Naval Postgraduate School

Computational Optimal Control

Theory & Tools Beyond Nonlinear Programming

Mike Ross
Distinguished Professor
& Program Director, Control and Optimization

3rd Workshop in Computational Issues in Nonlinear Control
October 8th, 2019, Monterey, CA

The views expressed in these slides are not the official views of the US Navy. These views are solely those of the presenter: I. M. Ross, Distinguished Professor & Program Director, Control & Optimization.

What is Optimal Control?

- **Not just LQR, LQG …**
- **Everything is nonlinear == conference theme**
  - Sample Problem:

$$
\overbrace{(B)}^{\textit{problem}}
\left\{
\begin{array}{lrl}
\text{Minimize} & J[\boldsymbol{x}(\cdot), \boldsymbol{u}(\cdot), t_0, t_f] := & \\
& E(\boldsymbol{x}_0, \boldsymbol{x}_f, t_0, t_f) + \displaystyle\int_{t_0}^{t_f} F(\boldsymbol{x}(t), \boldsymbol{u}(t), t)\, dt & \Big\} \textit{ cost} \\
\text{Subject to} & \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) & \} \textit{ dynamics} \\
& \boldsymbol{e}^L \leq \boldsymbol{e}(\boldsymbol{x}_0, \boldsymbol{x}_f, t_0, t_f) \leq \boldsymbol{e}^U & \} \textit{ events} \\
& \boldsymbol{h}^L \leq \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \leq \boldsymbol{h}^U & \} \textit{ path}
\end{array}
\right.
$$

Why Optimal Control?

- **Optimal control is the foundational science for:**
  - Nonlinear control
  - Machine learning; deep learning.
    - TOE
  - Economics
  - Pharmacology
  - Physics
  - Management science
  - …

- **Optimal control is also an emerging foundational science for:**
  - Nonlinear programming problems
  - Traveling salesman problems and beyond

Outline: Three Part Theory + One Part Tool

1. **Computational optimal control vs nonlinear programming**

2. **Optimal control programming**
   - Non-introduction introduction

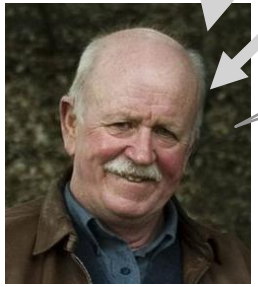
3. **Birkhoff methods for optimal control**
   - Very brief intro to some seemingly magical properties

4. **MATLAB Toolbox for optimal control**
   - How to solve hard problems easily

Theory

Tool

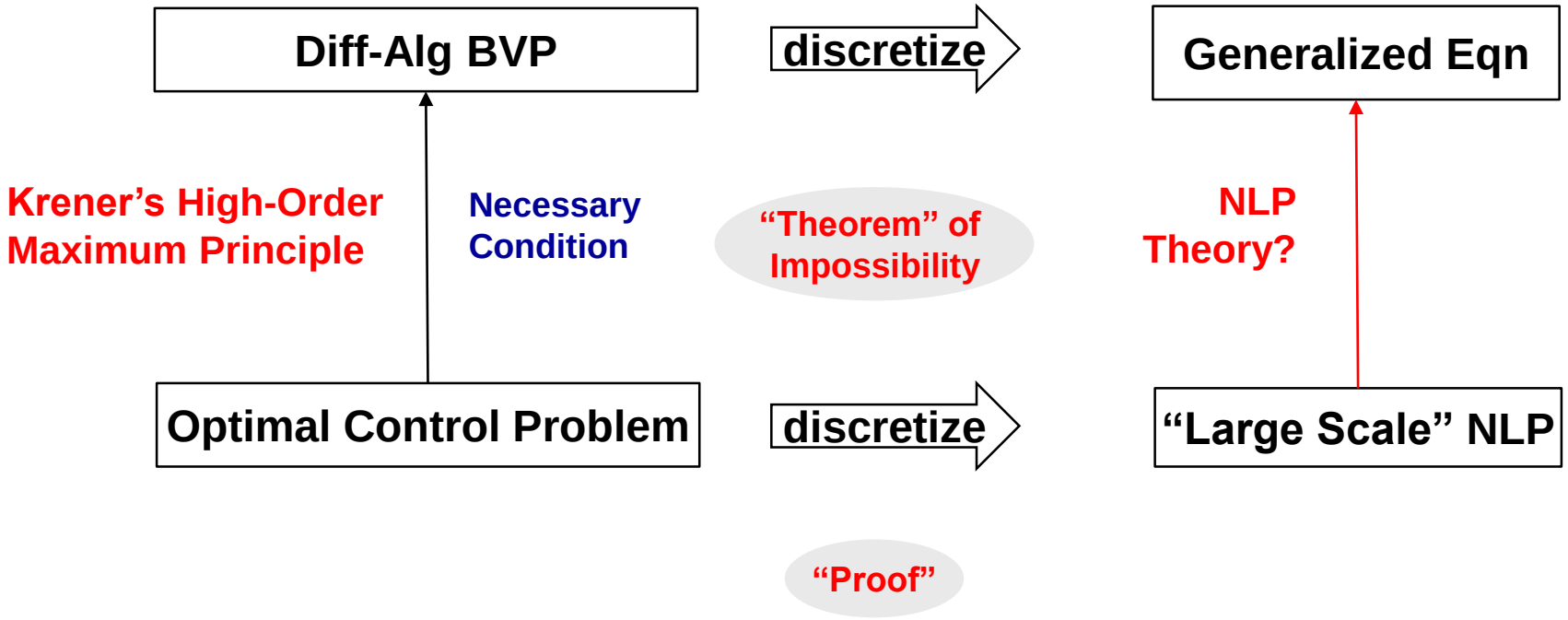
That's the conference theme!



