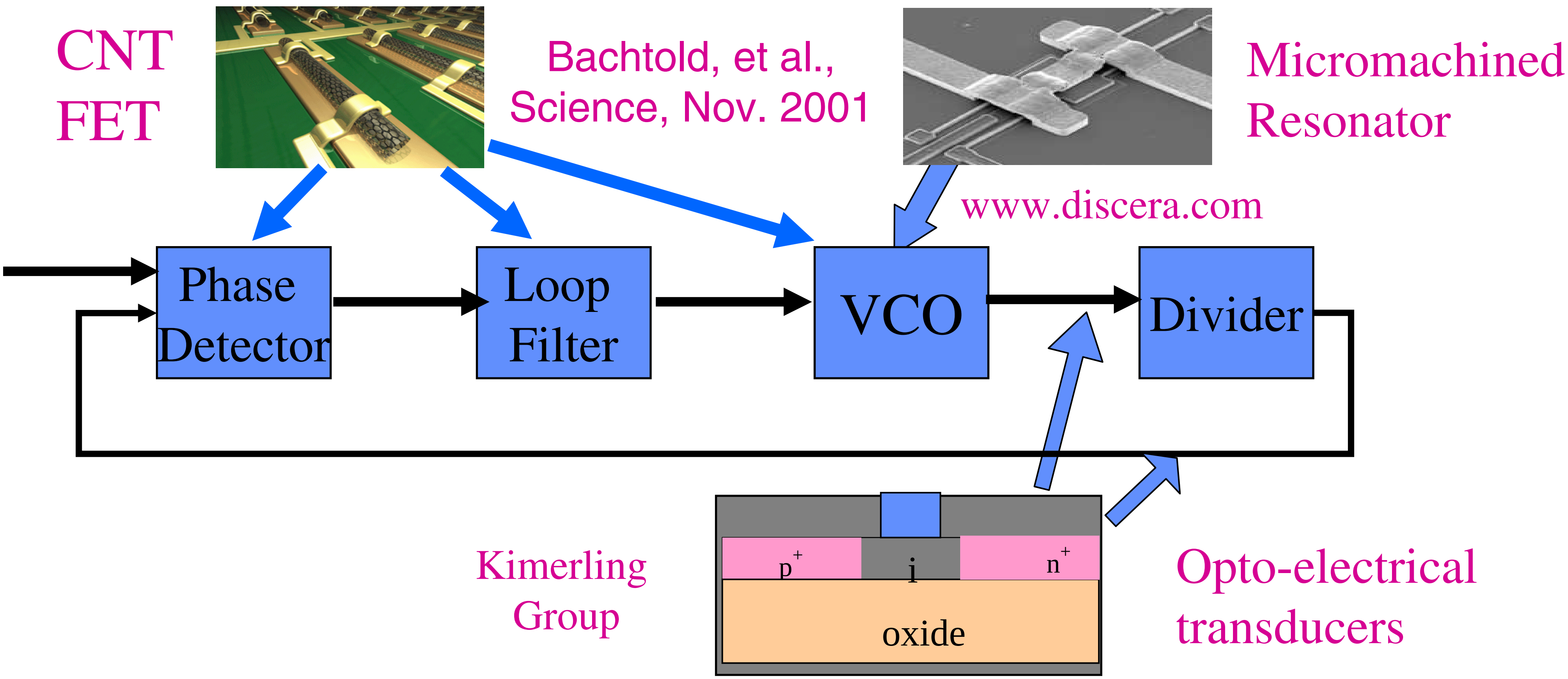
A Personal Perspective on Parameterized and Nonlinear MOR

J. White

Slides thanks to D. Luca, M. Reichelt, M. Rewienski, D. Vasilyev

A Multitechnology Phase-Locked Loop



Evaluating the New Technology

- What is system performance (capture, lock, noise, etc)?
- What is the impact of modifying technology parameters?
- How tight must manufacturing tolerances be?

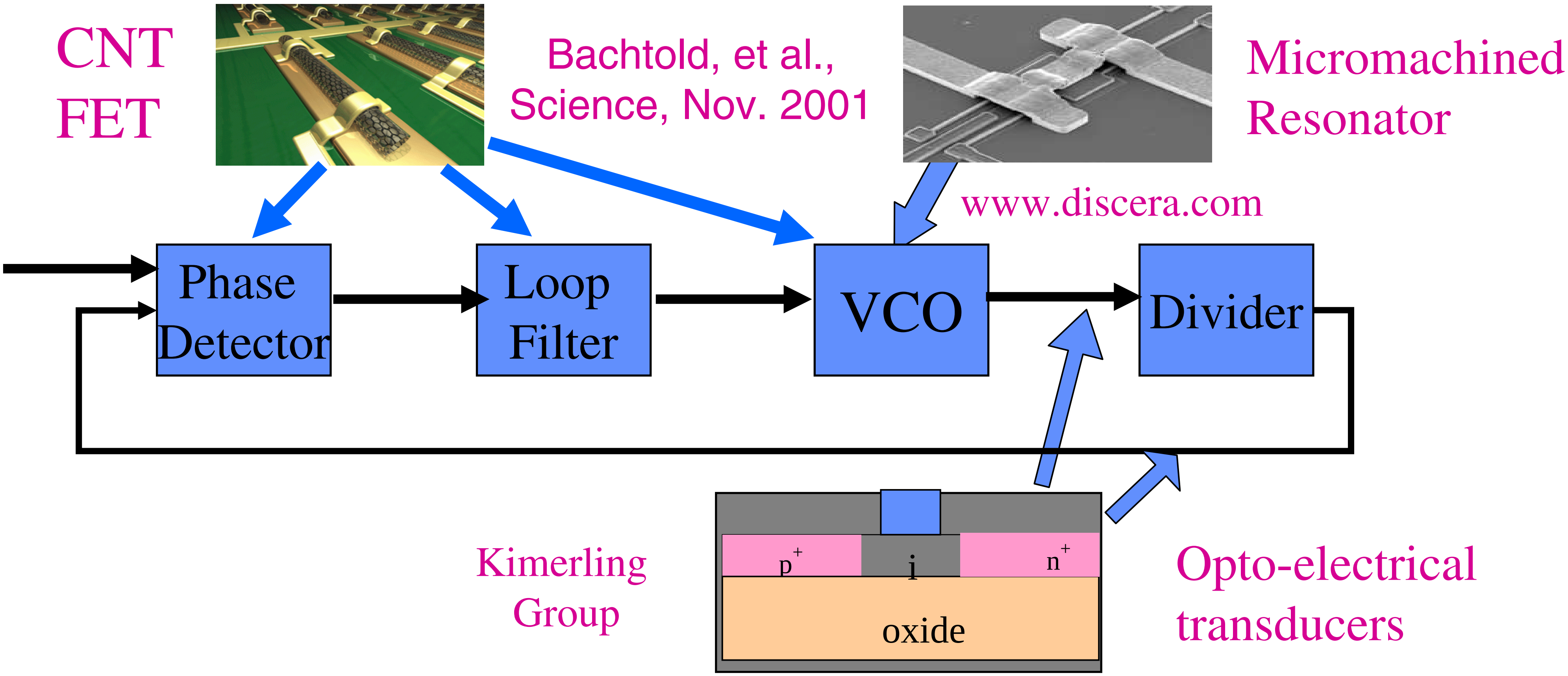
CAD for Diverse IC Technology

- Initial Assessment
  - **What is possible with a combination of technology?**
  - **Will new technology improve SYSTEM performance?**
  - **Requires a rough “optimization” step!**
- System Performance optimization
  - **Assess intra and inter technology trade-offs .**
  - **What is the impact of fabrication decisions?**
  - **Automate Analysis and Synthesis/Optimization**
- Manufacturability/Yield optimization
  - **Optimize design considering variations!**

Need to Assess and Optimize System Performance

- Hierarchical Simulation
  - Encapsulate the physics.
  - Automatically move between hierarchical levels.
  - Approach must apply given diverse technology.
- Hooks for Synthesis/Optimization
  - Compute Performance Sensitivities to:
    - Fabrication decisions
    - Layout modifications
    - Architectural Changes.
- Manufacturability/Yield
  - Optimize design considering variations!

Goal: Optimize Technology for the Application



Need to simulate ENTIRE system with dynamically accurate models for ALL the components

Capture Simulation will require thousands of oscillator cycles

Multiphysics Simulation Approach

- **Circuit**
  - Ordinary differential equation solver

+

- **Carbon Nanotube Transistor**
  - Molecular Dynamics or Atomistic Simulation

+

- **Microresonator**
  - Coupled 3-D Electro-Elasto-Fluidic Simulation

+

- **Optical Transducers**
  - 3-D Coupled Device-Optics Simulator

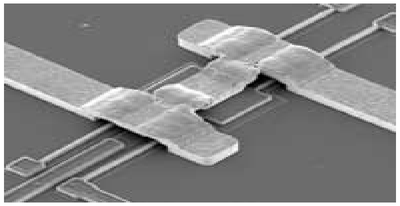
+

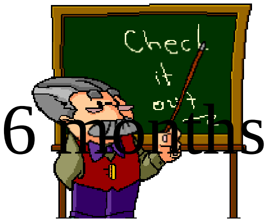
- **Interconnect + Substrate**
  - 3-D Full-Wave Simulation

Capture Simulation of thousands of cycles will never finish!

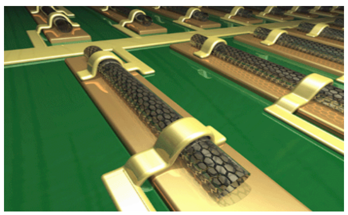
Must Generate Macromodels

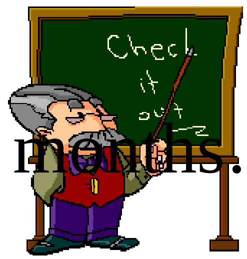
Macromodel Generation Now Done By Hand





6 months…

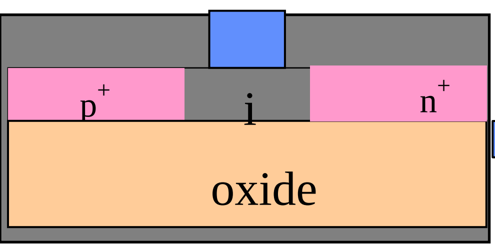


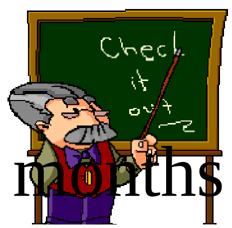


6 months…

$$\frac{dx_r(t)}{dt} = F(x_r(t)) + b_r u(t)$$

$$y(t) = c_r^T x_r(t)$$





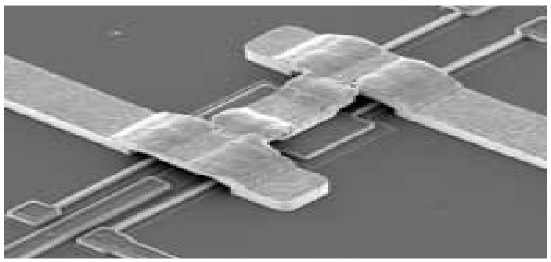
6 months…

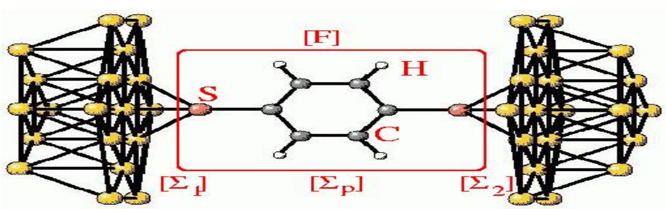
Model for the
System Simulator

Will Never Keep Up With Diverse Technology

The Numerical Macromodeling Paradigm

Generate a Reduced-Order Model Directly from 3-D Geometry and Physics





Lundstrom et al.

