anitescu, 2010]

Smoothed continuous-time OCP

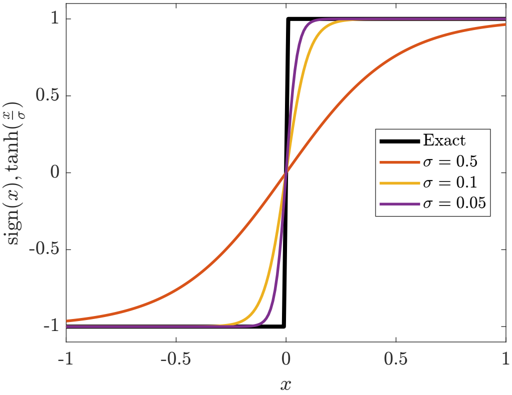
$$\min_{x(\cdot)\in\mathcal{C}^{\infty}([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \tanh\left(\frac{x(t)}{\sigma}\right), \quad t \in [0,2]$$

Equivalent reduced problem

$$\min_{x_0\in\mathbb{R}} V_\sigma(x_0)$$

- midpoint rule, with $h = 0.05$; $N = 40$
- solve smoothed OCP for different σ



Direct optimal control with a standard IRK discretization - smoothing

Tutorial example inspired by [Stewart & Anitescu, 2010]

Smoothed continuous-time OCP

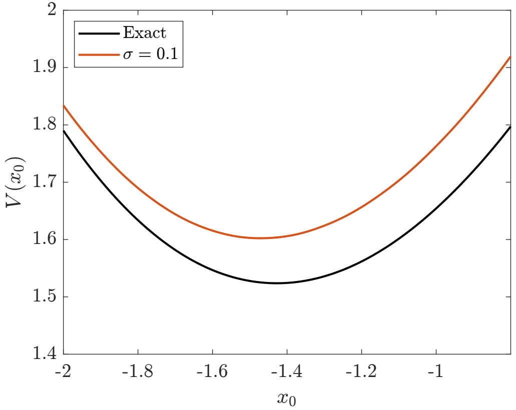
$$\min_{x(\cdot)\in\mathcal{C}^\infty([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \tanh\left(\frac{x(t)}{\sigma}\right), \quad t \in [0, 2]$$

Equivalent reduced problem

$$\min_{x_0 \in \mathbb{R}} V_\sigma(x_0)$$

- midpoint rule, with $h = 0.05;\ N = 40$
- solve smoothed OCP with $\sigma = 0.1$



Direct optimal control with a standard IRK discretization - smoothing

Tutorial example inspired by [Stewart & Anitescu, 2010]

Smoothed continuous-time OCP

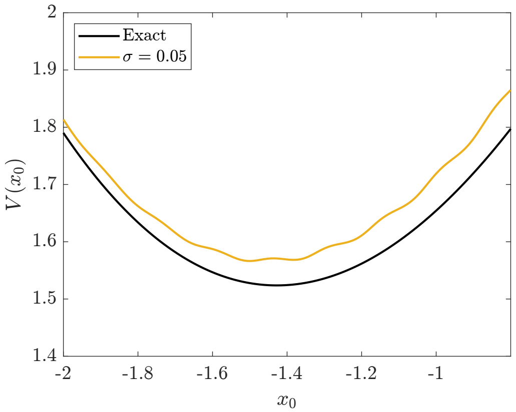
$$\min_{x(\cdot)\in\mathcal{C}^\infty([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \tanh\left(\frac{x(t)}{\sigma}\right), \quad t \in [0,2]$$

Equivalent reduced problem

$$\min_{x_0\in\mathbb{R}} V_\sigma(x_0)$$

- midpoint rule, with $h = 0.05;\ N = 40$
- solve smoothed OCP with $\sigma = 0.05$



Direct optimal control with a standard IRK discretization - smoothing

Tutorial example inspired by [Stewart & Anitescu, 2010]

Smoothed continuous-time OCP

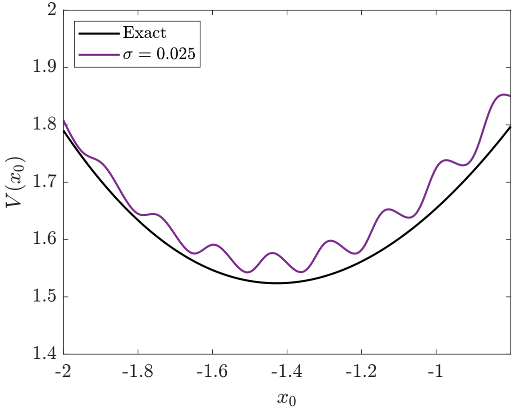
$$\min_{x(\cdot)\in\mathcal{C}^\infty([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \tanh\left(\frac{x(t)}{\sigma}\right), \quad t \in [0,2]$$

Equivalent reduced problem

$$\min_{x_0 \in \mathbb{R}} V_\sigma(x_0)$$

- midpoint rule, with $h = 0.05;\ N = 40$
- solve smoothed OCP with $\sigma = 0.025$



Direct optimal control with a standard IRK discretization - smoothing

Tutorial example inspired by [Stewart & Anitescu, 2010]

Smoothed continuous-time OCP

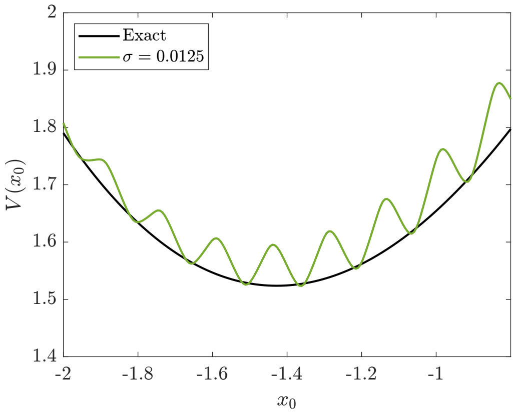
$$\min_{x(\cdot)\in\mathcal{C}^\infty([0,2])} \int_0^2 x(t)^2\mathrm{d}t + (x(2)-5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \tanh\left(\frac{x(t)}{\sigma}\right), \quad t \in [0,2]$$

Equivalent reduced problem

$$\min_{x_0\in\mathbb{R}} V_\sigma(x_0)$$

- midpoint rule, with $h = 0.05$; $N = 40$
- solve smoothed OCP with $\sigma = 0.0125$



Direct optimal control with a standard IRK discretization - smoothing

Tutorial example inspired by [Stewart & Anitescu, 2010]

Smoothed continuous-time OCP

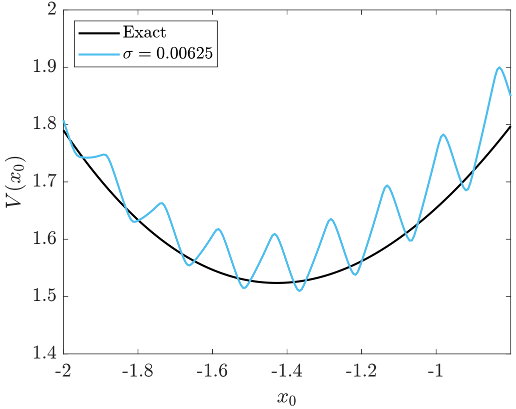
$$\min_{x(\cdot)\in\mathcal{C}^\infty([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \tanh\left(\frac{x(t)}{\sigma}\right), \quad t \in [0,2]$$

Equivalent reduced problem

$$\min_{x_0\in\mathbb{R}} V_\sigma(x_0)$$

- midpoint rule, with $h = 0.05;\ N = 40$
- solve smoothed OCP with $\sigma = 0.00625$



Direct optimal control with a standard IRK discretization - smoothing

Tutorial example inspired by [Stewart & Anitescu, 2010]