KEY ASSUMPTION

Necessary conditions are actually necessary!

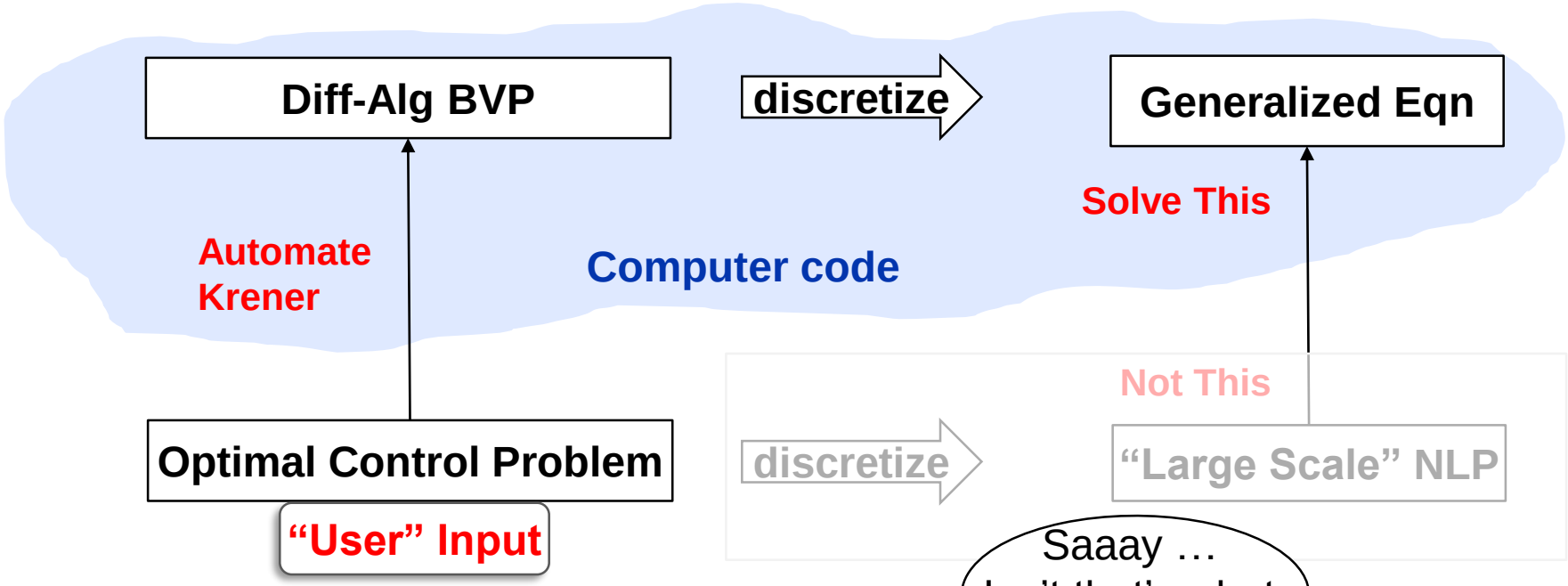
Optimal Control = Large Scale NLP?



- **Optimal control theory has an "infinitely" rich amount of information via:**
  - "Convexity" of time
  - Diffuse measure
  - "Infinite" differentiability wrt time

- **NLP theory can have "at most" countably infinite information.**
  - **Conjecture: Impossible to close the information gap**
  - There are no "Krener-type" conditions in NLP theory
    - Unlikely to discover one

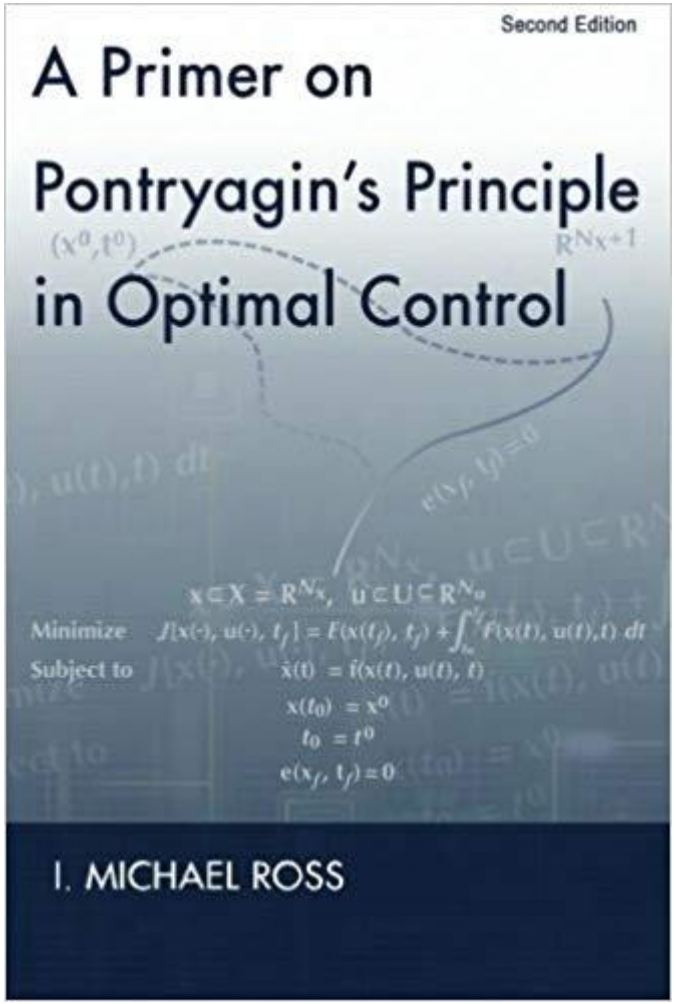
Optimal Control Programming: Big Pic Idea



- **Big Questions:**
  1. Is it possible to automate Krener?
     - No formulas for Krener's higher-order variations

  — Start with automatic Pontryagin; graduate to "low-Fi" K
     - Low-Fi Krener = Generalized Goh-Robbins condition – re "automatic" high order differentiation
  2. How to solve the generalized equation?

