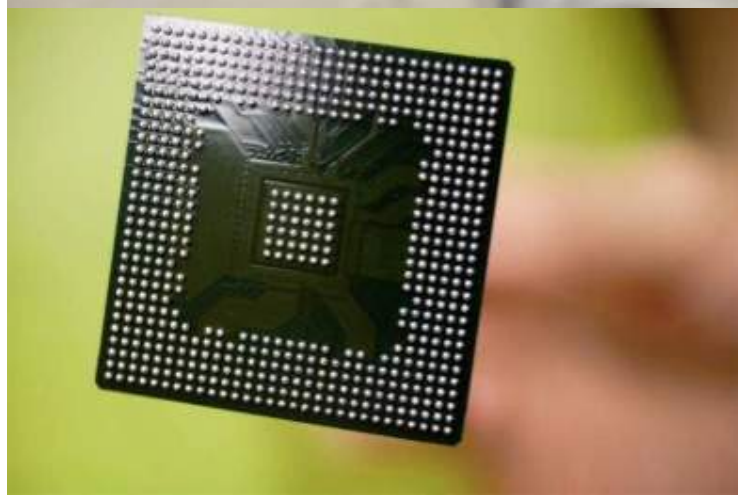
Model Order Reduction

Wil Schilders
NXP Semiconductors & TU Eindhoven
November 26, 2009

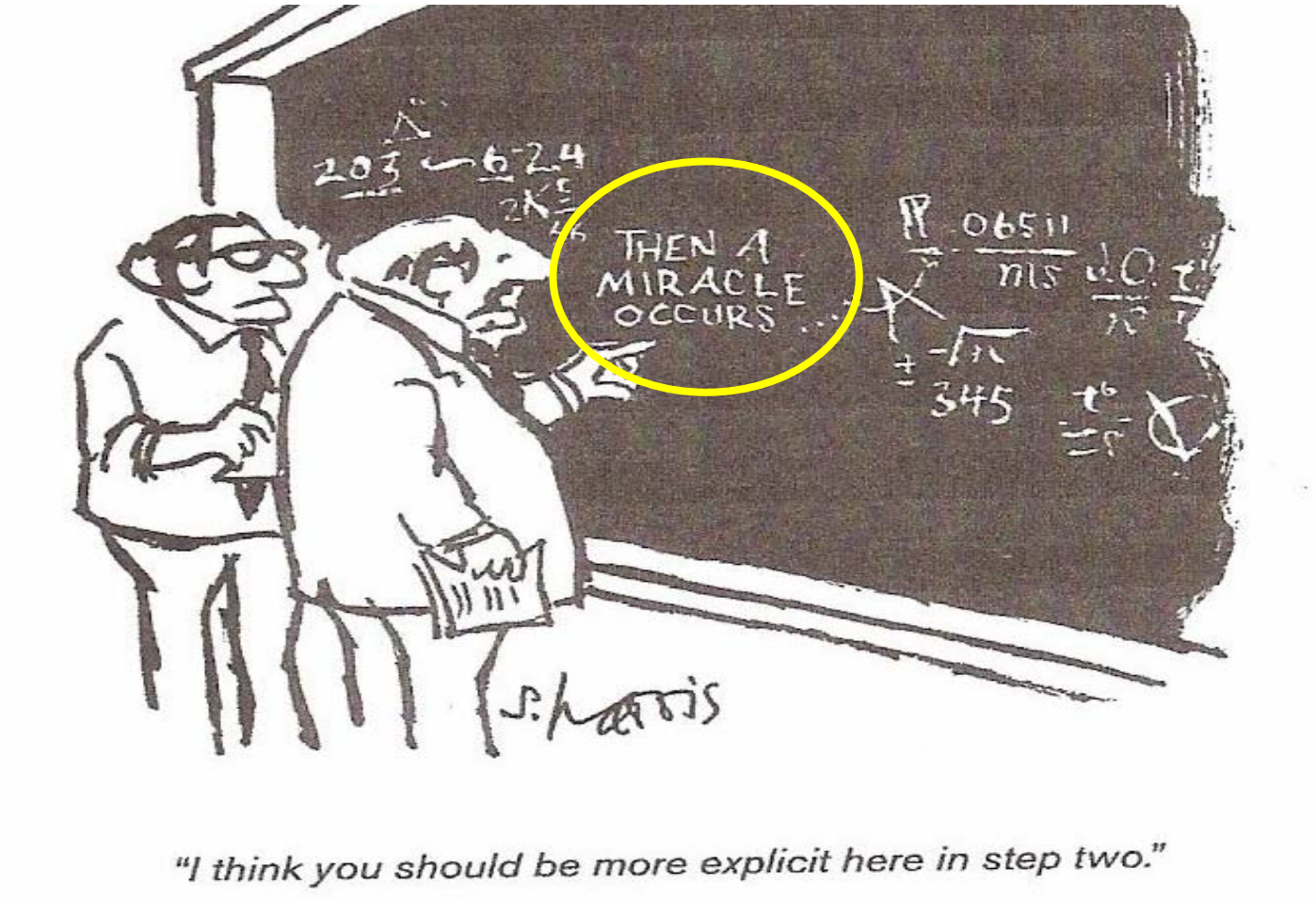
Utrecht University
Mathematics Staff Colloquium

Outline

- Introduction and motivation
- Preliminaries
- Model order reduction basics
- Challenges in MOR
- Conclusions

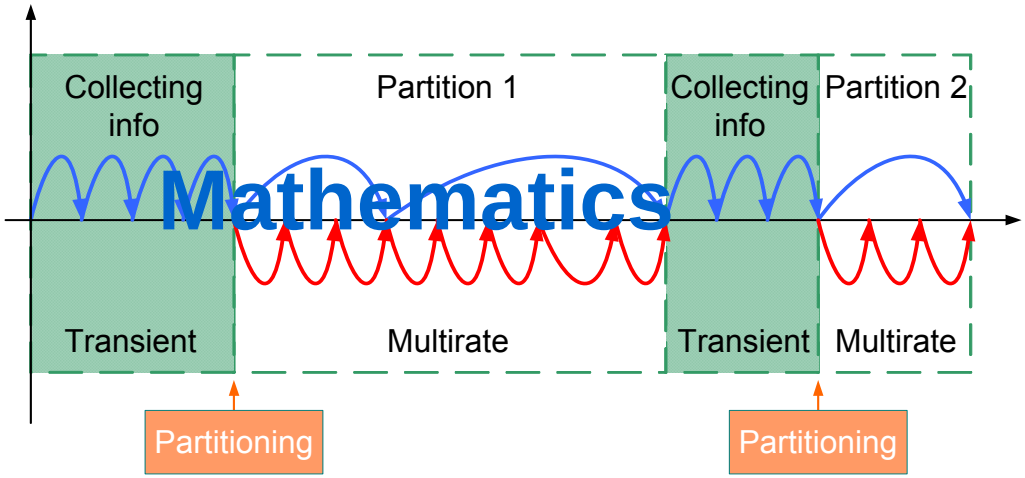


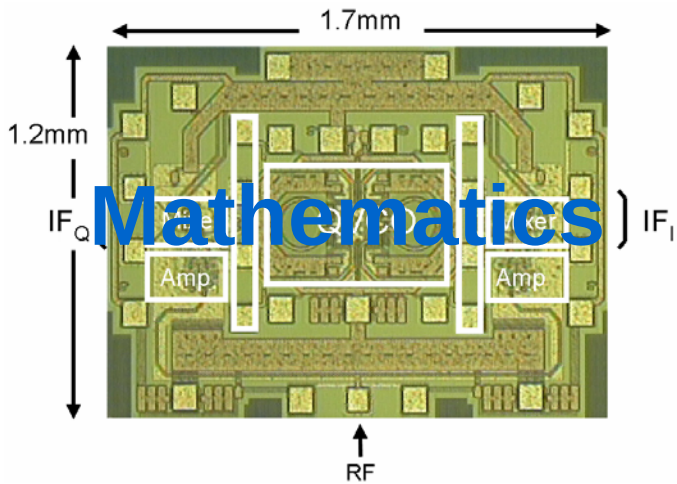
The work of mathematicians may not always be very transparant…….

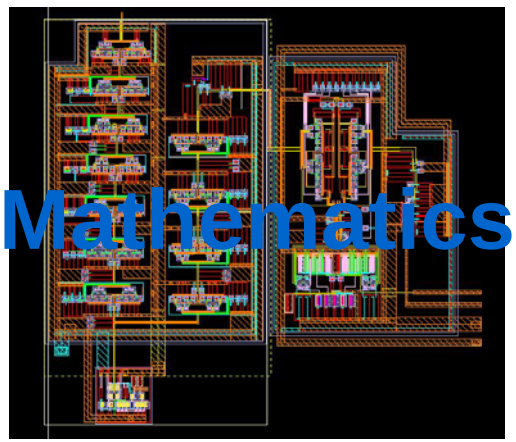


"I think you should be more explicit here in step two."

But it is present everywhere in our business

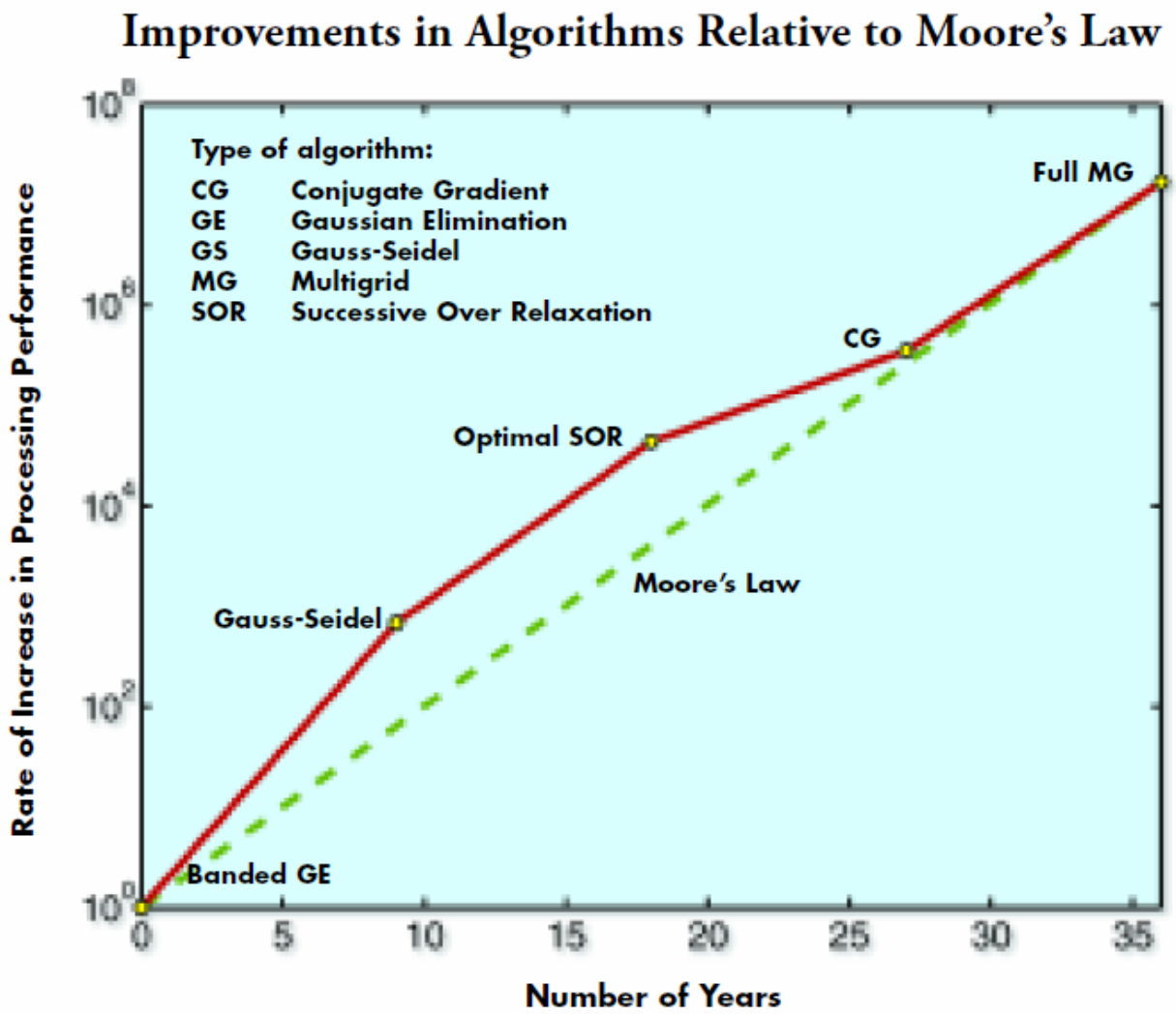




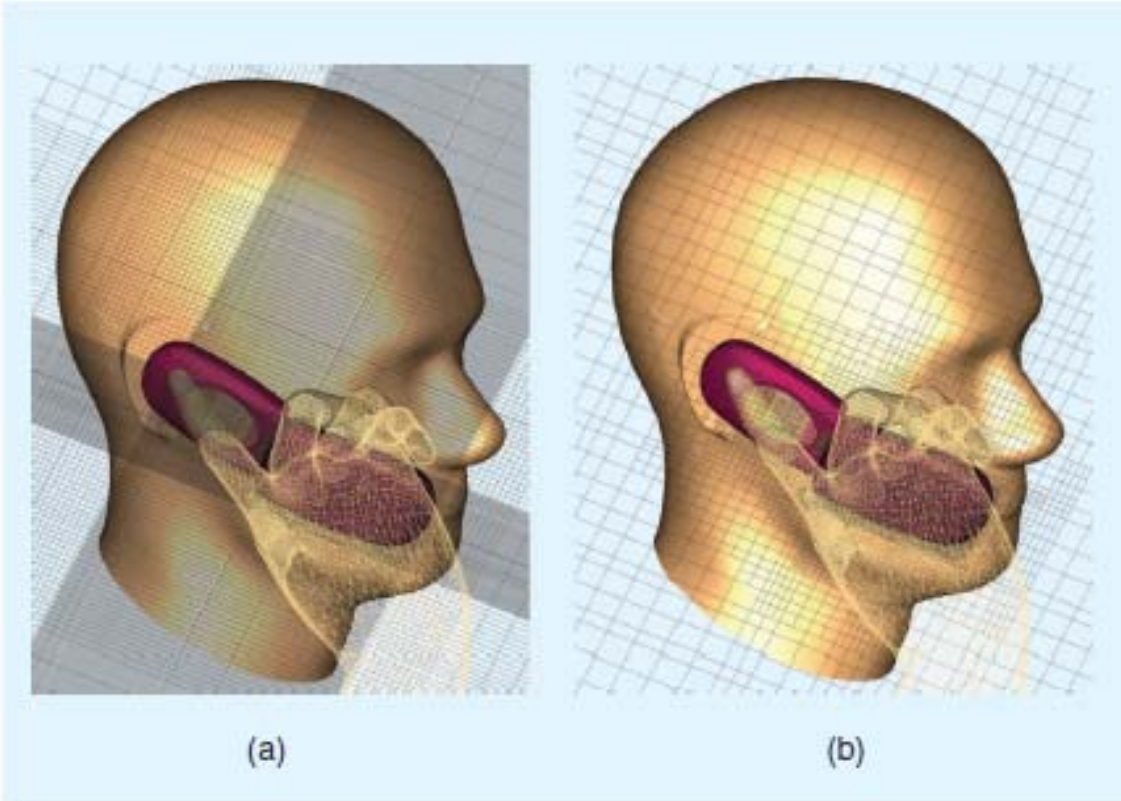


Invisible contribution, visible success

Moore's law also holds for numerical methods!



