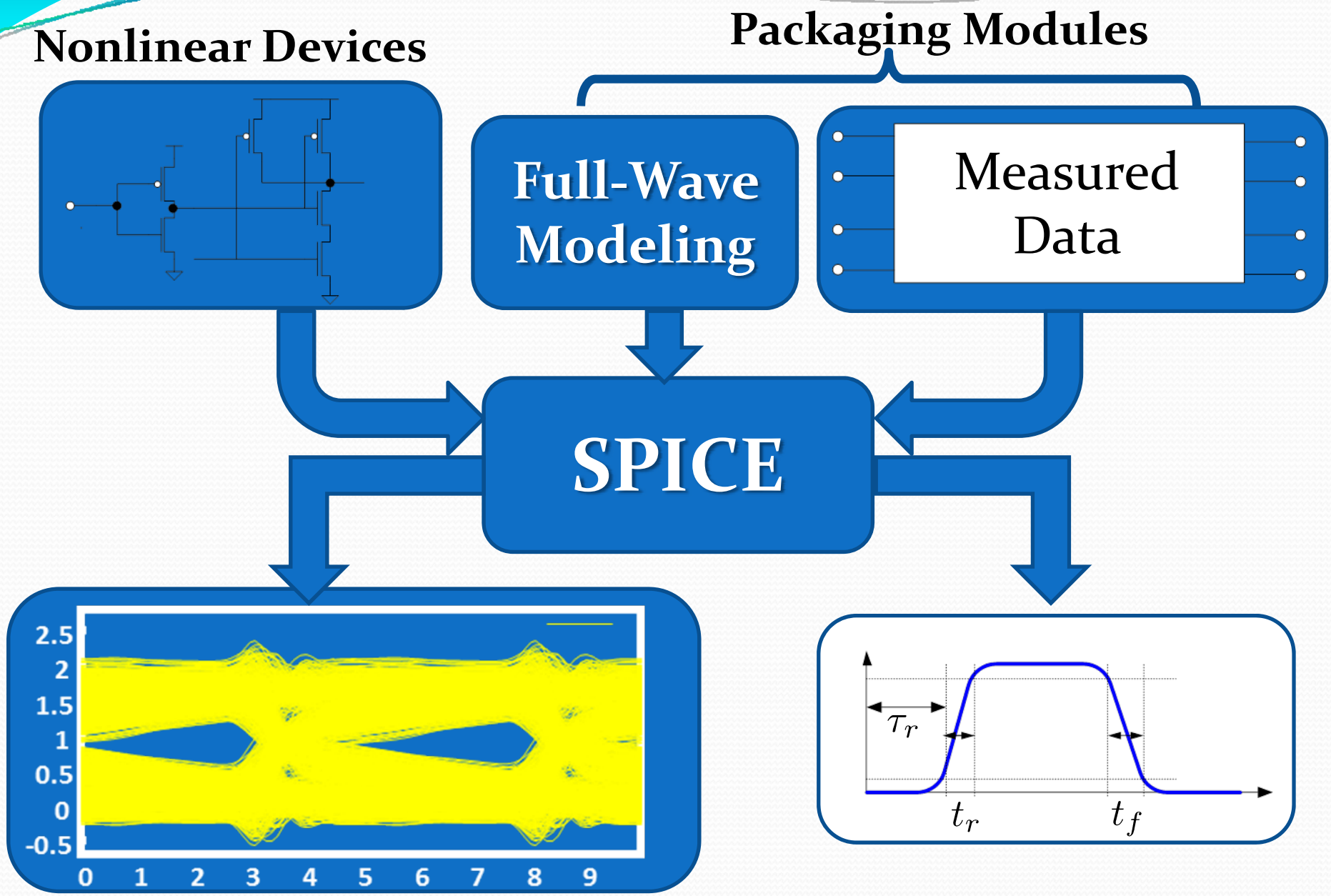
A Multi-Core High-Order A-stable and L-Stable Integration Methods For Fast Transient Simulation of High-Speed Interconnects and Transmission Line Circuits

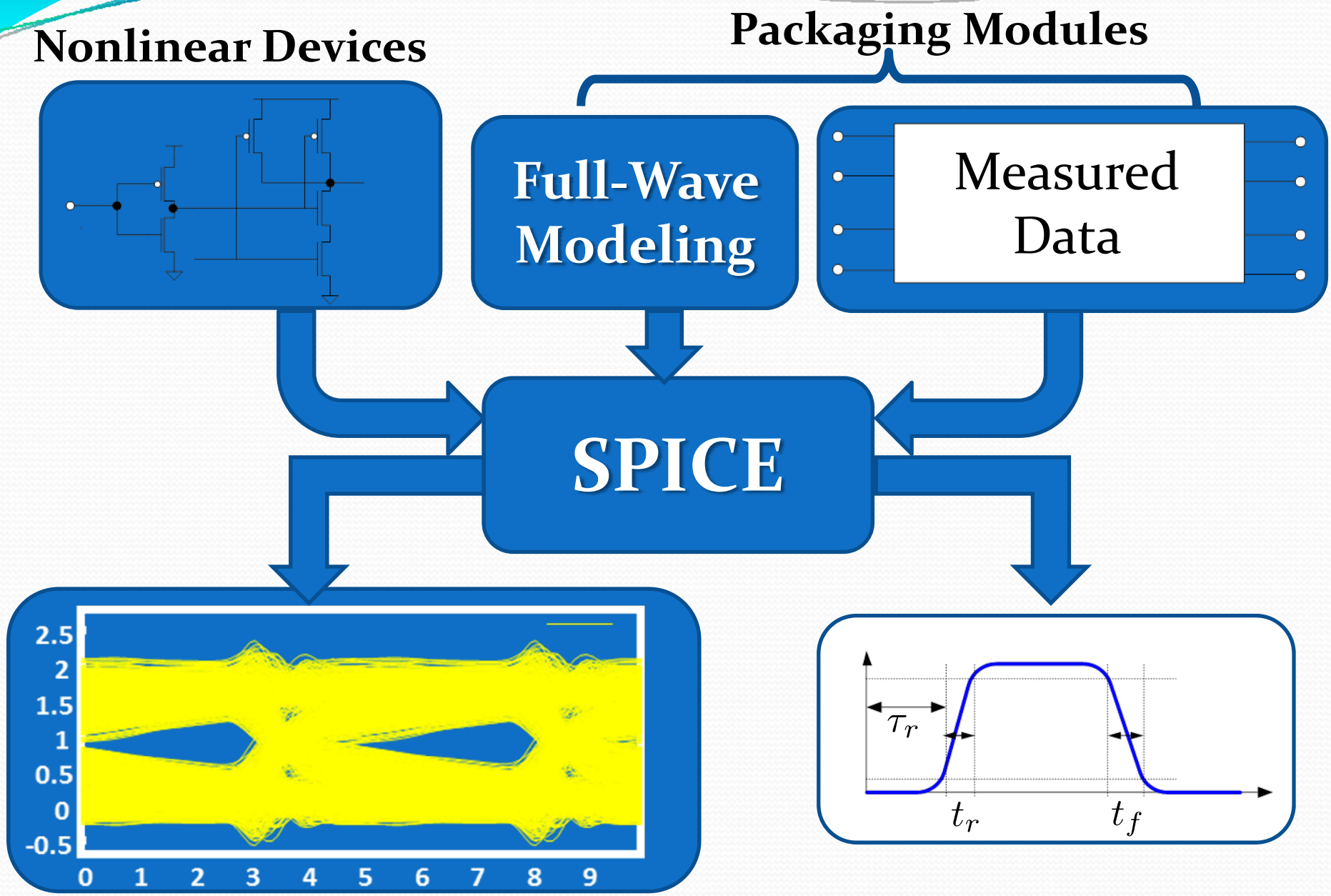
Mina Farhan[1], Emad Gad[2], Michel Nakhla[1], and Ram Achar[1]

1. Dept. of Electronics, Carleton University
2. School of Electrical Engineering and Computer Science, University of Ottawa

Background


Nonlinear Devices
Packaging Modules
Full-Wave Modeling
Measured Data
SPICE
2.5
2
1.5
1
0.5
0
-0.5
0 1 2 3 4 5 6 7 8 9
τ_r
t_r
t_f

Background


Nonlinear Devices
Packaging Modules
Full-Wave Modeling
Measured Data
SPICE
2.5
2
1.5
1
0.5
0
-0.5
0 1 2 3 4 5 6 7 8 9
τ_r
t_r
t_f

Agenda

- The troubles with classical SPICE
  - The dilemma of Order and Stability
- Recent Progress in a Stable Hi-order SPICE (HiSPICE)
  - Background, Concept, Advantages
- The Multi-Core Hi-SPICE
- Simulation Results
- Conclusion

The troubles with classical SPICE

The dilemma of Order and Stability in traditional SPICE engines.

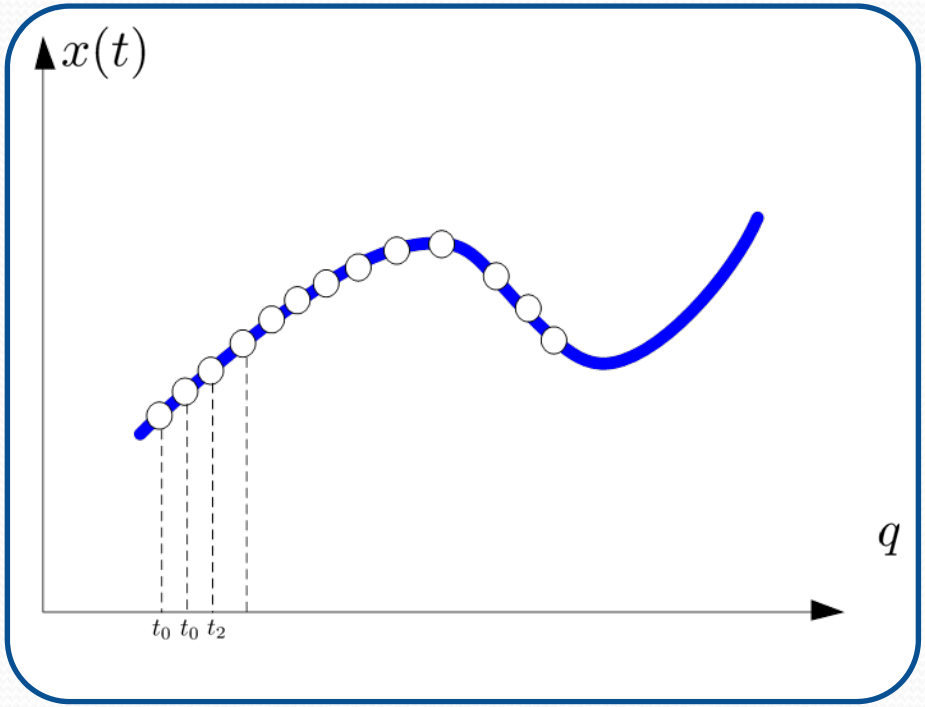
The troubles with classical SPICE

- SPICE : A Differential Equations (DE) Solver

General Circuit

$$C\frac{\mathrm{d}x(t)}{\mathrm{d}t} + Gx(t) + f(x(t)) = u(t)$$

A DE solver discretizes the time, using small step size, solving a system of nonlinear algebraic equations at each time point.