Extra, Extra: Read All About It Here!



Yes, Shameless Advertisement

Introduction To A Sacrilegious Idea

- **What is the best way to discretize a nonlinear ODE?**

$$\dot{x} = f(x, u, t)$$

- **What if?**

$$\dot{x} := v$$

Best way to discretize this?!

$$v = f(x, u, t)$$

- **DAE: Differential Algebraic Equation (= horror of horrors?)**

A Quick Pontryagin Math for Sacrilegiousness

$$x \in \mathbb{R}, \quad u \in \mathbb{R}, \quad \tau \in [-1, 1]$$

$$(B) \begin{cases} \textit{Minimize} & J[x(\cdot), u(\cdot)] = E(x(1)) \\ \textit{Subject to} & \dot{x}(\tau) = f(x(\tau), u(\tau)) \\ & x(-1) = 0 \end{cases}$$

$$x \in \mathbb{R}, \quad u \in \mathbb{R}, \quad \tau \in [-1, 1]$$

$$(B^\flat) \begin{cases} \textit{Minimize} & J^\flat[x(\cdot), u(\cdot), v(\cdot)] = E(x(1)) \\ \textit{Subject to} & \dot{x}(\tau) = v(\tau) \\ & f(x(\tau), u(\tau)) - v(t) = 0 \\ & x(-1) = 0 \end{cases}$$

Nonlinear Adjoint Equation ☹

$$
(B^{\flat,\lambda})\left\{
\begin{aligned}
\dot{x} &= v \\
-\dot{\lambda}^{\flat} &= \left[\partial_{\boldsymbol{x}} f(x,u)\right]^T \boldsymbol{\mu}^{\flat} \\
f(x,u) - v &= \mathbf{0} \\
\lambda^{\flat} - \mu^{\flat} &= 0 \\
\left[\partial_u f(x,u)\right]^T \mu^{\flat} &= \mathbf{0} \\
x_0 &= 0 \\
\lambda_f &= \partial_{x_f} E(x_f)
\end{aligned}
\right.
$$

Unnecessary equation

"Create" A New "Flat ♪" Set of Primal-Dual Eqns

$$
(B^{\lambda,\flat})\begin{cases}
\dot{x} - v = 0 \\
\dot{\lambda} - \omega = 0 \\
f(x,u) - v = 0 \\
\partial_x f(x,u)\lambda + \omega = 0 \\
\partial_u f(x,u)\lambda = 0 \\
x(-1) = 0 \\
\lambda(1) - \partial_{x(1)} E(x(1)) = 0
\end{cases}
$$