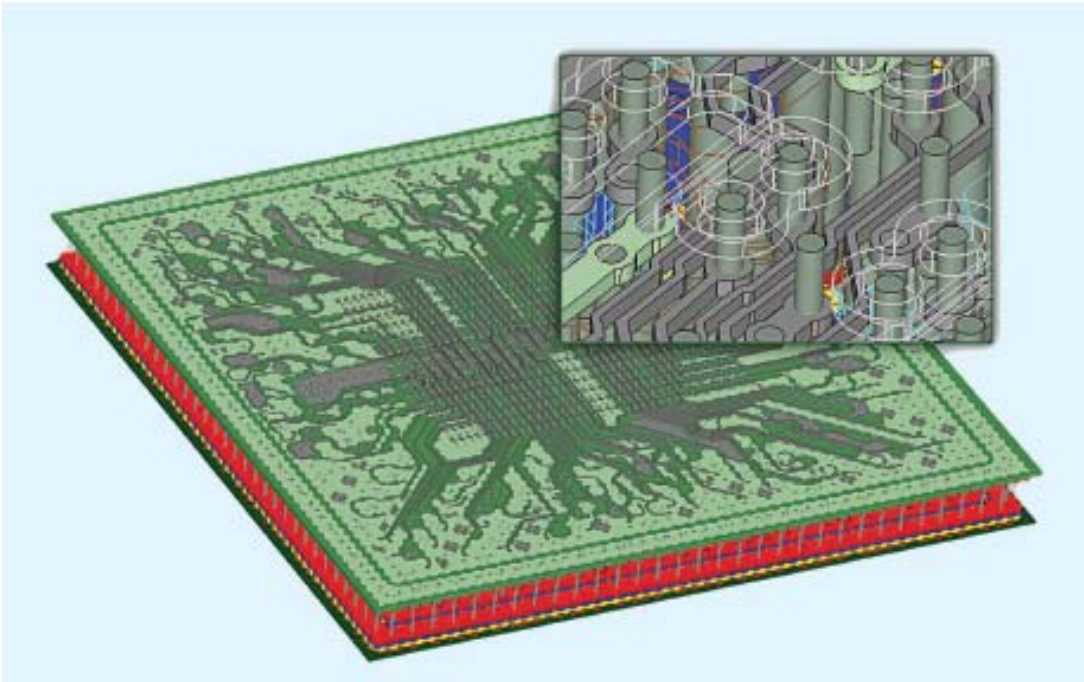
Very advanced simulations



Subgridding mechanisms reduce the number of mesh points in a simulation. In this example (a) the full grid is 20 times larger (35e6 mesh nodes) than (b) the subgridded version (1.75e6 mesh nodes).

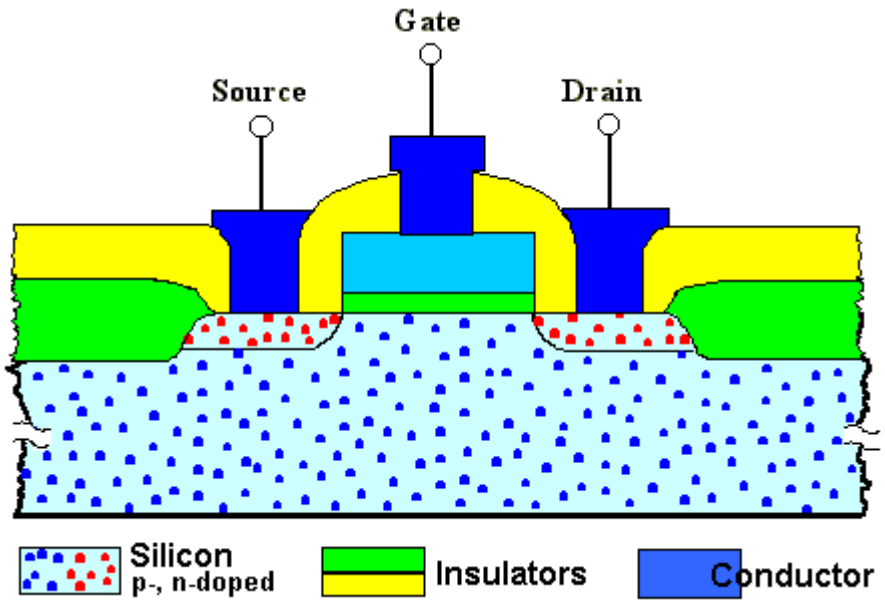
The impossible made possible…..

- *A complete package layout used for full wave signal integrity analysis*
- *8 metallization layers and 40,000 devices*
- *The FD solver used 27 million mesh nodes and 5.3 million tetrahedrons*
- *The transient solver model of the full package had 640 million mesh cells and 3.7 billion of unknowns*



Avoiding brute force

- The behaviour of this MOS transistor can be simulated by solving a system of 3 partial differential equations → discrete system for at least 30000 unknowns
- Insight?
- Even worse: an electronic circuit consists of 10^4-10^6 MOS devices
- Solution: **compact device model** (made by physicists/engineers based on many device simulations, ~50 unk)



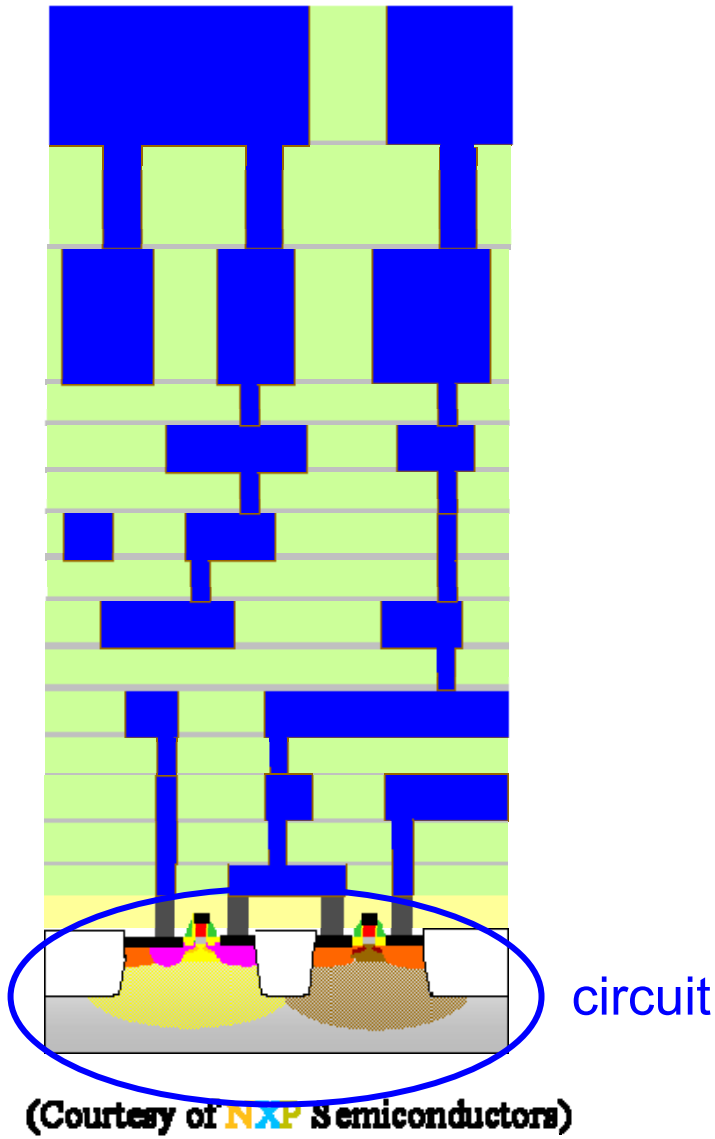
Can we construct such compact models in an automated way?

Another example

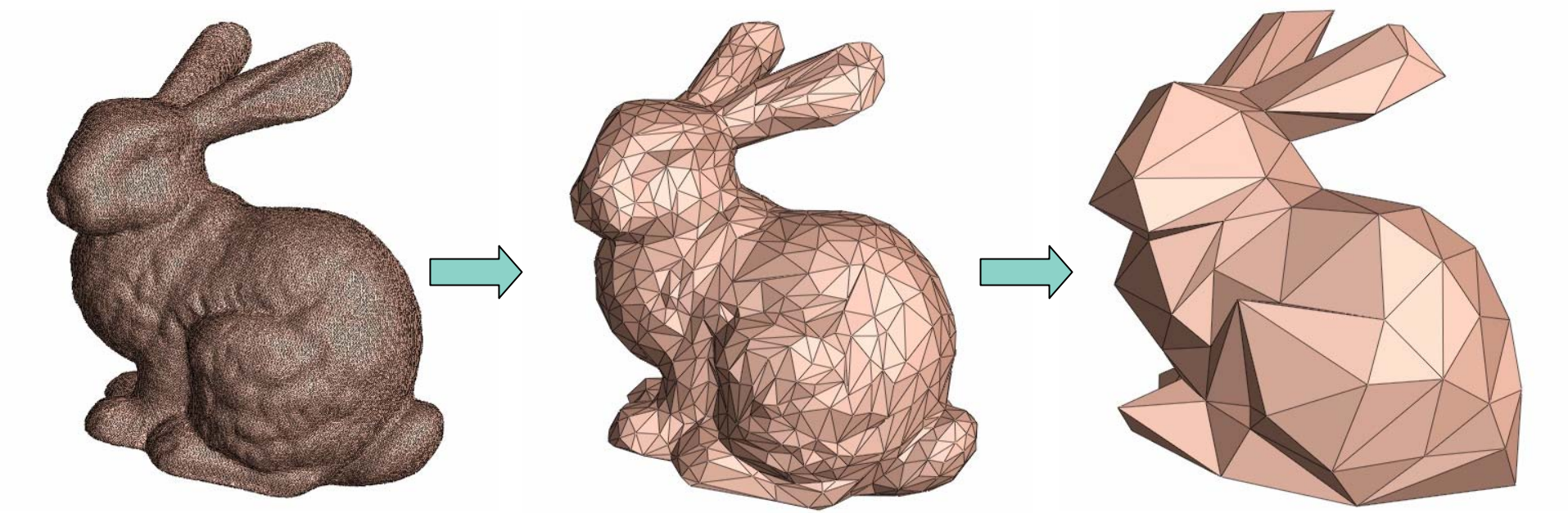
- Integrated circuits need 3-D structure for wiring
- 10 years ago
  - 1-2 layers of metal, no influence on circuit performance
- Present situation:
  - 8-10 layers of metal
  - Delay of signals, parasitic effects due to high frequencies
- 3-D solution of Maxwell equations leads to millions of extra unknowns

Gradual

Can we construct a compact model for the interconnect in an automated way?



Model Order Reduction is about capturing dominant features



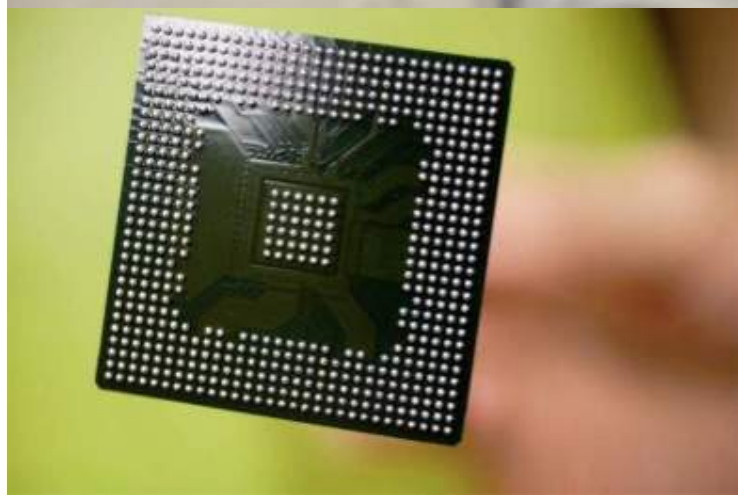
Strong link to numerical linear algebra

And, talking about motivation…..