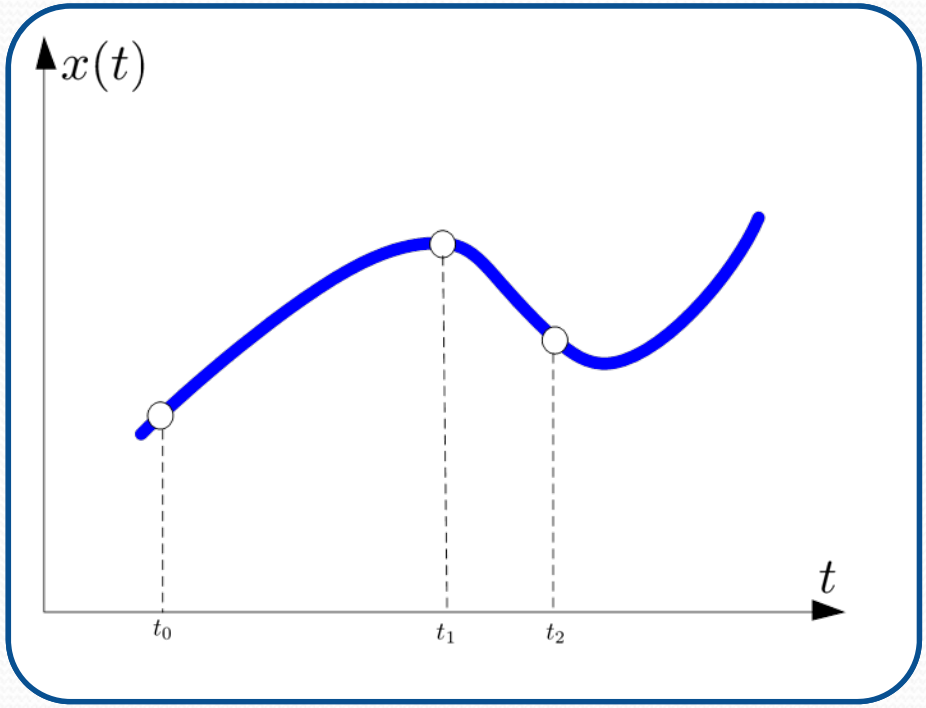
The troubles with classical SPICE

- SPICE : A Differential Equations (DE) Solver

General Circuit

$$C\frac{\mathrm{d}x(t)}{\mathrm{d}t} + Gx(t) + f(x(t)) = u(t)$$

But Why Small step sizes?
Why not Large Step Sizes?

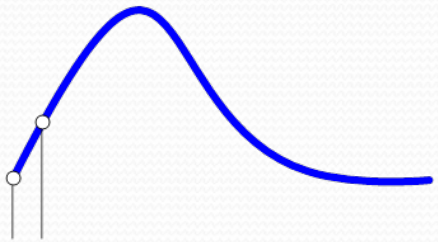


The troubles with classical SPICE

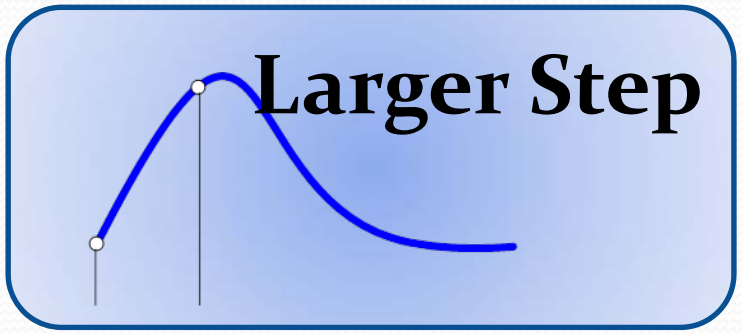
- **Classical SPICE engine:**
  - based on the Linear Multi-Step (**LMS**) methods
  - Backward Euler, Trapezoidal Rule, Gear's method
- **LMS Methods:**
  - Uses the past time steps to construct a low-order polynomials, $\sum_{i=0}^{P} a_i t^i$, to approximate the waveforms
- **Low-Order Polynomials**
  - Force small step sizes and larger than necessary time points