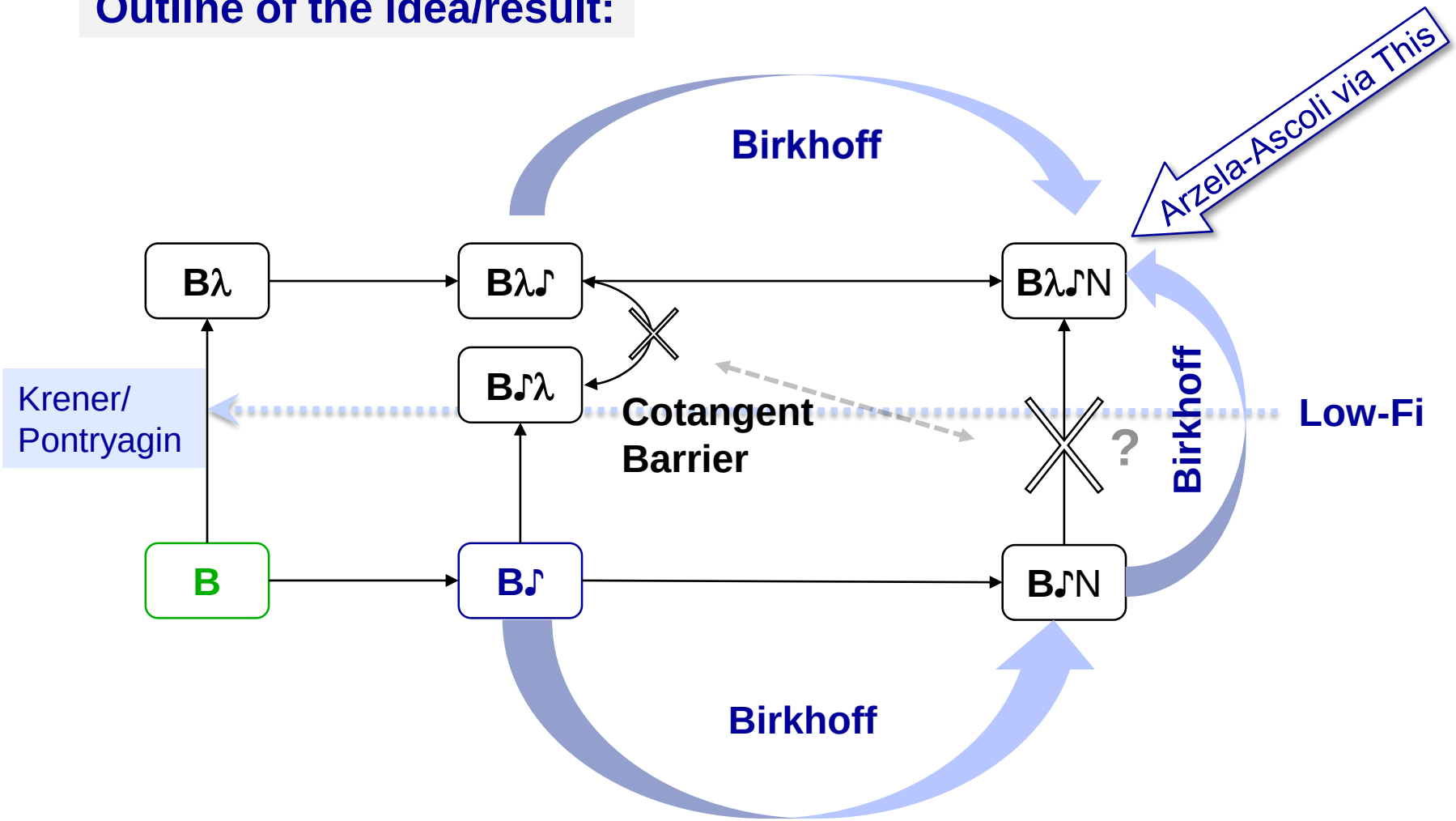
DAE Galore: "Inescapable" DAE

Discretize and Solve this Generalized Eqn via "Arzela-Ascoli" Sequencing

$$B^{\lambda,\flat} \not\equiv B^{\flat,\lambda}$$

Low-Fi Auto-Krener Via Birkhoff

Outline of the idea/result:



Brief Intro to Birkhoff Interpolation

- **Birkhoff generalizes Lagrange/Hermite:**

— Example from Wang, Samson, Zhao, SISC, 36 (3); 2014

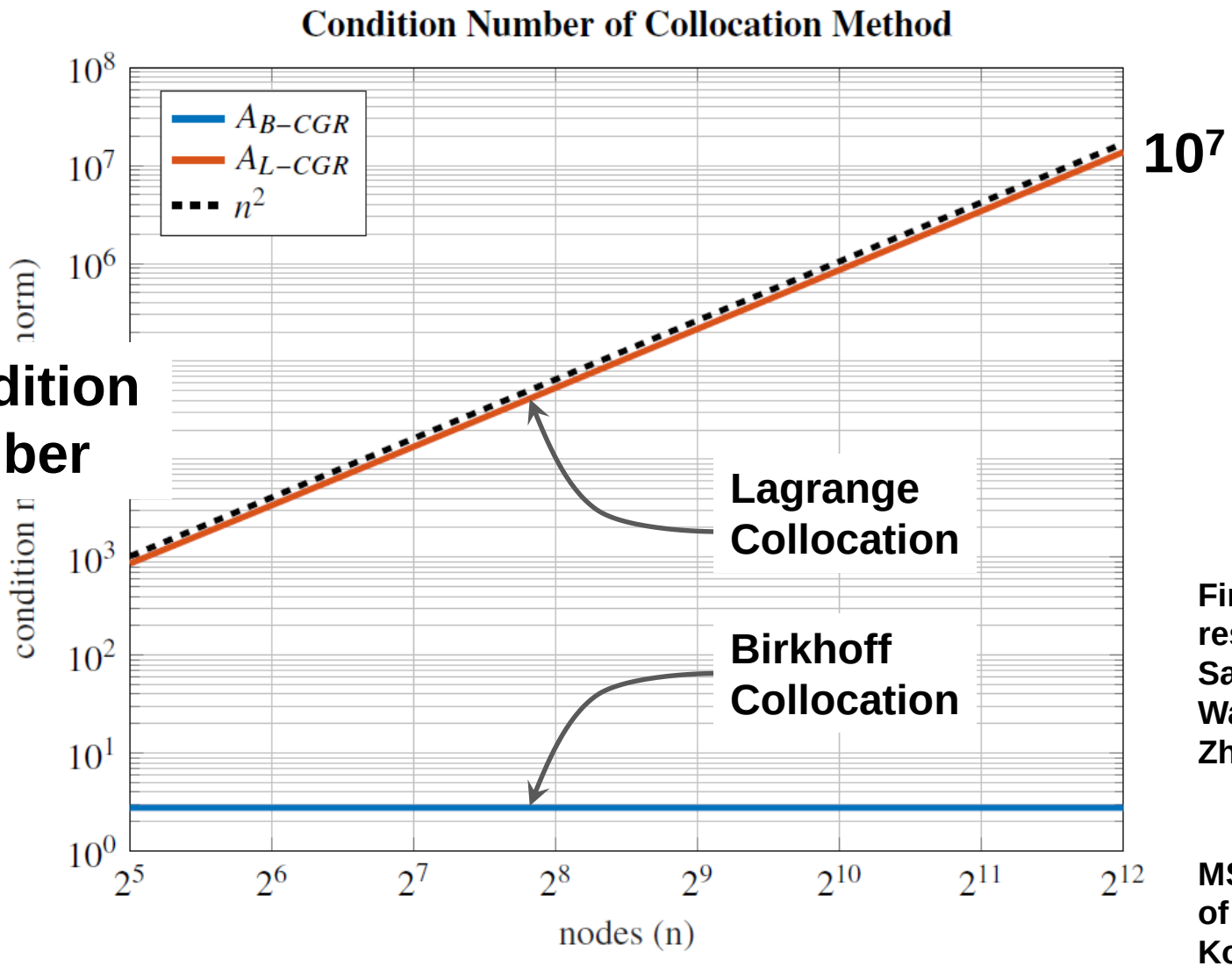
The Birkhoff interpolation polynomial p of u can be uniquely determined by

$$p(x) = u(-1)B_0(x) + \sum_{j=1}^{N-1} u''(x_j)B_j(x) + u(1)B_N(x), \quad x \in [-1,1],$$

if one can find $\{B_j\}_{j=0}^{N} \subseteq \mathbb{P}_N$, such that

$$\begin{aligned}
&B_0(-1) = 1, \quad B_0(1) = 0, \quad B_0''(x_i) = 0, \quad 1 \le i \le N-1;\\
&B_j(-1) = 0, \quad B_j(1) = 0, \quad B_j''(x_i) = \delta_{ij}, \quad 1 \le i, j \le N-1;\\
&B_N(-1) = 0, \quad B_N(1) = 1, \quad B_N''(x_i) = 0, \quad 1 \le i \le N-1.
\end{aligned}$$