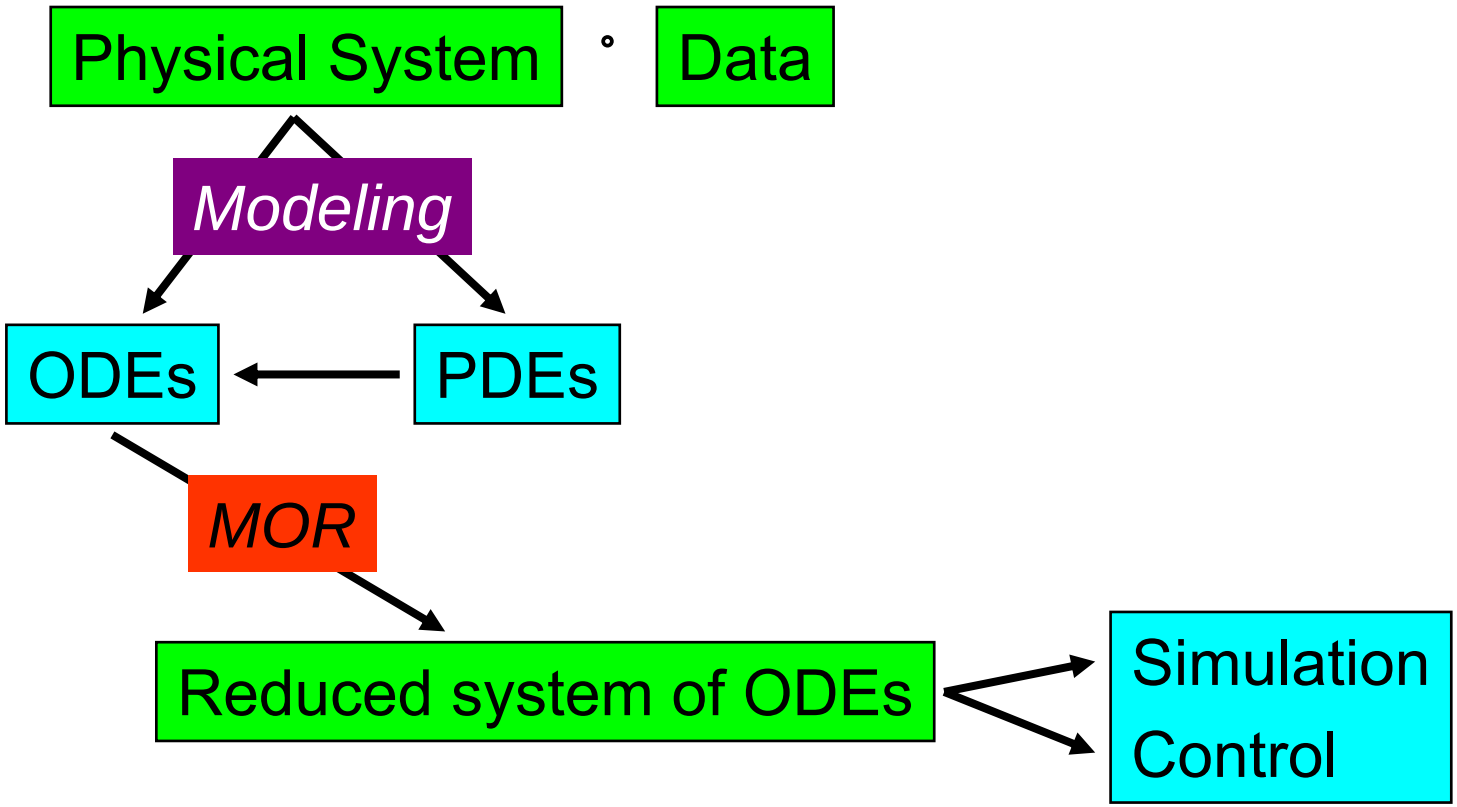
Mathematical Challenges!

Outline

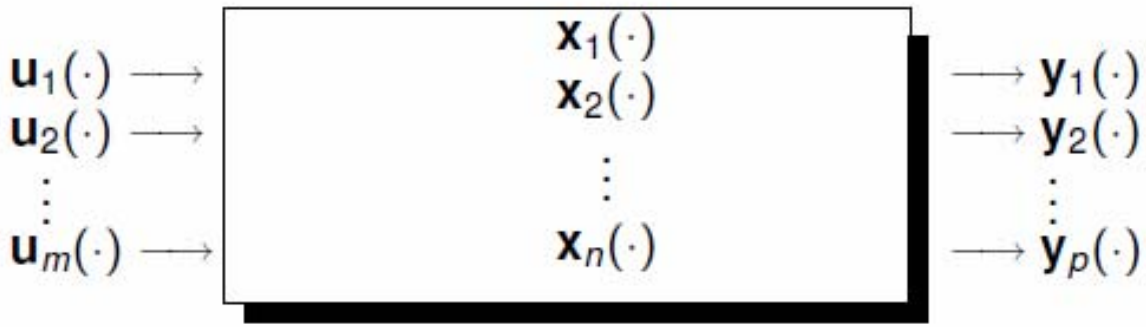
- Introduction and motivation
- Preliminaries
- Model order reduction basics
- Challenges in MOR
- Conclusions



Overall picture



Dynamical systems



We consider explicit state equations

$$\Sigma: \quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \ \mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t))$$

with **state** $\mathbf{x}(\cdot)$ of dimension $n \gg m, p$.

Problem statement

Given: dynamical system

$$\Sigma = (\mathbf{f}, \mathbf{h}) \text{ with: } \mathbf{u}(t) \in \mathbb{R}^m, \mathbf{x}(t) \in \mathbb{R}^n, \mathbf{y}(t) \in \mathbb{R}^p.$$

Problem: Approximate Σ with:

$$\hat{\Sigma} = (\hat{\mathbf{f}}, \hat{\mathbf{h}}) \text{ with : } \mathbf{u}(t) \in \mathbb{R}^m, \hat{\mathbf{x}}(t) \in \mathbb{R}^k, \hat{\mathbf{y}}(t) \in \mathbb{R}^p, k \ll n :$$

(1) Approximation error small - global error bound

(2) Preservation of stability/passivity

(3) Procedure must be computationally efficient

Approximation by projection

Unifying feature of approximation methods: **projections**.

Let $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n\times k}$, such that $\mathbf{W}^*\mathbf{V} = \mathbf{I}_k \Rightarrow \Pi = \mathbf{V}\mathbf{W}^*$ is a projection.
Define $\hat{\mathbf{x}} = \mathbf{W}^*\mathbf{x}$. Then

$$\hat{\Sigma}: \begin{cases} \frac{d}{dt}\hat{\mathbf{x}}(t) &= \mathbf{W}^*\mathbf{f}(\mathbf{V}\hat{\mathbf{x}}(t),\ \mathbf{u}(t)) \\ \mathbf{y}(t) &= \mathbf{h}(\mathbf{V}\hat{\mathbf{x}}(t),\ \mathbf{u}(t)) \end{cases}$$

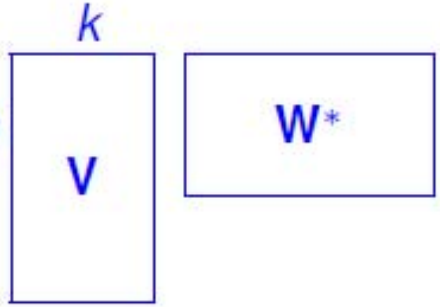
Thus $\hat{\Sigma}$ is "good" approximation of Σ, if $\mathbf{x} - \Pi\mathbf{x}$ is "small".

Special case: linear dynamical systems

$$\Sigma: \mathbf{E}\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t),\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)$$

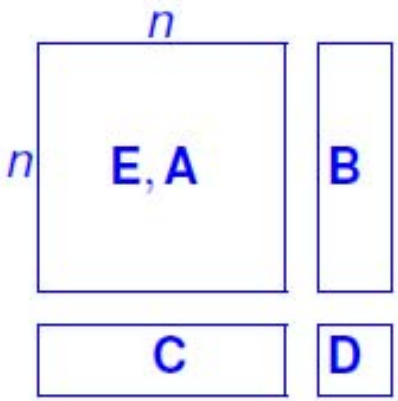
$$\Sigma = \left(\begin{array}{c|c} \mathbf{E}, \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right)$$

Problem: Approximate Σ by **projection**: $\Pi = \mathbf{V}\mathbf{W}^* =$

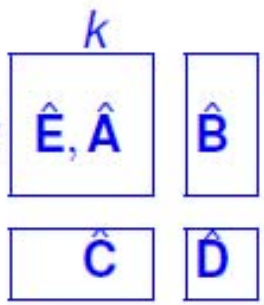


$$\hat{\Sigma} = \left(\begin{array}{c|c} \hat{\mathbf{E}}, \hat{\mathbf{A}} & \hat{\mathbf{B}} \\ \hline \hat{\mathbf{C}} & \hat{\mathbf{D}} \end{array}\right) = \left(\begin{array}{c|c} \mathbf{W}^*\mathbf{E}\mathbf{V}, \mathbf{W}^*\mathbf{A}\mathbf{V} & \mathbf{W}^*\mathbf{B} \\ \hline \mathbf{C}\mathbf{V} & \mathbf{D} \end{array}\right),\ k \ll n$$

Σ:



$\Rightarrow$



$: \hat{\Sigma}$

Norms:

- $\mathcal{H}_\infty$-norm:
worst output error
$\|\mathbf{y}(t) - \hat{\mathbf{y}}(t)\|$ for $\|\mathbf{u}(t)\| = 1$.
- $\mathcal{H}_2$-norm: $\|\mathbf{h}(t) - \hat{\mathbf{h}}(t)\|$

Singular value decomposition

$$A = U\Sigma V^* \in \mathbb{R}^{n\times m}$$

- Singular values: $\Sigma = \mathrm{diag}\,(\sigma_1, \cdots, \sigma_n)$, $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$

  $\Rightarrow \sigma_i = \sqrt{\lambda_i(A^*A)}$
- left singular vectors: $U = (u_1 \;\; u_2 \;\; \cdots \;\; u_n)$, $UU^* = I_n$
- right singular vectors: $V = (v_1 \;\; v_2 \;\; \cdots \;\; v_m)$, $VV^* = I_m$
- Dyadic decomposition:

$$A = \sigma_1 u_1 v_1^* + \sigma_2 u_2 v_2^* + \cdots + \sigma_n u_n v_n^*$$

- σ_1: 2-induced norm of A

Optimal approximation in 2-norm

- Given: $A \in \mathbb{R}^{n\times m}$
- find: $X \in \mathbb{R}^{n\times m}$, $\operatorname{rank} X = k < \operatorname{rank} A$
- Criterion: norm(error) is minimized, where
  error: $E = A - X$, norm: 2-norm

Theorem (*Schmidt-Mirsky, Eckart-Young*)

$$\min_{\operatorname{rank} X \leq k} \| A - X \|_2 = \sigma_{k+1}(A)$$

Minimizer (non-unique): truncation of dyadic decomposition of A:

$$X_{\#} = \sigma_1 u_1 v_1^* + \sigma_2 u_2 v_2^* + \cdots \sigma_k u_k v_k^*$$

Operators and norms

Impulse response

$$\Sigma = \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right) : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{cases} \Rightarrow \begin{cases} \mathbf{h}(t) = \mathbf{C}e^{\mathbf{A}t}\mathbf{B} + \mathbf{D}\delta(t), t \geq 0 \\ \mathbf{H}(s) = \mathbf{D} + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \end{cases}$$

Transfer function

Convolution operator $\mathcal{S}$:

$\mathcal{S} : \mathbf{u} \mapsto \mathbf{y}, \mathbf{y}(t) = \int_{-\infty}^{\infty} \mathbf{h}(t-\tau)\mathbf{u}(\tau)d\tau, t \in \mathbb{R}$

Singular values: let $\underline{\sigma} := \inf_\omega \sigma_{min}(\mathbf{H}(j\omega))$ and $\bar{\sigma} := \sup_\omega \sigma_{max}(\mathbf{H}(j\omega))$.

$$\sigma \text{ singular value of } \mathcal{S} \quad \Leftrightarrow \quad \sigma \in [\underline{\sigma}, \bar{\sigma}]$$

Hankel operator $\mathcal{H}$:

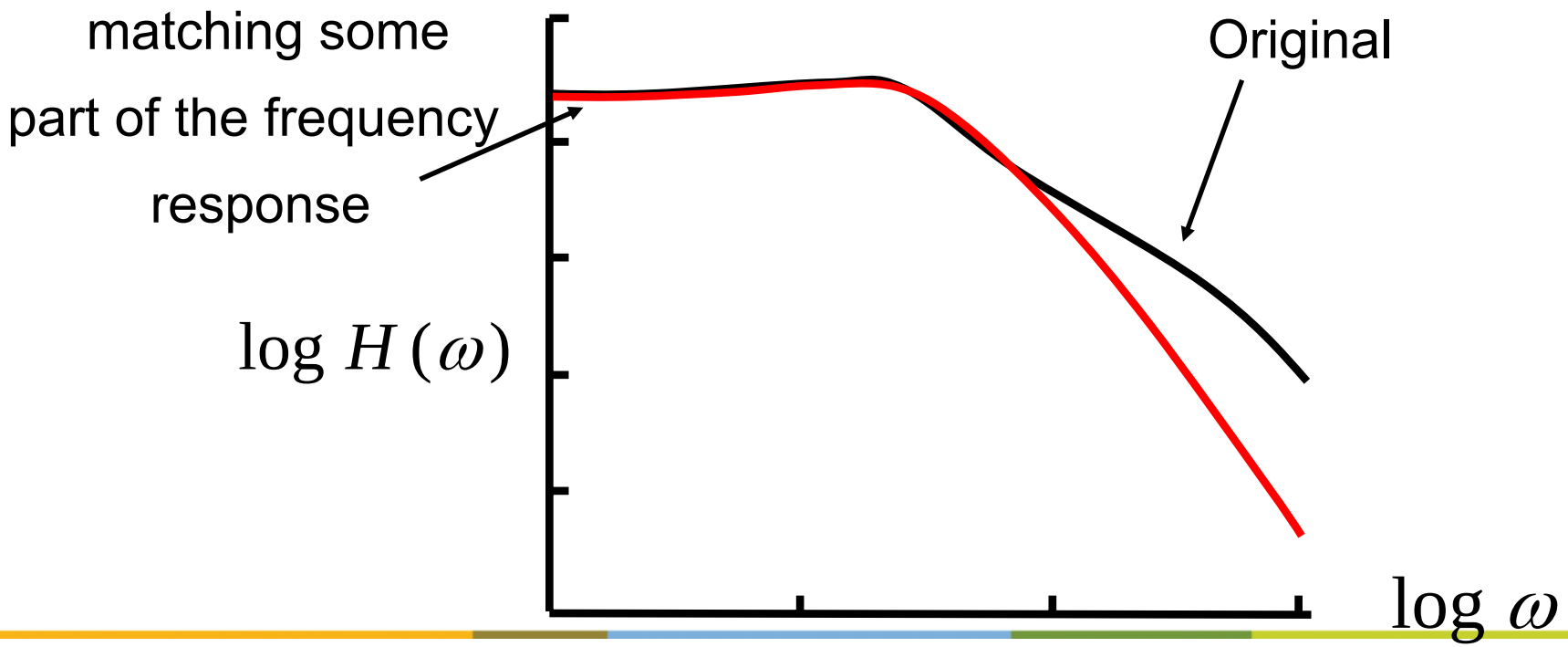
$\mathcal{H} : \mathbf{u}_- \mapsto \mathbf{y}_+, \mathbf{y}_+(t) = \int_{-\infty}^{0} \mathbf{h}(t-\tau)\mathbf{u}_-(\tau)d\tau, t \in \mathbb{R}_-$