Smoothed continuous-time OCP

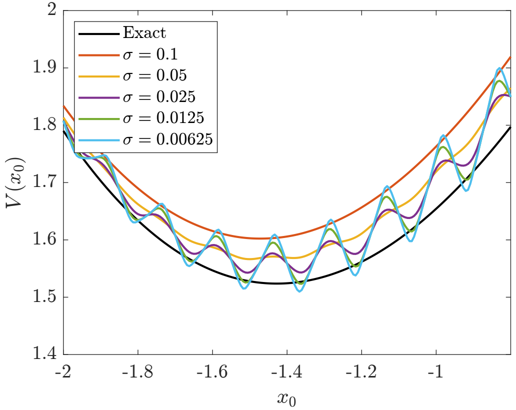
$$\min_{x(\cdot)\in\mathcal{C}^\infty([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \tanh\left(\frac{x(t)}{\sigma}\right), \quad t \in [0,2]$$

Equivalent reduced problem

$$\min_{x_0\in\mathbb{R}} V_\sigma(x_0)$$

- midpoint rule, with $h = 0.05;\ N = 40$



If $h \gg \sigma$, then the smooth approximation behaves **the same as the nonsmooth problem!**

Direct optimal control with a standard IRK discretization - smoothing

Tutorial example inspired by [Stewart & Anitescu, 2010]

Smoothed continuous-time OCP

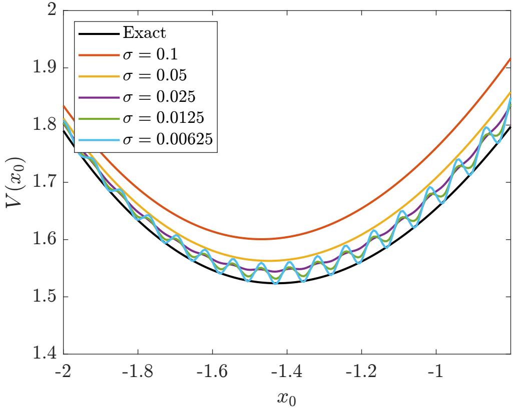
$$\min_{x(\cdot)\in\mathcal{C}^\infty([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2)-5/3)^2$$

$$\text{s.t.} \quad \dot{x}(t) = 2 - \tanh\left(\frac{x(t)}{\sigma}\right), \quad t \in [0,2]$$

Equivalent reduced problem

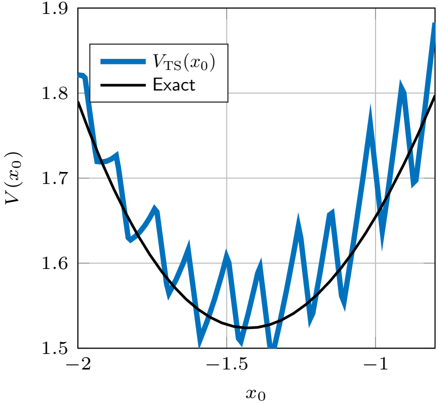
$$\min_{x_0\in\mathbb{R}} V_\sigma(x_0)$$

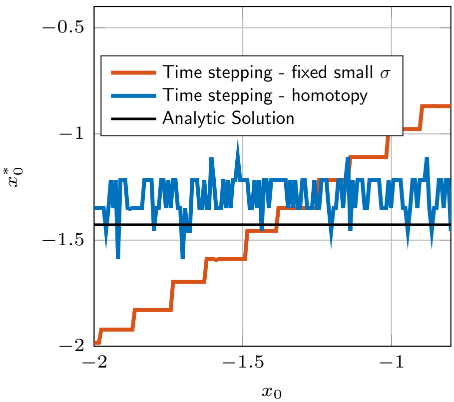
- midpoint rule, with $h = 0.025$; $N = 80$



If $h \gg \sigma$, then the smooth approximation behaves **the same as the nonsmooth problem!**

Direct optimal control with a standard time-stepping IRK discretization





- spurious local minima, optimizer gets trapped close to initialization
- sensitivity only correct if step sizes are smaller than smoothing parameter [Stewart & Anitescu, 2010]: homotopy improves convergence
- even for the best local minimizer, only $O(h)$ accuracy can be expected

Outline

Main ideas of FESD

Based on [Baumrucker & Biegler, 2009], [N. et. al, 2022, 2022a, 2023]

FESD overview

1. Transform Filippov DI into equivalent DCS - Stewart or Heaviside step (Lecture 5)

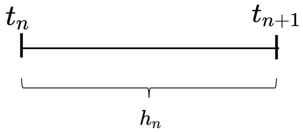
$$\dot{x} \in F_{\mathrm{F}}(x,u) \quad \iff \quad \begin{aligned} \dot{x} &= F(x,u)\theta \\ 0 &= G_{\mathrm{DCS}}(x,z,\theta) \end{aligned}$$

Main ideas of FESD

Based on [Baumrucker & Biegler, 2009], [N. et. al, 2022, 2022a, 2023]

FESD overview

1. Transform Filippov DI into equivalent DCS - Stewart or Heaviside step (Lecture 5)
2. Consider at least two integration intervals = finite elements

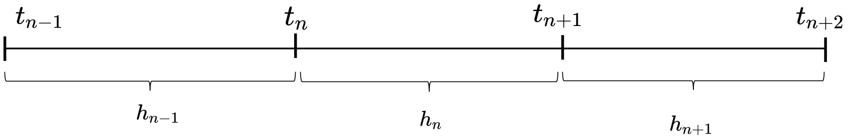


Main ideas of FESD

Based on [Baumrucker & Biegler, 2009], [N. et. al, 2022, 2022a, 2023]

FESD overview

1. Transform Filippov DI into equivalent DCS - Stewart or Heaviside step (Lecture 5)
2. Consider at least two integration intervals = finite elements

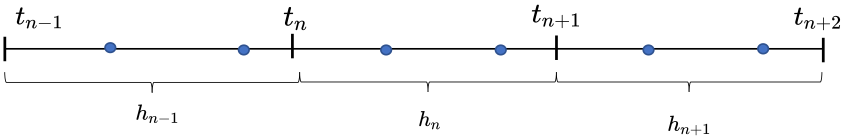


Main ideas of FESD

Based on [Baumrucker & Biegler, 2009], [N. et. al, 2022, 2022a, 2023]

FESD overview

1. Transform Filippov DI into equivalent DCS - Stewart or Heaviside step (Lecture 5)
2. Consider at least two integration intervals = finite elements
3. Use general implicit Runge-Kutta methods (Lectures 2 and 3)

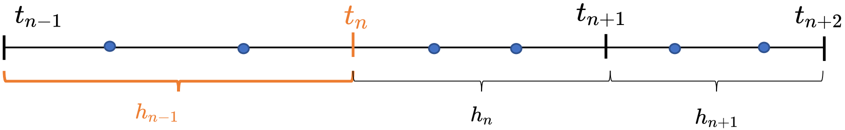


Main ideas of FESD

Based on [Baumrucker & Biegler, 2009], [N. et. al, 2022, 2022a, 2023]

FESD overview

1. Transform Filippov DI into equivalent DCS - Stewart or Heaviside step (Lecture 5)
2. Consider at least two integration intervals = finite elements
3. Use general implicit Runge-Kutta methods (Lectures 2 and 3)
4. Let step sizes h_n be degrees of freedom (under-determined system)

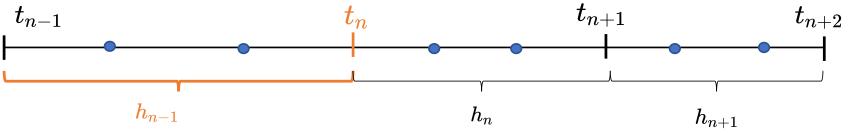


Main ideas of FESD

Based on [Baumrucker & Biegler, 2009], [N. et. al, 2022, 2022a, 2023]