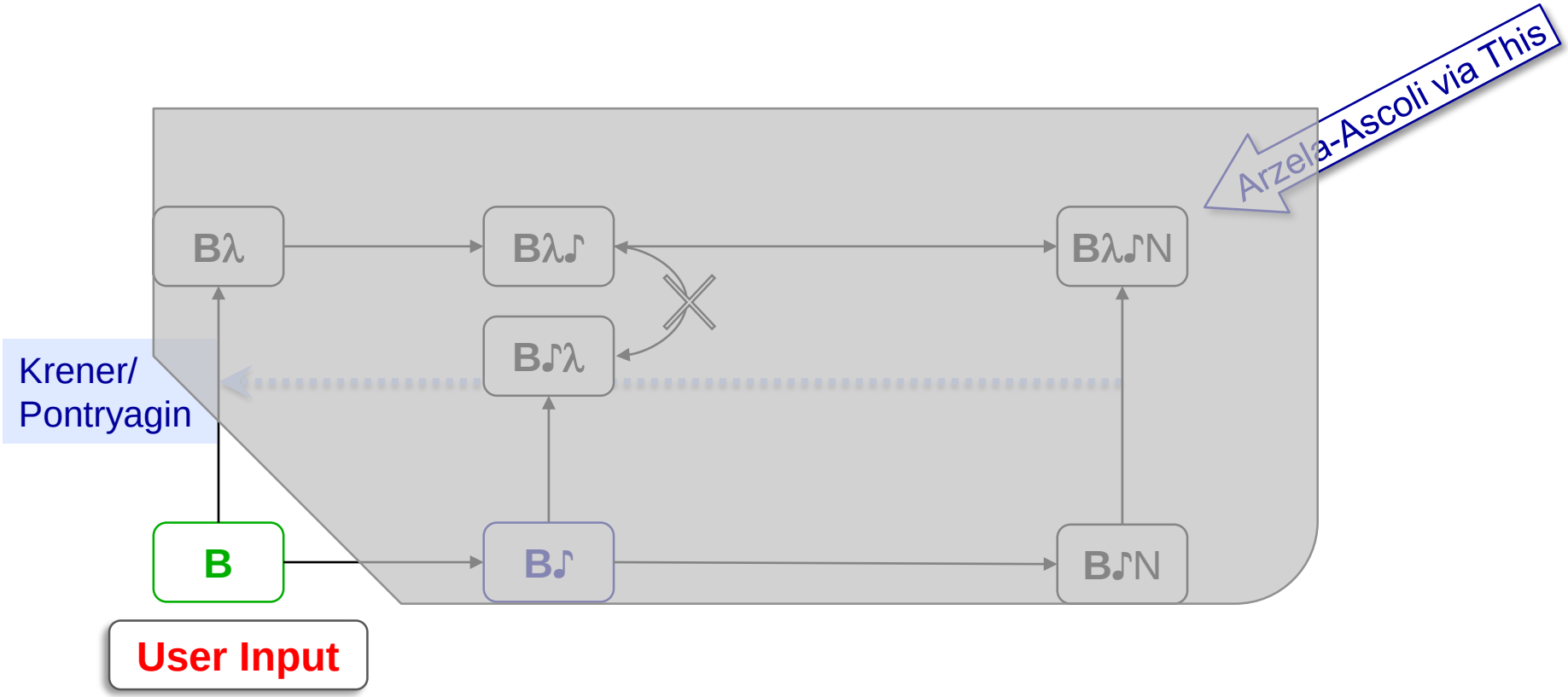
Yet Another Birkhoff "Magic"



First results by Samson, Wang & Zhao, 2014

MS Thesis of Capt Nick Koeppen

A “Turing Test” for Optimal Control Programming



This is DIDO©: A MATLAB® Toolbox for Solving Optimal Control Problems

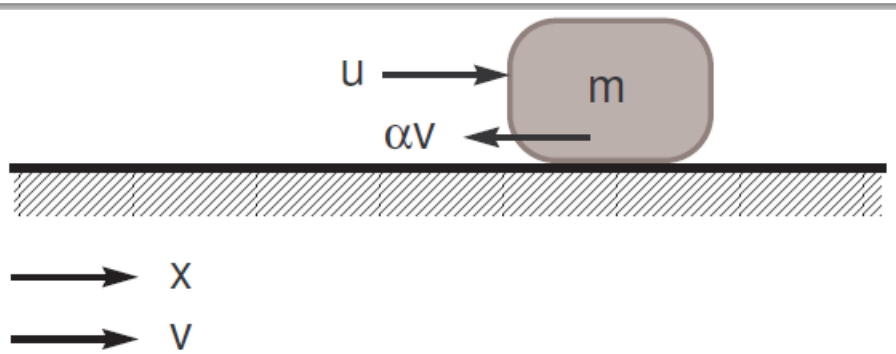
Not so shameless ad: proof of existence that this is do-able.