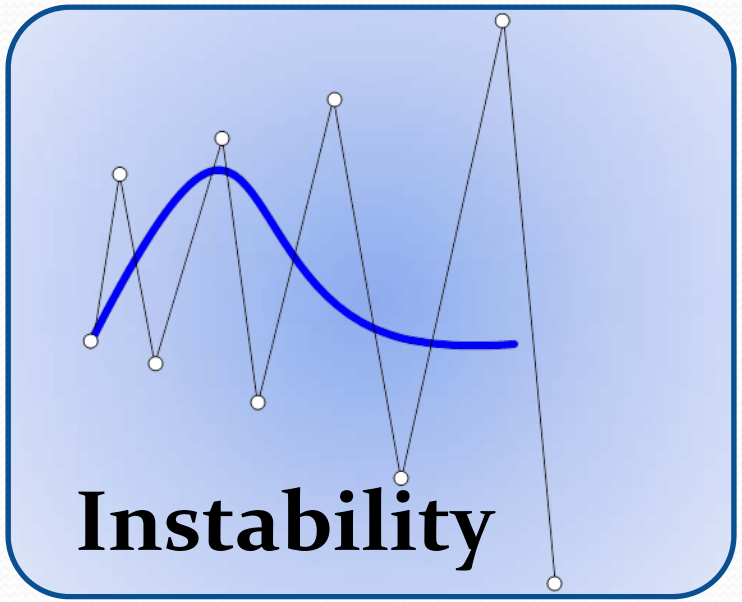
The troubles with classical SPICE

- **LMS using High-order polynomial?**
  - **Theoretically Unstable!**



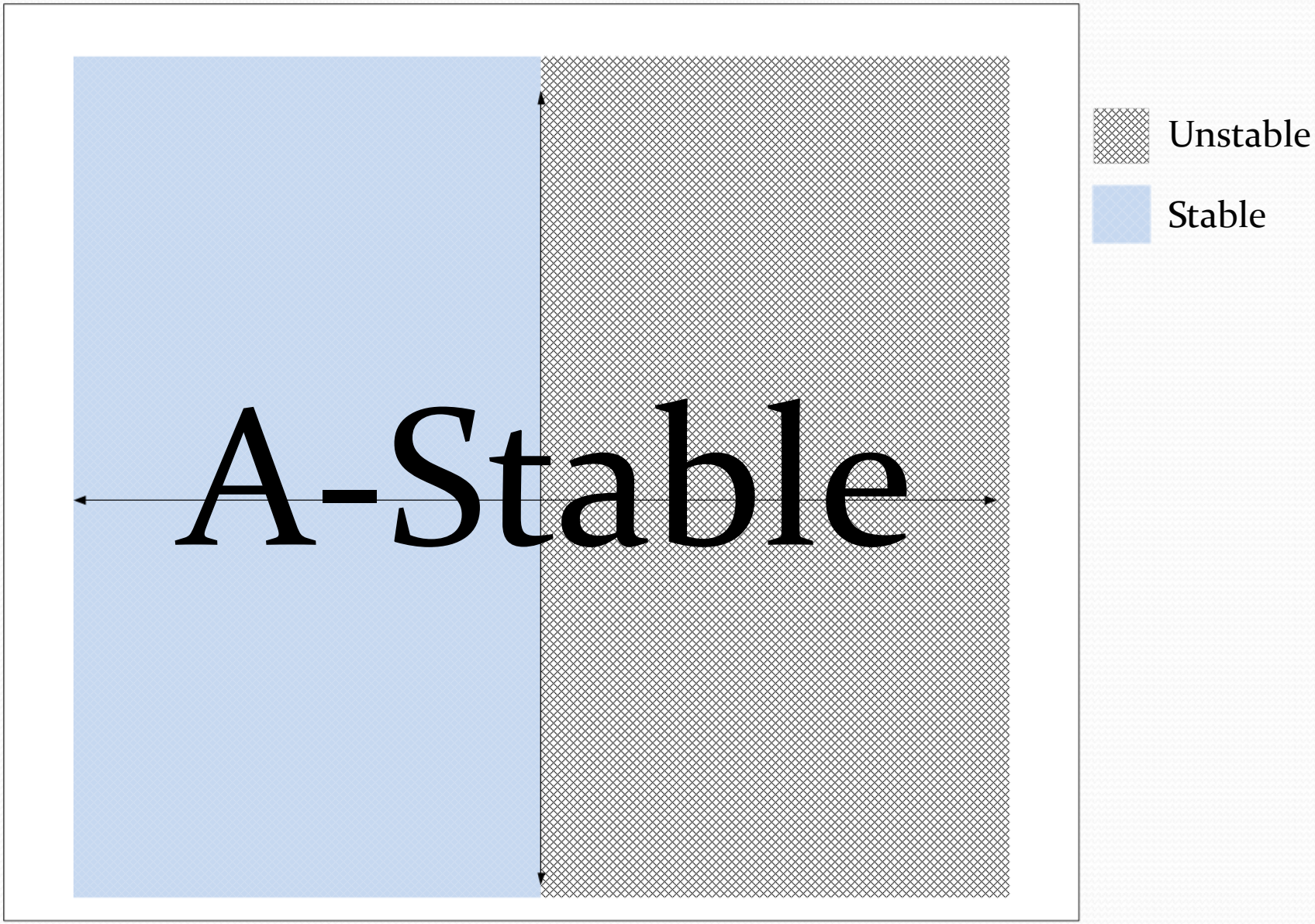
Classical SPICE (LMS) + High-Order





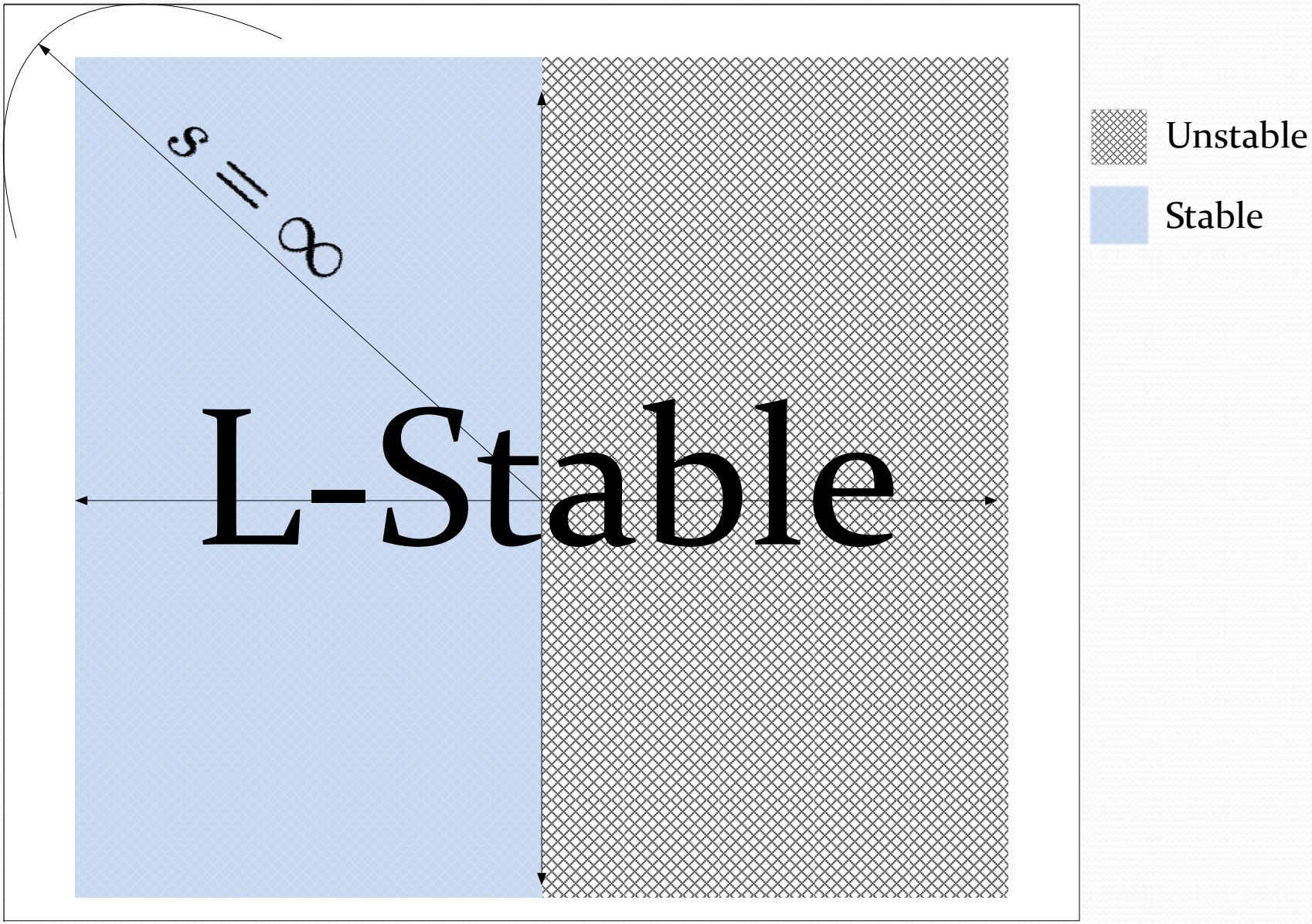
Stability in DE solvers

Laplace s-domain



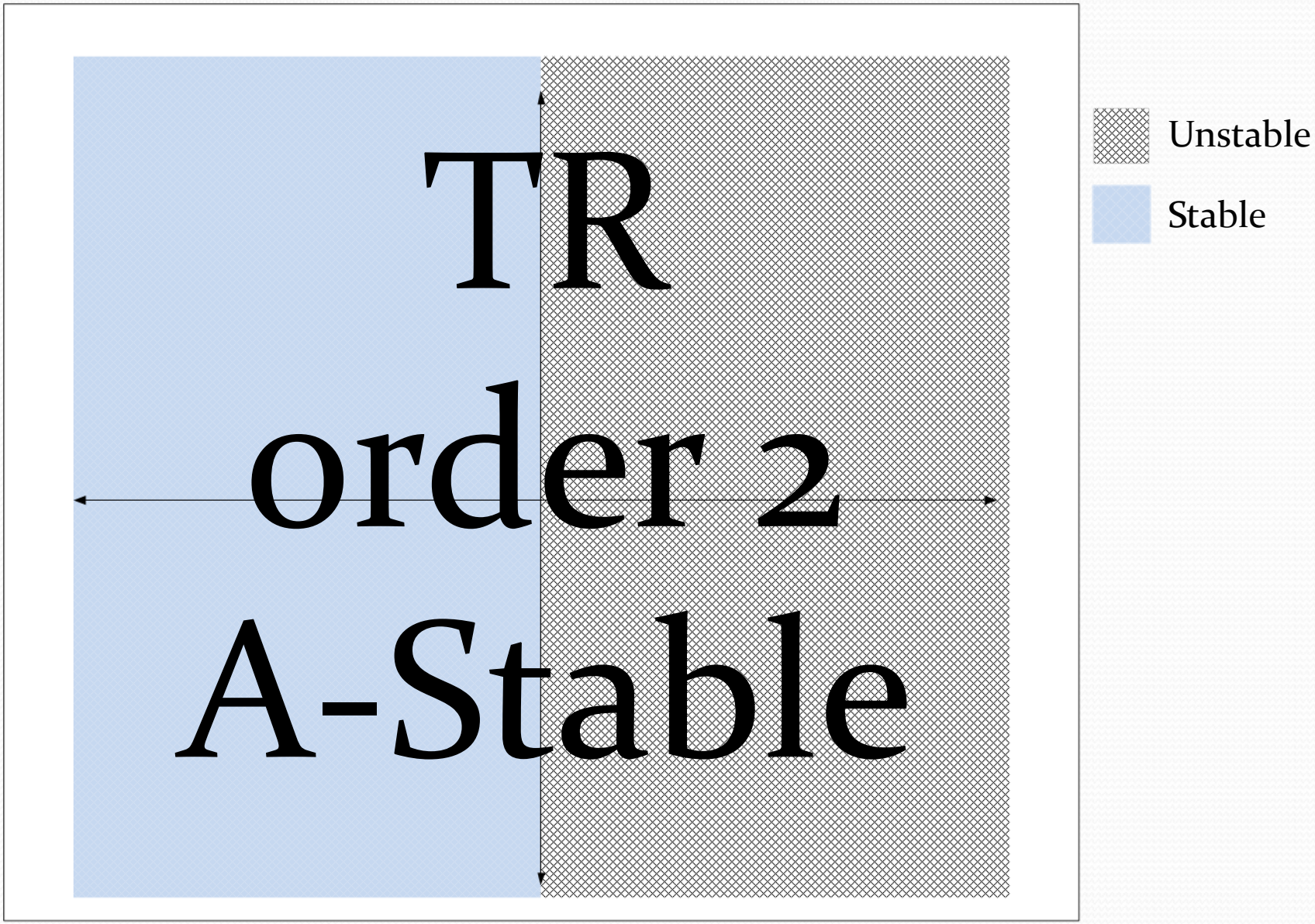
Stability in DE solvers

Laplace s-domain



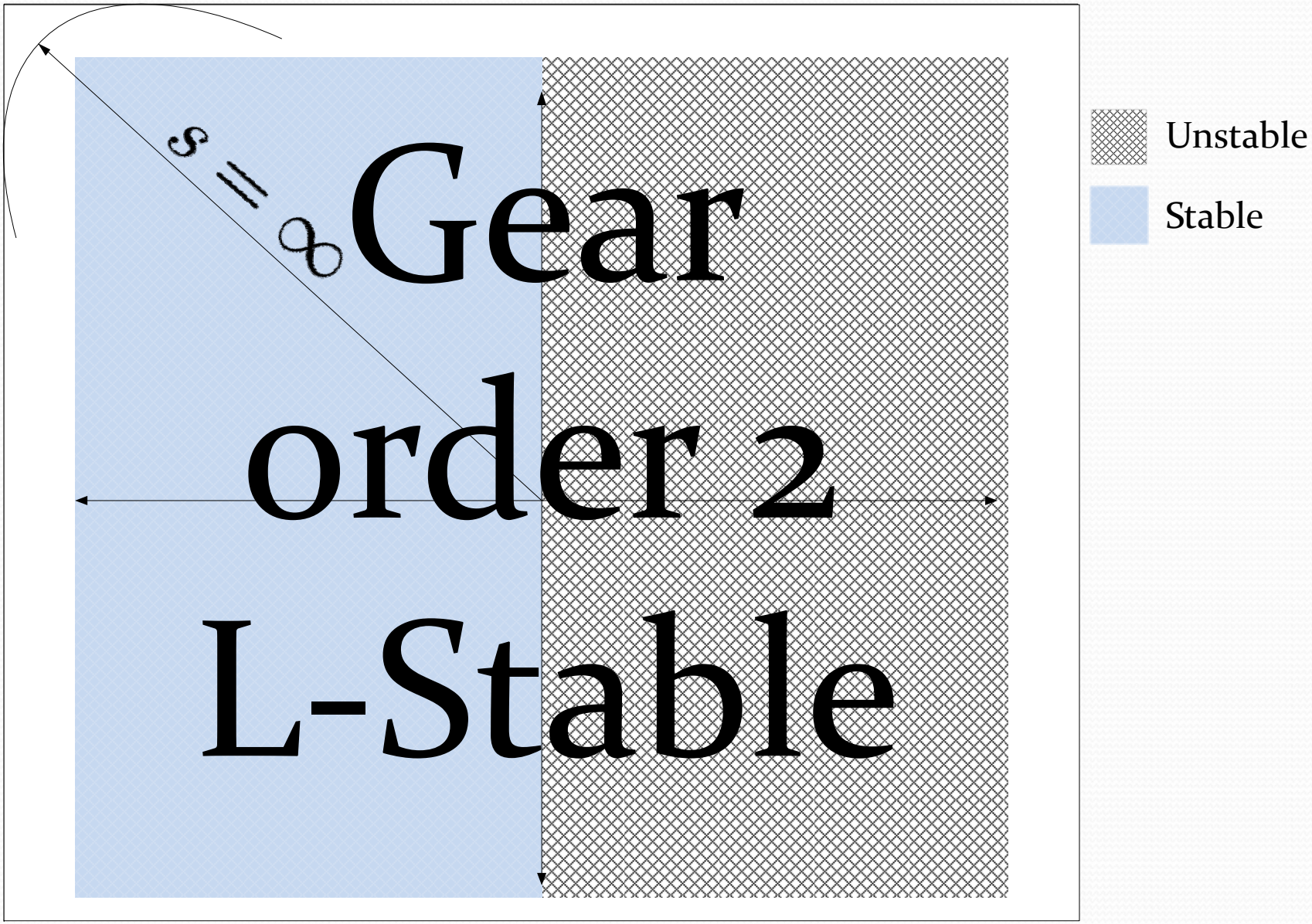
What kind of Stability in SPICE LMS

Laplace s-domain



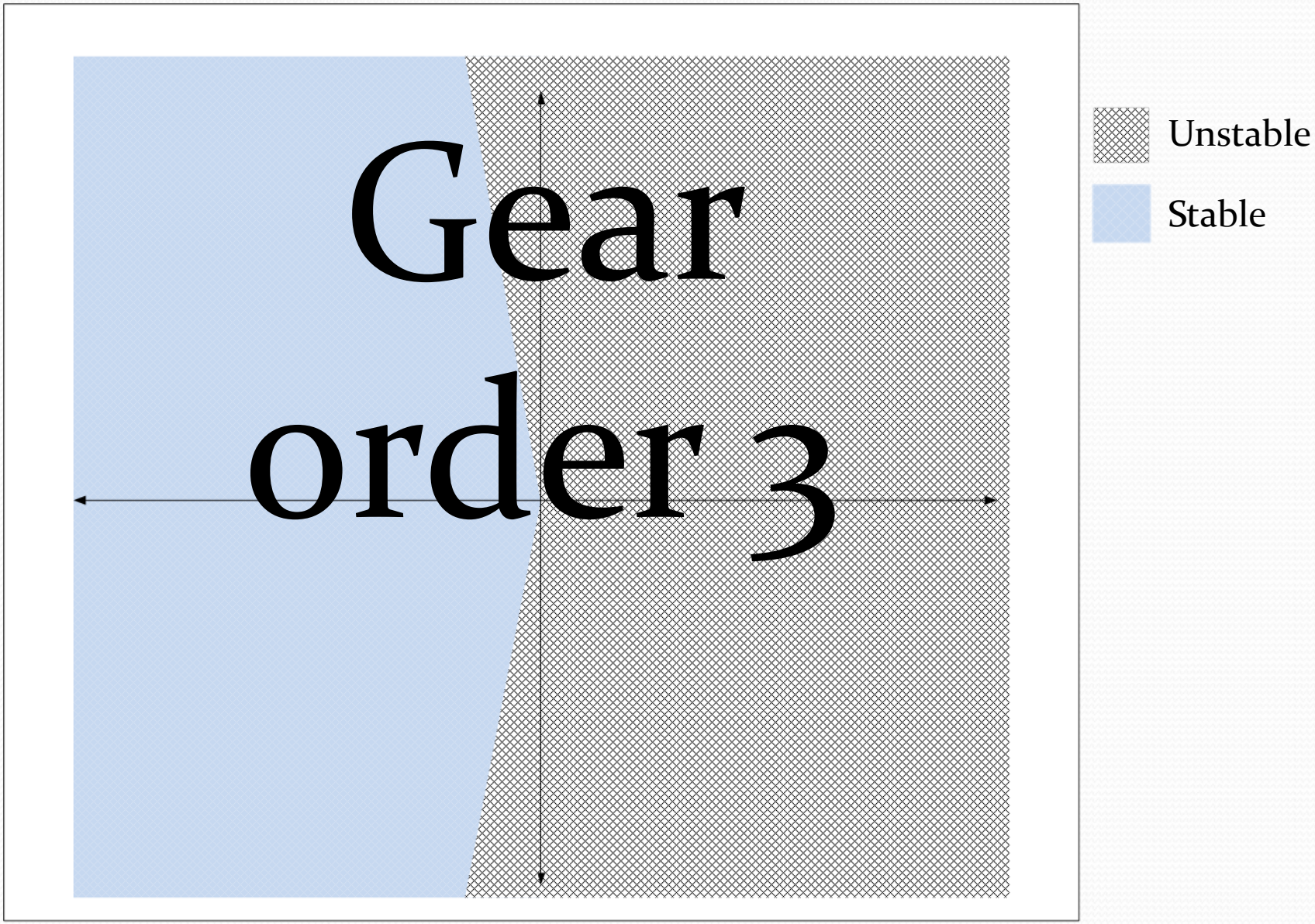
What kind of Stability in SPICE LMS

Laplace s-domain



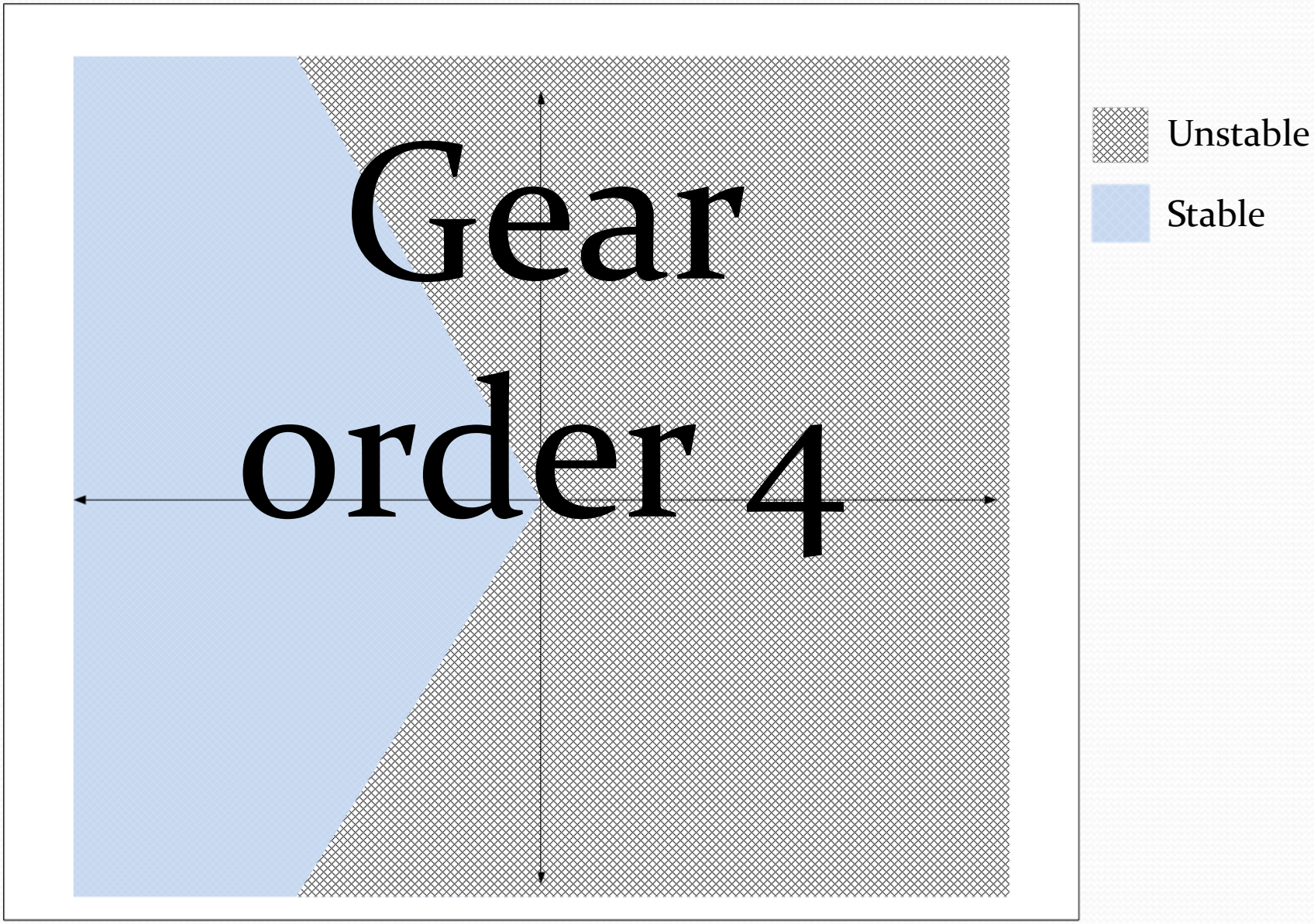
What kind of Stability in SPICE LMS

Laplace s-domain



What kind of Stability in SPICE LMS

Laplace s-domain



The troubles with classical SPICE

- **The Dahlquist Barriers:** **it was proved theoretically that LMS are incapable of high-order without losing stability.**

Classical SPICE LMS

High-Order

The Highest order of absolutely stable LMS method is 2!!

Agenda

- The troubles with classical SPICE
  - The dilemma of Order and Stability
- A-stable and L-stable High-order SPICE (HiSPICE), based on Obreshkov methods