Singular values: solve for continuous-time **Lyapunov equations**

$$\sigma_i^2(\mathcal{H}) = \lambda_i(\mathcal{H}^*\mathcal{H}) = \lambda_i(\mathcal{P}\mathcal{Q}) \quad \text{where} \quad \begin{cases} \mathbf{A}\mathcal{P} + \mathcal{P}\mathbf{A}^* + \mathbf{B}\mathbf{B}^* = \mathbf{0} \\ \mathbf{A}^*\mathcal{Q} + \mathcal{Q}\mathbf{A} + \mathbf{C}^*\mathbf{C} = \mathbf{0} \end{cases}$$

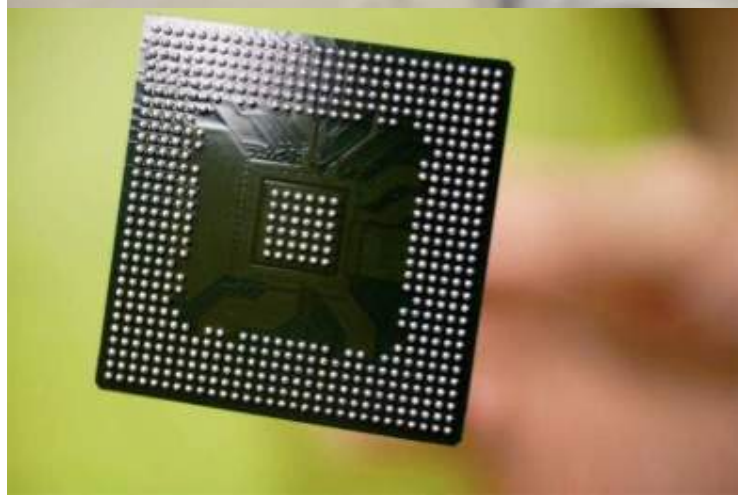
Important observation

- **Model Order Reduction methods approximate the transfer function**
- **In other words: not the entire internal characteristics of the problem, but only the relation between input and output (so-called external variables)**
- **No need to approximate the internal variables (“states”), but some of these may need to be kept to obtain good representation of input-output characteristics**

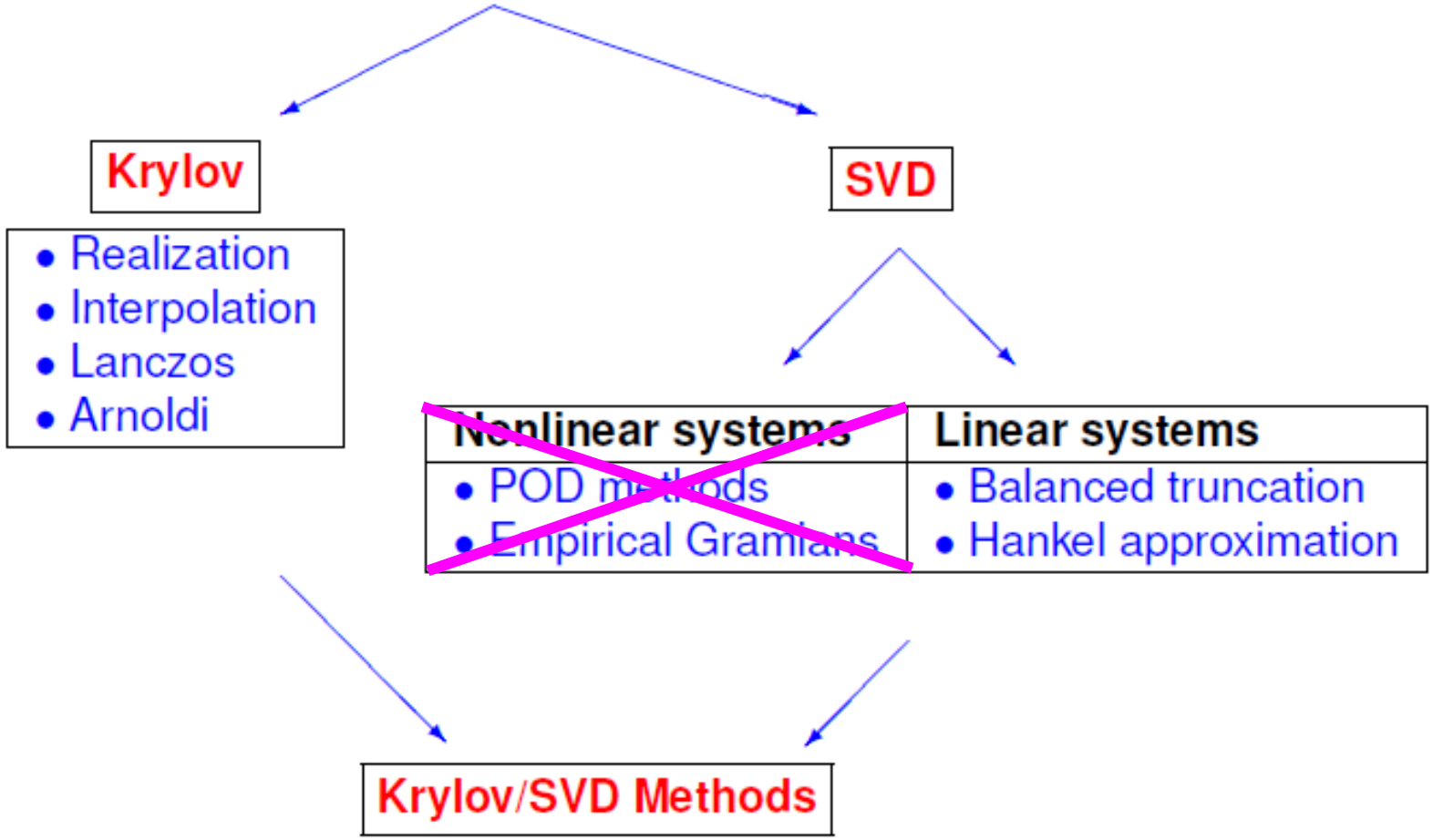


Outline

- Introduction and motivation
- Preliminaries
- Model order reduction basics
- Challenges in MOR
- Conclusions



Overview of MOR methods



Balanced truncation

Trade-off between accuracy and complexity for linear dynamical systems is provided by the **Hankel Singular Values**. Define the **gramians** as solutions of the Lyapunov equations

$$\left.\begin{aligned} \mathbf{AP} + \mathbf{PA}^* + \mathbf{BB}^* &= \mathbf{0}, \quad \mathbf{P} > \mathbf{0} \\ \mathbf{A}^*\mathbf{Q} + \mathbf{QA} + \mathbf{C}^*\mathbf{C} &= \mathbf{0}, \quad \mathbf{Q} > \mathbf{0} \end{aligned}\right\} \Rightarrow \boxed{\sigma_i = \sqrt{\lambda_i(\mathbf{PQ})}}$$

σ_i: **Hankel singular values** of the system. There exists balanced basis where $\mathbf{P} = \mathbf{Q} = \mathbf{S} = \operatorname{diag}(\sigma_1, \cdots, \sigma_n)$. In this basis partition:

$$\mathbf{A} = \left(\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{A}_{21} & \mathbf{A}_{22} \end{array}\right), \quad \mathbf{B} = \left(\begin{array}{c} \mathbf{B}_1 \\ \hline \mathbf{B}_2 \end{array}\right), \quad \mathbf{C} = (\mathbf{C}_1 \mid \mathbf{C}_2), \quad \mathbf{S} = \left(\begin{array}{c|c} \Sigma_1 & 0 \\ \hline 0 & \Sigma_2 \end{array}\right).$$

The reduced system is obtained by balanced truncation

$\left(\begin{array}{c|c} \mathbf{A}_{11} & B_1 \\ \hline C_1 & \end{array}\right)$, where Σ_2 contains the small Hankel singular values.

Interpretation of balanced truncation

Given state $\mathbf{x}$:

$\mathcal{E}_r$: min. input energy steering $\quad \mathbf{0} \rightarrow \mathbf{x} \quad \Rightarrow \quad \mathcal{E}_r = \mathbf{x}^* \mathcal{P}^{-1} \mathbf{x}$

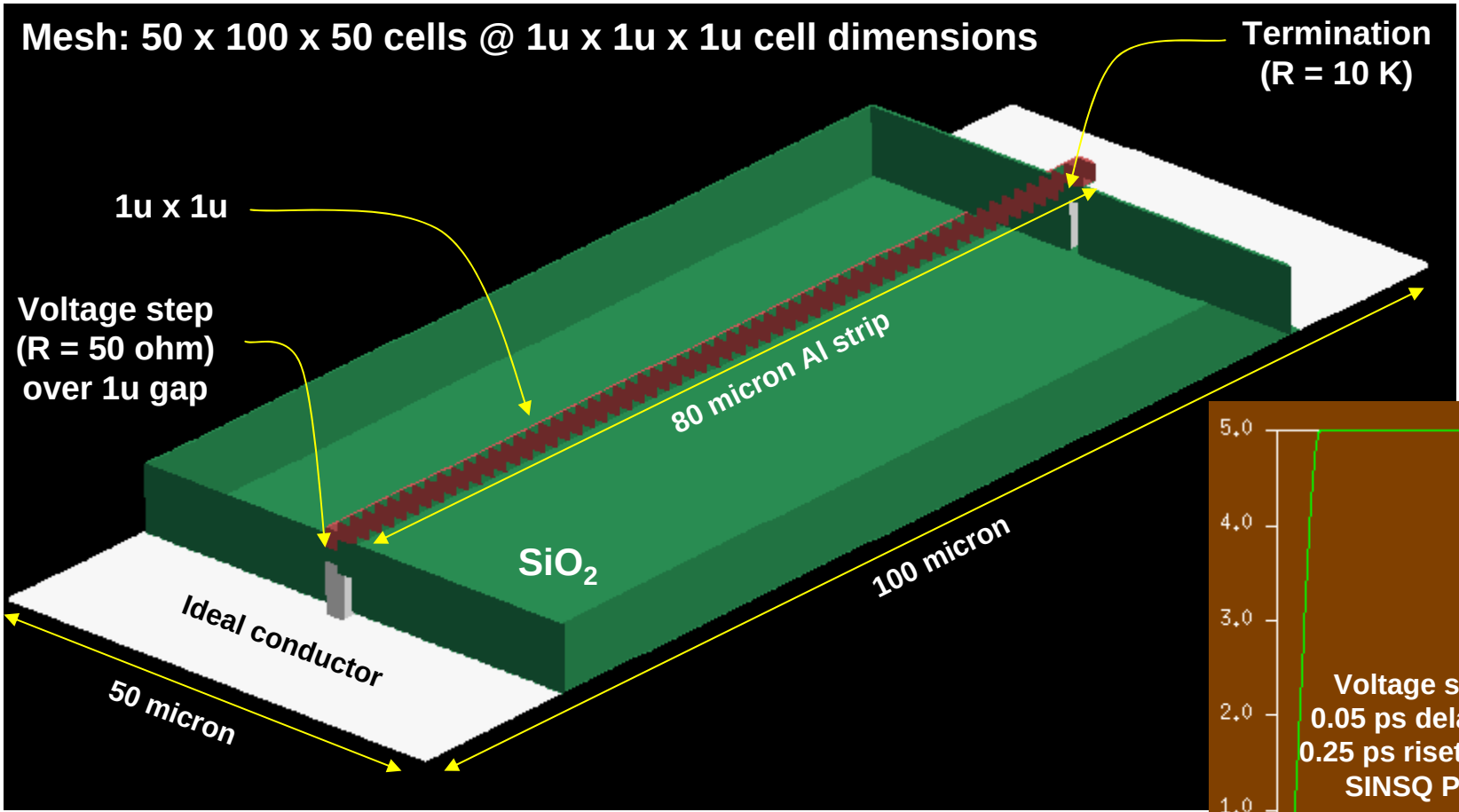
$\mathcal{E}_o$: output observation energy $\quad \mathbf{x} \rightarrow \mathbf{0} \quad \Rightarrow \quad \mathcal{E}_o = \mathbf{x}^* Q \mathbf{x}$