Illustrative Problem –1

Gong's Challenge Problem:

$$\mathbf{x} = (x, v) \in \mathbb{R}^2, \quad u \in \mathbb{R}$$

$$(G) \begin{cases} \text{Minimize} & J[\mathbf{x}(\cdot), u(\cdot)] = \displaystyle\int_0^1 v(t)u(t)\, dt \\ \text{Subject to} & \dot{x}(t) = v(t) \\ & \dot{v}(t) = -v(t) + u(t) \\ & v(t) \geq 0 \\ & 0 \leq u(t) \leq 2 \\ & (x(0), v(0)) = (0, 1) \\ & (x(1), v(1)) = (1, 1) \end{cases}$$

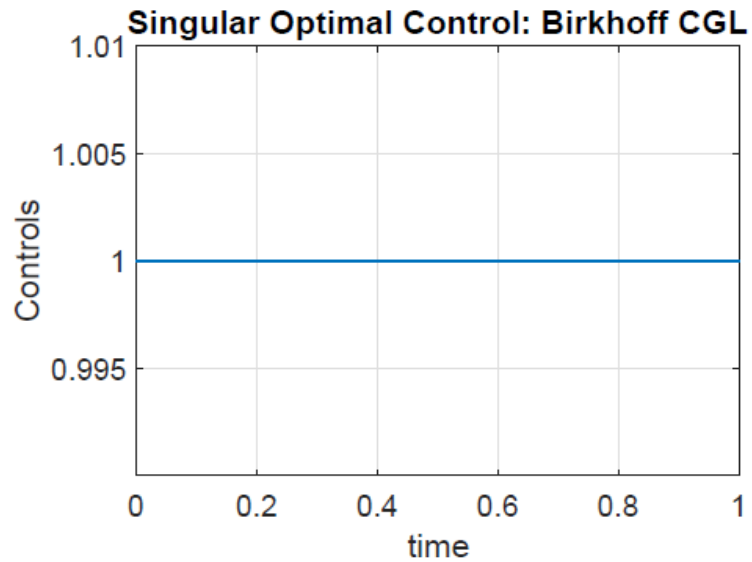


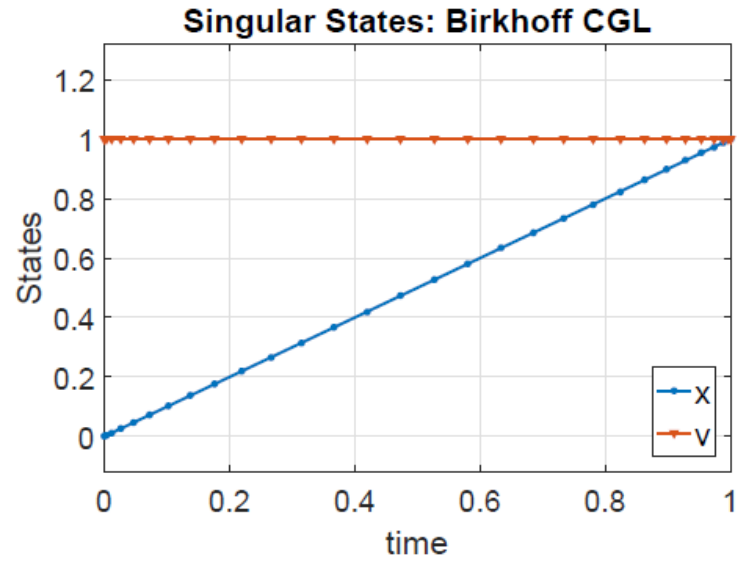
Solution via "Low-Fi" Krener:

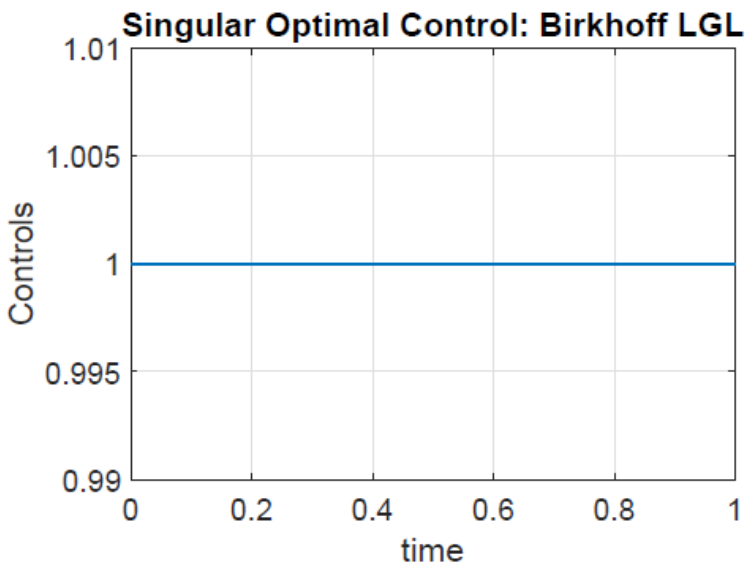
$$\begin{aligned} x^*(t) &= t \\ v^*(t) &= 1 \\ u^*(t) &= 1 \end{aligned}$$

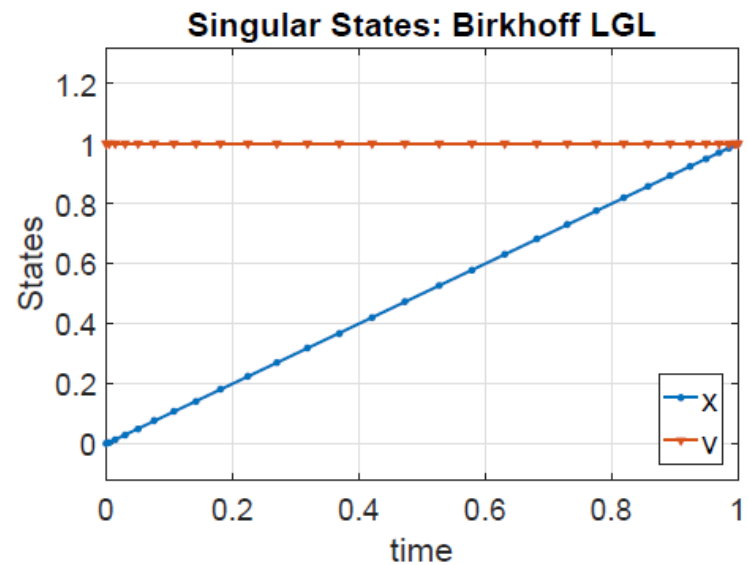
$$\begin{aligned} \lambda_x(t) &= -2 \\ \lambda_v(t) &= -1 \end{aligned}$$