  - Background, Concept, Advantages
- The Multi-Core Hi-SPICE
- Simulation Results
- Conclusion

The Proposed HiSPICE Algorithm

A background on previous approaches to HiSPICE

A linear circuit with N unknown waveforms is represented by

$$C\frac{\mathrm{d}x(t)}{\mathrm{d}t} + Gx(t) = u(t)$$

- C and G are N x N matrices
- $u(t)$ are the independent sources

Assumed Known

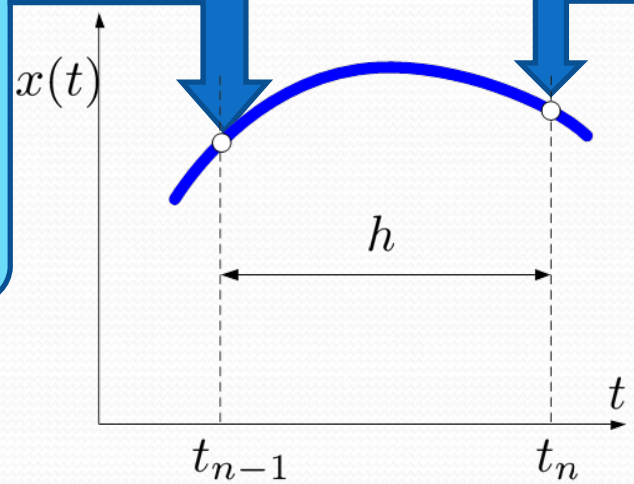
$$x(t_{n-1}) \equiv x_{n-1}$$

$$h\left.\frac{\mathrm{d}x(t)}{\mathrm{d}t}\right|_{t_{n-1}} \equiv x_{n-1}^{(1)}$$

$$\vdots$$

$$h^i\left.\frac{\mathrm{d}^i x(t)}{\mathrm{d}t^i}\right|_{t_{n-1}} \equiv x_{n-1}^{(i)}$$

Time Stepping



?

Need to be computed

$$x(t_n) \equiv x_n$$

Added Unknowns

$$h\left.\frac{\mathrm{d}x(t)}{\mathrm{d}t}\right|_{t_n} \equiv x_n^{(1)}$$

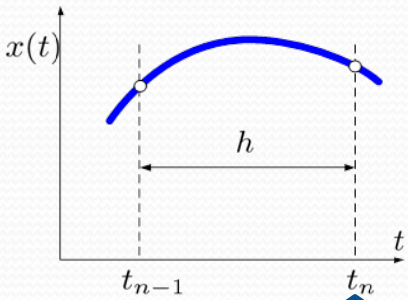
$$\vdots$$

$$h^j\left.\frac{\mathrm{d}^j x(t)}{\mathrm{d}t^j}\right|_{t_n} \equiv x_n^{(j)}$$

The Proposed HiSPICE Algorithm

A background on previous approaches to HiSPICE

At each time step, we will need to solve an enlarged system of algebraic equations: *j* times the size of the original system (*Nj*)