Complicated Geometry, Coupled Physics, possibly even statistical

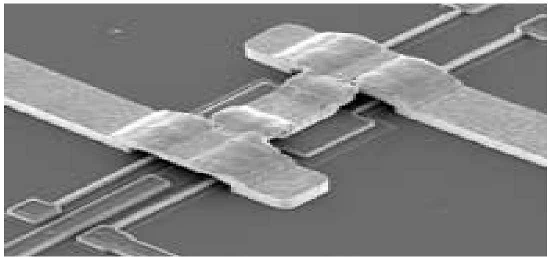
Automatic

$$\frac{dx_r(t)}{dt} = F(x_r(t)) + b_r u(t)$$

$$y(t) = c_r^T x_r(t)$$

Low order state-space model which captures input (u)/output(y) behavior

What's Needed For Numerical Macromodeling



$$\frac{dx_r(t)}{dt} = F(x_r(t)) + b_r u(t)$$

$$y(t) = c_r^T x_r(t)$$

1) Fast Coupled Domain 3-D Solvers

- Fluids, EM Fields, mechanics, Transport
- Must handle ENTIRE Devices!

2) Model-Order Reduction

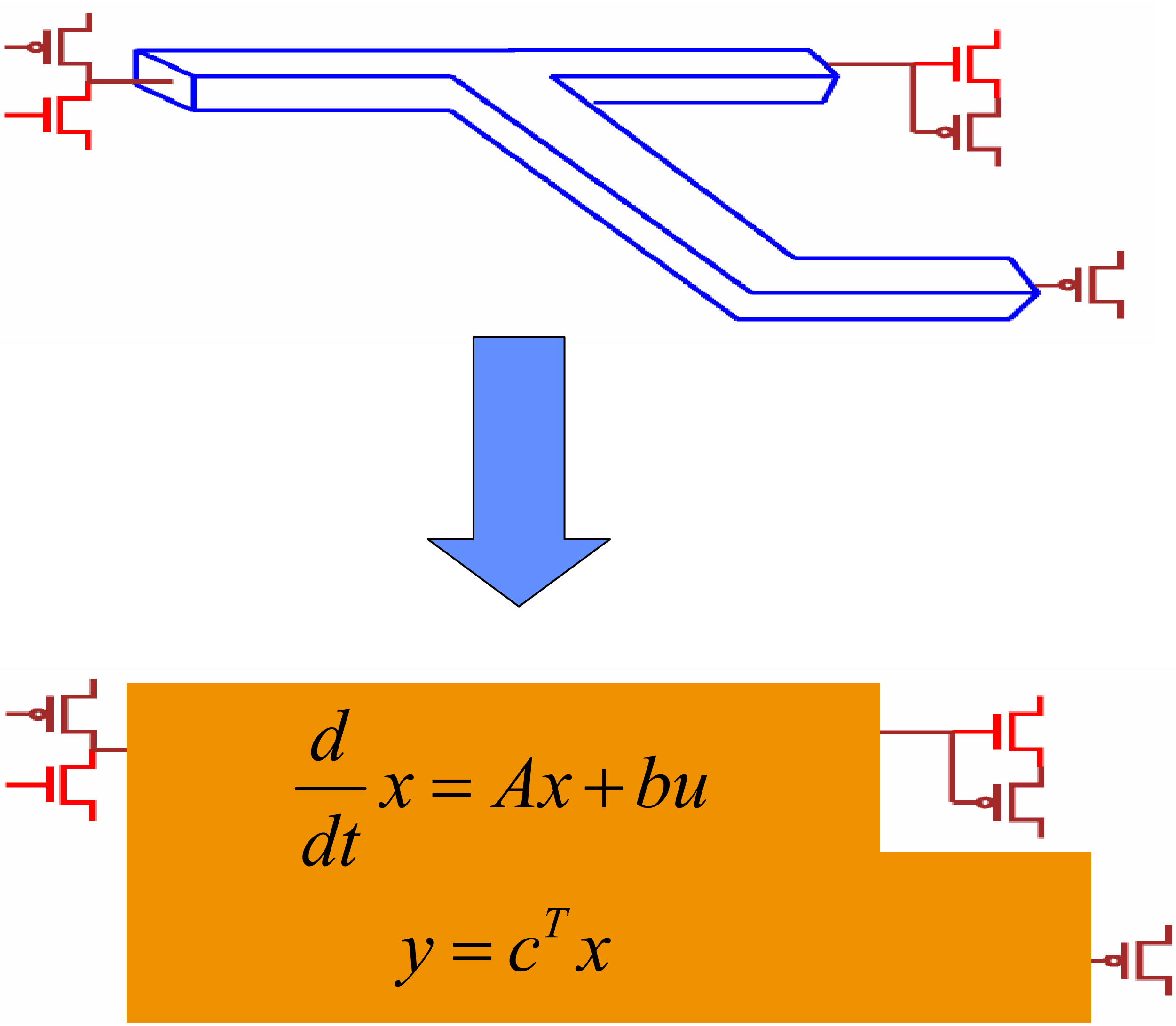
- Start with a Meshed 3-D Structure (>100,000 DOF's)
- Or Start with molecular positions
- Automatic generation of low-order model (<100 DOF's)

3) Hooks for Optimization

- Model Should be parameterized

Where Are We Now?

Linear, Few Port Problem is Getting there.



$$\frac{d}{dt}x = Ax + bu$$

$$y = c^T x$$

- **Fast 3-D E-M Solvers**
  - Multipole, Hierarchical SVD, Precorrected-FFT, Wavelets
- **Efficient MOR**
  - Krylov, Krylov-TBR, Projection methods, Frequency Domain POD, PMTBR, etc
- **Still Issues**
  - Passivity
  - Performance for Distributed Systems

State-Space Description

- **Original Dynamical System - Single Input/Output**

$$\frac{dx(t)}{dt} = \underbrace{A}_{NxN} x(t) + \underbrace{b}_{Nx1} \underbrace{u(t)}_{\substack{scalar\\input}} \qquad \underbrace{y(t)}_{\substack{scalar\\output}} = \underbrace{c}_{Nx1}{}^{T} x(t)$$

- **Reduced Dynamical System q << N, but I/O preserved**

$$\frac{dx_r(t)}{dt} = \underbrace{A_r}_{qxq} x(t) + \underbrace{b_r}_{qx1} \underbrace{u(t)}_{\substack{scalar\\input}} \qquad \underbrace{y_r(t)}_{\substack{scalar\\output}} = \underbrace{c_r}_{qx1}{}^{T} x_r(t)$$

Projection Framework

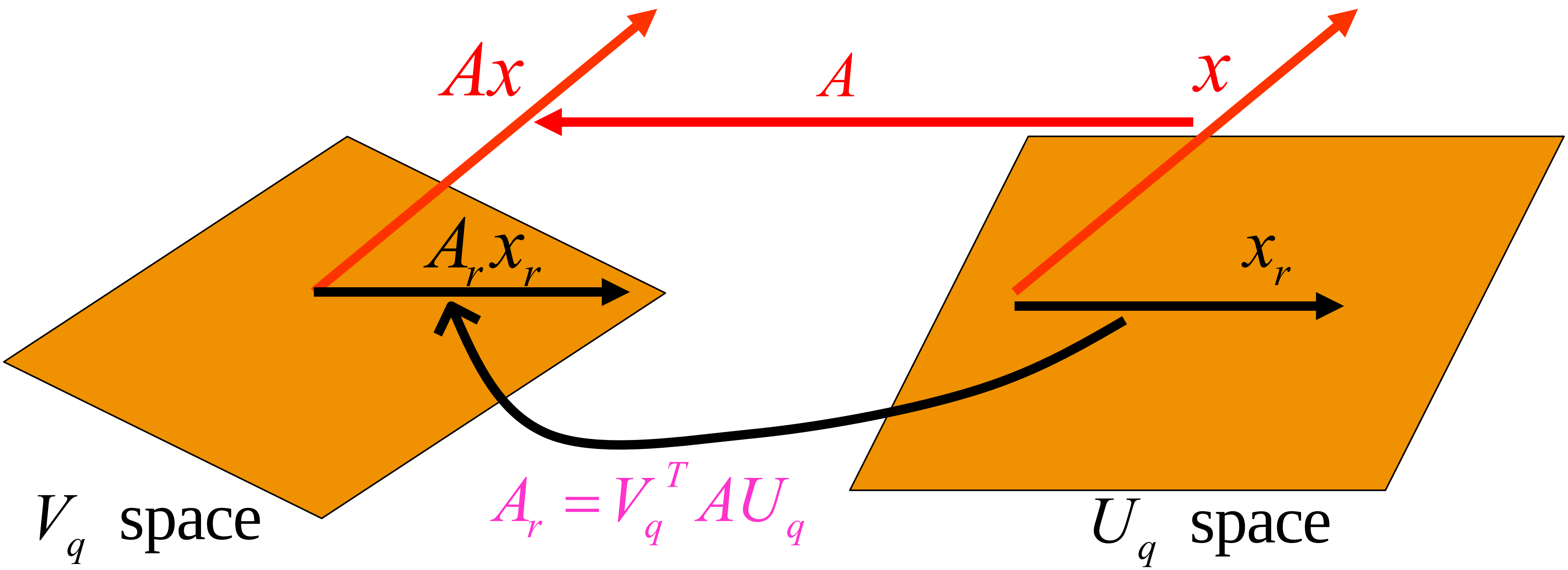
$$\dot{x} = Ax + bu,\ \ y = c^T x \Rightarrow \dot{x}_r = A_r x_r + b_r u,\ \ y_r = c_r^T x$$

<u>Equation Testing</u>

$$V_q^T Ax \approx A_r x_r$$

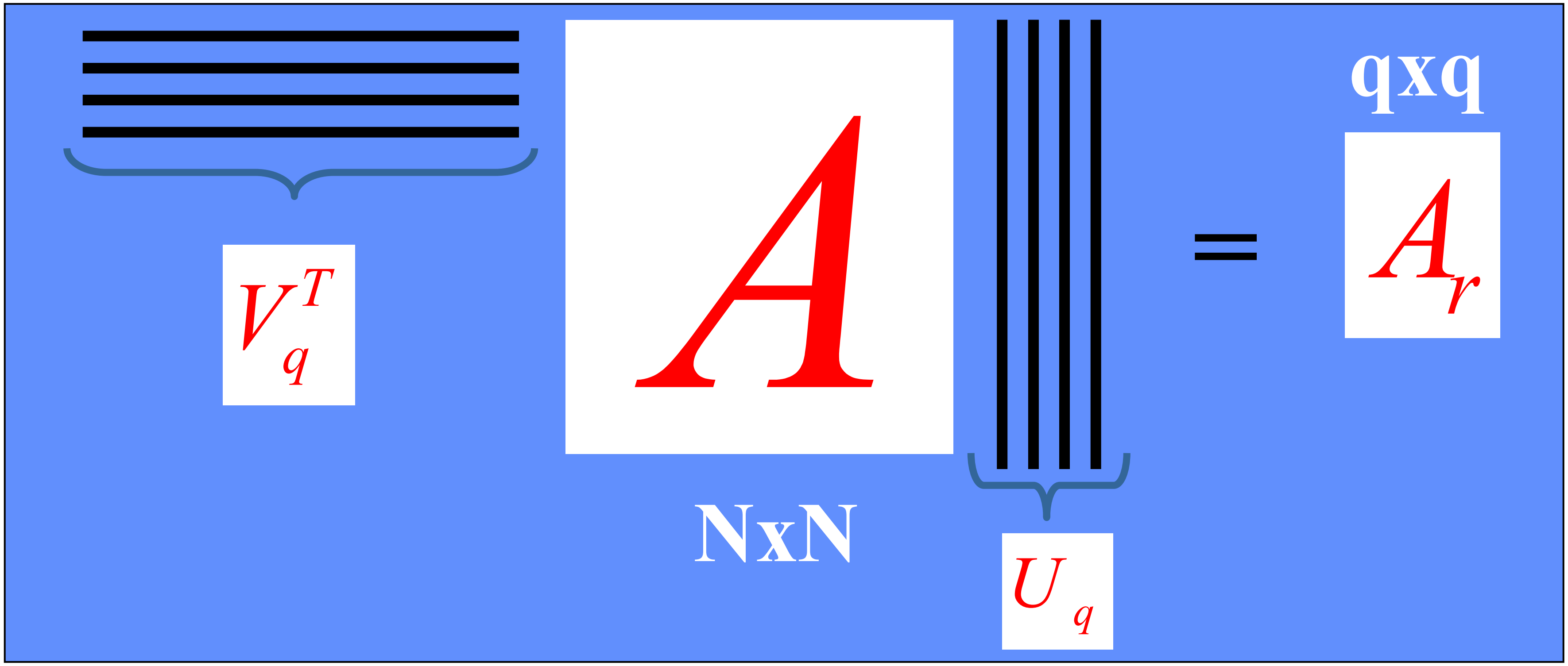
<u>Change of variables</u>

$$x \approx U_q x_r$$



Galerkin $\rightarrow V_q$ space = U_q space

Forming the Reduced Matrix



- **No explicit A need, Only Matrix-vector products**

For each column of U_q

Multiply by A, then dot result with columns of V_q

Picking U and V

- **Use Eigenvectors**
- **Use Time Series Data**
  - **Compute**
  - **Use the SVD to pick *q* < *k* important vectors**