Solution Via Birkhoff



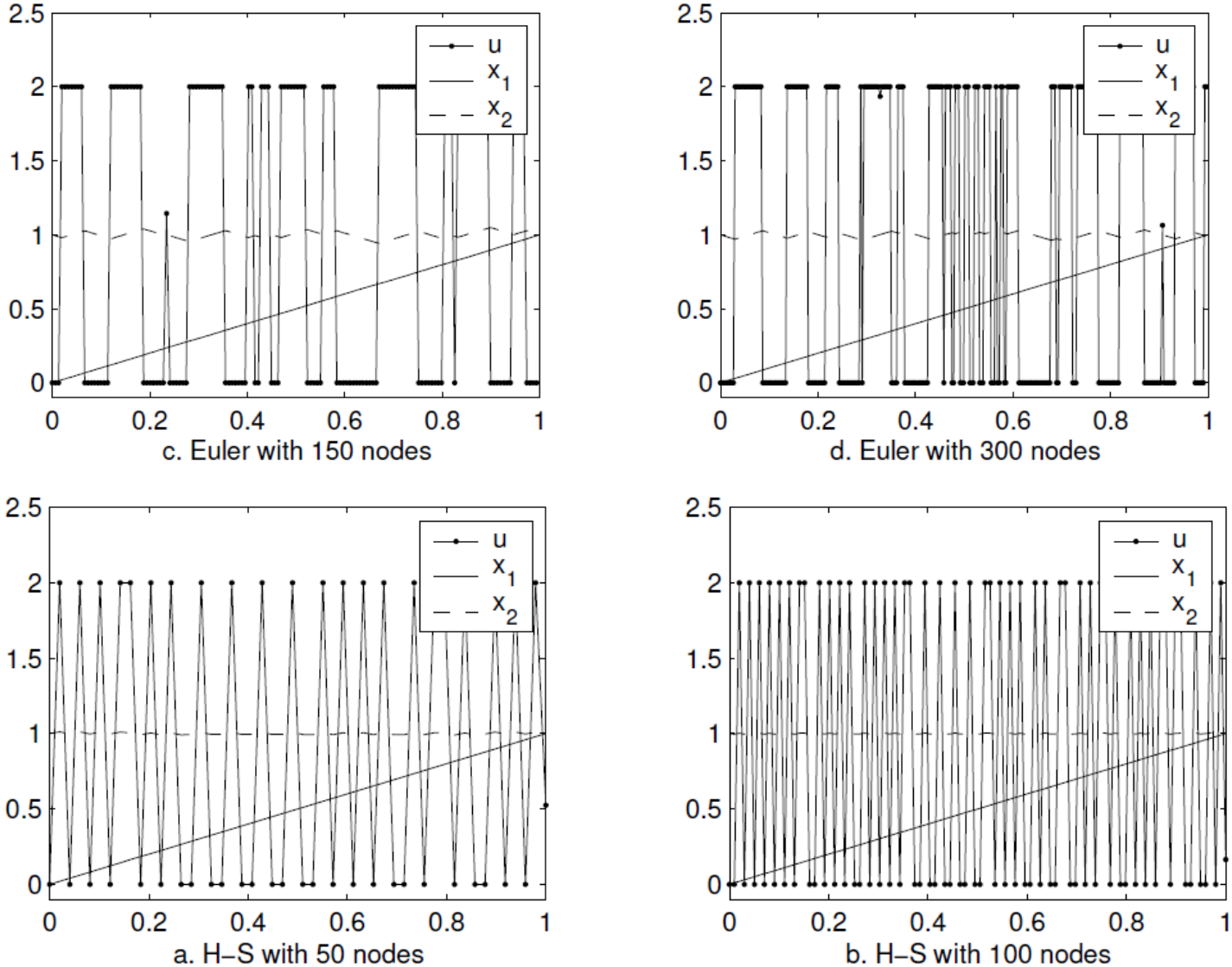






Numerics by Ron Proulx

“Solution” via Runge-Kutta



Converges in Lebesgue, not Sobolev norm

Numerics by Qi Gong

Illustrative Problem –215

- **A DAE Challenge Problem**
  - Betts, Campbell et al (2016, 2018)

$$J(x,u) = \int_0^T cu^2 + d(x - L\sin(t+\alpha))^2 + d(y - L\cos(t+\alpha))^2\, dt \qquad (1)$$

subject to the constraints:

"Index 3 DAE"

$$(D): \begin{cases} \ddot{x} = -\lambda x - a\dot{x} + uy \\ \ddot{y} = -\lambda y - a\dot{y} - g - ux \\ 0 = x^2 + y^2 - L^2 \\ x(0) = 0 \;\; y(0) = L \;\; \dot{x}(0) = 0 \;\; \dot{y}(0) = 0 \end{cases} \qquad (2)$$