FESD overview

1. Transform Filippov DI into equivalent DCS - Stewart or Heaviside step (Lecture 5)
2. Consider at least two integration intervals = finite elements
3. Use general implicit Runge-Kutta methods (Lectures 2 and 3)
4. Let step sizes h_n be degrees of freedom
5. Cross complementarity conditions - adapt h_n for switch detection

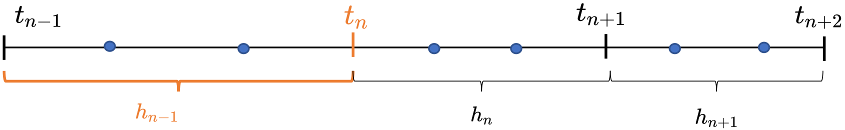


Main ideas of FESD

Based on [Baumrucker & Biegler, 2009], [N. et. al, 2022, 2022a, 2023]

FESD overview

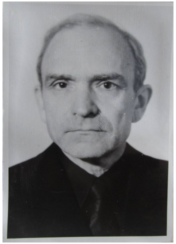
1. Transform Filippov DI into equivalent DCS - Stewart or Heaviside step (Lecture 5)
2. Consider at least two integration intervals = finite elements
3. Use general implicit Runge-Kutta methods (Lectures 2 and 3)
4. Let step sizes h_n be degrees of freedom
5. Cross complementarity conditions - adapt h_n for switch detection
6. Step equilibration - remove degrees of freedom if no switch



Switched ODE not 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

$$\dot{x} \in F_{\mathrm{F}}(x,u) := \Big\{ \sum_{i=1}^{n_f} f_i(x,u)\,\theta_i \;\Big|\; \sum_{i=1}^{n_f} \theta_i = 1, \quad \theta_i \geq 0, \quad i = 1,\dots n_f, \quad \theta_i = 0, \quad \text{if } x \notin \overline{R_i} \Big\}$$



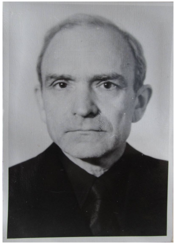
Aleksei F. Filippov (1923-2006)
image source: wikipedia

Recap on Filippov Convexification

Switched ODE not 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

$$\dot{x} \in F_{\mathrm{F}}(x,u) := \Big\{ \sum_{i=1}^{n_f} f_i(x,u)\,\theta_i \;\Big|\; \sum_{i=1}^{n_f} \theta_i = 1, \quad \theta_i \geq 0, \quad i = 1,\dots n_f, \quad \theta_i = 0, \quad \text{if } x \notin \overline{R_i} \Big\}$$



Aleksei F. Filippov
(1923-2006)
image source: wikipedia

- 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

From Filippov to dynamic complementarity systems

Using the KKT conditions of the parametric LP

LP representation

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

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

where

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

From Filippov to dynamic complementarity systems

Using the KKT conditions of the parametric LP

LP representation

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

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

where

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

Express equivalently by optimality conditions:

Dynamic Complementarity System (DCS)

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

Compact notation

$$\dot{x} = F(x,u)\,\theta$$
$$0 = G_{\mathrm{LP}}(x,\theta,\lambda,\mu),$$

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

Conventional discretization by Implicit Runge Kutta (IRK) method

Continuous time DCS

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

Conventional discretization by Implicit Runge Kutta (IRK) method

Continuous time DCS

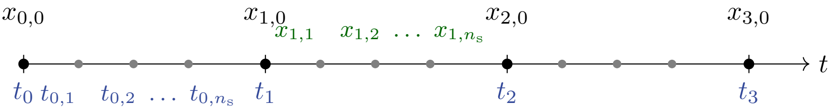
$$
\begin{aligned}
x(0) &= \bar{x}_0, \\
\dot{x}(t) &= v(t) \\
v(t) &= F(x(t), u(t))\,\theta(t) \\
0 &= g(x(t)) - \lambda(t) - e\mu(t) \\
0 &\leq \theta(t) \perp \lambda(t) \geq 0 \\
1 &= e^\top \theta(t), \quad t \in [0, T]
\end{aligned}
$$

Discrete time IRK-DCS equation

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

Notation: $x_{n,i} \in \mathbb{R}^{n_x}, \theta_{n,i} \in \mathbb{R}^m$ etc. RK stage values with:

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



Conventional time stepping - illustrative example

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

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

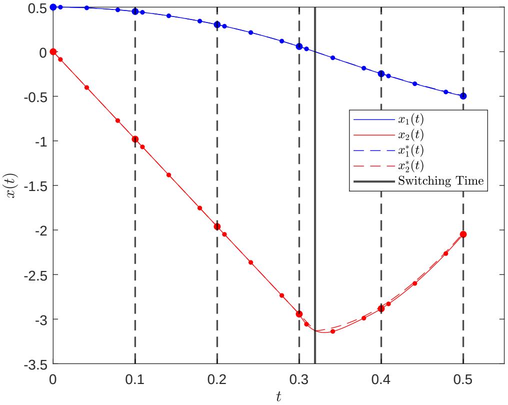
$$f_1(x) = \begin{pmatrix} x_2 \\ -a \end{pmatrix}, \; f_2(x) = \begin{pmatrix} x_2 \\ -kx_1 - cx_2 \end{pmatrix}$$

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

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

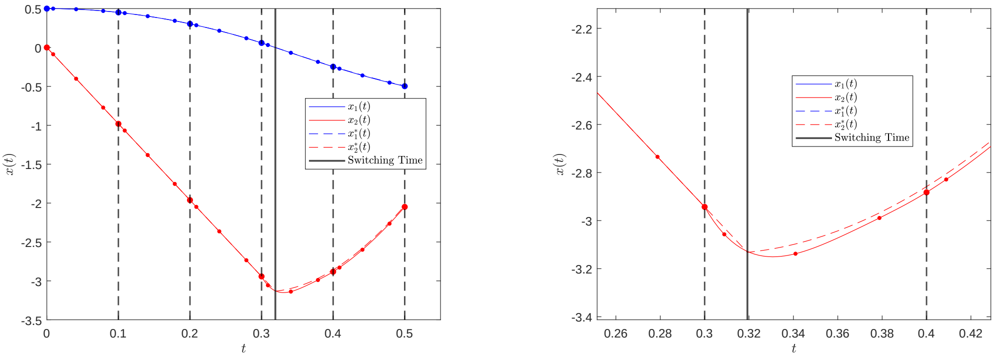
Solve with IRK Radau IIA method of **order 7**

$s = 4, \; N = 5, \; T = 0.5, \; h = 0.1$



Conventional time stepping - illustrative example

Zoom in



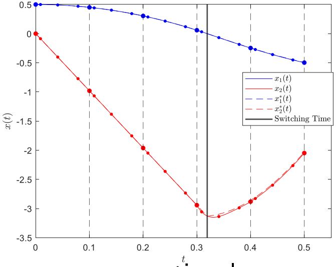
High integration accuracy of 7th order IRK method is lost in fourth time step.
Reason: we try to approximate a nonsmooth function by a (smooth) polynomial.