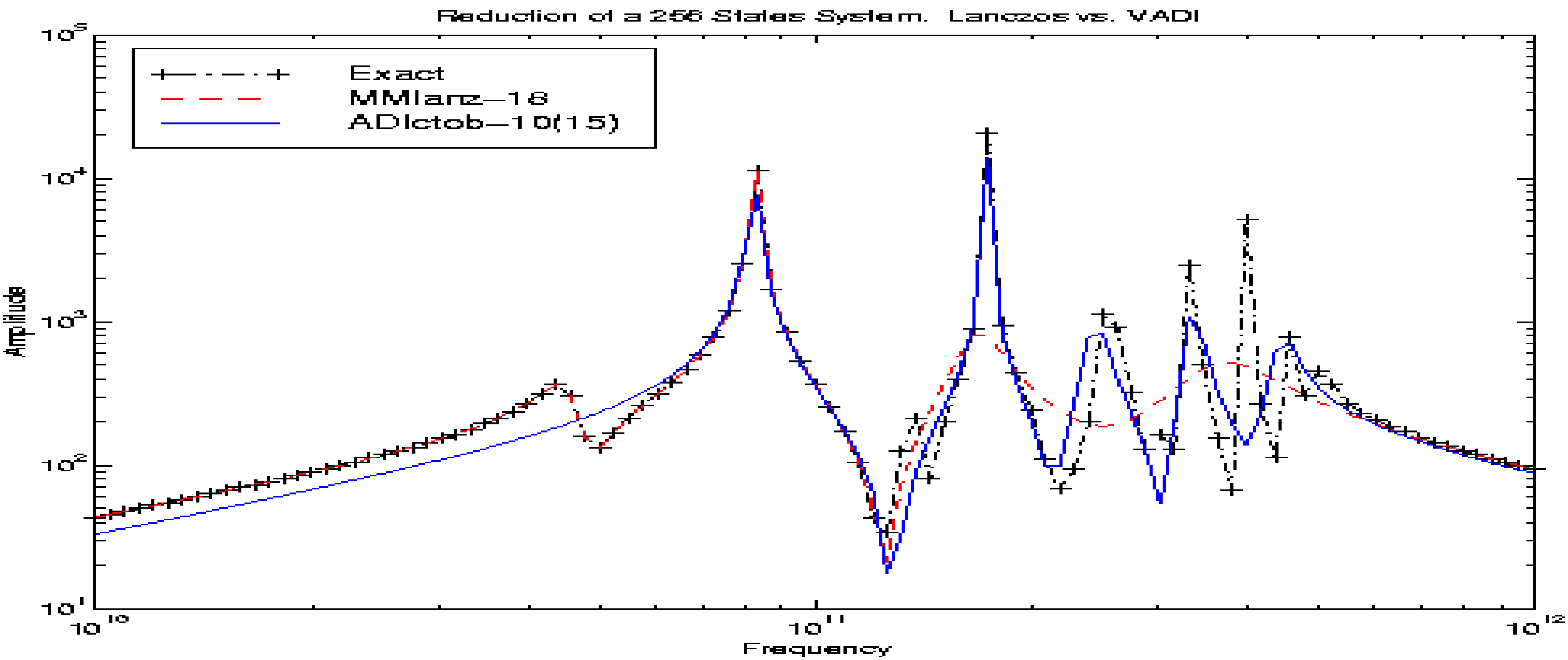
$$x(t_0), x(t_1), \cdots, x(t_k)$$

- **Use Frequency Domain Data**
  - **Compute**
  - **Use the SVD to pick *q* < *k* important vectors**

$$X(s_1), X(s_2), \cdots, X(s_k)$$

- **Krylov subspace Vectors**
- **Use Singular Vectors of System Grammians**

Easy to Model Even Complicated Frequency Behavior



- **Krylov subspace methods (red)**
  - **Excellent match over a narrow range of frequencies**
- **SVD of Hankel Operator (~TBR) (blue)**
  - **Minimizes worst case frequency domain error**
  - **Recently developed fast algorithms (CFADI).**

Persistent Problem with Projection

- **Can to get most observable and most controllable modes**

$$X_c(j\omega_i) = (j\omega_i I - A)^{-1} b, \quad i = \{1, \ldots, q\}$$

$$X_o(j\omega_i) = (j\omega_i I - A)^{-T} c, \quad i = \{1, \ldots, q\}$$

- **Harder to get the modes with the largest transfer product, if different**
  - **Happens in More Nonsymmetric problems (RLC)**
  - **Help to work with 2nd Order Systems?**

Transfer Function Point of View

Reduced Model Dynamical System

$$\frac{dx_r(t)}{dt} = \underbrace{A_r}_{qxq} x(t) + \underbrace{b_r}_{qx1} \underbrace{u(t)}_{\text{scalar input}} \qquad \underbrace{y_r(t)}_{\text{scalar output}} = \underbrace{c_r}_{qx1}{}^T x_r(t)$$

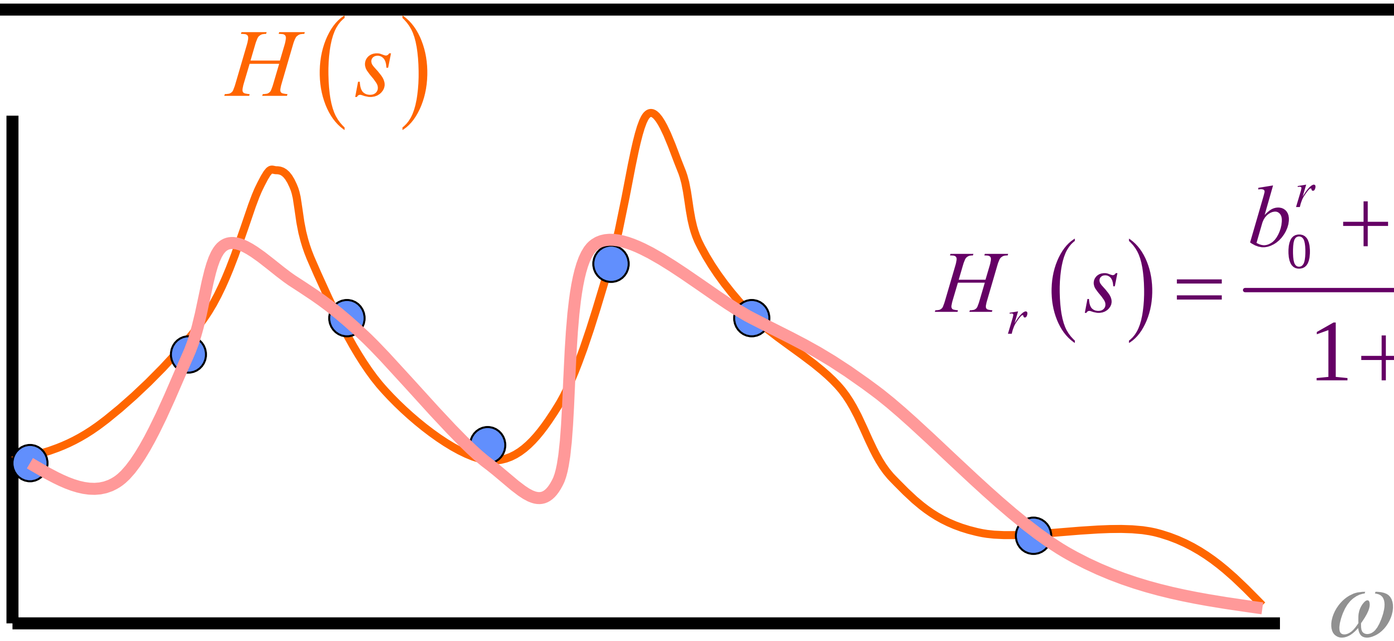
$2q + q^2$ coefficients

Reduced Model Transfer Function

$$H_r(s) = \frac{b_0^r + b_1^r s + \cdots + b_{q-1}^r s^{q-1}}{1 + a_1^r s + \cdots + a_q^r s^q}$$

$2q$ coefficients

Why Not Just fit the data



$$H_r(s) = \frac{b_0^r + b_1^r s + \cdots + b_{q-1}^r s^{q-1}}{1 + a_1^r s + \cdots + a_q^r s^q}$$

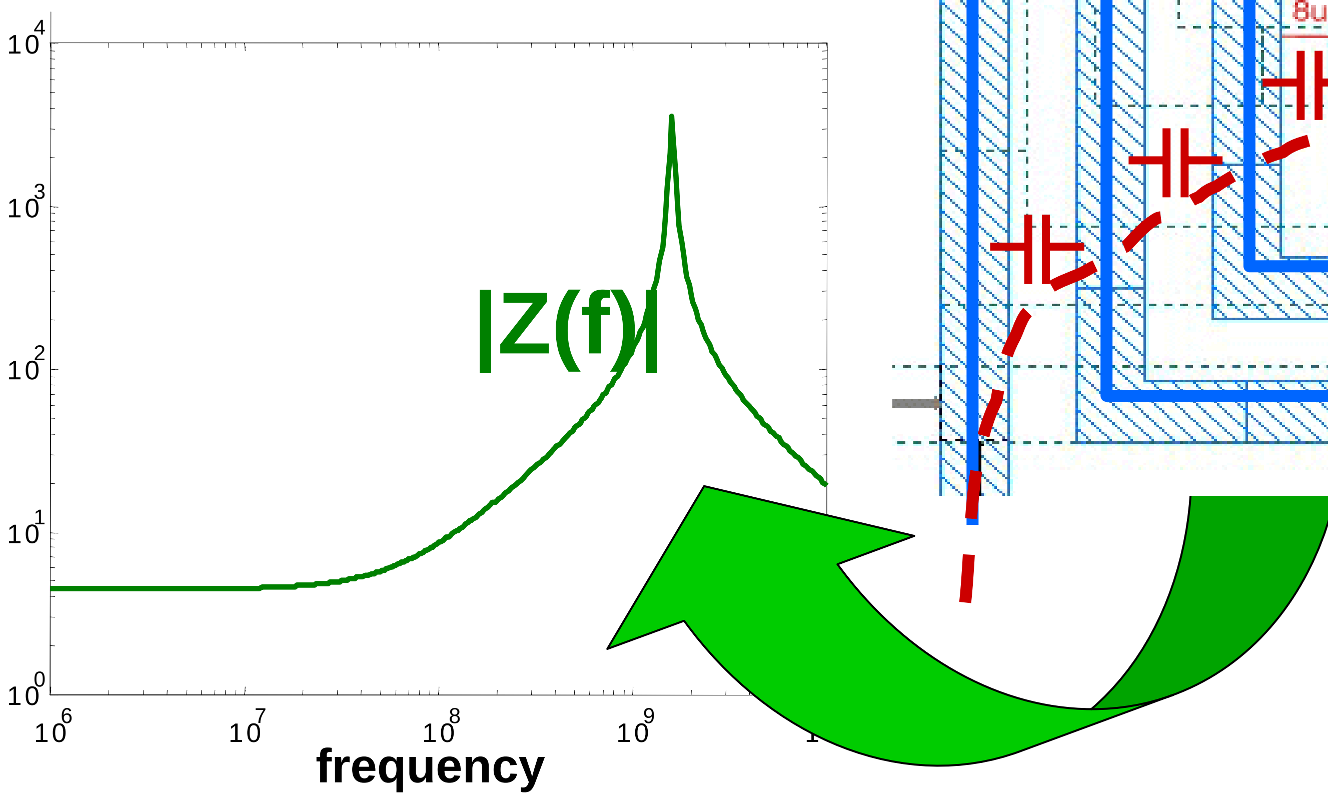
$$\left(1 + a_1^r s_i + \cdots + a_q^r s_i^q\right) H(s_i) - \left(b_0^r + b_1^r s + \cdots + b_{q-1}^r s^{q-1}\right) \approx 0$$