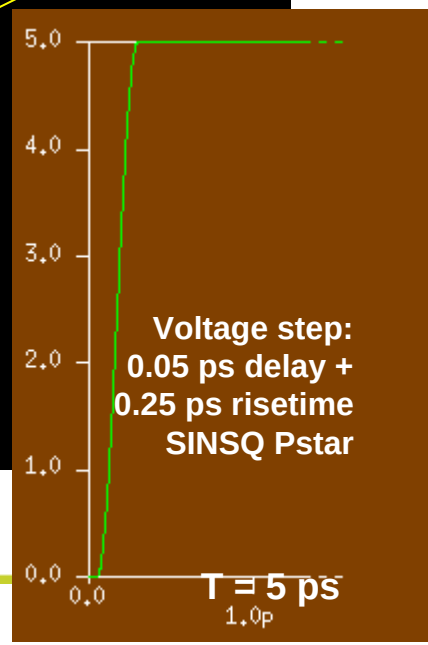
Properties of balanced truncation

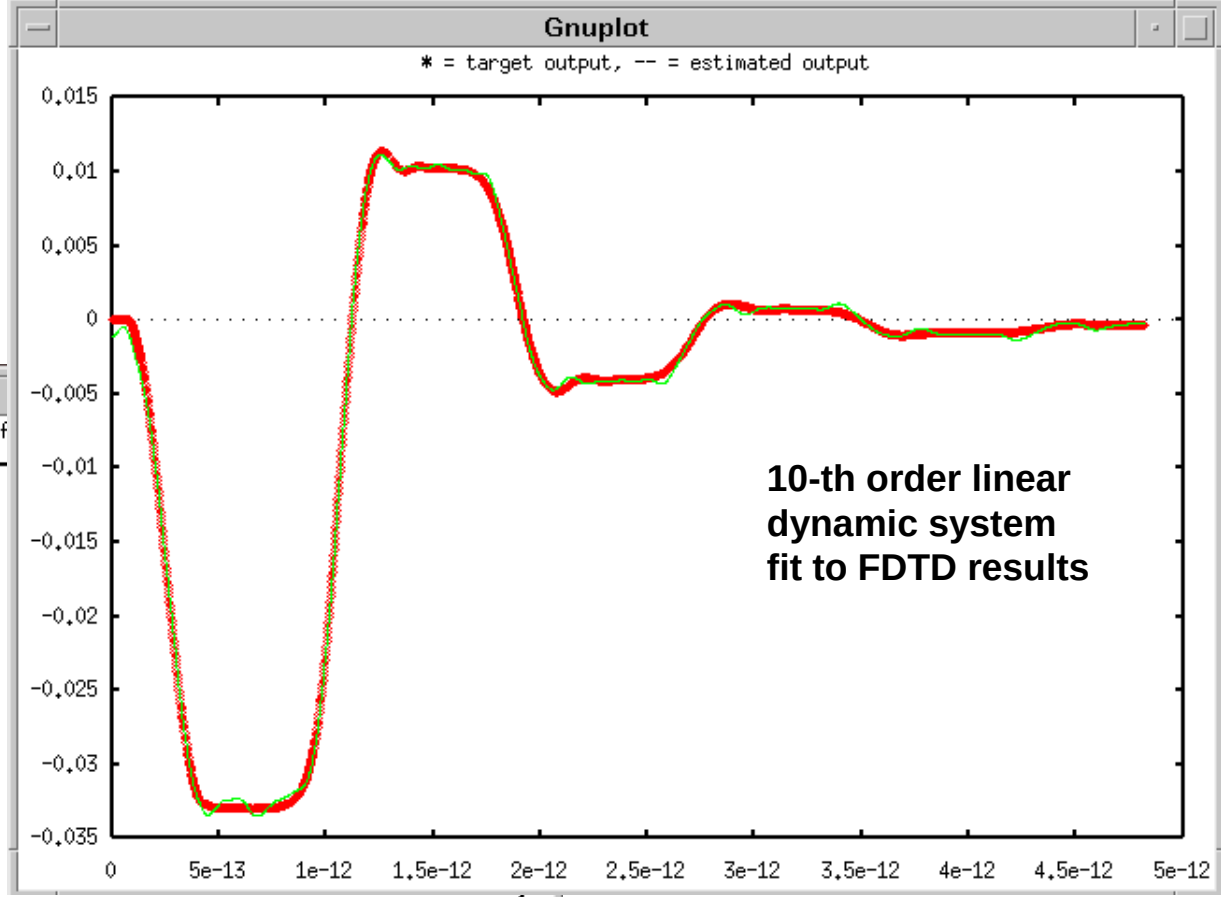
1. **Stability is preserved**
2. **Global error bound:**

$$\sigma_{k+1} \leq \| \Sigma - \hat{\Sigma} \|_\infty \leq 2(\sigma_{k+1} + \cdots + \sigma_n)$$

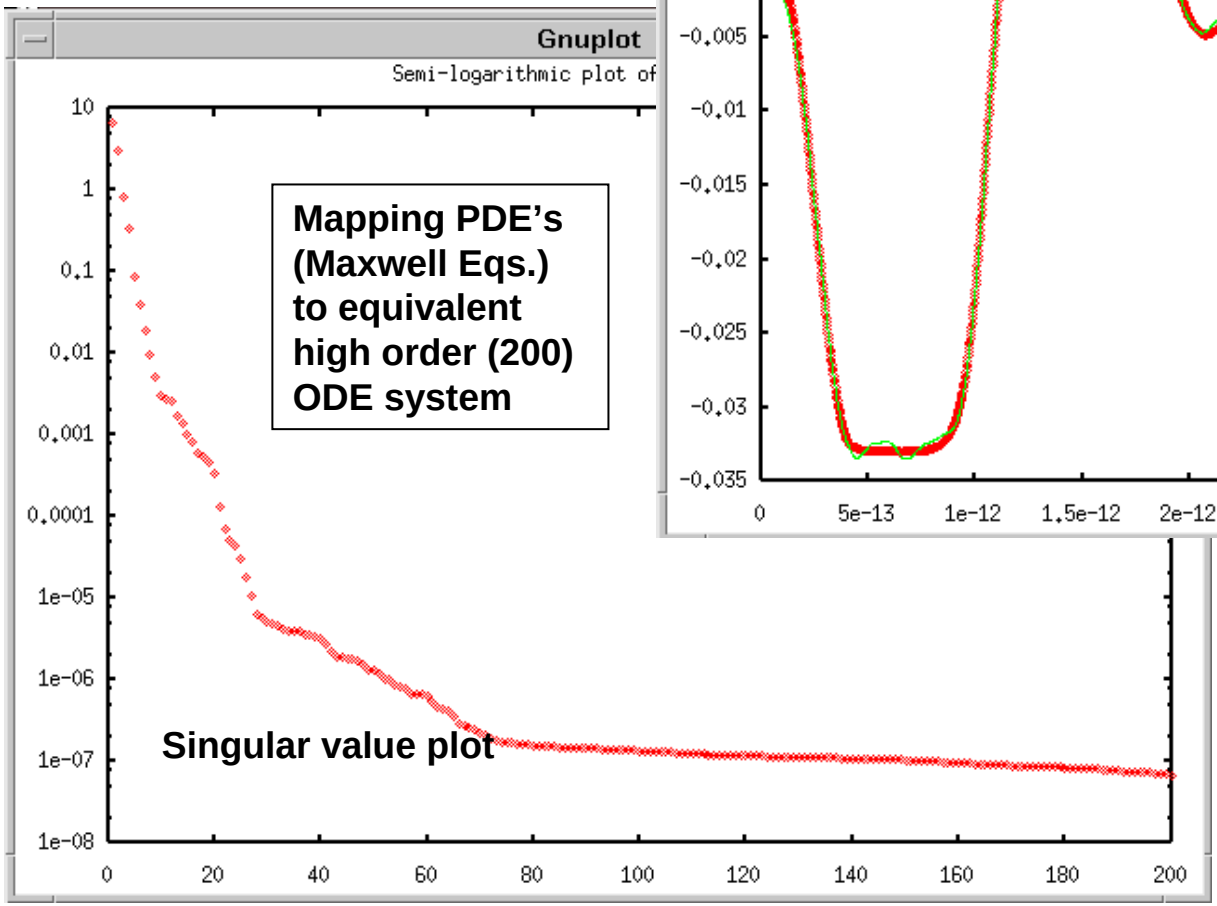
Simple example







10-th order linear dynamic system fit to FDTD results



Mapping PDE's (Maxwell Eqs.) to equivalent high order (200) ODE system

Singular value plot

Moment matching (AWE, 1991)

Given $\Sigma = \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right)$, expand the transfer function around s_0:

$$\mathbf{H}(s) = \eta_0 + \eta_1(s - s_0) + \eta_2(s - s_0)^2 + \eta_3(s - s_0)^3 + \cdots$$

Moments at s_0: η_j, $j \geq 0$. Find $\hat{\Sigma} = \left(\begin{array}{c|c} \hat{\mathbf{A}} & \hat{\mathbf{B}} \\ \hline \hat{\mathbf{C}} & \hat{\mathbf{D}} \end{array}\right)$, with

$$\hat{\mathbf{H}}(s) = \hat{\eta}_0 + \hat{\eta}_1(s - s_0) + \hat{\eta}_2(s - s_0)^2 + \hat{\eta}_3(s - s_0)^3 + \cdots$$

such that for appropriate k:

$$\boxed{\eta_j = \hat{\eta}_j,\ j = 1, 2, \cdots, k}$$

Asymptotic Waveform Evaluation (AWE)

- Direct calculation of moments suffers from severe problems:
  - As #moments increases, the matrix in linear system becomes extremely ill-conditioned → at most 6-8 poles accurately
  - Approximate system often has instable poles (real part > 0)
  - AWE does not guarantee passivity

But:
basis for many new developments in MOR!

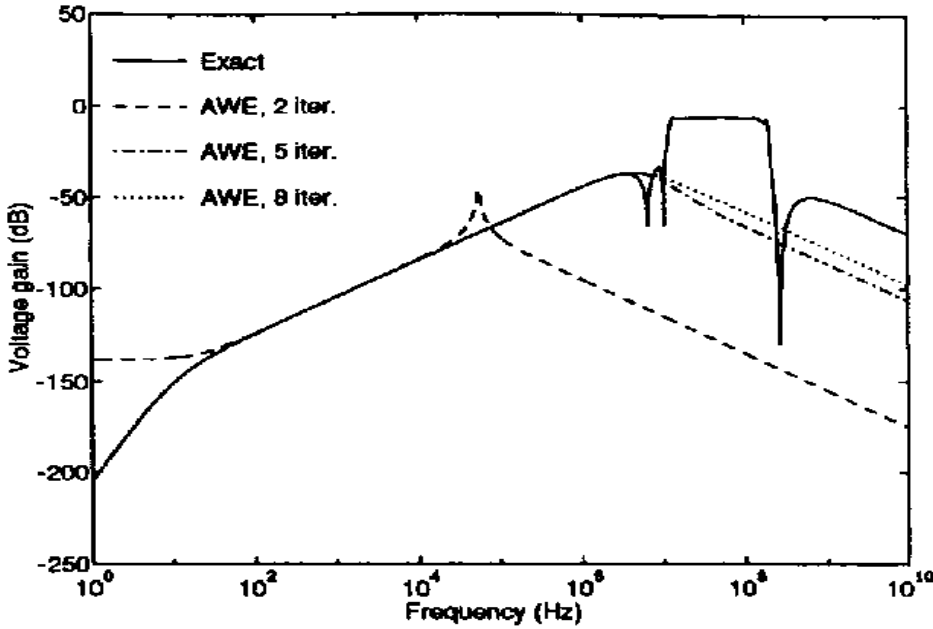


Fig. 1. Results for simulation of voltage gain with AWE.

Moment matching (AWE, 1991)

Given $\Sigma = \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right)$, expand the transfer function around s_0:

$$\mathbf{H}(s) = \eta_0 + \eta_1(s - s_0) + \eta_2(s - s_0)^2 + \eta_3(s - s_0)^3 + \cdots$$

Moments at s_0: η_j, $j \geq 0$. Find $\hat{\Sigma} = \left(\begin{array}{c|c} \hat{\mathbf{A}} & \hat{\mathbf{B}} \\ \hline \hat{\mathbf{C}} & \hat{\mathbf{D}} \end{array}\right)$, with

$$\hat{\mathbf{H}}(s) = \hat{\eta}_0 + \hat{\eta}_1(s - s_0) + \hat{\eta}_2(s - s_0)^2 + \hat{\eta}_3(s - s_0)^3 + \cdots$$

such that for appropriate k:

$$\boxed{\eta_j = \hat{\eta}_j,\ j = 1, 2, \cdots, k}$$

Moment matching methods can be implemented in a **numerically stable** and **efficient way**.

Arnoldi and Lanczos methods

Efficient Linear Circuit Analysis by Padé Approximation via the Lanczos Process

Peter Feldmann, *Member, IEEE*, and Roland W. Freund

***Abstract*—In this paper, we introduce PVL, an algorithm for computing the Padé approximation of Laplace-domain transfer functions of large linear networks via a Lanczos process. The PVL algorithm has significantly superior numerical stability, while retaining the same efficiency as algorithms that compute the Padé approximation directly through moment matching, such as AWE [1], [2] and its derivatives. As a consequence, it produces more accurate and higher-order approximations, and it renders unnecessary many of the heuristics that AWE and its derivatives had to employ. The algorithm also computes an error bound that permits to identify the true poles and zeros of the original network. We present results of numerical experiments with the PVL algorithm for several large examples.**

I. Introduction

CIRCUIT simulation tasks, such as the accurate prediction of interconnect effects at the board and chip level, or analog circuit analysis with full accounting of parasitic elements, may require the solution of large linear networks. These networks can become extremely large, especially when circuits

Despite its spectacular success, AWE suffers from a number of fundamental numerical limitations. In particular, each run of AWE produces only a fairly small number of accurate poles and zeros. The proposed remedial techniques, such as scaling, frequency shifting, and complex frequency hopping, are sometimes heuristic, hard to apply automatically, and may be computationally expensive. Another shortcoming of AWE is the absence of a theoretically solid procedure to predict the accuracy of the approximating reduced-order model [3], [6], [7].

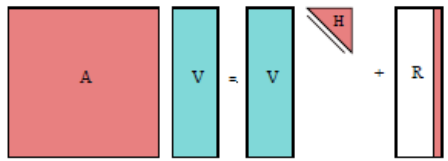
In this paper, we introduce a new, numerically stable algorithm that computes the Padé approximation of a linear circuit via the Lanczos process [8]. This algorithm, called PVL (**P**adé **V**ia **L**anczos), can be used to generate an arbitrary number of poles and zeros (even all of them) with little numerical degradation. Moreover, PVL computes a quality measure for the poles and zeros it produces. The computational cost per order of approximation is practically the same as for AWE.

The Arnoldi method

Given is $\mathbf{A} \in \mathbb{R}^{n\times n}$, and $\mathbf{b} \in \mathbb{R}^n$. Let $\mathcal{R}_k(\mathbf{A}, \mathbf{b}) \in \mathbb{R}^{n\times k}$ be the reachability or Krylov matrix. It is *assumed* that $\mathcal{R}_k$ has full column rank equal to k.

Devise a process which is iterative and at the k^{th} step we have

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_k\mathbf{H}_k + \mathbf{R}_k,\ \mathbf{V}_k,\ \mathbf{R}_k \in \mathbb{R}^{n\times k},\ \mathbf{H}_k \in \mathbb{R}^{k\times k},\ k = 1, 2, \cdots, n$$



These quantities have to satisfy the following conditions at each step.

- The columns of $\mathbf{V}_k$ are orthonormal: $\mathbf{V}_k^*\mathbf{V}_k = \mathbf{I}_k,\ k = 1, 2, \cdots, n.$
- $\text{span col}\,\mathbf{V}_k = \text{span col}\,\mathcal{R}_k(\mathbf{A}, \mathbf{b}),\ k = 1, 2, \cdots, n$
- The residual R_k satisfies the Galerkin condition: $\mathbf{V}_k^*\mathbf{R}_k = 0$, $k = 1, 2, \cdots, n.$

This problem can be solved by the **Arnoldi procedure**.