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

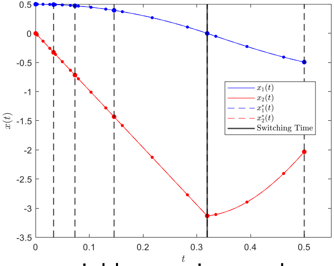
Finite Elements with Switch Detection (FESD)

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

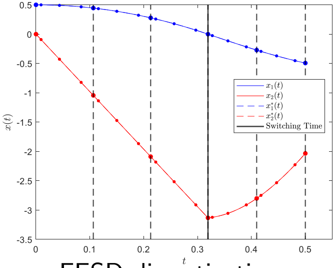
- make stepsizes h_n free, ensure $\sum_{n=0}^{N-1} h_n = T$ [cf. Baumrucker & Biegler, 2009]
- allow switches only at element boundaries, enforce via *cross-complementarities*
- remove spurious degrees of freedom via *step equilibration*



conventional discretization



variable stepsizes and cross-complementarities



FESD discretization with step equilibration

Conventional DCS and FESD discretization without step equilibration

Time-stepping discretization

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

for $i = 1, \dots, n_\mathrm{s}$

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

FESD discretization without step equilibration

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

for $i = 1, \dots, n_\mathrm{s}$ and $n = 0, \dots, N-1$

and $i' = 0, 1, \dots, n_\mathrm{s}$

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

Conventional DCS and FESD discretization with step equilibration

Time-stepping discretization

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

for $i = 1, \dots, n_s$

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

FESD discretization with step equilibration

$$
\begin{aligned}
x_{0,0} &= \bar{x}_0, \quad \sum_{n=0}^{N-1} h_n = T \\
x_{n+1,0} &= x_{n,0} + h_n \sum_{i=1}^{n_s} b_i v_{n,i} \\
x_{n,i} &= x_{n,0} + h_n \sum_{j=1}^{n_s} a_{i,j} v_{n,j} \\
v_{n,i} &= F(x_{n,i}, u_{n,i})\, \theta_{n,i} \\
0 &= g(x_{n,i'}) - \lambda_{n,ii'} - e\mu_{n,i'} \\
0 &\leq \theta_{n,i} \perp \lambda_{n,i'} \geq 0 \quad \text{(cross-complementarities)} \\
1 &= e^\top \theta_{n,i} \\
0 &= \nu(\theta_{n'}, \theta_{n'+1}, \lambda_{n'}, \lambda_{n'+1}) \cdot (h_{n'} - h_{n'+1})
\end{aligned}
$$

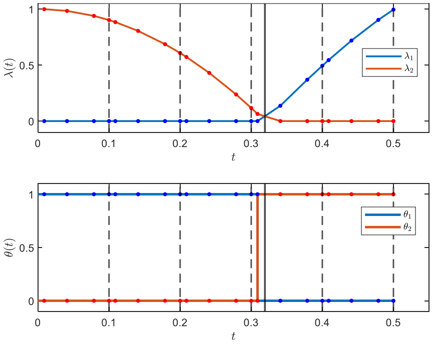
for $i = 1, \dots, n_s$ and $n = 0, \dots, N-1$

and $i' = 0, 1, \dots, n_s$ and $n' = 0, \dots, N-2$

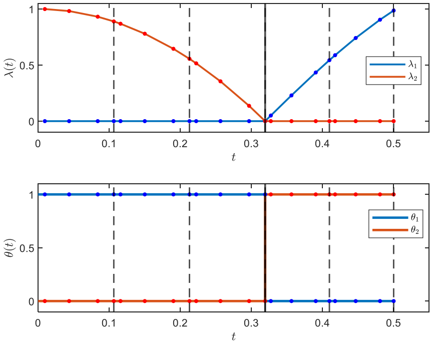
- N extra variables $(h_0, \dots, h_{N-1})$ restricted by one extra equality
- Additional multipliers $\lambda_{n,0}, \mu_{n,0}$ are uniquely determined
- Indicator function $\nu(\theta_{n'}, \theta_{n'+1}, \lambda_{k'}, \lambda_{k'+1})$ only zero if a switch occurs

Multipliers in conventional and FESD discretization

Time stepping discretization:



FESD discretization:



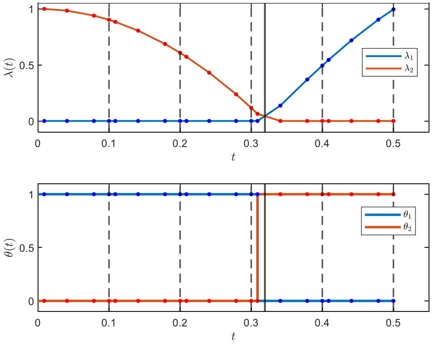
Lemma (Cross complementarity)

If any $\theta_{n,j,i}$ with $j = 1, \ldots, n_\mathrm{s}$ is positive, then all $\lambda_{n,j',i}$ with $j' = 0, \ldots, n_\mathrm{s}$ must be zero. Conversely, if any $\lambda_{n,j',i}$ is positive, then all $\theta_{n,j,i}$ are zero.

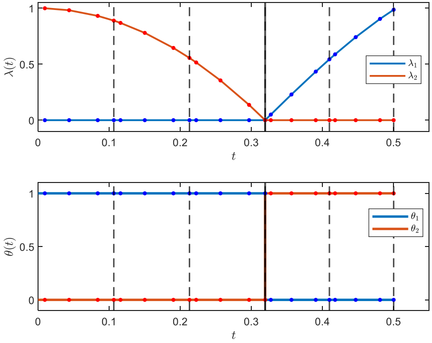
Notation $\lambda_{n,j,i}$ - n - finite element, j - RK stage, i - component of vector

Multipliers in conventional and FESD discretization

Time stepping discretization:



FESD discretization:



FESD's cross-complementarities exploit the fact that the multiplier $\lambda_i(t)$ is continuous in time.
On boundary, $\lambda_i(t_n)$ must be zero if $\theta_i(t) > 0$ for any $t \in [t_{n-1}, t_{n+1}]$ on the adjacent intervals.
This implicitly imposes the constraint $g_i(x_n) - g_j(x_n) = 0$.
$\implies$ h_n adapts for exact switch detection

Step equilibration

- if no switches happen, cross complementarity implied by standard complementarity
- spurious degrees of freedom in h_n: more degrees of freedom than equations

Step equilibration

- if no switches happen, cross complementarity implied by standard complementarity
- spurious degrees of freedom in h_n: more degrees of freedom than equations
- exploit complementarity of θ_n, λ_n to encode switching logic
- define (very complicated) switch indicator function ν (cf. PhD Nurkanović):

$$\nu(\theta_n, \theta_{n+1}, \lambda_n, \lambda_{n+1}) := \begin{cases} \text{positive}, & \text{if no switch at } t_{n+1} \\ 0, & \text{if switch at } t_{n+1} \end{cases}$$

Step equilibration

- if no switches happen, cross complementarity implied by standard complementarity
- spurious degrees of freedom in h_n: more degrees of freedom than equations
- exploit complementarity of θ_n, λ_n to encode switching logic
- define (very complicated) switch indicator function ν (cf. PhD Nurkanović):

$$\nu(\theta_n, \theta_{n+1}, \lambda_n, \lambda_{n+1}) := \begin{cases} \text{positive}, & \text{if no switch at } t_{n+1} \\ 0, & \text{if switch at } t_{n+1} \end{cases}$$

- step equilibration:

$$0 = \nu(\theta_n, \theta_{n+1}, \lambda_n, \lambda_{n+1}) \cdot (h_n - h_{n+1}), \quad n = 0, \dots, N-2$$

Step equilibration

- if no switches happen, cross complementarity implied by standard complementarity
- spurious degrees of freedom in h_n: more degrees of freedom than equations
- exploit complementarity of θ_n, λ_n to encode switching logic
- define (very complicated) switch indicator function ν (cf. PhD Nurkanović):

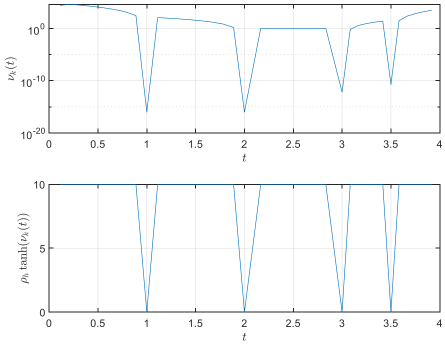
$$\nu(\theta_n, \theta_{n+1}, \lambda_n, \lambda_{n+1}) := \begin{cases} \text{positive}, & \text{if no switch at } t_{n+1} \\ 0, & \text{if switch at } t_{n+1} \end{cases}$$

- step equilibration:

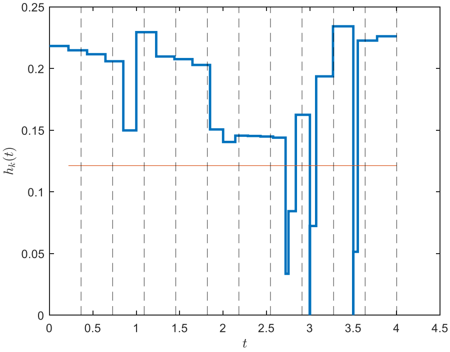
$$0 = \nu(\theta_n, \theta_{n+1}, \lambda_n, \lambda_{n+1}) \cdot (h_n - h_{n+1}), \qquad n = 0, \ldots, N-2$$

- Summary:
  - If switch happens, then h_n is determined by cross complementarity.
  - If no switch happens, then h_n is determined by step equilibration.