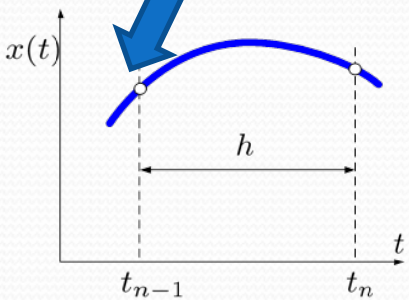


unknowns $\tilde{x}_n \in \mathbb{R}^{Nj}$

$$\left(\tilde{C} + \tilde{G}\right)\tilde{x}_n = \tilde{u}_n$$

Obtained from $u(t)$, its derivatives, and the past time point

$$\tilde{C}, \tilde{G} \in \mathbb{R}^{Nj \times Nj}$$

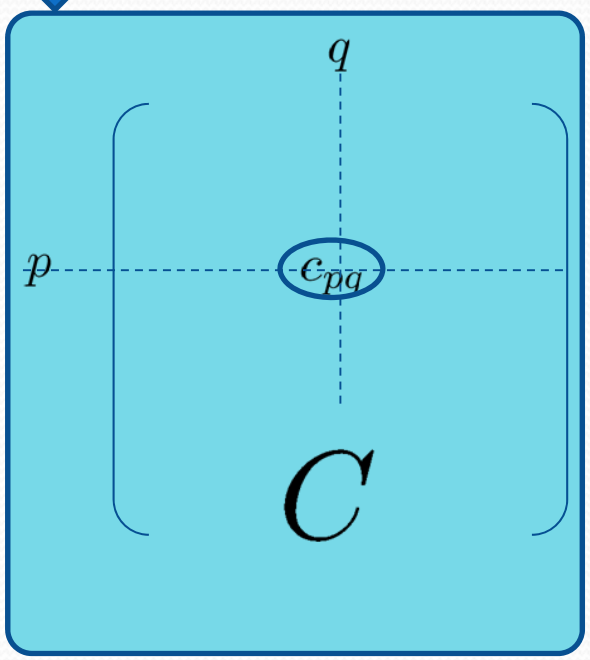


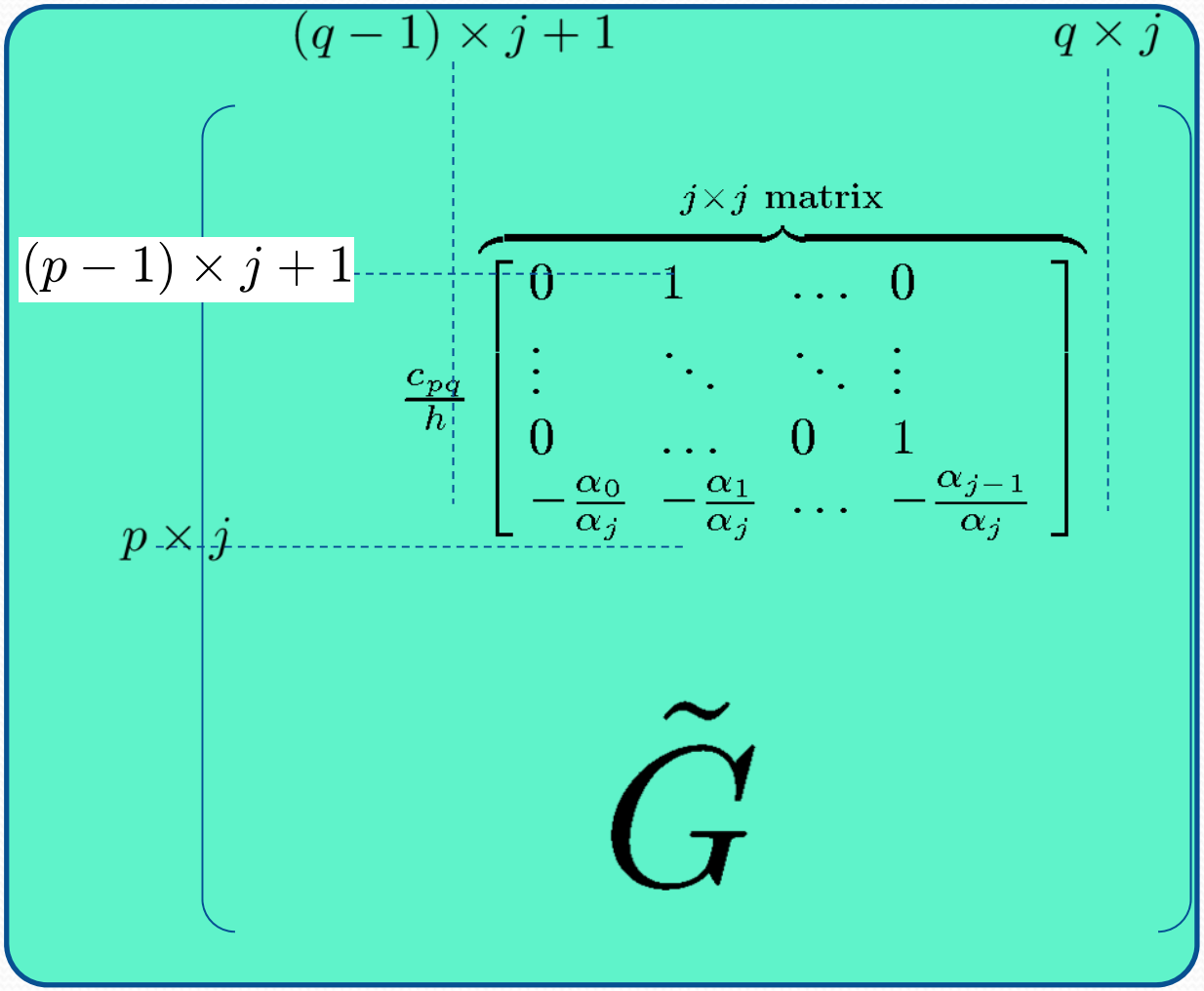
The Proposed HiSPICE Algorithm

A background on previous approaches to HiSPICE

The block structure of the augmented matrices $\tilde{C}, \tilde{G} \in \mathbb{R}^{Nj \times Nj}$

$$C\frac{\mathrm{d}x(t)}{\mathrm{d}t} + Gx(t) = u(t)$$





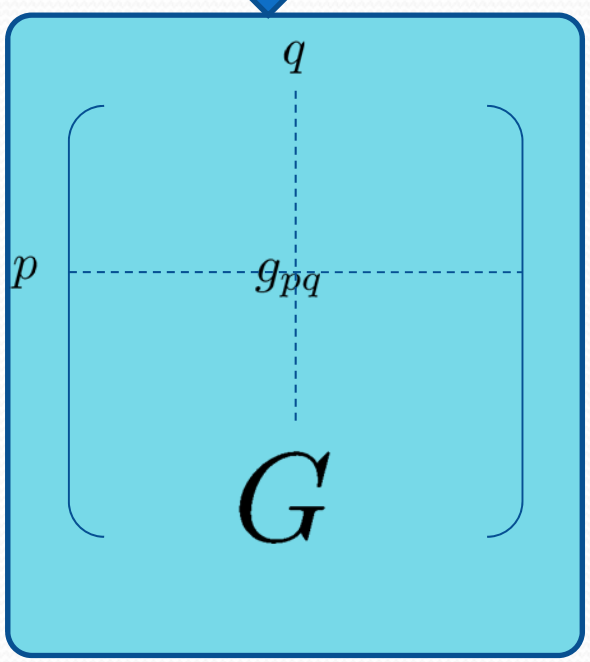
$$\alpha_k = (-1)^{(j)} \frac{(i+j-k)!j!}{i!(i+j)!(j-k)!}$$

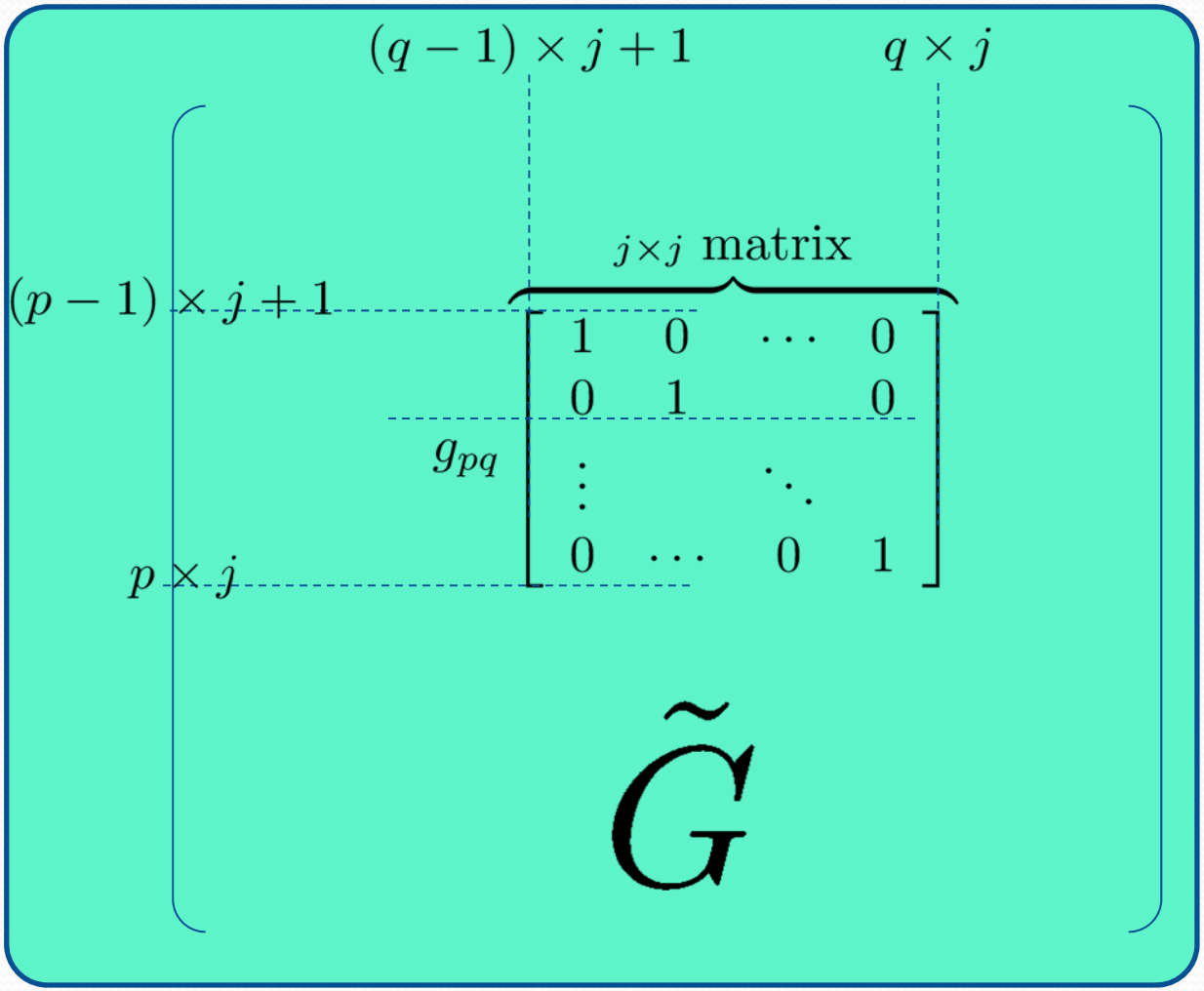
The Proposed HiSPICE Algorithm

A background on previous approaches to HiSPICE

The block structure of the augmented matrices $\tilde{C}, \tilde{G} \in \mathbb{R}^{Nj \times Nj}$

$$C\frac{\mathrm{d}x(t)}{\mathrm{d}t} + Gx(t) = u(t)$$



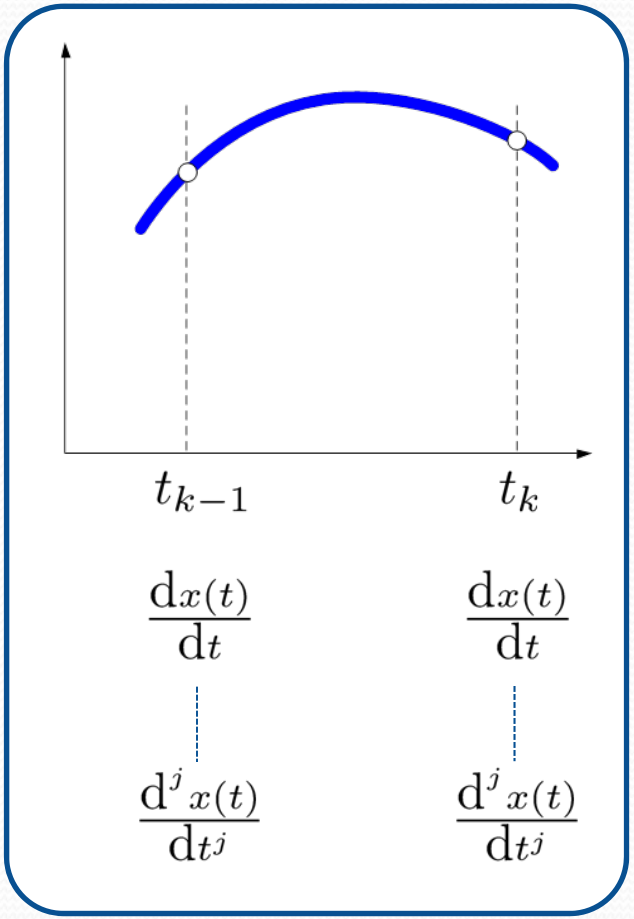


A-Stable High-order SPICE (HiSPICE):