The Arnoldi method (2)

Given: $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{b} \in \mathbb{R}^n$

Find: $\mathbf{V} \in \mathbb{R}^{n\times k}$, $\mathbf{f} \in \mathbb{R}^n$, and $\mathbf{H} \in \mathbb{R}^{k\times k}$, such that

$$\mathbf{AV} = \mathbf{VH} + \mathbf{f}\mathbf{e}_k^* \quad \text{where} \quad \mathbf{H} = \mathbf{V}^*\mathbf{AV},\ \mathbf{V}^*\mathbf{V} = \mathbf{I}_k,\ \mathbf{V}^*\mathbf{f} = 0,$$

with $\mathbf{H}$ in *upper Hessenberg* form.

1. $\mathbf{v}_1 = \frac{\mathbf{b}}{\|\mathbf{b}\|}$, $\mathbf{w} = \mathbf{A}\mathbf{v}_1$; $\alpha_1 = \mathbf{v}_1^*\mathbf{w}$
   $\mathbf{f}_1 = \mathbf{w} - \mathbf{v}_1\alpha_1$; $\mathbf{V}_1 = (\mathbf{v}_1)$; $\mathbf{H}_1 = (\alpha_1)$
2. For $j = 1, 2, \cdots, k-1$
   1. $\beta_j = \| \mathbf{f}_j \|$, $\mathbf{v}_{j+1} = \frac{\mathbf{f}_j}{\beta_j}$
   2. $\mathbf{V}_{j+1} = (\mathbf{V}_j \ \ \mathbf{v}_{j+1})$, $\hat{\mathbf{H}}_j = \begin{pmatrix} \mathbf{H}_j \\ \beta_j \mathbf{e}_j^* \end{pmatrix}$
   3. $\mathbf{w} = \mathbf{A}\mathbf{v}_{j+1}$, $\mathbf{h} = \mathbf{V}_{j+1}^*\mathbf{w}$, $\mathbf{f}_{j+1} = \mathbf{w} - \mathbf{V}_{j+1}\mathbf{h}$
   4. $\mathbf{H}_{j+1} = \begin{pmatrix} \hat{\mathbf{H}}_j & \mathbf{h} \end{pmatrix}$

The Lanczos method

If $\mathbf{A} = \mathbf{A}^*$ then the Arnoldi procedure is the same as the symmetric Lanczos procedure. In this case $\mathbf{H}_k$ is tridiagonal:

$$\mathbf{H}_k = \begin{bmatrix} \alpha_1 & \beta_2 & & & & \\ \beta_2 & \alpha_2 & \beta_3 & & & \\ & \beta_3 & \alpha_3 & & & \\ & & \ddots & \ddots & \ddots & \\ & & & & \alpha_{k-1} & \beta_k \\ & & & & \beta_k & \alpha_k \end{bmatrix}$$

This matrix shows that the vectors in the Lanczos procedure satisfy a *three term recurrence* relationship

$$\mathbf{A}\mathbf{v}_i = \beta_{i+1}\mathbf{v}_{i+1} + \alpha_i\mathbf{v}_i + \beta_i\mathbf{v}_{i-1}, \; i = 1, 2, \cdots, k-1$$

Remark. If the remainder $\mathbf{r}_k = 0$, the procedure has terminated, in which case if $(\lambda, \mathbf{x})$ is an eigenpair of $\mathbf{H}_k$, $(\lambda, \mathbf{V}_k x)$ is an eigenpair of $\mathbf{A}$ (since $\mathbf{H}_k\mathbf{x} = \lambda\mathbf{x}$ implies $\mathbf{A}\mathbf{V}_k\mathbf{x} = \mathbf{V}_k\mathbf{H}_k\mathbf{x} = \lambda\mathbf{V}_k\mathbf{x}$).

Two-sided Lanczos

The two-sided Lanczos procedure. Given $\mathbf{A} \in \mathbb{R}^{n\times n}$ which is not symmetric, and two vectors $\mathbf{b}, \mathbf{c}^* \in \mathbb{R}^n$, devise a process which is iterative and the k^{th} step there holds:

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_k\mathbf{H}_k + \mathbf{R}_k, \ \mathbf{A}^*\mathbf{W}_k = \mathbf{W}_k\mathbf{H}_k + \mathbf{S}_k, \ \ k = 1, 2, \cdots, n.$$

- Biorthogonality: $\mathbf{W}_k^*\mathbf{V}_k = \mathbf{I}_k$,
- span col $\mathbf{V}_k$ = span col $\mathcal{R}_k(\mathbf{A}, \mathbf{b})$, span col $\mathbf{W}_k$ = span col $\mathcal{R}_k(\mathbf{A}^*, \mathbf{c}^*)$,
- Galerkin conditions: $\mathbf{V}_k^*\mathbf{S}_k = 0$, $\mathbf{W}_k^*\mathbf{R}_k = 0$, $k = 1, 2, \cdots, n$.

Moment matching using Arnoldi (1)

The Arnoldi factorization can be used for model reduction as follows. Recall the QR factorization of the reachability matrix $\mathcal{R}_k \in \mathbb{R}^{n\times k}$; a projection $\mathbf{V}\mathbf{V}^*$ can then be attached to this factorization:

$$\mathcal{R}_k = \mathbf{V}\mathbf{U} \;\Rightarrow\; \mathbf{V} = \mathcal{R}_k\mathbf{U}^{-1}$$

where $\mathbf{V} \in \mathbb{R}^{n\times k}$, $\mathbf{V}^*\mathbf{V} = \mathbf{I}_k$, and $\mathbf{U}$ is upper triangular. The reduced order system is:

$$\bar{\Sigma} = \left(\begin{array}{c|c} \bar{\mathbf{A}} & \bar{\mathbf{B}} \\ \hline \bar{\mathbf{C}} & \end{array}\right) \quad \text{where} \quad \boxed{\bar{\mathbf{A}} = \mathbf{V}^*\mathbf{A}\mathbf{V}}, \quad \boxed{\bar{\mathbf{B}} = \mathbf{V}^*\mathbf{B}}, \quad \boxed{\bar{\mathbf{C}} = \mathbf{C}\mathbf{V}}$$

Theorem. $\bar{\Sigma}$ as defined above satisfies the equality of the Markov parameters $\hat{\eta}_i = \eta_i$, $i = 1, \cdots, k$. Furthermore, $\bar{\mathbf{A}}$ is in Hessenberg form, and $\bar{\mathbf{B}}$ is a multiple of the unit vector $\mathbf{e}_1$.

Moment matching using Arnoldi (2)

Proof. First notice that since $\mathbf{U}$ is upper triangular, $\mathbf{v}_1 = \frac{\mathbf{B}}{\|\mathbf{B}\|}$, and since $\mathbf{V}^*\mathcal{R}_k = \mathbf{U}$ it follows that $\bar{\mathbf{B}} = \mathbf{u}_1 = \| \mathbf{B} \| \mathbf{e}_1$; therefore $\bar{\mathbf{B}} = \mathbf{V}^*\mathbf{B}$. $\mathbf{V}\mathbf{V}^*\mathbf{B} = \mathbf{V}\bar{\mathbf{B}} = \mathbf{B}$, hence $\bar{\mathbf{A}}\bar{\mathbf{B}} = \mathbf{V}^*\mathbf{A}\mathbf{V}\mathbf{V}^*\mathbf{B} = \mathbf{V}^*\mathbf{A}\mathbf{B}$; in general, since $\mathbf{V}\mathbf{V}^*$ is a projection along the columns of $\mathcal{R}_k$, we have $\mathbf{V}\mathbf{V}^*\mathcal{R}_k = \mathcal{R}_k$; moreover: $\hat{\mathcal{R}}_k = \mathbf{V}^*\mathcal{R}_k$; hence

$$(\hat{\eta}_1 \ \cdots \ \hat{\eta}_k) = \hat{\mathbf{C}}\hat{\mathcal{R}}_k = \mathbf{C}\mathbf{V}\mathbf{V}^*\mathcal{R}_k = \mathbf{C}\mathcal{R}_k = (\eta_1 \ \cdots \ \eta_k)$$

Finally, the upper triangularity of $\mathbf{U}$ implies that $\mathbf{A}$ is in Hessenberg form. ∎

Remark.

Similarly, one can show that reduction by means the two-sided Lanczos procedure preserves $2k$ Markov parameters.

Example: MOR for transmission line (RC)

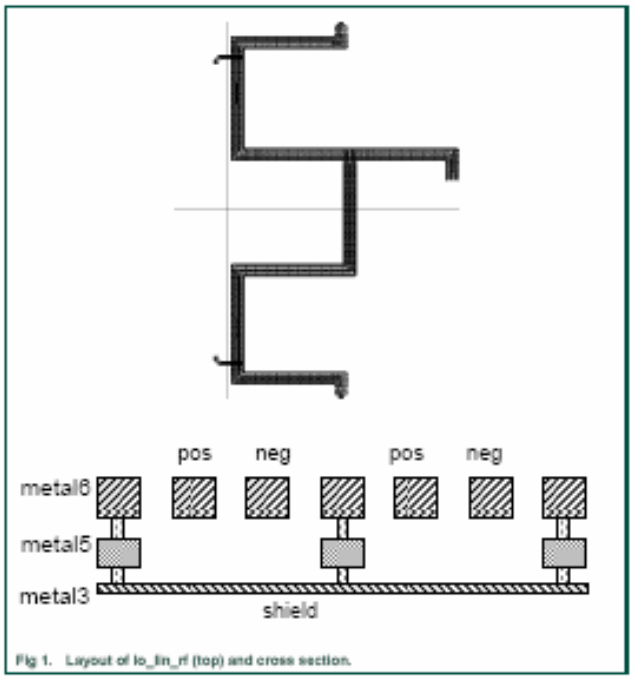
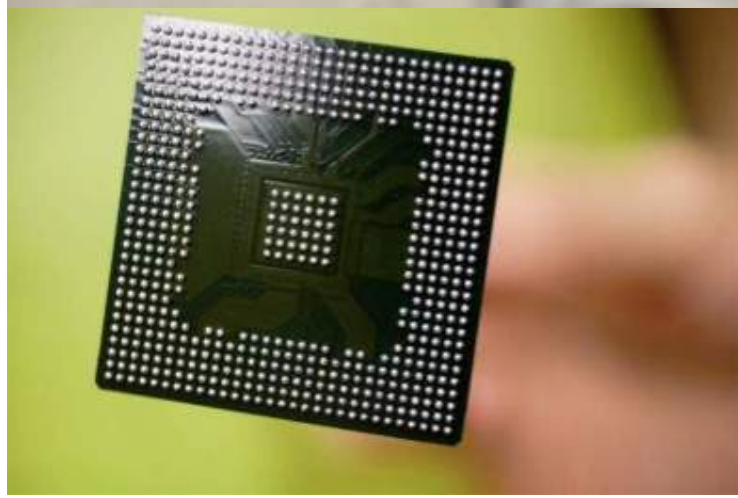


Fig 1. Layout of lo_lin_rf (top) and cross section.

	Original	Reduced
Netlist size (Mb)	0.53	0.003
# ports	22	22
# internal nodes	3231	12
# resistors	5892	28
# capacitors	3065	97

Outline

- Introduction and motivation
- Preliminaries
- Model order reduction basics
- Challenges in MOR
- Conclusions



Start of a new era!

- The Pade-via-Lanczos algorithm was published in 1994-1995 by Freund and Feldmann, and was a very important invention
- Not only did the method (PVL) solve the problems associated with AWE
- It also sparked a multitude of new developments, and a true explosion regarding the field of Model Order Reduction

AWE 1990 → **PVL 1995** →

- **Lanczos methods**
- **Arnoldi methods**
- **Laguerre methods**
- **Vector fitting**
- **……**

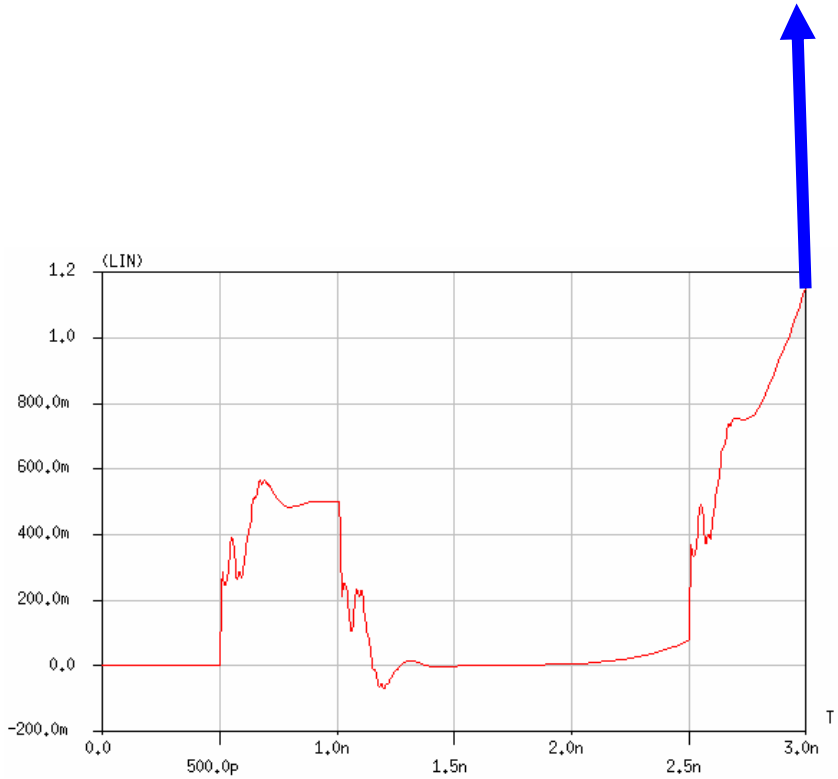
1995-2009

Topic 1: passivity

- A linear state space system is **passive** if its transfer function H(s) is **positive real**
  - H has no poles in C^+
  - H is real for real s
  - Real part of $\mathbf{z}^*H(s)\mathbf{z}$ non-negative for all s, **z**
- Passivity is important in practice (no energy generated)
- Lanczos-based methods such as PVL turned out not to be passive! →problem!!!
- Search started for methods that guarantee (provable) passivity
- **Passive MOR:**
  - PRIMA (Passive Reduced-order Interconnect Macromodeling Algorithm)
  - SVD-Laguerre (Knockaert & Dezutter)
  - Laguerre with intermediate orthogonalisation (Heres & Schilders)