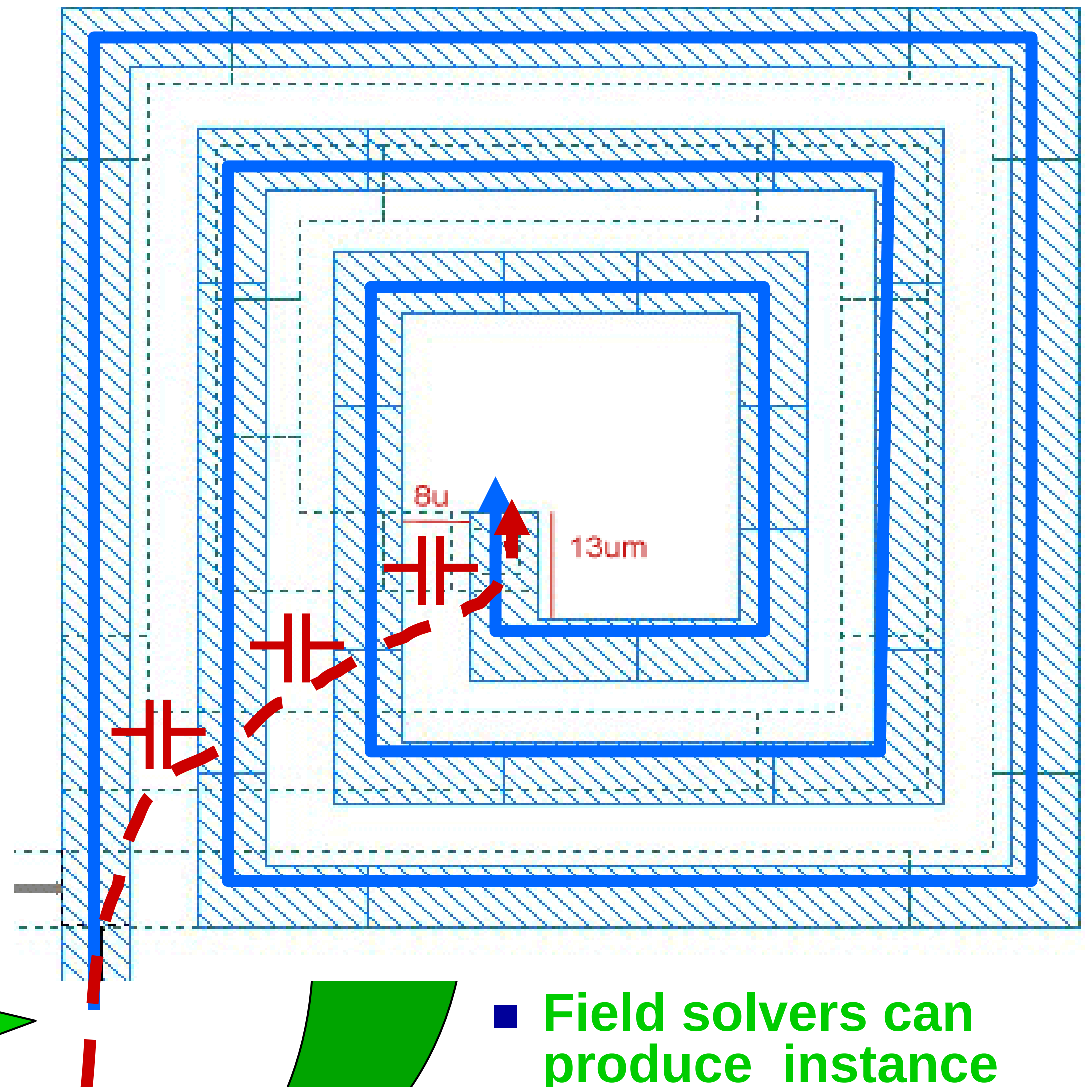
Key New Result – Matching Real Part (or Imaginary part) is a Convex Optimization Problem

Is Projection A Waste Of Time?

Motivation Example: RF micro-inductor

- How are the **substrate eddy currents** affecting the quality factor of the inductor?
- How are the **displacement currents** affecting the resonance of the inductor?
- Need to capture **all 2nd order effects**





- Field solvers can produce instance impedance vs. frequency curves.

Model Order Reduction for LINEARLY Parameterized Systems

- Given a large parameterized linear system:

$$\left(E_0 + s_1 E_1 + \ldots + s_p E_p\right) x = b\,u$$

$$y = c^T x$$

$$\left(\hat{E}_0 + s_1 \hat{E}_1 + \ldots + s_p \hat{E}_p\right) \hat{x} = \hat{b}\,u$$

$$\hat{y} = \hat{c}^T \hat{x}$$

- Projection Preserves the Smoothness of the Parameter Variation!
- Will Rational Fitting?

Many State Spaces per Transfer function

Reduced Model Transfer Function

$$\frac{dx_r(t)}{dt} = A_r x(t) + b_r u(t) \quad y_r(t) = c_r^T x_r(t)$$

$$\Rightarrow H(s) = c_r^T (sI - A_r)^{-1} b_r$$

Similarity (x = Sw) Transformed Transfer Function

$$\frac{dw_r(t)}{dt} = S^{-1} A_r S w(t) + S^{-1} b_r u(t) \quad y_r(t) = c_r^T S w_r(t)$$

$$\Rightarrow H(s) = c_r^T S \left(sI - S^{-1} A_r S\right)^{-1} S^{-1} b_r = c_r^T (sI - A_r)^{-1} b_r$$

Many Dynamical Systems have the same transfer function, only one retains smoothness

Interpolation Approaches Generalize

$$\left[s_1 E_1 + \ldots + s_p E_p - I \right] x = b u$$

$$y = c^T x$$

$$x = -\left[I - \left(s_1 E_1 + \cdots + s_p E_p \right) \right]^{-1} b u = \sum_{m=0}^{\infty} \left(s_1 E_1 + \cdots + s_p E_p \right)^m b \quad u$$

- **It is a p-variables Taylor series expansion**

$$x \in span \left\{ \underbrace{b, E_1 b, E_2 b, \cdots, E_p b}, E_1^2 b, \left(E_1 E_2 + E_2 E_1 \right) b, \cdots \right\}$$

$$\left[x \right] = \left[U_q \right] \left[\hat{x} \right]$$

Pick vectors to match points or derivatives, or both, for each parameter

Picking U and V

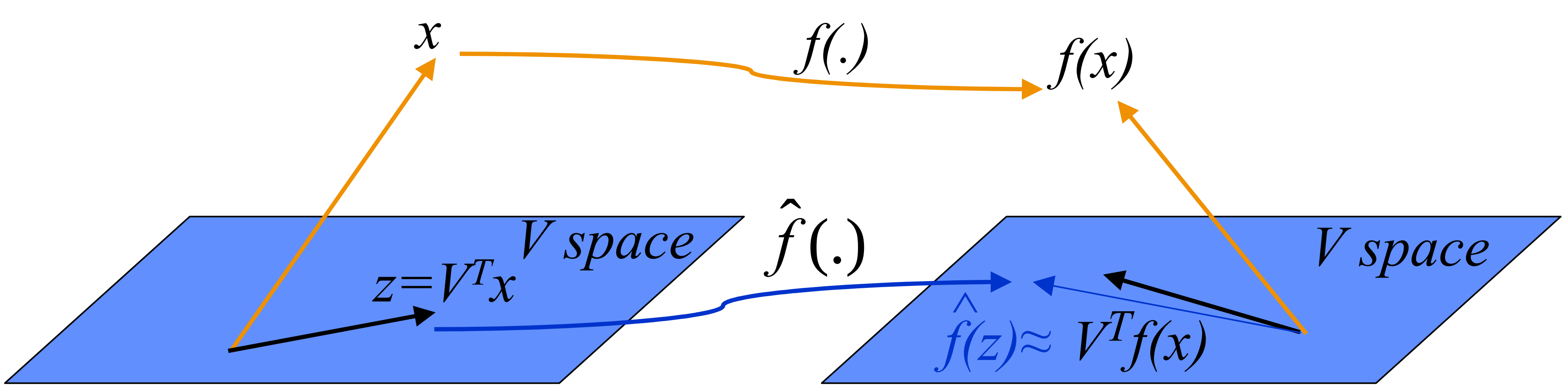
- **Use Eigenvectors of Many Systems**
- **Use Frequency Domain and Parameter Domain Data**
  - **Compute state for lots of points**
  - **Use the SVD to pick $q < k$ important vectors**
- **Krylov subspace Vectors**
  - **Get a combinatorial explosion**
- **Use Singular Vectors of Compromise System Grammians**
  - **Solve many simultaneous Lyapunov Inequalities**

Nonlinear MOR – Representation Problem

- Nonlinear dynamical systems:

$$\frac{dx}{dt} = f(x) + Bu \qquad y = C^T x \qquad x \in R^n$$

- Projection of the nonlinear operator *f(x)*:



- How to find $\hat{f}(.)$?

Problems with MOR for nonlinear

- Substitute: $x = Vz$ to $\dfrac{dx}{dt} = f(x) + Bu$
- Reduced system: $$\frac{dz}{dt} = \overbrace{V^T f(Vz)}^{\hat{f}(z)} + V^T Bu$$
- A problem: $V^T f(Vz)$: $\underset{\text{small}}{R^q} \to \underset{\text{large}}{R^N} \to \underset{\text{large}}{R^N} \to \underset{\text{small}}{R^q}$

  $q=10$ $N=10^4$ $N=10^4$ $q=10$
- Using $V^T f(Vz)$ is too expensive!

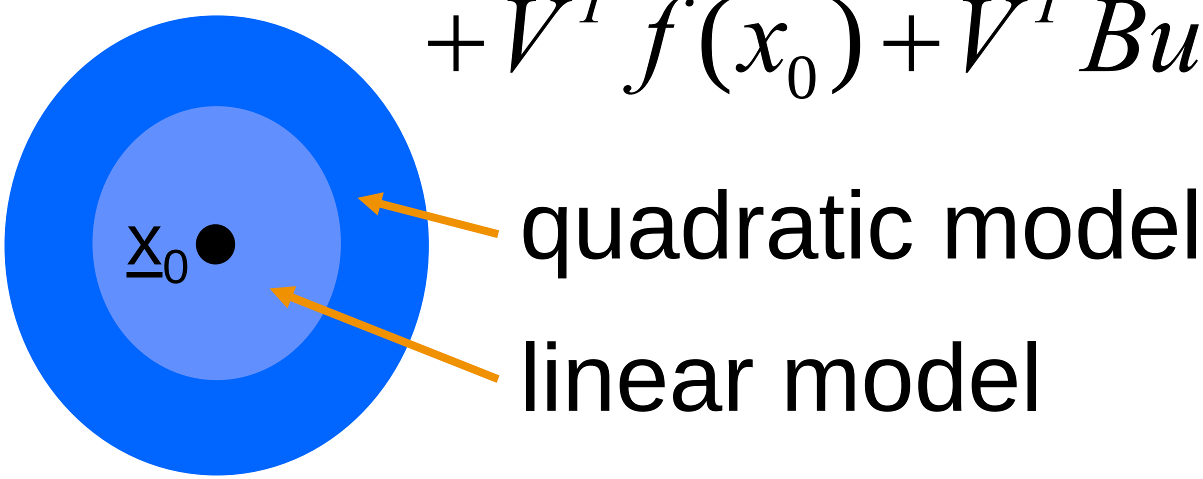
Volterra Approach

- Use Taylor's expansions to approximate *f(x)*:

$$f(x) = f(x_0) + J(x - x_0) + W((x - x_0) \otimes (x - x_0)) + \ldots$$

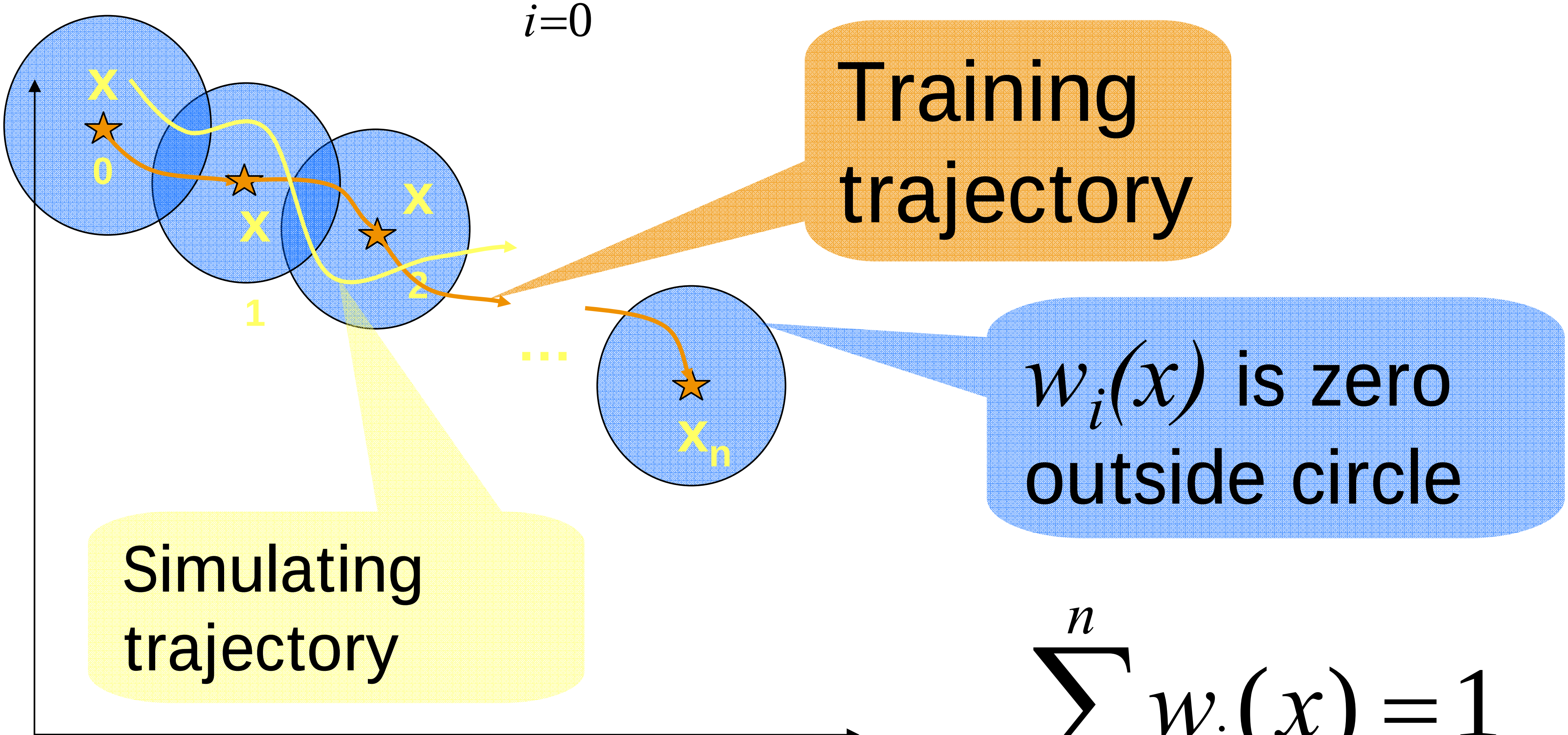
- Linear, quadratic reduced order models [Chen, Phillips 2000]:

$$\frac{dz}{dt} = \overbrace{V^T J V}^{\hat{J}}(z - z_0) + \overbrace{V^T W V \otimes V}^{\hat{W}}(z - z_0) \otimes (z - z_0) + V^T f(x_0) + V^T B u$$



Trajectory Piecewise Linear approximation of *f*.

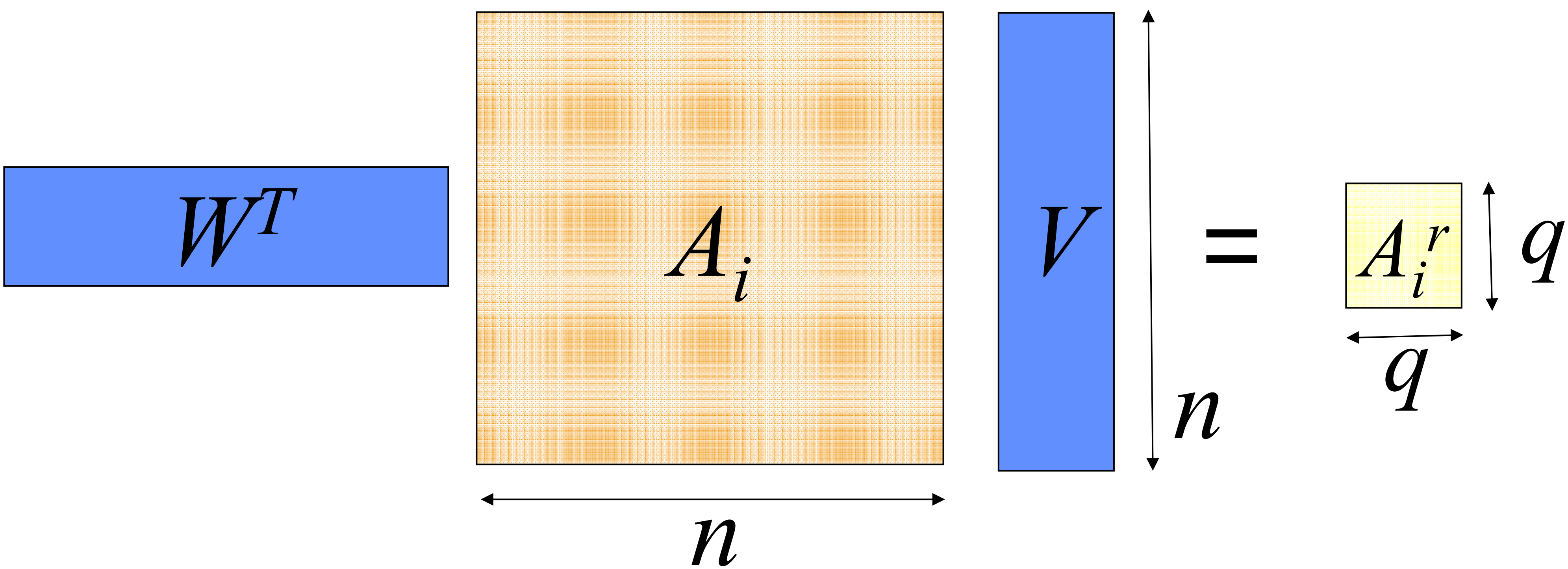
$$f_{TPWL}(x) = \sum_{i=0}^{n} w_i(x)(f(x_i) + A_i(x - x_i))$$



$$\sum_{i=0}^{n} w_i(x) = 1$$

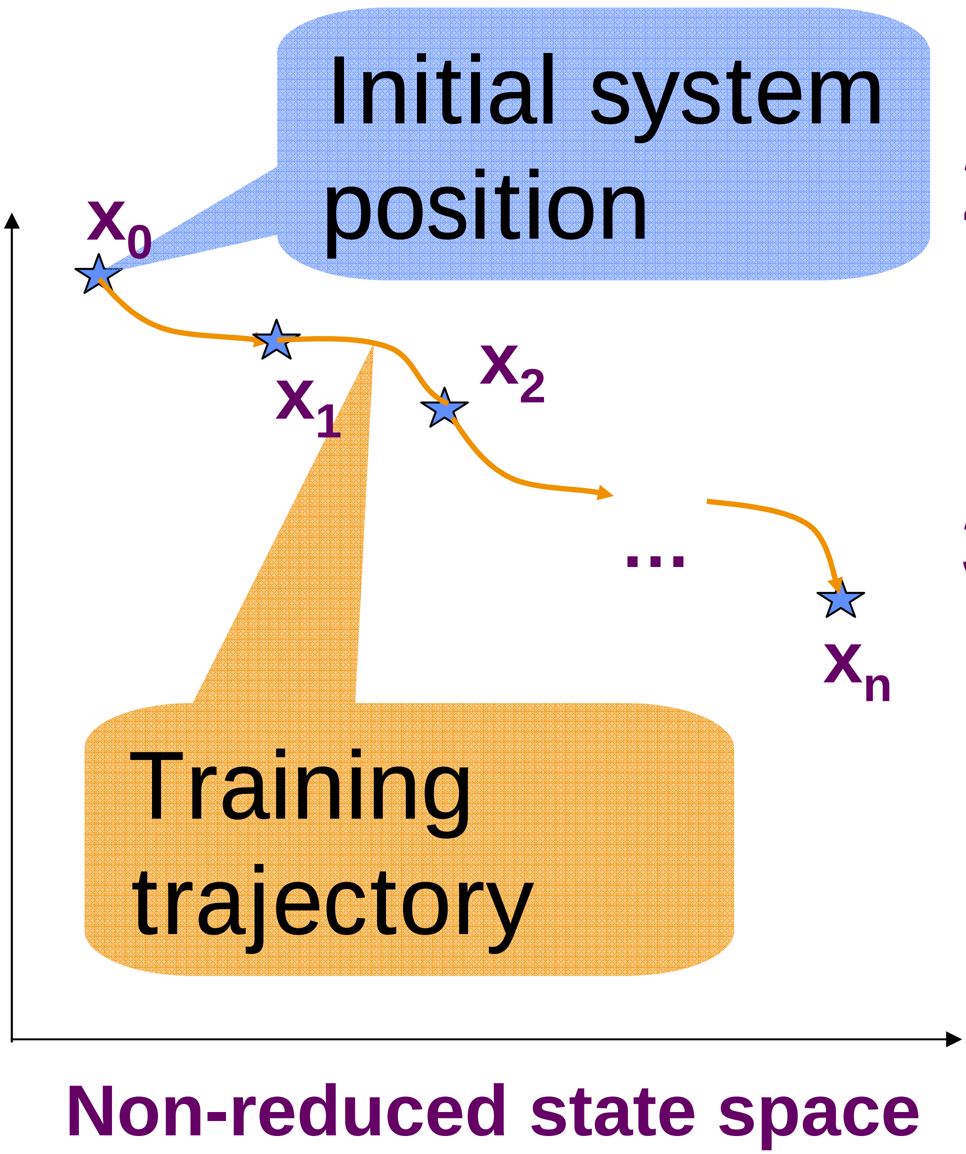
Projection and TPWL approximation yields efficient f^r

$$f^r_{TPWL}(x^r) = \sum_{i=0}^{n} w_i(x^r)\Big(\underbrace{W^T f(x_i)}_{q \times 1} + \underbrace{W^T A_i V}_{A_i^r}(x^r - W^T x_i)\Big)$$