Indicator function over time:



Step size over time:

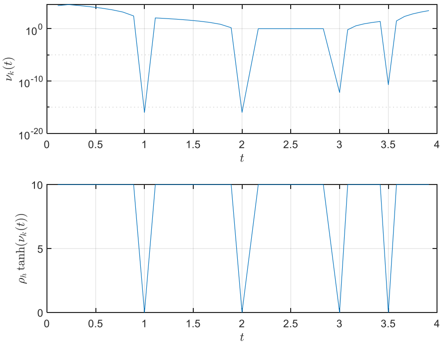


Optimizer varies step size randomly, potentially playing with integration errors.

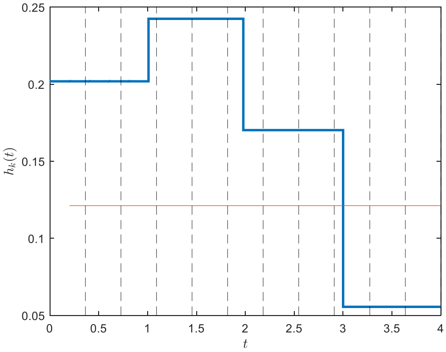
Numerical solution with equilibration

Example with four switches

Indicator function over time:



Step size over time:



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

Summary of theoretical results

1. An FESD problem needs to solve a nonlinear complementarity problem (NCP) to advance the integration. The solutions of these NCP are locally unique.
   - For a given point determine which constraint cross comp. and step eq. are binding, and which implicitly satisfied.
   - Obtain square system and apply implicit function theorem.

2. Convergence of the FESD method to a Filippov solution of the underlying system with accuracy $O(h^p)$ is proven. Here, p is the order of the underlying smooth IRK method.
   - Solution approximation and true solution predict same active set.
   - Switching time accuracy also $O(h^p)$.

Summary of theoretical results

1. An FESD problem needs to solve a nonlinear complementarity problem (NCP) to advance the integration. The solutions of these NCP are locally unique.
   - For a given point determine which constraint cross comp. and step eq. are binding, and which implicitly satisfied.
   - Obtain square system and apply implicit function theorem.

2. Convergence of the FESD method to a Filippov solution of the underlying system with accuracy $O(h^p)$ is proven. Here, p is the order of the underlying smooth IRK method.
   - Solution approximation and true solution predict same active set.
   - Switching time accuracy also $O(h^p)$.

3. Convergence of numerical sensitivities to the true value with $O(h^p)$ is given.
   - Cross. comp. implicitly enforce switching condition and lead to correct sensitivities.
   - The Stewart & Anitescu problem is solved.

Integration order plots for FESD and IRK time stepping

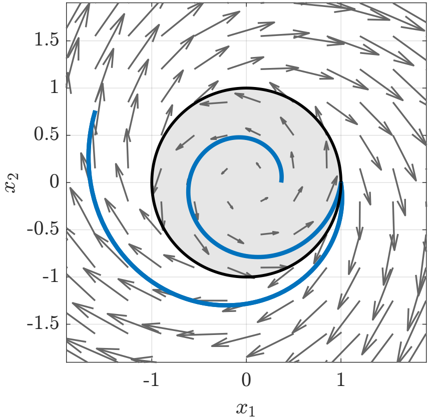
Revisit example from Lecture 4

Tutorial example

$$\dot{x} = \begin{cases} A_1 x, & \|x\|_2^2 < 1, \\ A_2 x, & \|x\|_2^2 > 1, \end{cases}$$

with $A_1 = \begin{bmatrix} 1 & 2\pi \\ -2\pi & 1 \end{bmatrix}$, $A_2 = \begin{bmatrix} 1 & -2\pi \\ 2\pi & 1 \end{bmatrix}$

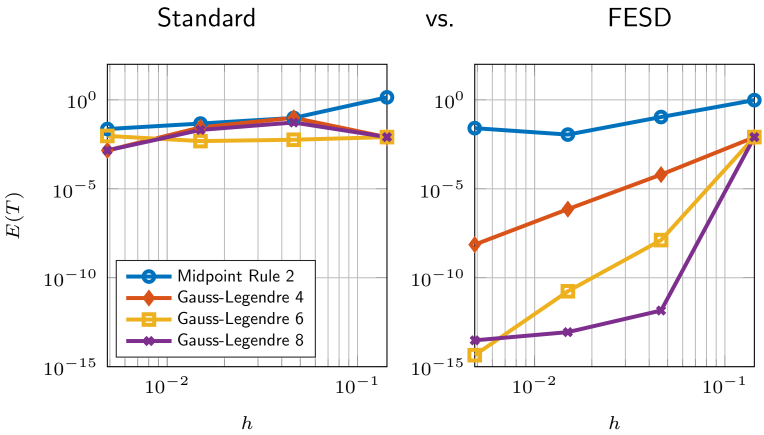
$x(0) = (e^{-1}, 0)$ for $t \in [0, \frac{\pi}{2}]$.



Compute global integration error $E(T)$ using different strategies.
Compute solution approximation:

1. With fixed step size IRK methods (time-stepping).
2. FESD with same underlying IRK methods.

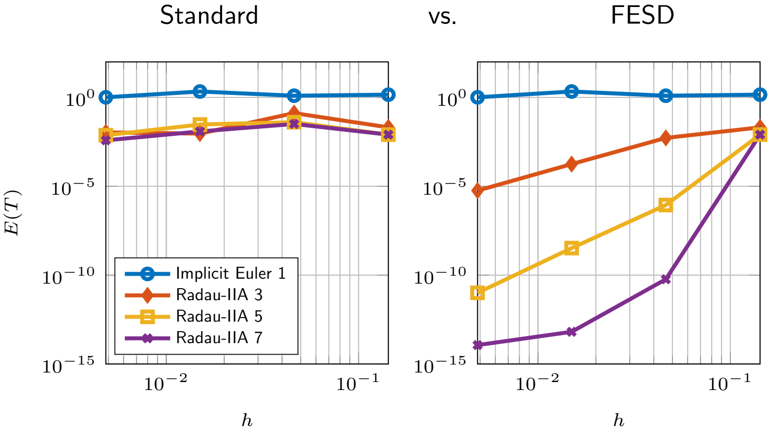
FESD recovers high integration order for switched systems



Integration error $E(T)$ at time $T = \pi/2$ vs. step-size h, for different IRK methods.