The main concept

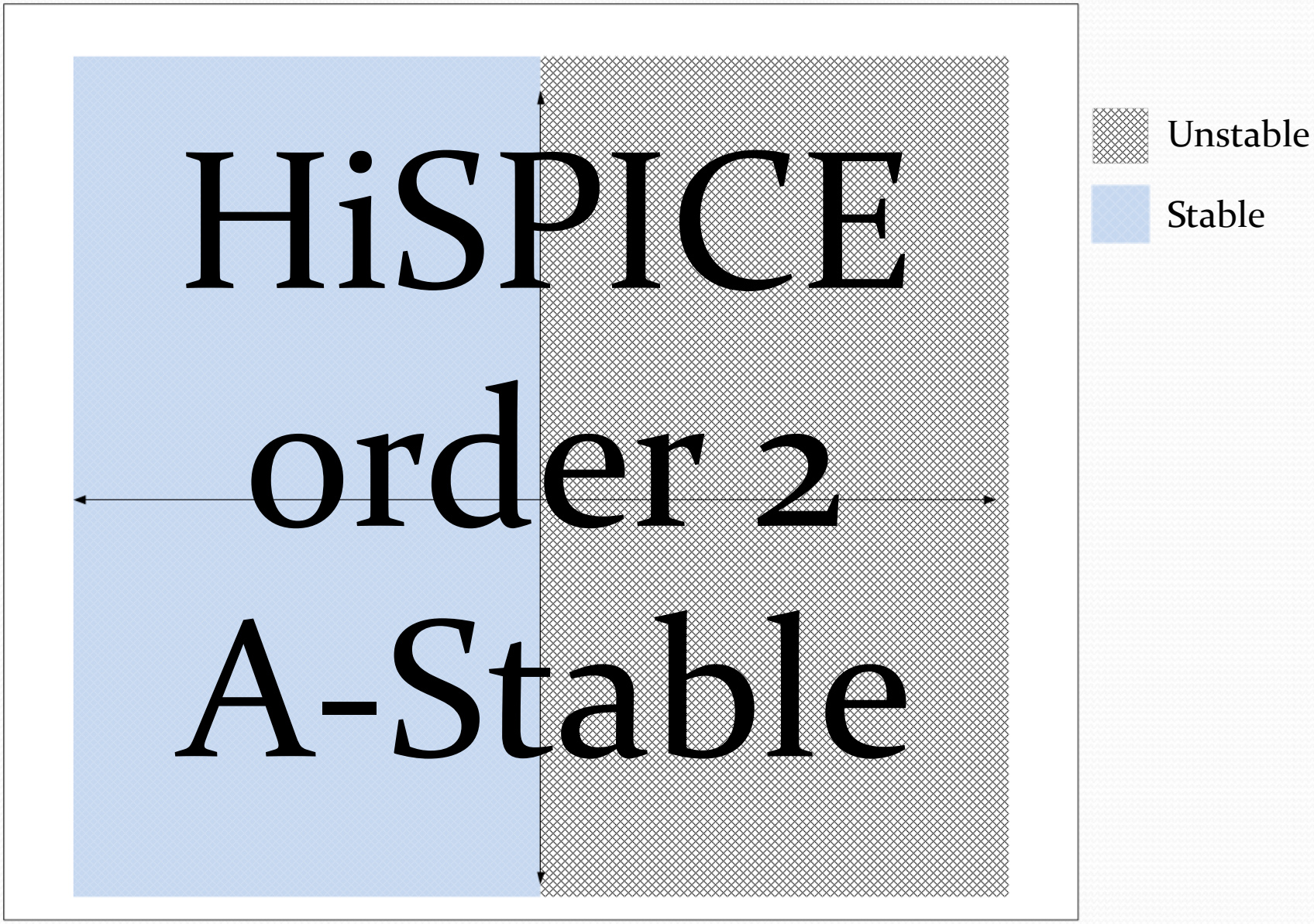
- Order? $i + j$

- Stability?
  - Necessary and sufficient condition for A-stability is $j - 2 \leq i < j$
  - Necessary and sufficient condition for L-stability is $j - 2 \leq i < j$

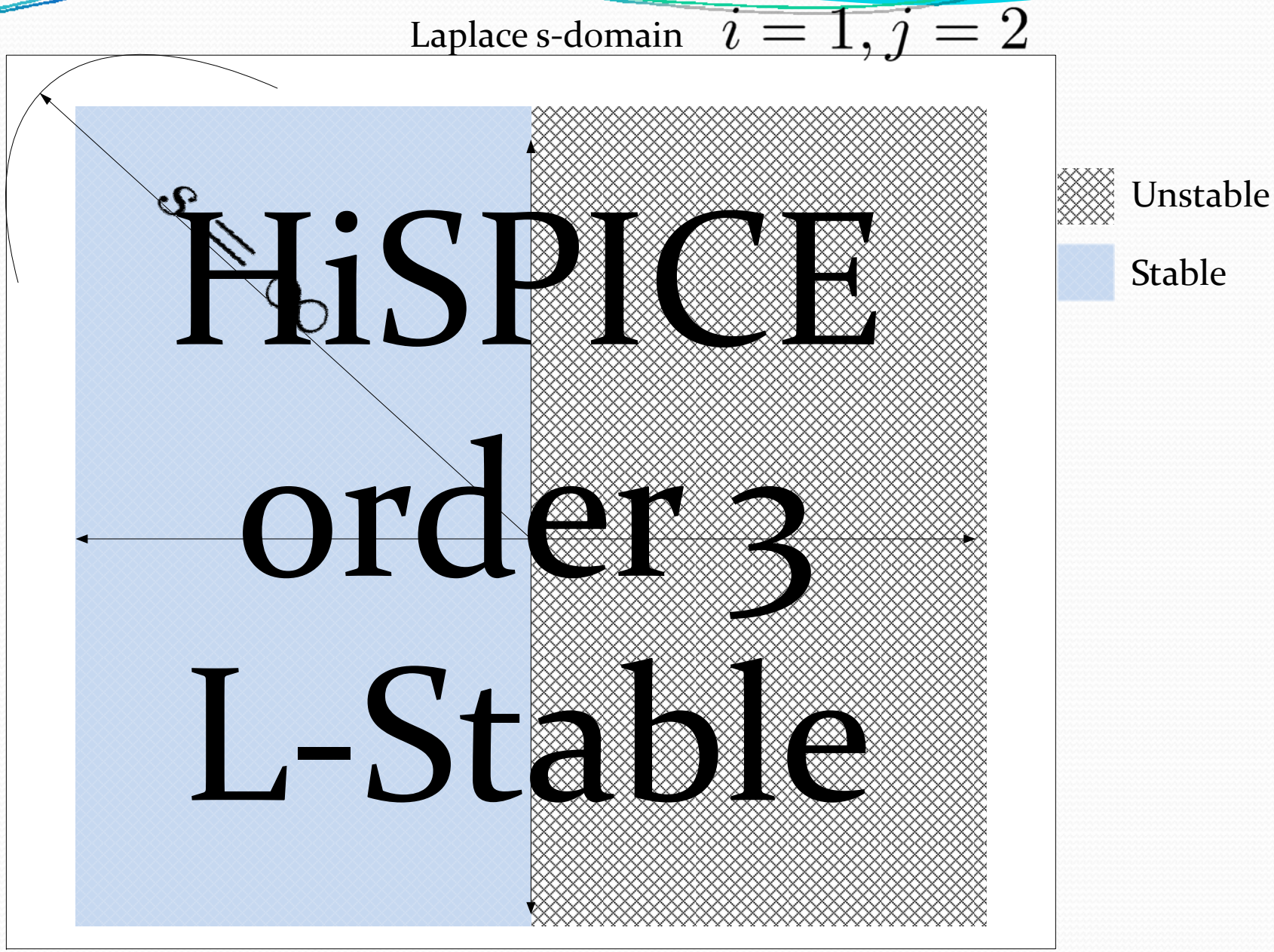


What kind of Stability in HiSPICE

Laplace s-domain $i = j = 1$

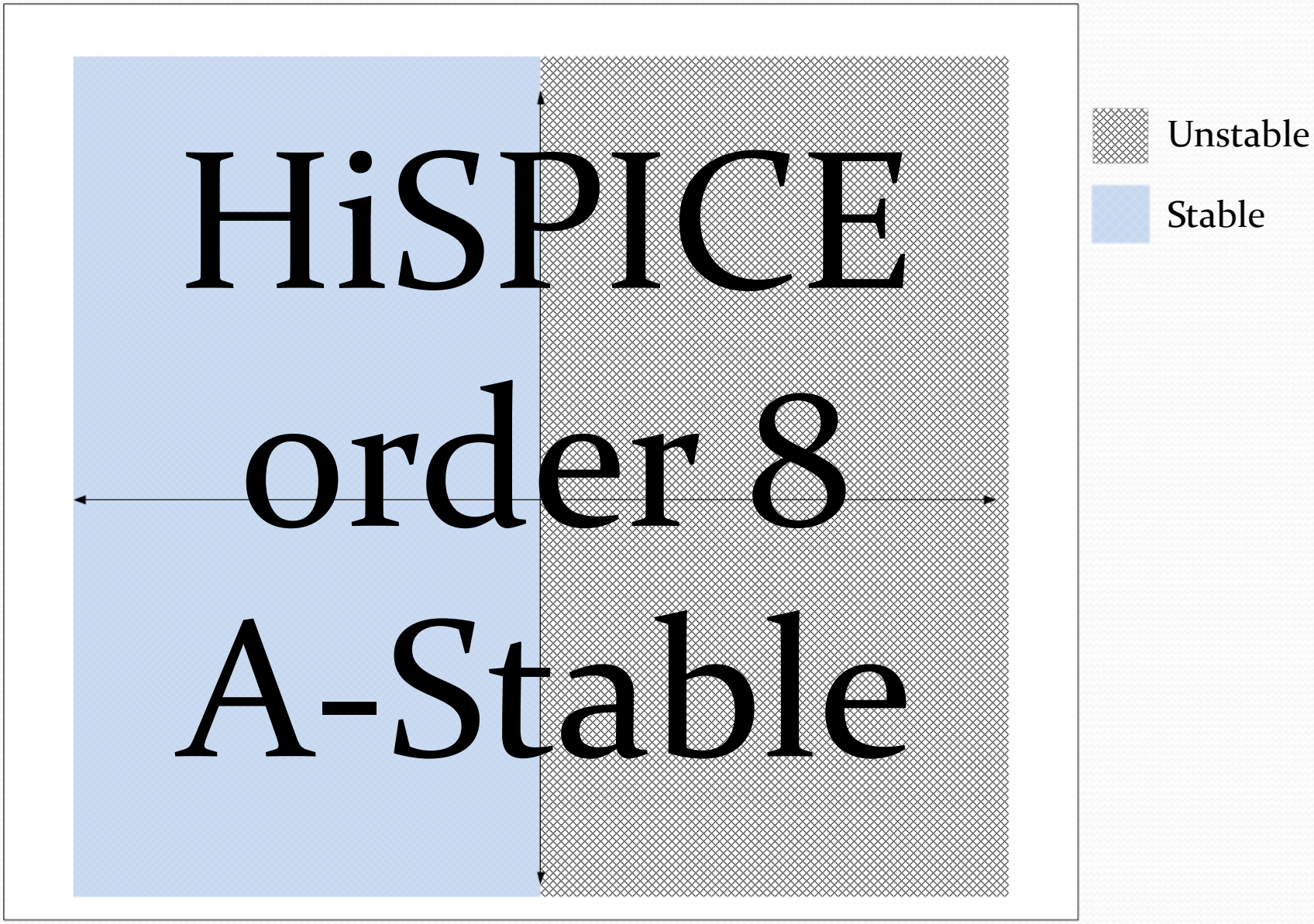


Stability in DE solvers



What kind of Stability in HiSPICE

Laplace s-domain $i = j = 4$



A-Stable High-order SPICE (HiSPICE):