Evaluating f^r_{TPWL} only order q^2 operations

TPWL approximation of *f*. Extraction algorithm



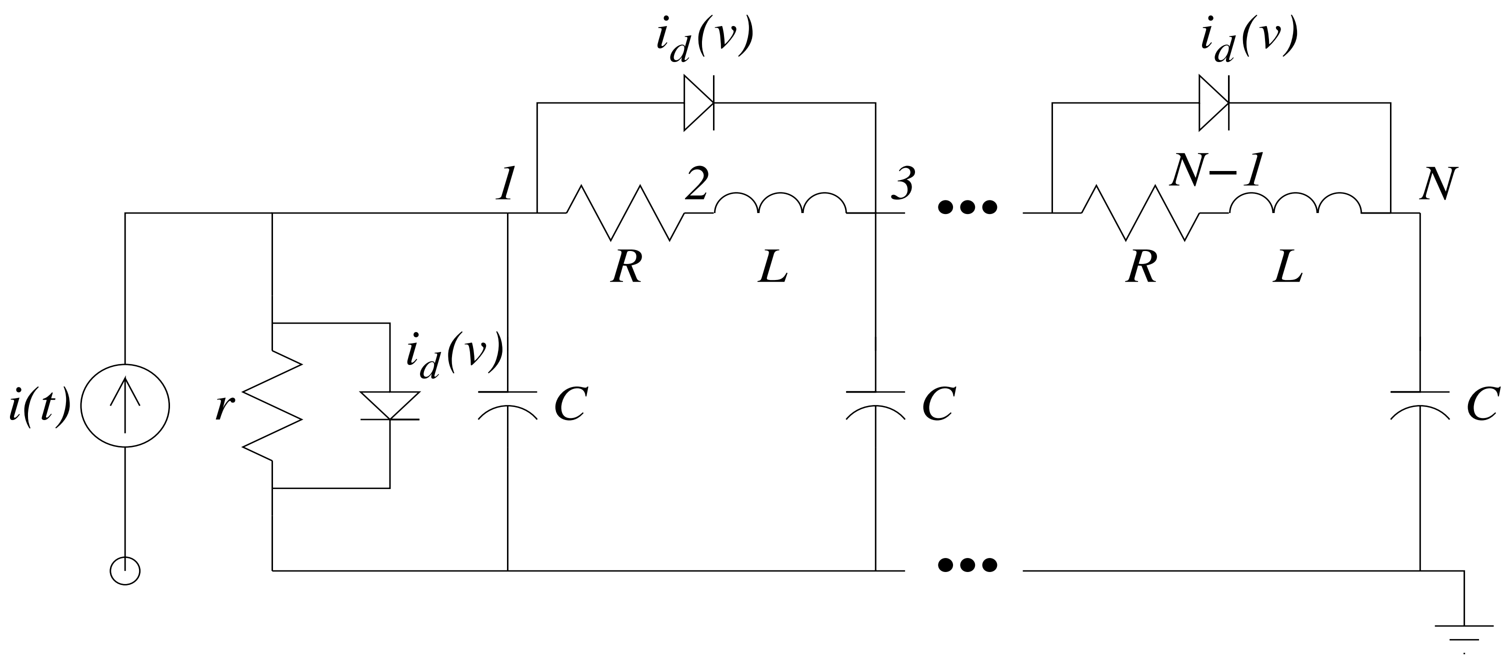
1. Compute A_1
2. Obtain W_1 and V_1 using linear reduction for A_1
3. Simulate training input, collect and reduce linearizations

$$A_i^r = W_1^T A_i V_1$$

$$f^r(x_i) = W_1^T f(x_i)$$

Example problem

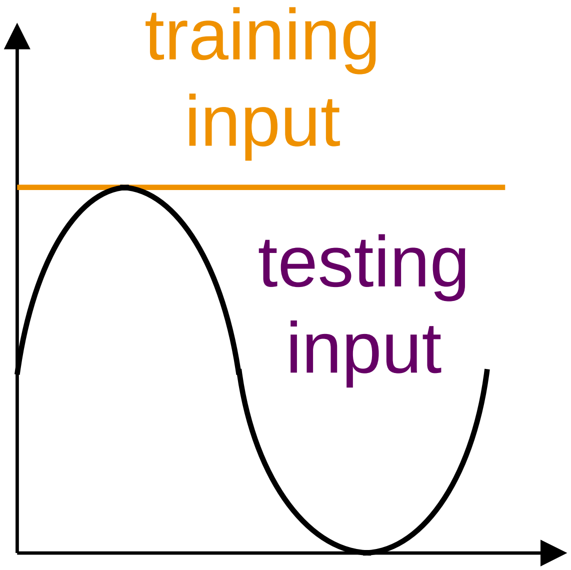
RLC line



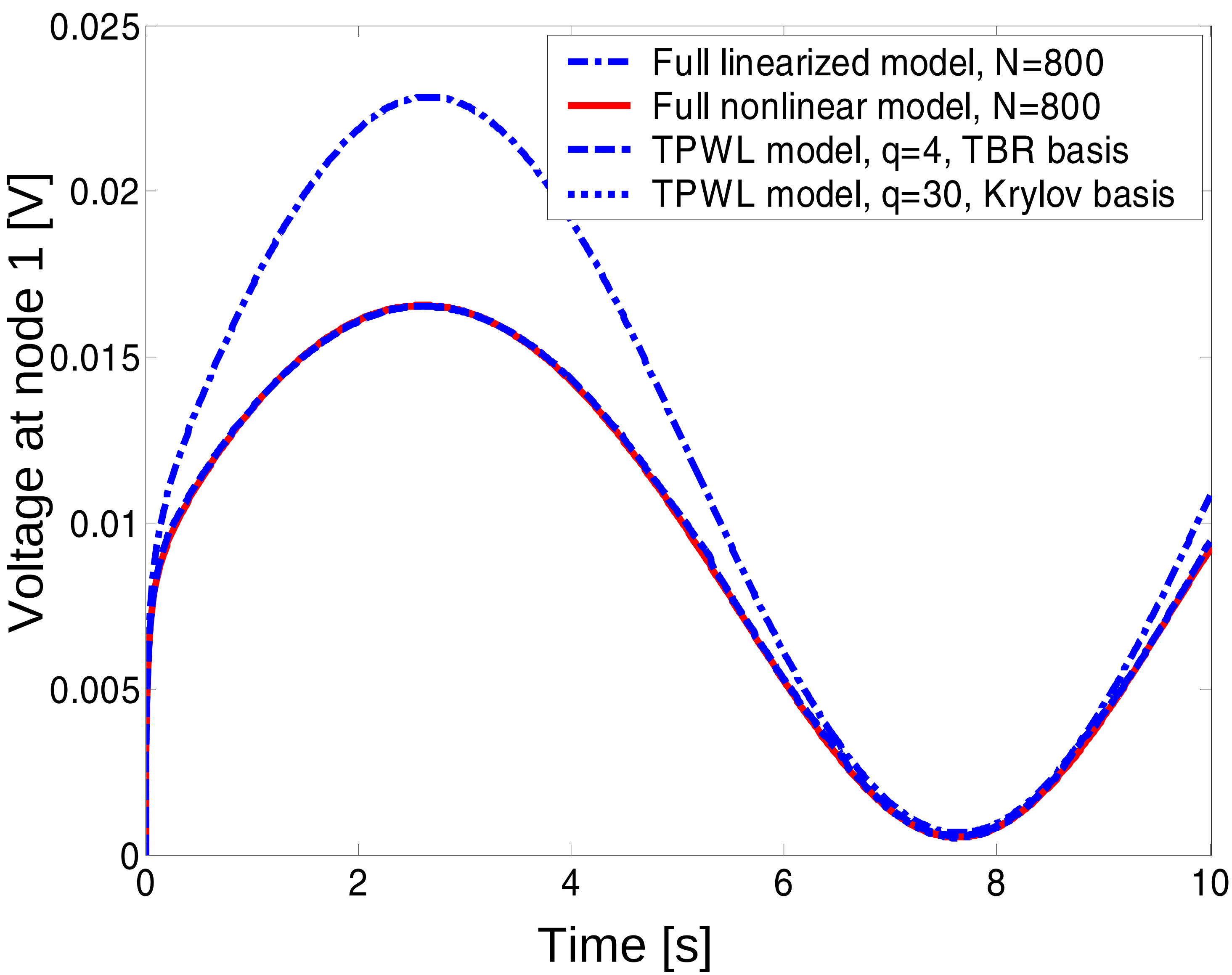
Linearized system has nonsymmetric, indefinite Jacobian

Numerical results – nonlinear RLC transmission line

- **Input:**



System response for input current i(t) = (sin(2π/10)+1)/2



Key issue: choosing projection

$$\begin{cases} \dfrac{dx(t)}{dt} = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases}$$

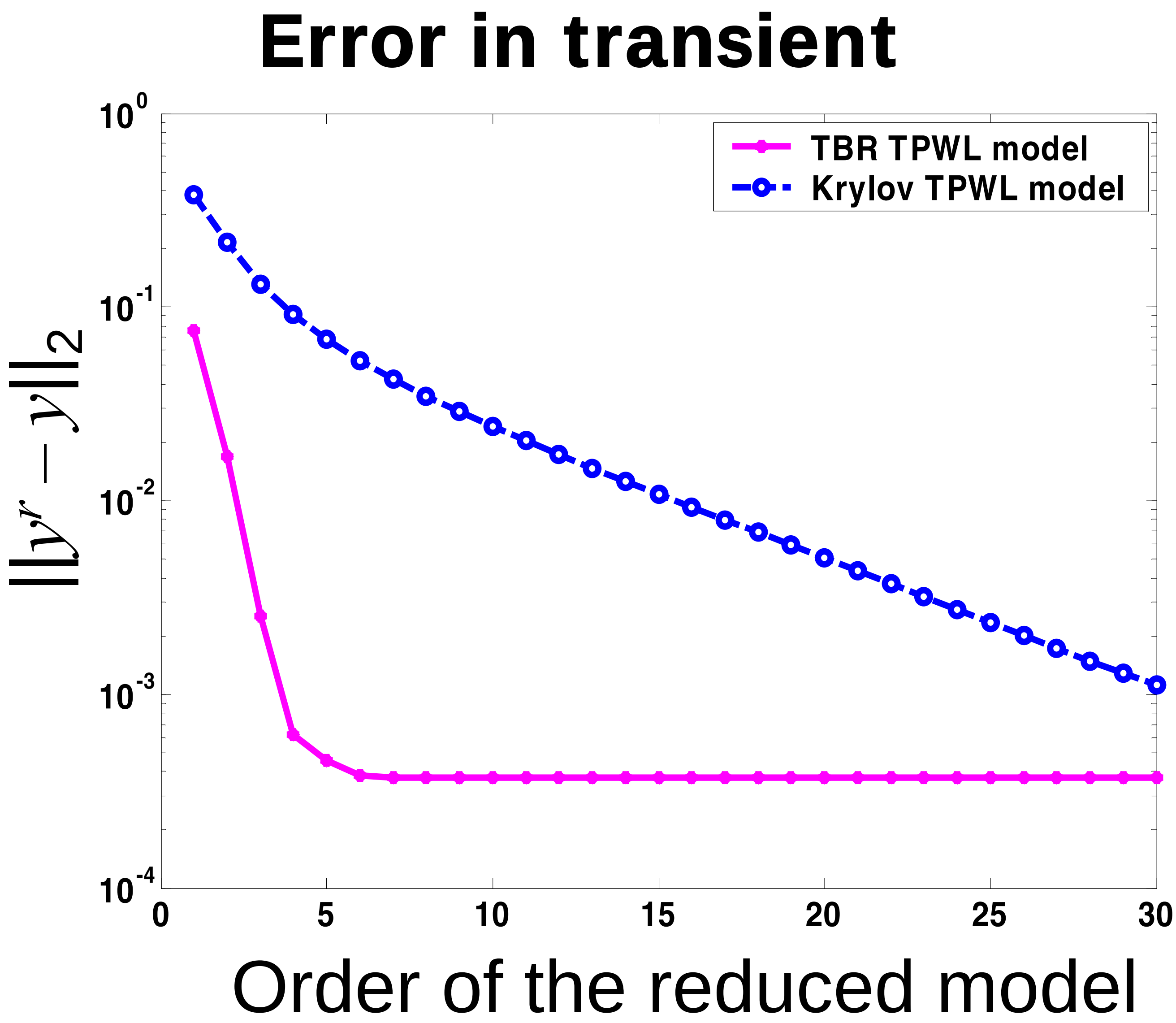
Krylov-subspace methods

Balanced-truncation methods

Result:

projection matrices W and V

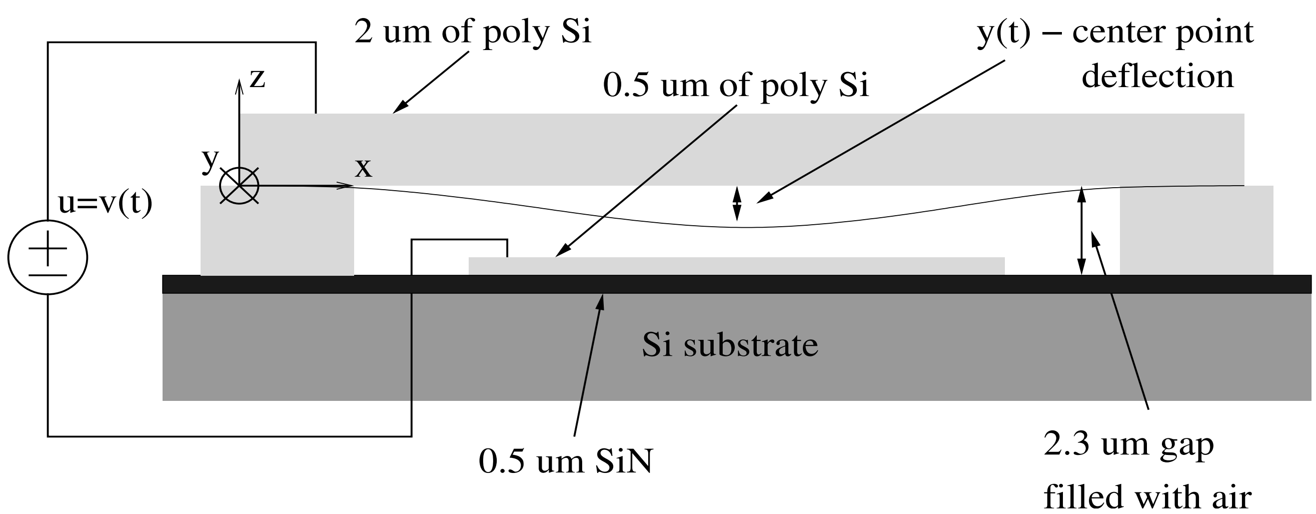
Numerical results – RLC transmission line