Recent developments w.r.t. passivity

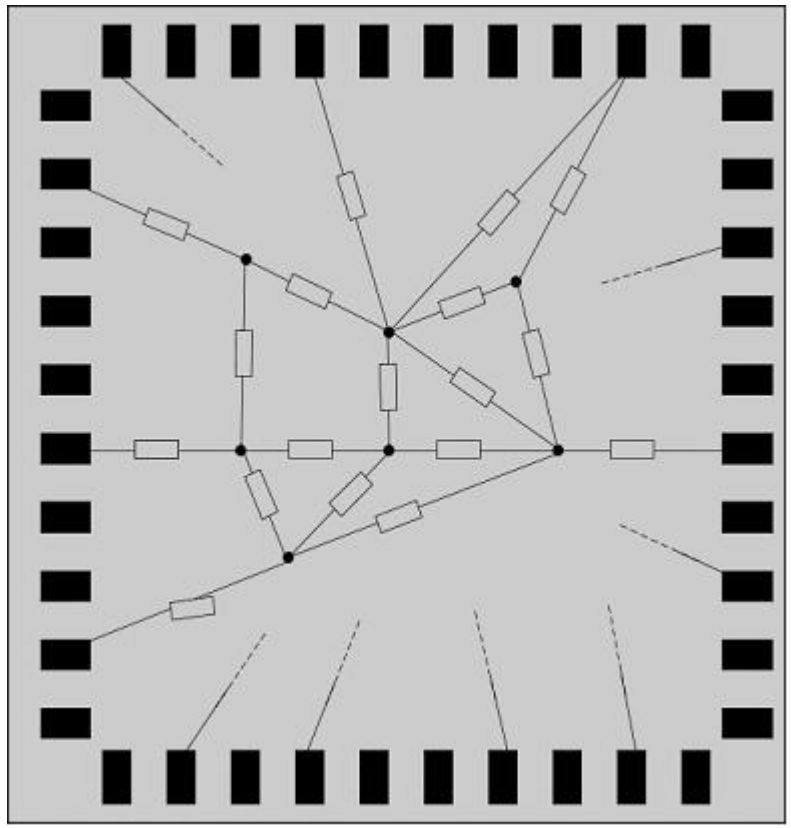
- Several passivity-preserving methods have been developed
- However: if the original discretized system is not passive, or if the data originate from experiments, passivity of the reduced order model cannot be guaranteed
- This happens frequently in practical situations: EM software generating an equivalent circuit model that is not passive
- In the frequency domain, this may not be too problematic, just an accuracy issue (S-parameters may exceed 1)
- However, it may become really problematic in time domain simulations



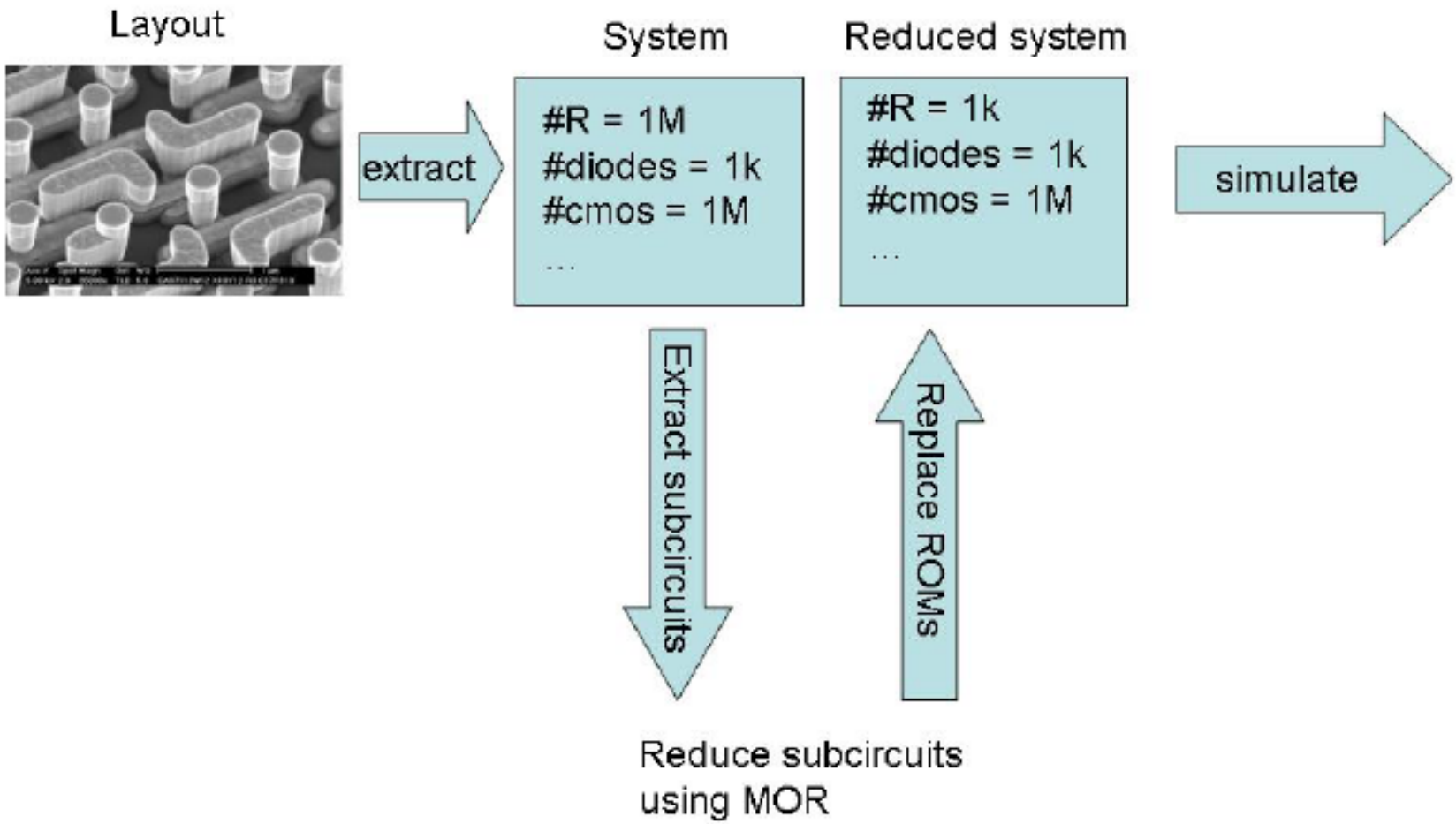
- **Passivity enforcement methods**
- **Often: trading accuracy for passivity**

Topic 2: Large resistor networks

- Obtained from extraction programs to model substrate and interconnect
- Networks are typically extremely large, up to millions of resistors and thousands of inputs/outputs
- Network typically contains:
  - Resistors
  - Internal nodes (“state variables”)
  - External nodes (connection to outside world, often to diodes)
- Model Order Reduction needed to drastically reduce number of internal nodes and resistors



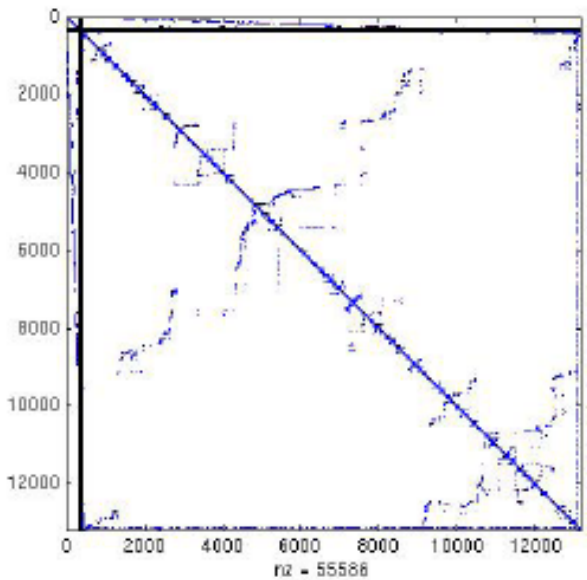
Reduction of resistor networks

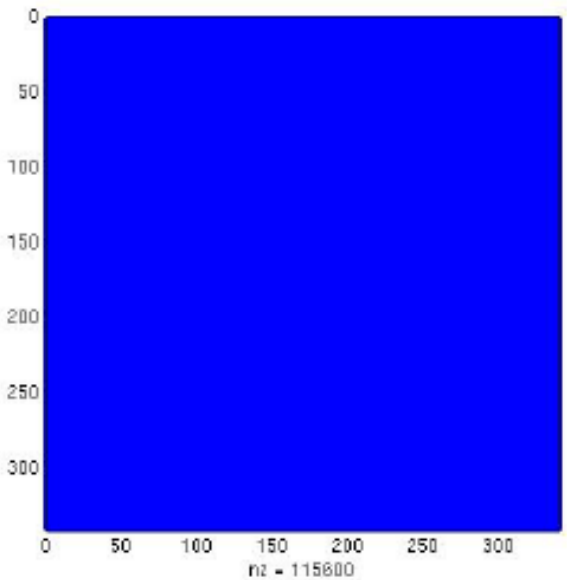


Reduction of resistor networks

Eliminating all internal nodes not an option:

- satisfies conditions (a)–(c),
- but violates (d) and (e): $\hat{r} = (m^2 - m)/2$ resistors in ROM!
- example with 12738 nodes, 340 terminals, 21209 resistors





ROM has 0 internal nodes, same terminals, but 57630 resistors!

General idea: exploit structure

- Resistor networks in ESD are extracted networks with structure related to underlying layout
- Unfortunately, structure may be hard to recover
- Use tools from graph theory to recover part of the structure

Note:

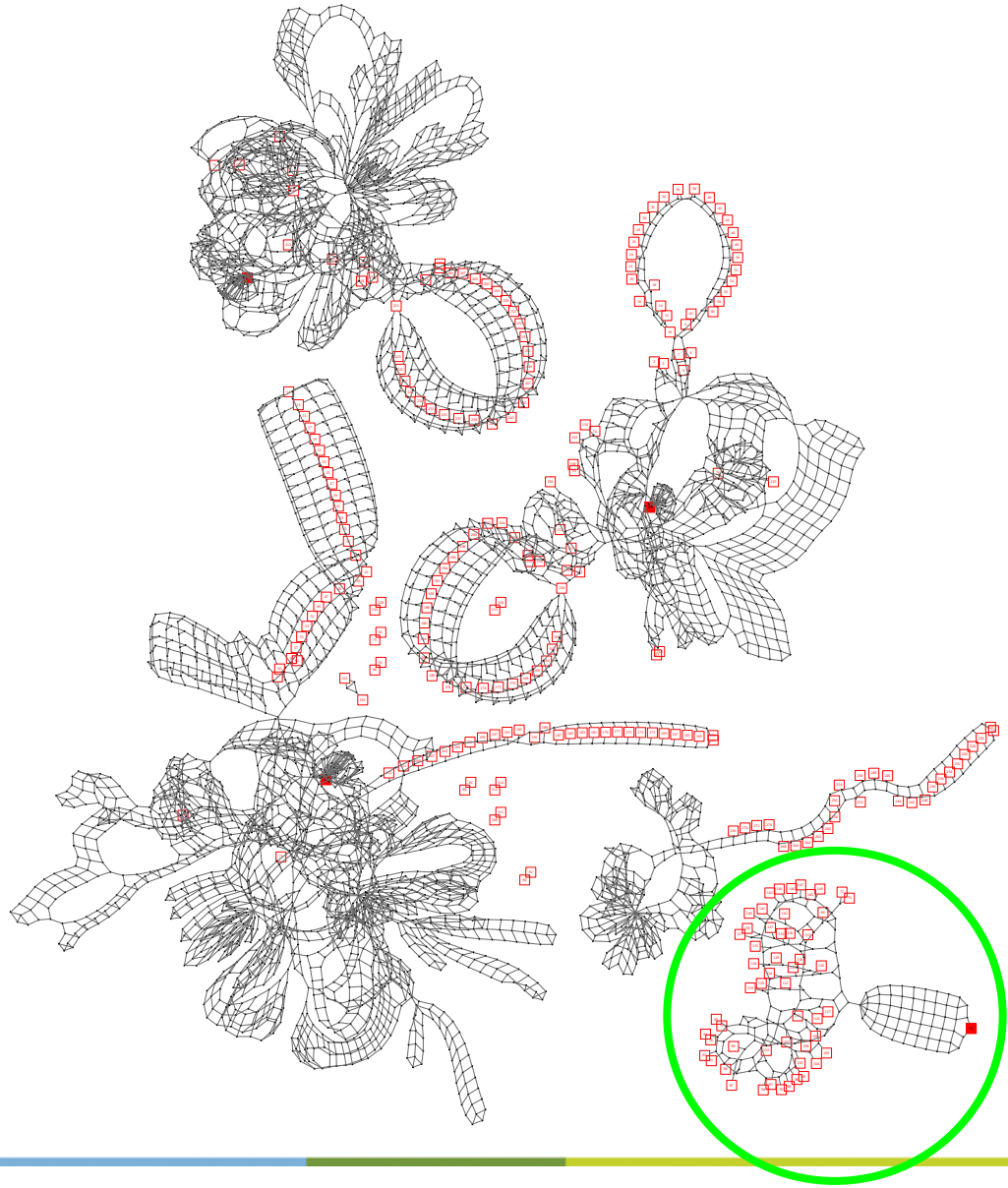
- With structure we refer to topology of network
- In our applications, reduced network should have fewer elements than and same topology as original