FESD discretization recovers high integration order

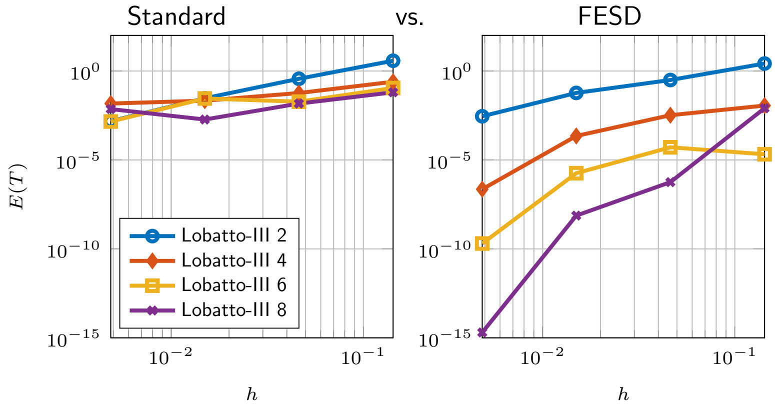
FESD recovers high integration order for switched systems



Integration error $E(T)$ at time $T = \pi/2$ vs. step-size h, for different IRK methods.

FESD discretization recovers high integration order

FESD recovers high integration order for switched systems



Integration error $E(T)$ at time $T = \pi/2$ vs. step-size h, for different IRK methods.
FESD discretization recovers high integration order

Outline

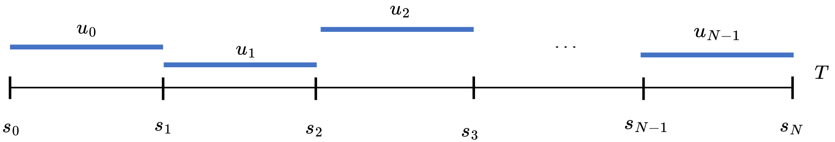
Discretizing optimal control problems with FESD

Discretized optimal control problem

$$\min_{s,z,u} \sum_{k=0}^{N-1} \Phi_L(s_k, z_k, u_k) + E(s_N)$$
$$\text{s.t.} \quad s_0 = \bar{x}_0$$
$$s_{k+1} = \Phi_f(s_k, z_k, u_k)$$
$$0 = \Phi_{\text{int}}(s_k, z_k, u_k)$$
$$0 \geq h(s_k, u_k), \ k = 0, \dots, N-1$$
$$0 \geq r(s_N)$$

- States at control grid points $s = (s_0, \dots, s_N)$
- Piecewise controls $u = (u_0, \dots, u_{N-1})$
- FESD with N_{FE} finite elements applied on every control interval

Control horizon $[0, T]$ with N control stages



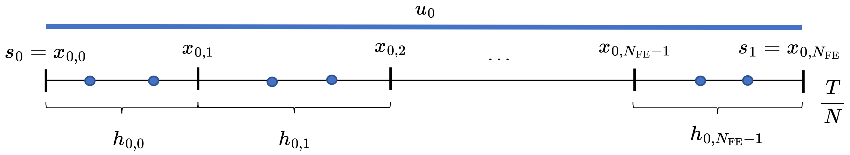
Discretizing optimal control problems with FESD

Discretized optimal control problem

$$
\begin{aligned}
\min_{s,z,u} \quad & \sum_{k=0}^{N-1} \Phi_L(s_k, z_k, u_k) + E(s_N) \\
\text{s.t.} \quad & s_0 = \bar{x}_0 \\
& s_{k+1} = \Phi_f(s_k, z_k, u_k) \\
& 0 = \Phi_{\text{int}}(s_k, z_k, u_k) \\
& 0 \geq h(s_k, u_k), \; k = 0, \dots, N-1 \\
& 0 \geq r(s_N)
\end{aligned}
$$

- States at control grid points $s = (s_0, \dots, s_N)$
- Piecewise controls $u = (u_0, \dots, u_{N-1})$
- FESD with N_{FE} finite elements applied on every control interval
- Φ_{int} summarizes all internal FESD equations: RK, cross complementarity, step equilibration,...

Control horizon $[0, T]$ with N control stages



Discretizing optimal control problems with FESD

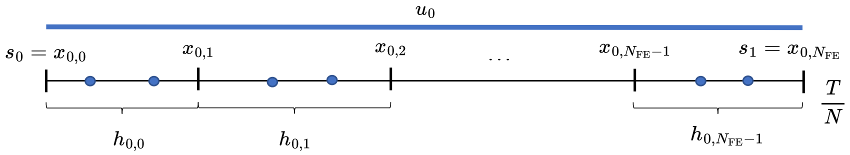
Discretized optimal control problem

$$\min_{s,z,u} \sum_{k=0}^{N-1} \Phi_L(s_k, z_k, u_k) + E(s_N)$$

$$\begin{aligned} \text{s.t.} \quad s_0 &= \bar{x}_0 \\ s_{k+1} &= \Phi_f(s_k, z_k, u_k) \\ 0 &= \Phi_{\text{int}}(s_k, z_k, u_k) \\ 0 &\geq h(s_k, u_k), \; k = 0, \ldots, N-1 \\ 0 &\geq r(s_N) \end{aligned}$$

- States at control grid points $s = (s_0, \ldots, s_N)$
- Piecewise controls $u = (u_0, \ldots, u_{N-1})$
- FESD with N_{FE} finite elements applied on every control interval
- Φ_{int} summarizes all internal FESD equations: RK, cross complementarity, step equilibration,...
- $z = (z_0, \ldots, z_{N-1})$ - all interval variables: internal states, stage values of states and multipliers, step sizes, ...

Control horizon $[0, T]$ with N control stages



FESD-discretized optimal control problems are MPCC

Discretized optimal control problem

$$
\begin{aligned}
\min_{s,z,u} \quad & \sum_{k=0}^{N-1} \Phi_L(s_k, z_k, u_k) + E(s_N) \\
\text{s.t.} \quad & s_0 = \bar{x}_0 \\
& s_{k+1} = \Phi_f(s_k, z_k, u_k) \\
& 0 = \Phi_{\text{int}}(s_k, z_k, u_k) \\
& 0 \geq h(s_k, u_k), \; k = 0, \ldots, N-1 \\
& 0 \geq r(s_N)
\end{aligned}
$$

Collect $w = (s, z, u) \in \mathbb{R}^{n_w}$
Mathematical programs with complementarity constraints (MPCC) are more difficult than standard NLPs

NLP with Complementarity Constraints

$$
\begin{aligned}
\min_{w \in \mathbb{R}^{n_w}} \quad & F(w) \\
\text{s.t.} \quad & 0 = G(w) \\
& 0 \geq H(w) \\
& 0 \leq G_1(w) \perp G_2(w) \geq 0
\end{aligned}
$$

Standard and cross complementarity constraints summarized in

$$
0 \leq G_1(w) \perp G_2(w) \geq 0
$$

Nonlinear Programs (NLP)

Newton-type methods generate a sequence $w_0, w_1, w_2, \ldots$ by linearizing and solving convex subproblems.

Summarized NLP

$$\begin{aligned} \min_{w \in \mathbb{R}^{n_w}} \quad & F(w) \\ \text{s.t.} \quad & 0 = G(w) \\ & 0 \geq H(w) \end{aligned}$$

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

Nonlinear Programs (NLP)

Newton-type methods generate a sequence $w_0, w_1, w_2, \ldots$ by linearizing and solving convex subproblems.

Summarized NLP

$$\begin{aligned} \min_{w \in \mathbb{R}^{n_w}} \quad & F(w) \\ \text{s.t.} \quad & 0 = G(w) \\ & 0 \geq H(w) \end{aligned}$$

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

NLP with complementarity constraints

$$\begin{aligned} \min_{w \in \mathbb{R}^{n_w}} \quad & F(w) \\ \text{s.t.} \quad & 0 = G(w) \\ & 0 \geq H(w) \\ & 0 \leq G_1(w) \perp G_2(w) \geq 0 \end{aligned}$$

There is significant nonconvex and nonsmooth structure in the NLP.