TBR-based TPWL beat Krylov-based

4-th order TBR TPWL reaches the limit of TPWL representation

Micromachined device example



$$\hat{E}I\frac{\partial^4 u}{\partial x^4} - S\frac{\partial^2 u}{\partial x^2} = F_{elec} + \int_0^w (p - p_a)dy - \rho\frac{\partial^2 u}{\partial t^2}$$

$$\nabla((1+6K)u^3 p\nabla p) = 12\mu\frac{d(pu)}{dt}$$

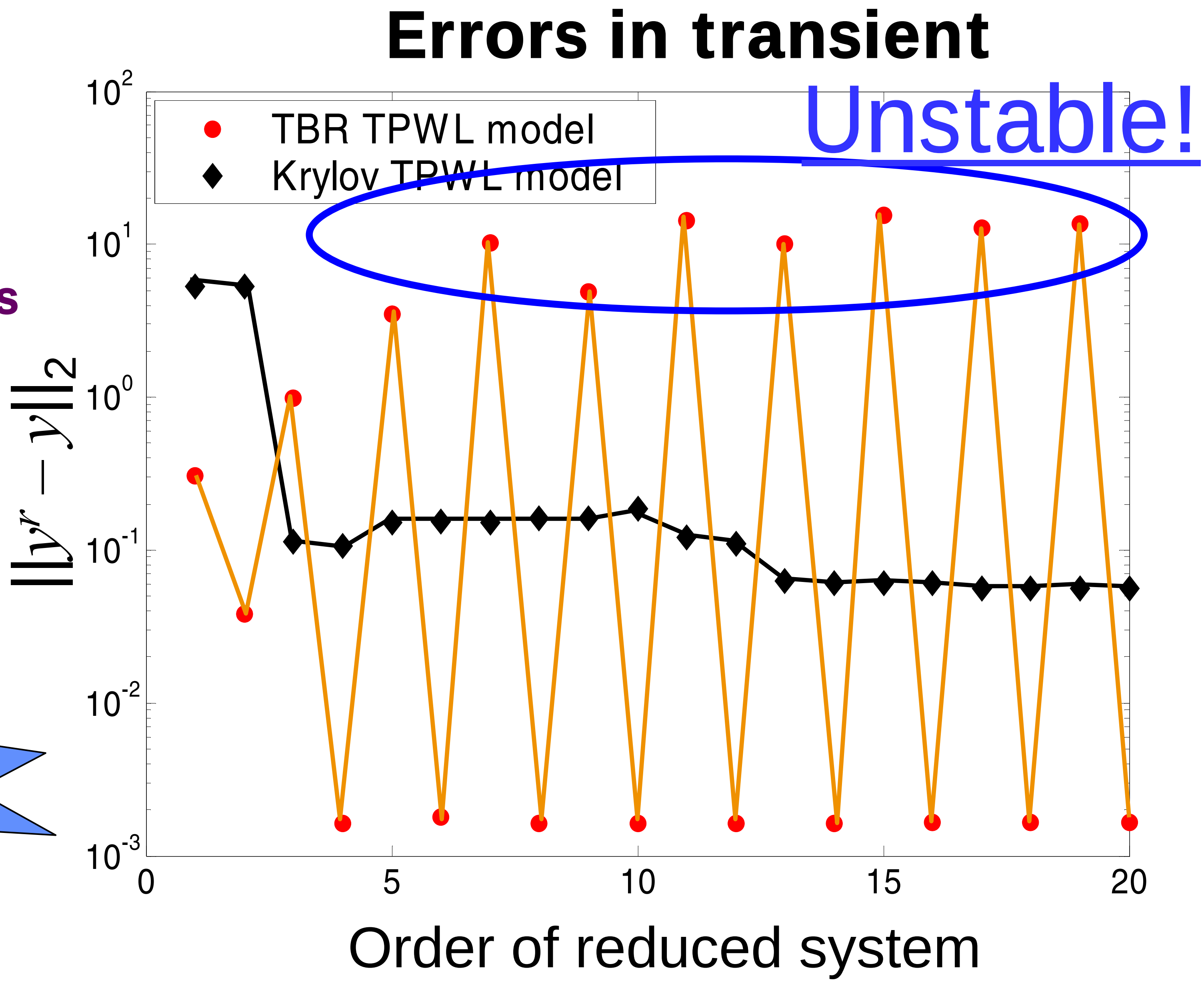
FD model

non-symmetric indefinite Jacobian

TPWL-TBR results – MEMS switch example

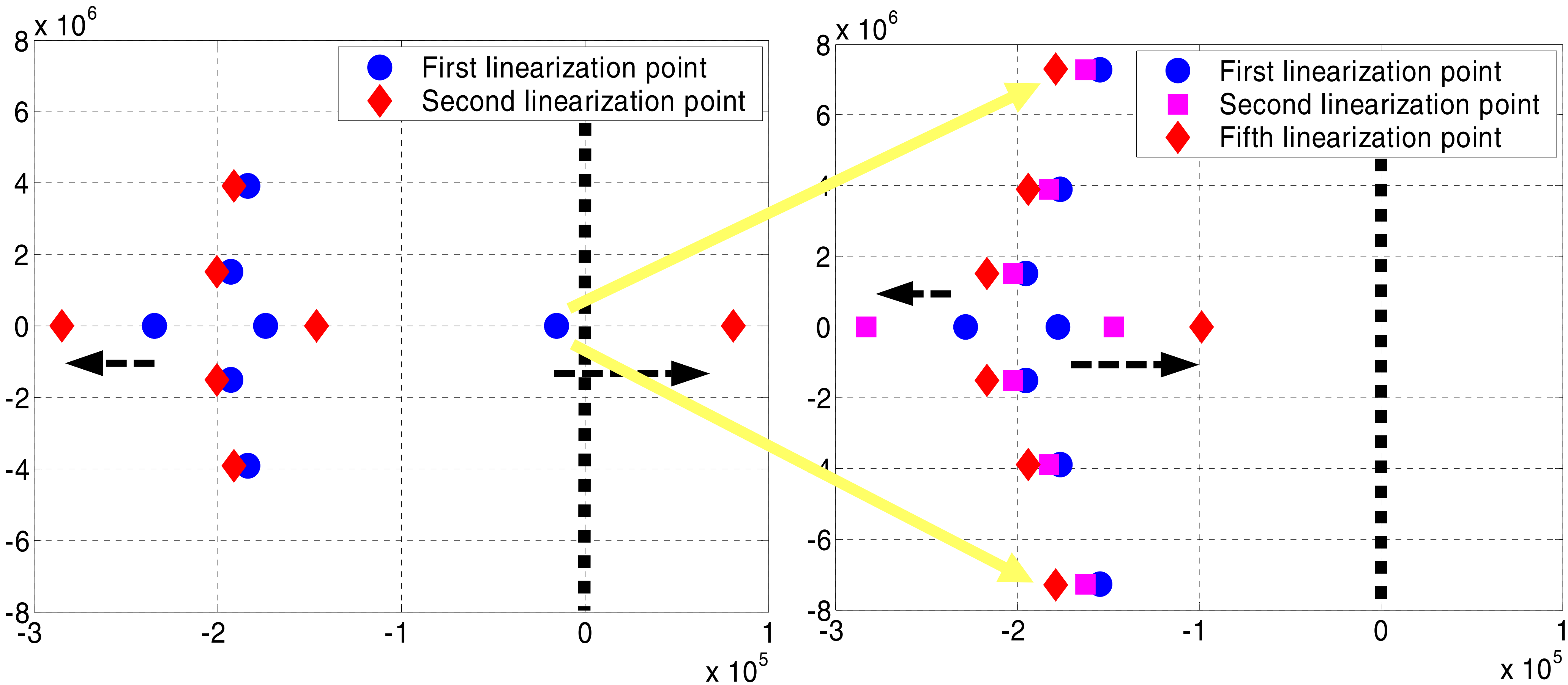
Odd order models unstable!

Even order models beat Krylov



Eigenvalue behavior of linearized models

Eigenvalues of reduced Jacobians, **q=7** Eigenvalues of reduced Jacobians, **q=8**



TBR is adding complex-conjugate pair

Explanation of even-odd effect – Problem statement

Consider two LTI systems:

Initial: (A, B, C)

TBR reduction

Projection basis V

Perturbed: $(\tilde{A}, \tilde{B}, \tilde{C})$

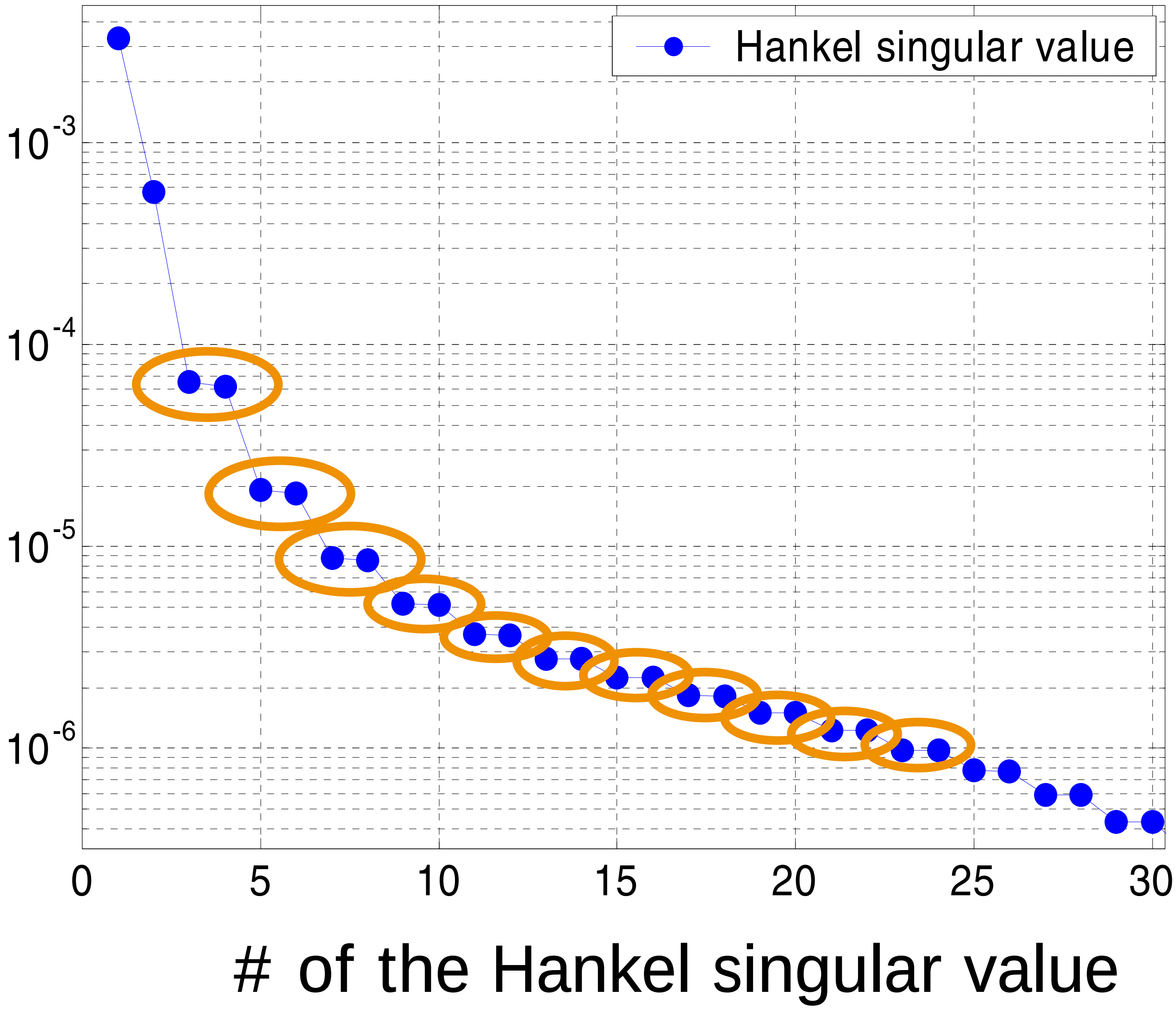
TBR reduction

Projection basis $\tilde{V}$

Define our problem:

How perturbation in the initial system affects projection basis?