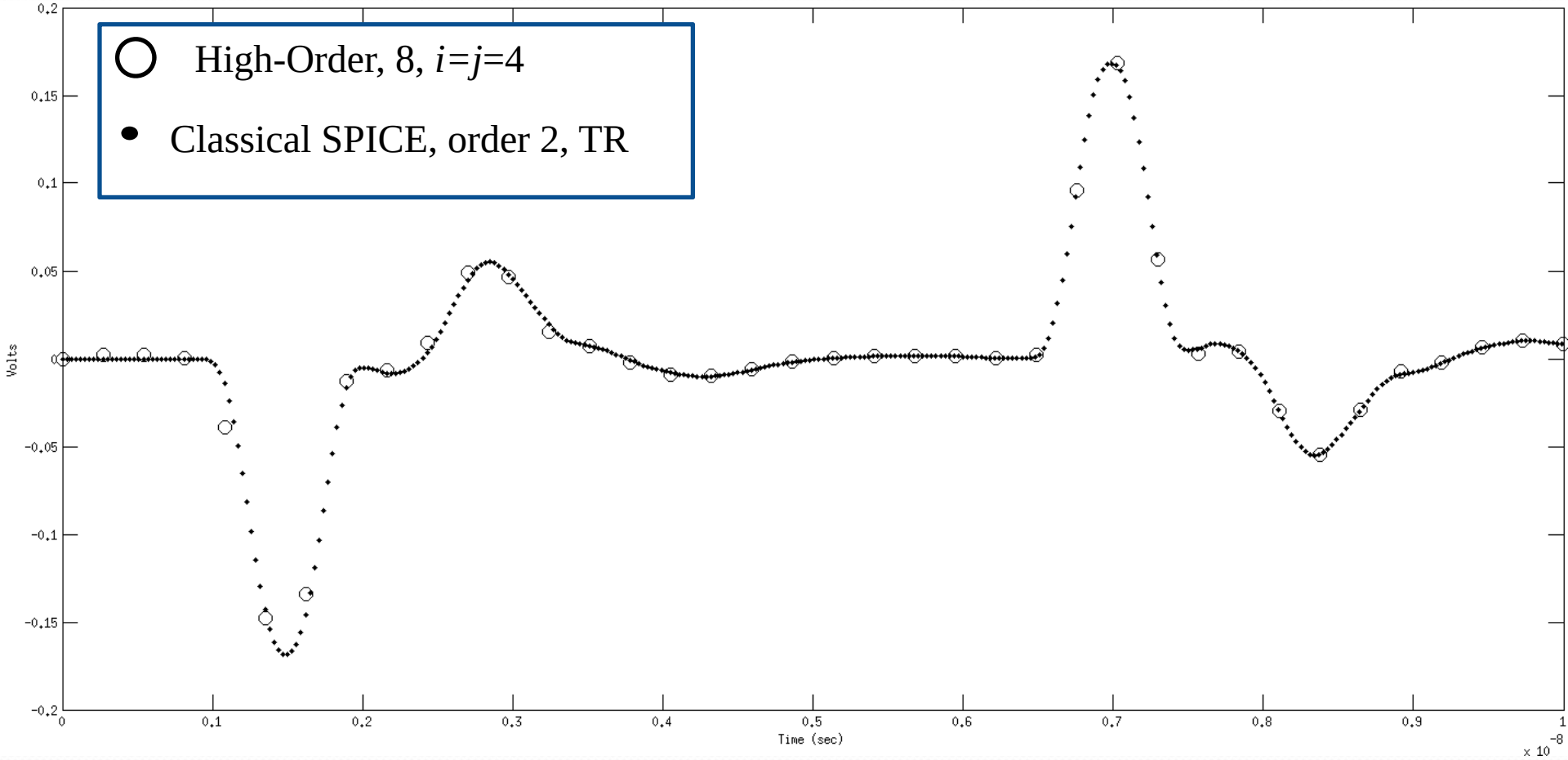
A sample of the results

Order	# time points	Speedup	Error
Classical SPICE (2)	1771	-	< 0.001
HiSPICE (4)	**101**	**6.5**	< 0.001
HiSPICE (6)	**36**	**10.3**	< 0.001

Order	# time points	Speedup	Error
Classical SPICE (2)	5751	-	< 0.0001
HiSPICE (4)	**185**	**12**	< 0.0001
HiSPICE(6)	**59**	**21.4**	< 0.0001

A-Stable High-order SPICE (HiSPICE):

A sample of the results



A-Stable High-order SPICE (HiSPICE):

- To Summarize
  - At the expense of **solving a larger system** of equations, compared to the low-order method, we are able to save on the CPU time significantly, in addition to solving the difficult problem of stability.
- This is previous work that was published [1],[2]

1) Emad Gad, Michel S. Nakhla, Ramachandra Achar, Yinghong Zhou: A-Stable and L-Stable High-Order Integration Methods for Solving Stiff Differential Equations. ***IEEE Trans. on CAD of Integrated Circuits and Systems 28(9): 1359-1372 (2009)***.
2) Yinghong Zhou, Emad Gad, Michel S. Nakhla, Ramachandra Achar: Structural Characterization and Efficient Implementation Techniques for A-Stable High-Order Integration Methods. ***IEEE Trans. on CAD of Integrated Circuits and Systems 31(1): 101-108 (2012)***.

This Work

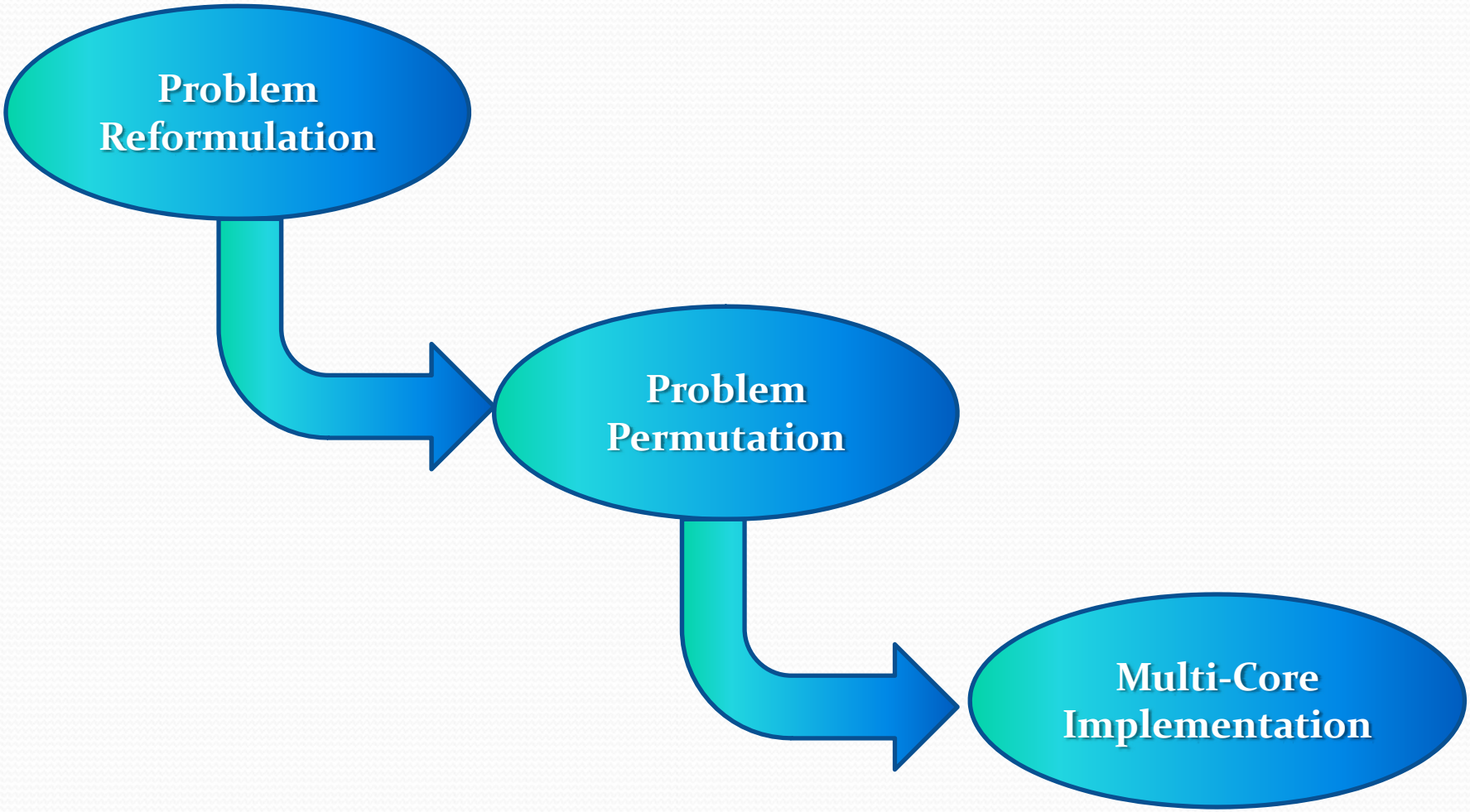
Presents a new way to reduce the computational cost per time step for linear circuits.

Agenda

- The troubles with classical SPICE
  - The dilemma of Order and Stability
- A-stable and L-stable High-order SPICE (HiSPICE), based on Obreshkov methods
  - Background, Concept, Advantages
- The Proposed MultiCore HiSPICE
- Simulation Results
- Conclusion

The Proposed HiSPICE Algorithm

Proposed Multi-Core approach to HiSPICE



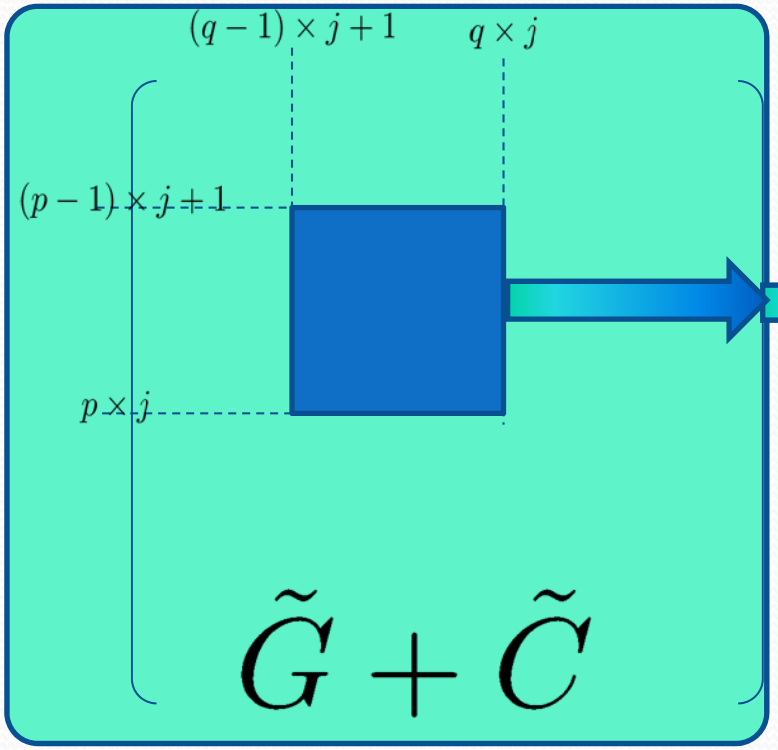
The Proposed HiSPICE Algorithm

Proposed Multi-Core approach to HiSPICE

Problem Reformulation

Problem Permutation

Multi-Core Implementation



$Nj \times Nj$ matrix

$$V\left(g_{pq}I + c_{pq}\lambda\right)V^{-1}$$

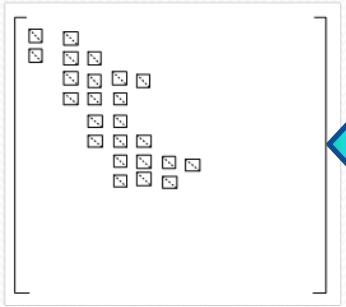
- ***V* is common to all the blocks in the matrix**
- ***λ* is a diagonal matrix**

$$\tilde{G}+\tilde{C} = \hat{V}\tilde{J}\hat{V}^{-1}$$

$$\hat{V} = \begin{bmatrix} V & 0 & \cdots & 0 \\ 0 & V & & 0 \\ \vdots & & \ddots & \\ 0 & \cdots & 0 & V \end{bmatrix}$$

Block Diagonal

$\tilde{J} =$



Diagonal blocks

The Proposed HiSPICE Algorithm

Proposed Multi-Core approach to HiSPICE

Problem Reformulation

Problem Permutation

Multi-Core Implementation

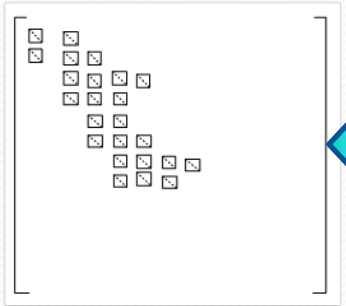
$$\left(\tilde{C} + \tilde{G}\right) \tilde{x}_n = \tilde{u}_n$$