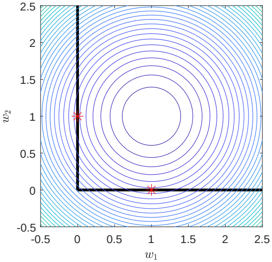
Mathematical Programs with Complementarity Constraints (MPCC)

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

MPCC as an NLP

$$\min_{w \in \mathbb{R}^{n_w}} \; F(w)$$
$$\text{s.t.} \quad 0 = G(w)$$
$$0 \geq H(w)$$
$$0 \leq G_1(w)$$
$$0 \leq G_2(w)$$
$$0 \geq G_1(w)^\top G_2(w)$$

Convex J, H and smooth F. Smooth G_1, G_2.



Due to complementarity constraints, MPCC are nonsmooth and nonconvex.

Toy MPCC example:

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

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

MPCCs treated in detail in three lectures by C. Kirches.

Revisiting the OCP example - now with FESD

Tutorial example inspired by [Stewart & Anitescu, 2010]

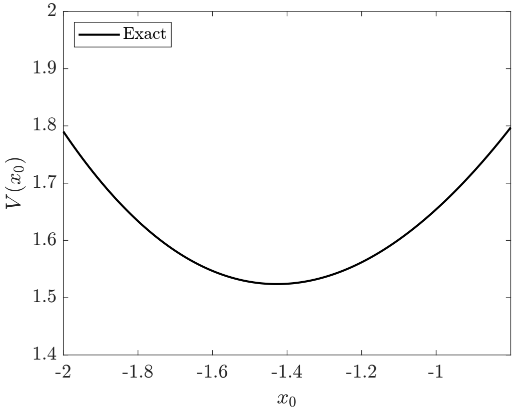
Continuous-time OCP

$$\min_{x(\cdot)\in\mathcal{C}^0([0,2])} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2$$
$$\text{s.t.} \quad \dot{x}(t) = 2 - \mathrm{sign}(x(t)), \quad t \in [0,2]$$

Free initial value $x(0)$ is the effective degree of freedom.

Equivalent reduced problem

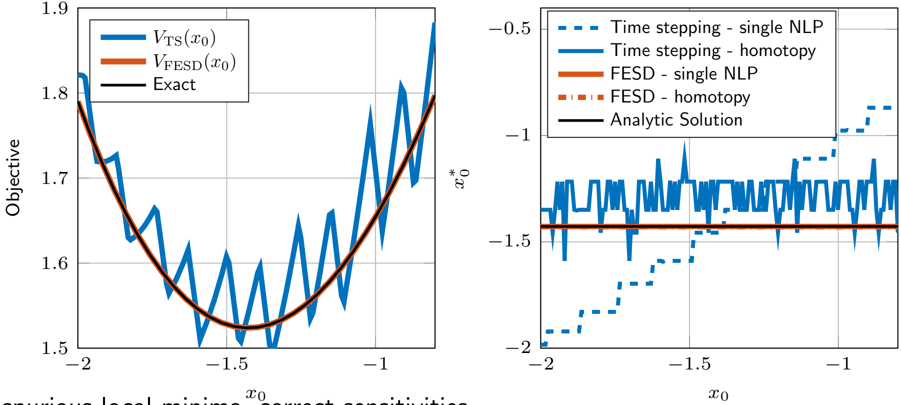
$$\min_{x_0\in\mathbb{R}} V(x_0)$$



- Denote by $V(x_0)$ the nonsmooth objective value for the unique feasible trajectory starting at $x(0) = x_0$.

Revisiting the OCP example - now with FESD

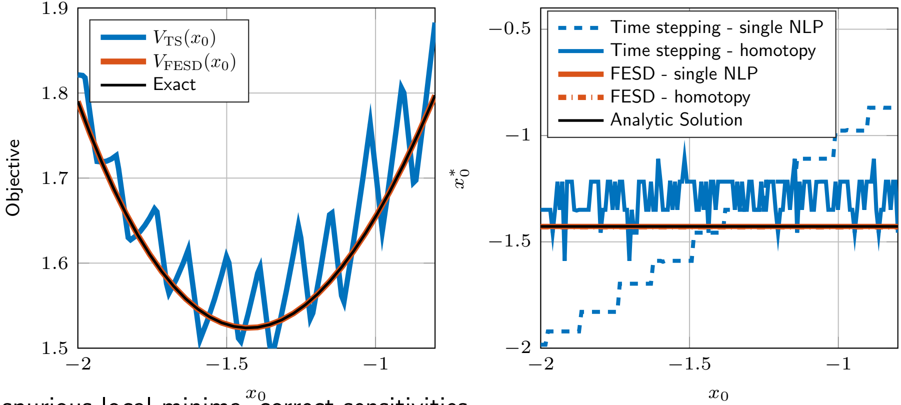
Tutorial example inspired by [Stewart & Anitescu, 2010]



- no spurious local minima, correct sensitivities
- convergence to the "true" local minimum, both with homotopy and without it
- accuracy of order $O(h^p)$, in contrast to standard approach with only $O(h)$

Revisiting the OCP example - now with FESD

Tutorial example inspired by [Stewart & Anitescu, 2010]



- no spurious local minima, correct sensitivities
- convergence to the "true" local minimum, both with homotopy and without it
- accuracy of order $O(h^p)$, in contrast to standard approach with only $O(h)$
- FESD solves the accuracy and convergence issues

OCP example

Benchmark example with entering/leaving sliding mode

OCP with sliding modes

$$\min_{x(\cdot),u(\cdot)} \int_0^4 u(t)^\top u(t) + v(t)^\top v(t)\,\mathrm{d}t$$

$$\text{s.t.} \quad x(0) = (\frac{2\pi}{3}, \frac{\pi}{3}, 0, 0)$$

$$\dot{x}(t) = \begin{bmatrix} -\text{sign}(c(x(t))) + v(t) \\ u(t) \end{bmatrix}$$

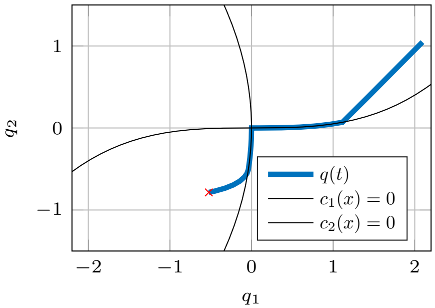
$$-2e \leq v(t) \leq 2e$$

$$-10e \leq u(t) \leq 10e \quad t \in [0, 4],$$

$$q(T) = (-\frac{\pi}{6}, -\frac{\pi}{4})$$

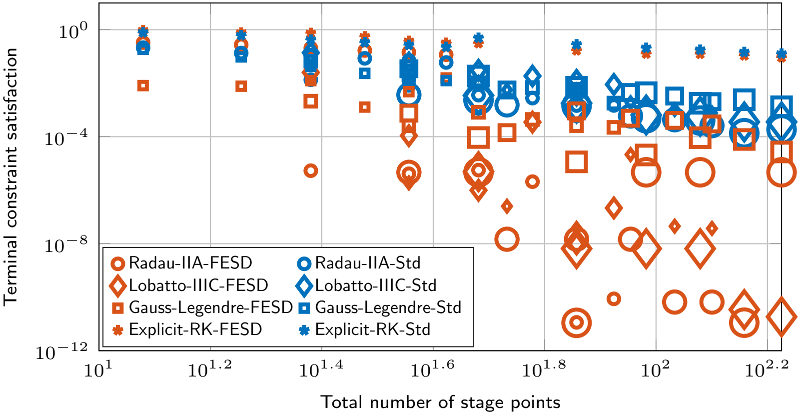
States $q, v \in \mathbb{R}^2$ and control $u \in \mathbb{R}^2$, $x = (q, v)$

Switching functions $c(x) = \begin{bmatrix} q_1 + 0.15 q_2^2 \\ 0.05 q_1^3 + q_2 \end{bmatrix}$