Hankel singular values, MEMS beam example



This is the key to the problem.

Singular values are arranged in pairs!

Explaining even-odd behavior

$$c_i^k = \frac{(e_k^0)^T \Delta e_i^0}{\lambda_k^0 - \lambda_i^0}, k \neq i$$

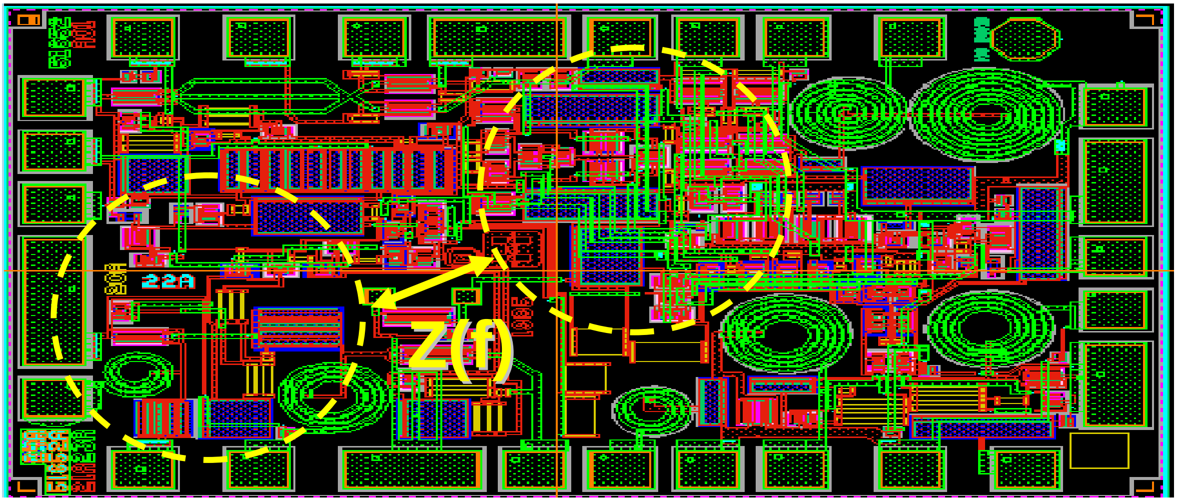
The closer Hankel singular values lie to each other, the more corresponding eigenvectors of V tend to intermix!

- **Analysis implies simple recipe for using TBR**
  - **Pick reduced order to insure**
    - **Remaining Hankel singular values are small enough**
    - **The last kept and first removed Hankel Singular Values are well separated**
  - **Helps insure that all linearizations stably reduced**

Many Methods Under Investigation

- Projection Methods
- Data Mining
- Support Vector Machines
- Nonlinear Generalizations of Controllability and Observability
- Finite-State Automata
- Sophisticated Sampling and Fitting

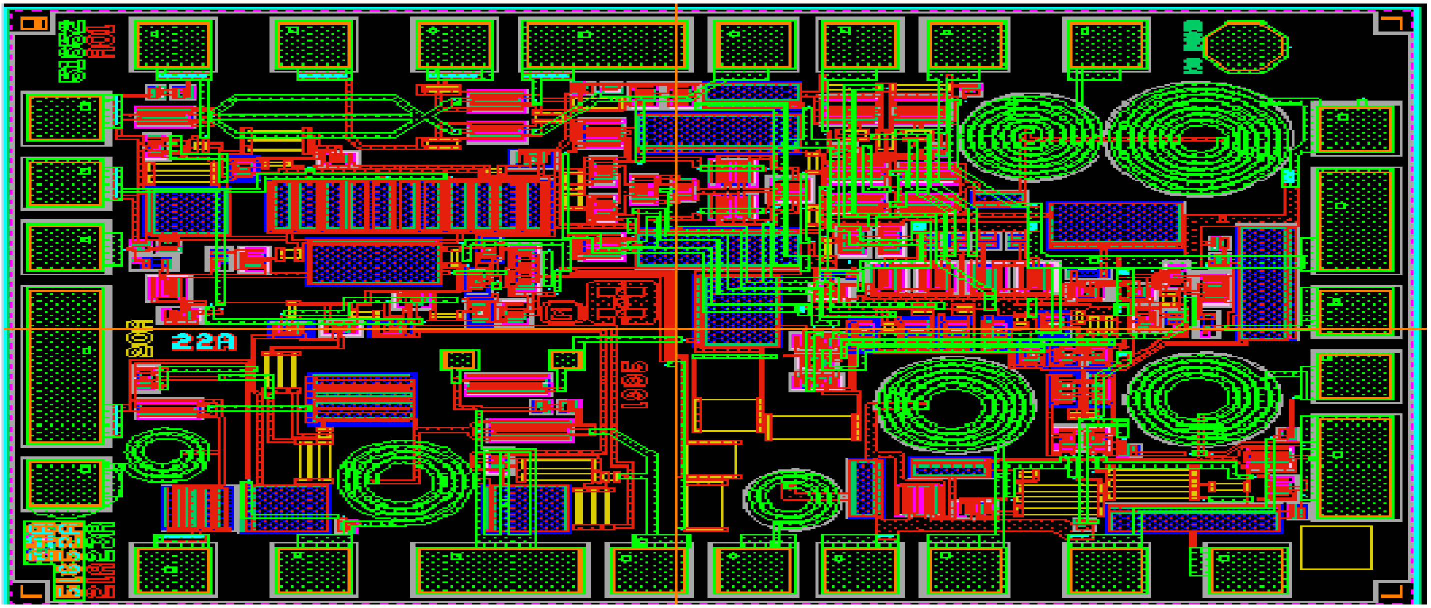
Massively Coupled Effects



Courtesy of Harris Semiconductor

- **Digital** – Narrow Signal Range **– 20db**
  - **Effective to Screen Small couplings**
- **Analog** – Wide Signal Dynamic Range **– 80db**
  - **Small couplings must be retained**
- **Analog Block – 1000's of interacting interconnect lines**
  - **Millions of Coupling terms** ⇦ **Massively Coupled Problem!**

Still to Come: Massively Coupled Interconnect Analysis



Courtesy of Harris Semiconductor

- Need to draw a box and extract everything
  - Including all the small couplings
  - Extracted Result must be efficient in a simulator
- Will try to use SVD based methods plus model order reduction
  - SVD for the geometric coupling
  - MOR for the frequency dependence

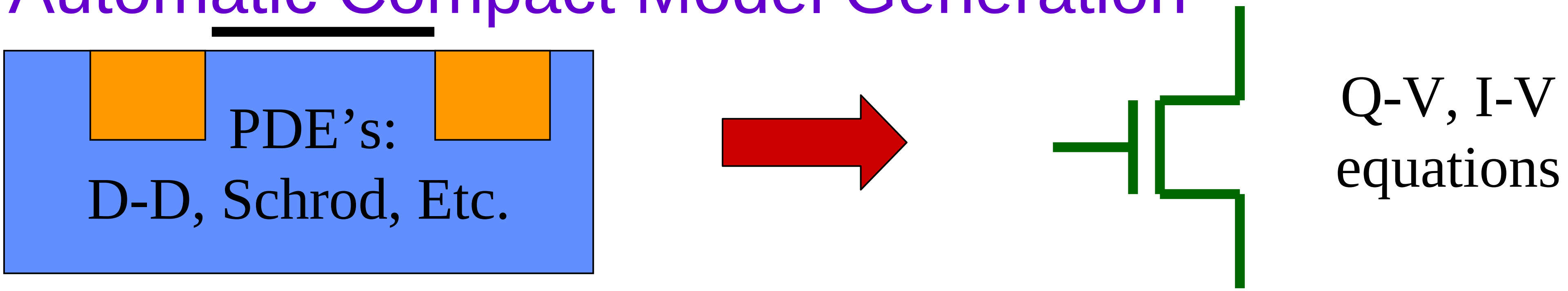
Still Massively Coupled Problem-- But New Approaches!

The role of fitting versus projection?

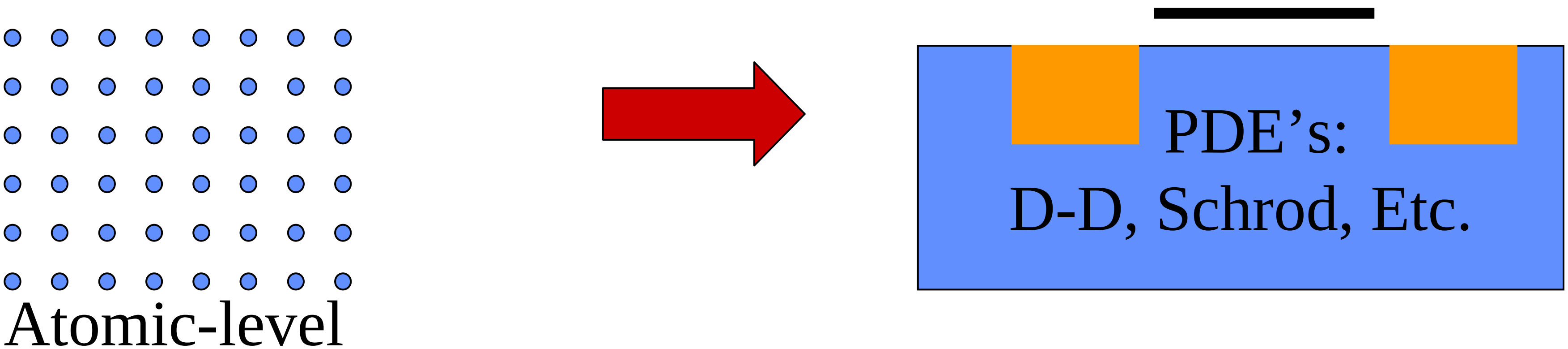
- **Fitting only uses I/O data**
  - **Convex optimization procedures**
  - **No smoothness between models**
- **Projection uses the system description**
  - **Has more information, what good is that info**
  - **Can pick out the state space that preserves smoothness**
- **For projection, how to get Observe/Control**
  - **Robustness demands we get x's large in transfer behavior**
- **Will Lyapunov Inequalities help?**

Impact of Reliable nonlinear MOR

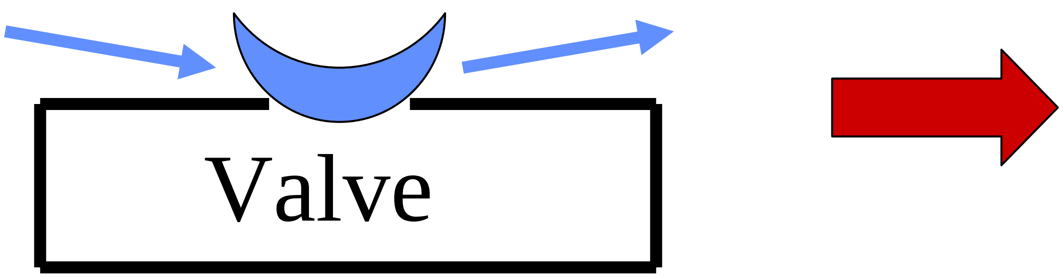
- Automatic Compact Model Generation



- Multiscale modeling?



- New device/technology models



$$\frac{dx_r(t)}{dt} = F(x_r(t)) + b_r u(t)$$

$$y(t) = c_r^T x_r(t)$$