Using a change of variables

$$\tilde{\rho}_n \leftarrow \hat{V}^{-1} \tilde{x}_n$$

$$\tilde{J} \tilde{\rho}_n = \hat{V}^{-1} \tilde{u}_n$$

$\tilde{J} =$



Diagonal blocks

The Proposed HiSPICE Algorithm

Proposed Multi-Core approach to HiSPICE

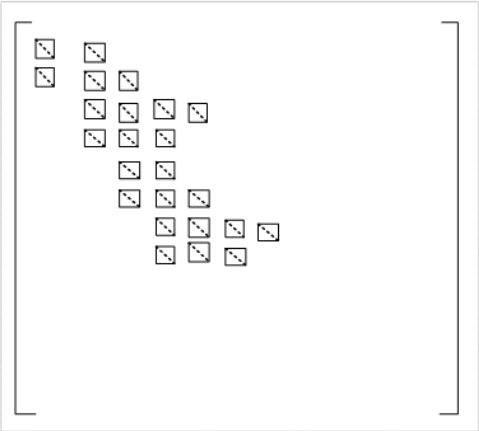
Problem Reformulation

Problem Permutation

Multi-Core Implementation

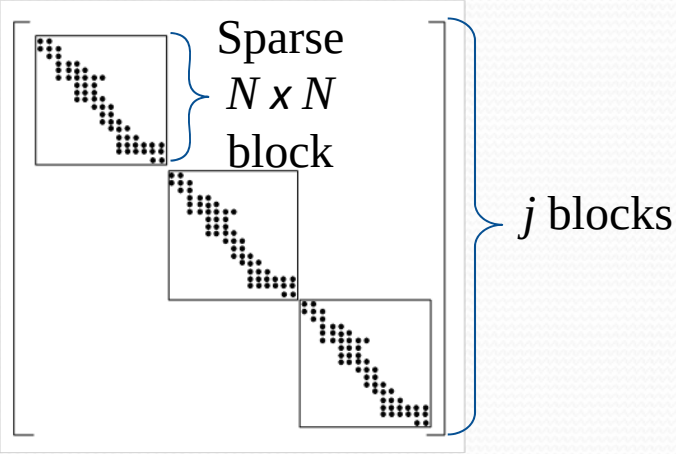
$$\tilde{J}\tilde{\rho}_n = \hat{V}^{-1}\tilde{u}_n$$

$\tilde{J} =$



Diagonal blocks

Problem Permutation



Block Diagonal

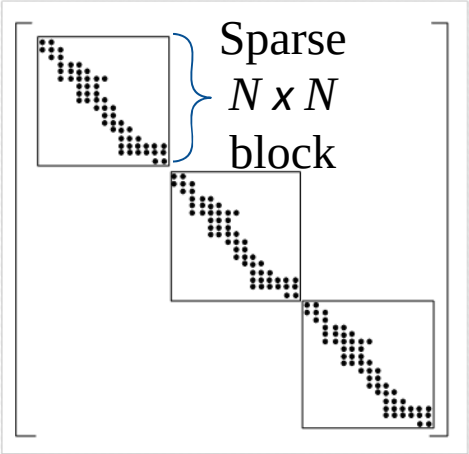
The Proposed HiSPICE Algorithm

Proposed Multi-Core approach to HiSPICE

Problem Reformulation

Problem Permutation

Multi-Core Implementation



Block Diagonal

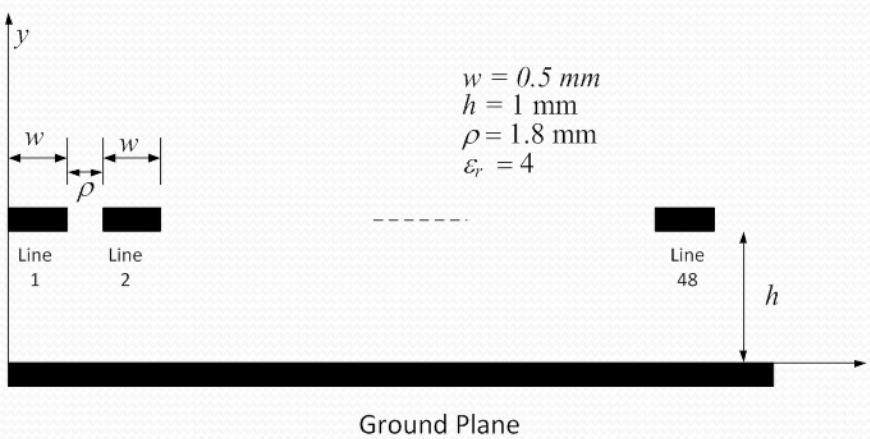
- **Blocks can be factorized using separate threads executed in parallel**
- **All the blocks have identical structures**
  - Sparse ordering is done only once
- **Some of the blocks are complex, some are real**
- **Complex blocks are conjugate pairs**
  - Needs complex factorization libraries (klu_zl_xxx)
  - Conjugate-ness is exploited to save computations

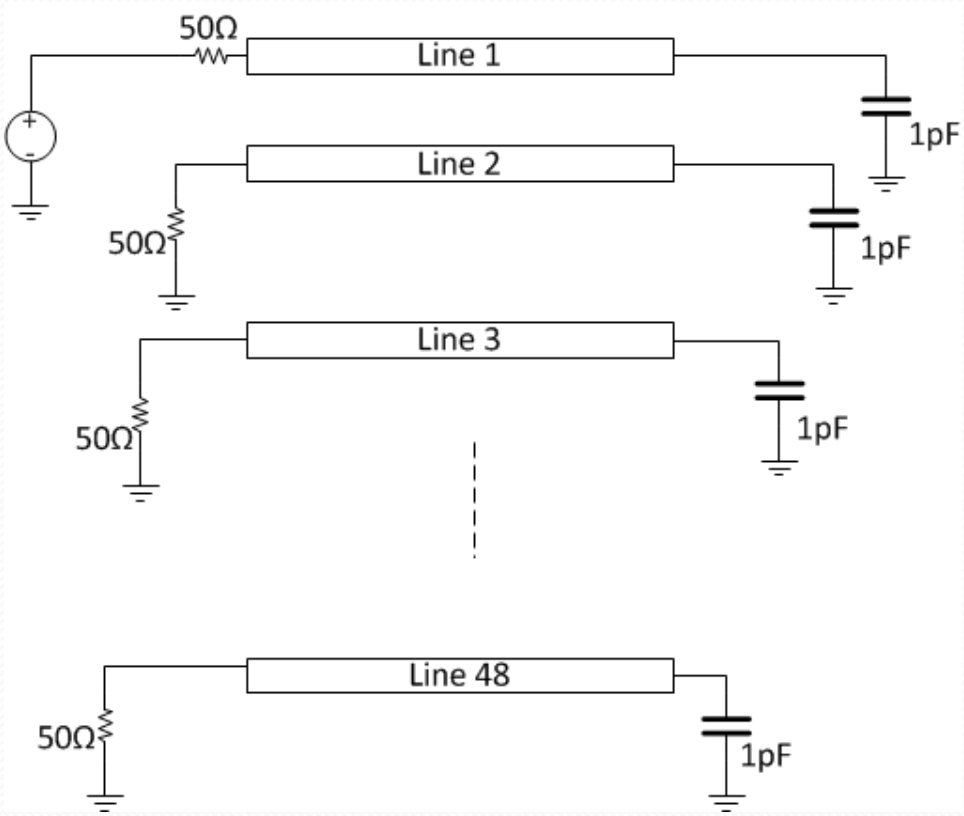
Agenda

- The troubles with classical SPICE
  - The dilemma of Order and Stability
- A-stable and L-stable High-order SPICE (HiSPICE), based on Obreshkov methods
  - Background, Concept, Advantages
- The Proposed MultiCore HiSPICE
- Simulation Results
- Conclusion

Simulation Results

A circuit with 48 coupled conductor transmission lines

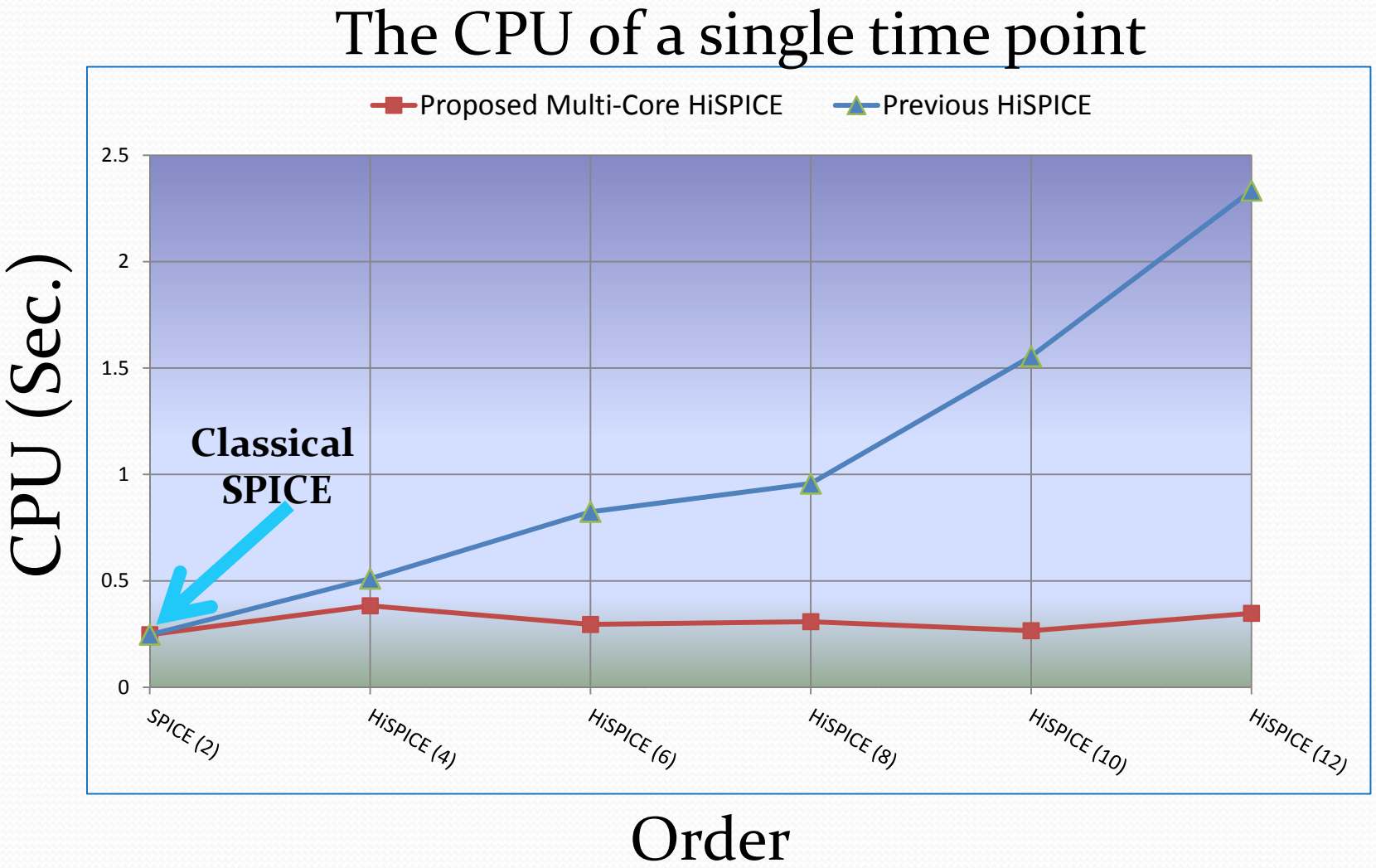




Simulation Results

Order	CPU (seconds)	Speedup
Classical SPICE (2)	145.17	-
HiSPICE (10)	11.20	**13**

Simulation Results



Conclusion

- High-Order A-stable and L-stable SPICE (HiSPICE)
  - Taking advantage of multicore architecture
- Cost per time point:
  - Independent from the order
  - Comparable to Low Order Cost
- The algorithm is developed for linear circuits arising in package models.

Thank You