FESD vs standard IRK - number of function evaluations

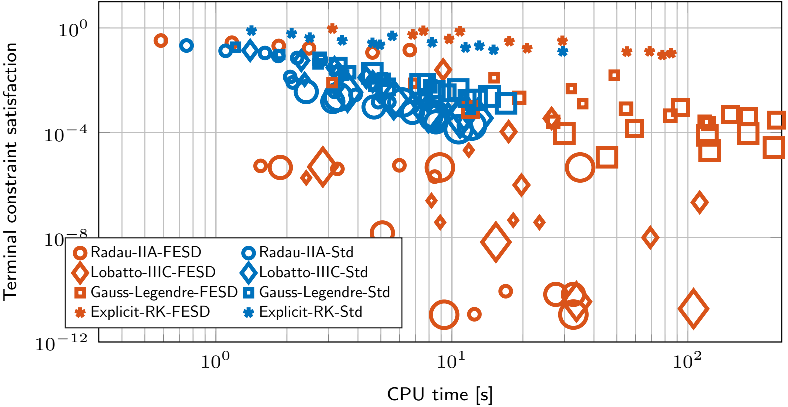
Benchmark on an optimal control problem with nonlinear sliding modes



Terminal constraint satisfaction vs. number of stage points

FESD vs standard IRK - CPU Time

Benchmark on an optimal control problem with nonlinear sliding modes



Terminal constraint satisfaction vs. CPU time

FESD one million times more accurate than Std. for CPU time of ≈ 2 s

Conclusions and summary

- Finite Elements with Switch Detection (FESD) allow highly accurate simulation and optimal control for nonsmooth systems of level NSD2
- Following similar lines, FESD can be derived for the Heaviside step reformulation

Conclusions and summary

- Finite Elements with Switch Detection (FESD) allow highly accurate simulation and optimal control for nonsmooth systems of level NSD2
- Following similar lines, FESD can be derived for the Heaviside step reformulation
- Key ideas: make step sizes degrees of freedom and introduce implicit relations that locate the switches
- Switch detection not only essential for high accuracy, **but also for correct sensitivities** (no spurious solutions)

Conclusions and summary

- Finite Elements with Switch Detection (FESD) allow highly accurate simulation and optimal control for nonsmooth systems of level NSD2
- Following similar lines, FESD can be derived for the Heaviside step reformulation
- Key ideas: make step sizes degrees of freedom and introduce implicit relations that locate the switches
- Switch detection not only essential for high accuracy, **but also for correct sensitivities** (no spurious solutions)
- FESD solves many of the issues that standard methods have: integration accuracy, convergence of sensitivities
- Main practical difficulty: solving Mathematical Programs with Complementarity Constraints (MPCC)

References

- Brian T. Baumrucker and Lorenz T. Biegler. MPEC strategies for optimization of a class of hybrid dynamic systems. Journal of Process Control, 19(8):1248–1256, 2009.
- David E Stewart and Mihai Anitescu. Optimal control of systems with discontinuous differential equations. Numerische Mathematik, 114(4):653–695, 2010.
- Armin Nurkanović, Mario Sperl, Sebastian Albrecht, and Moritz Diehl. Finite Elements with Switch Detection for Direct Optimal Control of Nonsmooth Systems. Submitted to Numerische Mathematik 2022.
- Armin Nurkanović, Sebastian Albrecht, and Moritz Diehl. Limits of MPCC Formulations in Direct Optimal Control with Nonsmooth Differential Equations. In 2020 European Control Conference (ECC), pages 2015–2020, 2020.
- Armin Nurkanović and Moritz Diehl. NOSNOC: A software package for numerical optimal control of nonsmooth systems. IEEE Control Systems Letters, 2022.
- Armin Nurkanović, Anton Pozharskiy, Jonathan Frey, and Moritz Diehl. Finite elements with switch detection for numerical optimal control of nonsmooth dynamical systems with set-valued step functions. arXiv preprint arXiv:2307.03482, 2023.
- Armin Nurkanović, Jonathan Frey, Anton Pozharskiy, and Moritz Diehl. Finite elements with switch detection for direct optimal control of nonsmooth systems with set-valued step functions. In Conference on Decision on Control, 2023.

Switch detection - example

Suppose that $x(t)$ crosses from R_1 to R_2 and recall that $\mu = \min_j g_j(x)$
Continuous time:

- Before switch: $\theta_1(t) > 0, \lambda_1(t) = 0$, and $\theta_2(t) = 0, \lambda_2 \geq 0$
- After switch: $\theta_1(t) = 0, \lambda_1(t) \geq 0$, and $\theta_2(t) > 0, \lambda_2 = 0$

Switch detection - example

Suppose that $x(t)$ crosses from R_1 to R_2 and recall that $\mu = \min_j g_j(x)$
Discrete time (switch between the n-th and $n+1$-st finite element):

- Before switch: $\theta_{n,j,1}(t) > 0, \lambda_{n,j,1}(t) = 0$, and $\theta_{n,j,2}(t) = 0, \lambda_{n,j,2} \geq 0$
- After switch: $\theta_{n,j,1}(t) = 0, \lambda_{n,j,1}(t) > 0$, and $\theta_{n,j,2}(t) > 0, \lambda_{n,j,2} = 0$

Switch detection - example

Suppose that $x(t)$ crosses from R_1 to R_2 and recall that $\mu = \min_j g_j(x)$
Discrete time (switch between the n-th and $n+1$-st finite element):

- Before switch: $\theta_{n,j,1}(t) > 0, \lambda_{n,j,1}(t) = 0$, and $\theta_{n,j,2}(t) = 0, \lambda_{n,j,2} \geq 0$
- After switch: $\theta_{n,j,1}(t) = 0, \lambda_{n,j,1}(t) > 0$, and $\theta_{n,j,2}(t) > 0, \lambda_{n,j,2} = 0$

From Lemma 1 it follows that $\lambda_{n,n_\mathrm{s},1} = \lambda_{n,n_\mathrm{s},2} = 0$

Switch detection conditions

$$g_1(x_{n+1}) = \lambda_{n,n_\mathrm{s},1} - \mu_{n,n_\mathrm{s}}$$

Switch detection - example

Suppose that $x(t)$ crosses from R_1 to R_2 and recall that $\mu = \min_j g_j(x)$
Discrete time (switch between the n-th and $n+1$-st finite element):

- Before switch: $\theta_{n,j,1}(t) > 0, \lambda_{n,j,1}(t) = 0$, and $\theta_{n,j,2}(t) = 0, \lambda_{n,j,2} \geq 0$
- After switch: $\theta_{n,j,1}(t) = 0, \lambda_{n,j,1}(t) > 0$, and $\theta_{n,j,2}(t) > 0, \lambda_{n,j,2} = 0$

From Lemma 1 it follows that $\lambda_{n,n_\mathrm{s},1} = \lambda_{n,n_\mathrm{s},2} = 0$

Switch detection condition

$$g_1(x_{n+1}) = 0 - g_2(x_{n+1})$$

Switch detection - example

Suppose that $x(t)$ crosses from R_1 to R_2 and recall that $\mu = \min_j g_j(x)$
Discrete time (switch between the n-th and $n+1$-st finite element):

- Before switch: $\theta_{n,j,1}(t) > 0, \lambda_{n,j,1}(t) = 0$, and $\theta_{n,j,2}(t) = 0, \lambda_{n,j,2} \geq 0$
- After switch: $\theta_{n,j,1}(t) = 0, \lambda_{n,j,1}(t) > 0$, and $\theta_{n,j,2}(t) > 0, \lambda_{n,j,2} = 0$

From Lemma 1 it follows that $\lambda_{n,n_\mathrm{s},1} = \lambda_{n,n_\mathrm{s},2} = 0$

Switch detection conditions

$$0 = g_1(x_{n+1}) - g_2(x_{n+1}) = \psi_{12}(x_{n+1})$$

Implies constraint such that h_n must adapt for exact switch detection!