- **Not solvable via several software (e.g., SOCX)**
  - INDEPENDENTLY VERIFIED

"Low-Fi" Krener Conditions for the "DAE" Problem

- **The DAE challenge problem is also singular**
  - Krener-type necessary conditions can be easily derived:
  - Repeated diff … GLC, "Goh-Robbins"

$$\frac{d^k}{dt^k}(\partial_u H) = 0$$

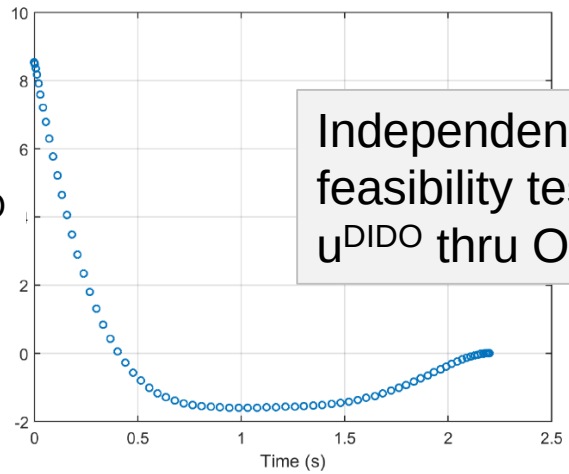
One of many reasons why we want convergence in Sobolev spaces

Sample singular integral of motion

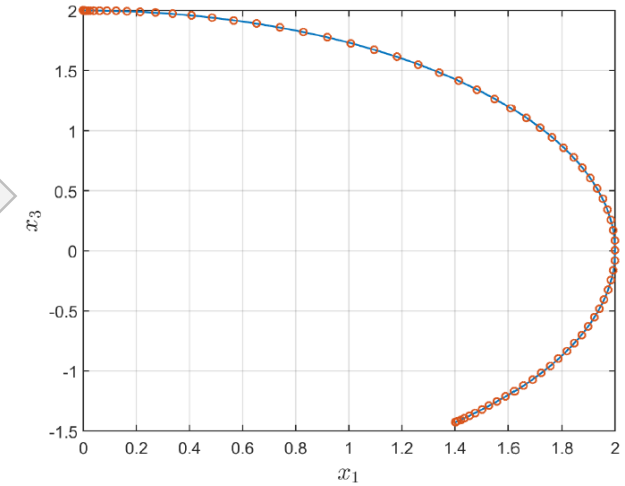
$$-\lambda_{x_2} x_1 - \lambda_{x_4} x_3 = 0$$

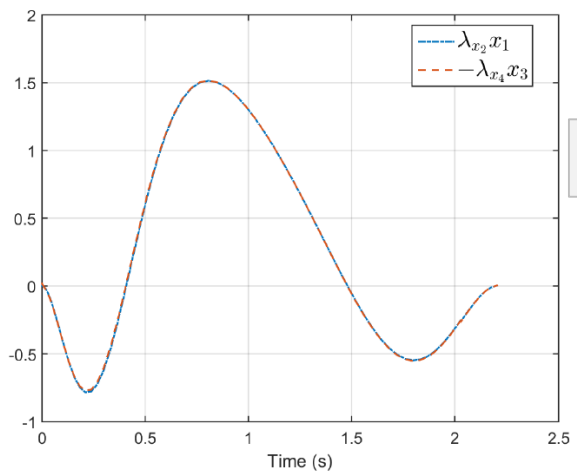
Solution via DIDO w/ Independent V&V

u^{DIDO}



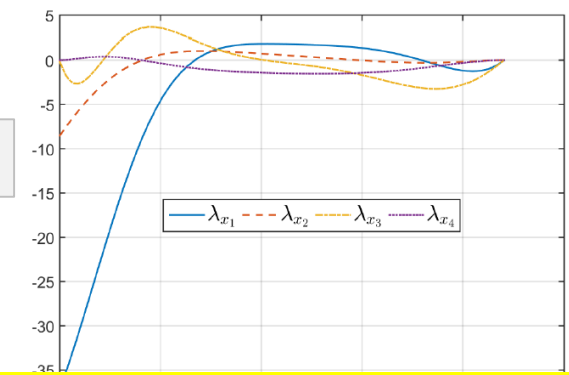
Independent primal feasibility test: propagate u^{DIDO} thru ODE





Dual feasibility tests

$$-\lambda_{x_2} x_1 - \lambda_{x_4} x_3 = 0$$



Proof of the possible!

- **DIDO requires no *a priori* "index reduction"**
  - DIDO solution agrees w/ SOCX solution generated after a laborious process of index reduction

Numerics by Mark Karpenko

Do The Principles Hold For "Regular" Problems?

- **Short answer is yes!**
  - Krener's HMP was an "easy way" to prove flaws in some arguments
- **Pontryagin's Principle (non-singular problems)**
  - Hamiltonian minimization condition
    - = NLP within the collection of necessary conditions
    - Pointwise condition
  - Discretization generates necessary conditions for a necessary condition!
    - Via KKT
- **Vast number of "nuances" that separate optimal control from "large scale NLPs"**
  - HMC
  - States and controls must be treated differently; NLP is agnostic to this distinction
  - x_{k+1} and x_k are "near each other" via Lipschitz; in NLP they are "random"
  - Scaling each x_k generates "additional" dynamics that are not in the optimal control problem
    - Scaling and balancing in optimal control problems
  - Scaling u_k can generate accuracy problems in optimality
  - Many more!

Take Aways

- **A significant body of work beyond NLP theory/tools remains to be done**
  - A vast number of theoretical issues are open
    - PhD topics (& beyond) for research!
- **"Turing software" based on some/many necessary conditions are possible**
  - DIDO = proof of the possible
- **Presented theory and tools are actively used by NASA, DoD and some industrial sectors**
  - Certificate program at the Naval Postgraduate School (Cert # 299)
- **"Hard" problems are not as hard (as advertised)**
  - Practical problems are mathematically rich: "singular," "high-index DAEs," "highly constrained" etc.
- **Computational optimal control based on necessary conditions is unquestionably better than one without it**
  - Necessary conditions might actually be necessary!

Thank You!