“Simple” network

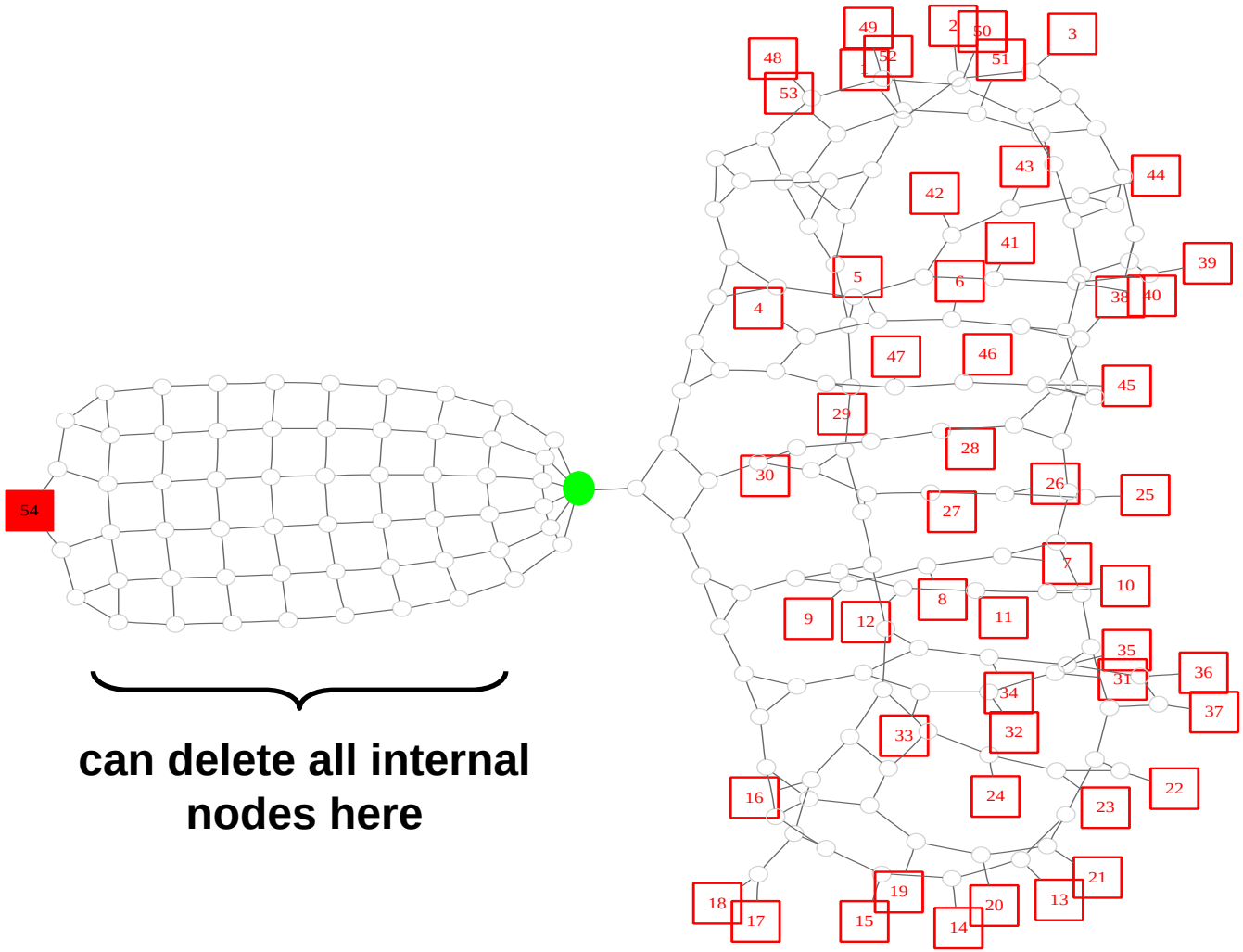
- 274 external nodes (pads, in red)
- 5384 internal nodes
- 8007 resistors/branches

Can we reduce this network by deleting internal nodes and resistors, still guaranteeing accurate approximations to the path resistances between pads?

NOTE: there are strongly connected sets of nodes (independent subsets), so the problem is to reduce each of these strongly connected components individually



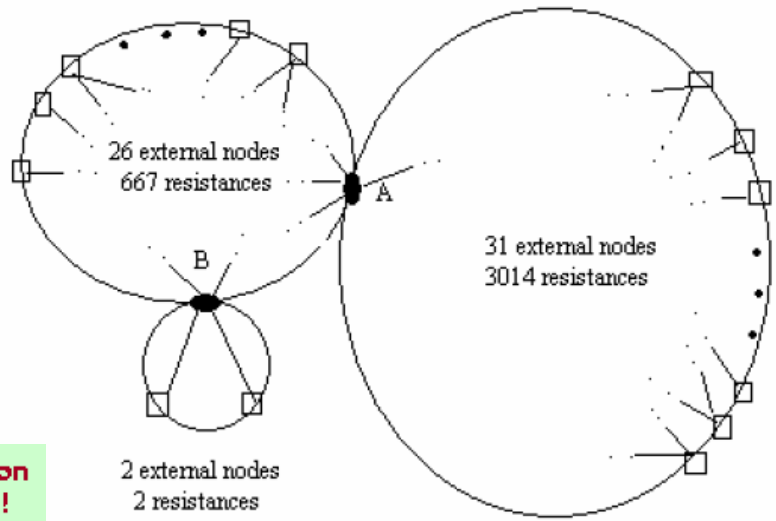
Two-Connected Components

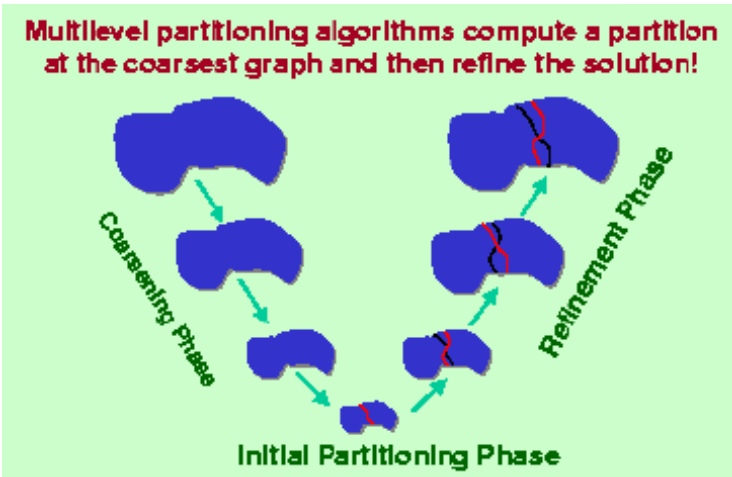


can delete all internal nodes here

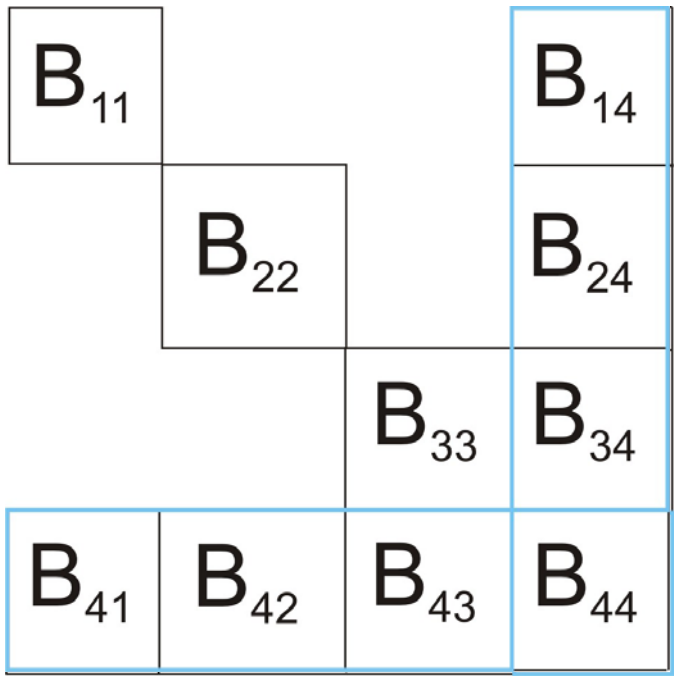
How to reduce the strongly connected components

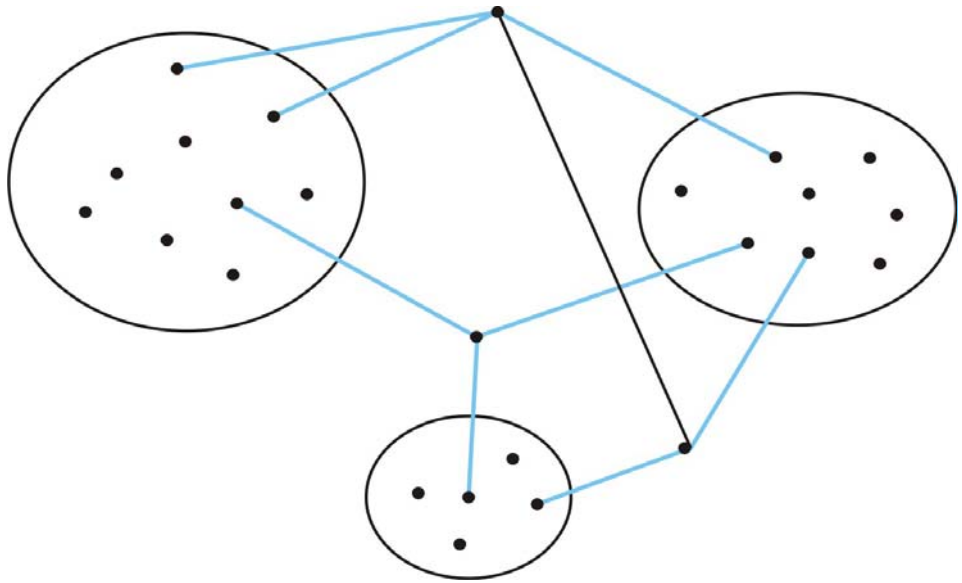
- Graph theoretical techniques, such as graph dissection algorithms (but: rather time consuming)
- Numerical methods such as re-ordering via AMD (approximate minimum degree)
- “Commercially” available tools like METIS and SCOTCH (not satisfactory)





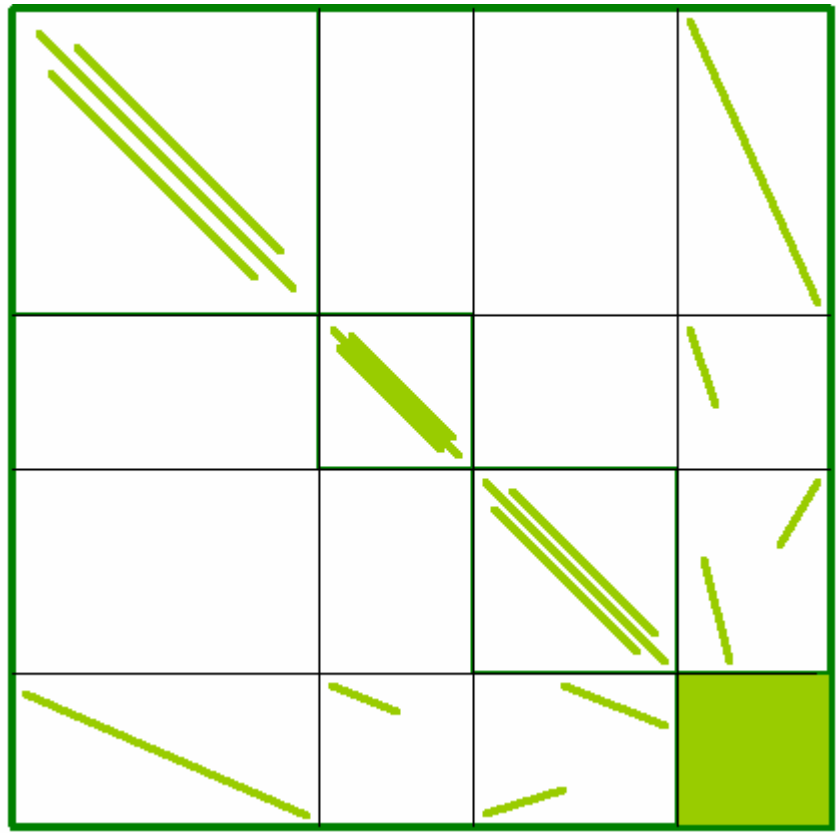
Border Block Diagonal Form





Block bordered diagonal form

- Algorithm was developed to put each of the strongly connected components into BBD form (see figure)
- Internal nodes can be deleted in the diagonal blocks, keeping only the external nodes and crucial internal nodes
- The reduced BBD matrix allows extremely fast calculation of path resistances (work of Duff et al)

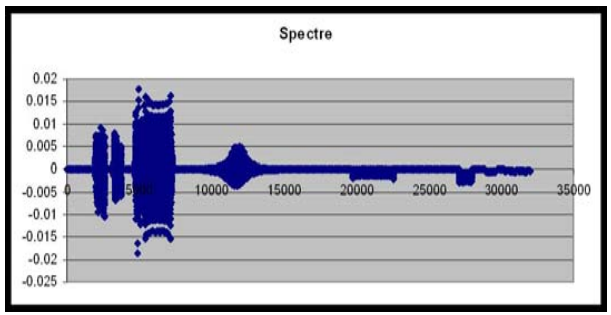


Reduction results

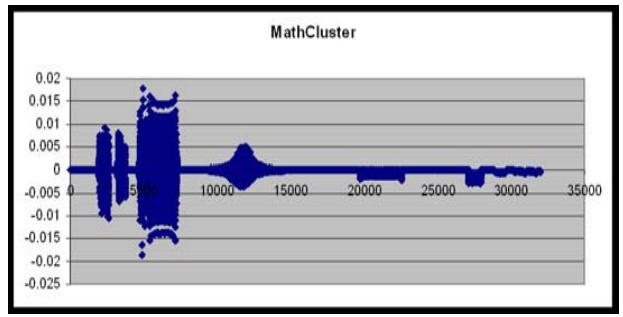
- Results for four resistor networks
- ESD analysis of realistic layouts (I–III)
- Parasitic interconnect reduction (IV)

	Network I		Network II		Network III		Network IV	
	Orig	ROM	Orig	ROM	Orig	ROM	Orig	ROM
#terminals	3260		1978		15299		8000	
#int nodes	99k	8k	101k	1888	1M	180k	46k	6k
#resistors	161k	56k	164k	39k	1.5M	376k	67k	26k
#other devs	1874		1188		8250		29k	
#other nodes	0		0		0		11k	
CPU red	130 s		140 s		1250 s		75 s	
CPU sim	67h	6h	20h	2h	–	120h	–	392s
Speed up	11x		10x		∞		∞	

Results are exact



No loss of accuracy!
←Spectre New →



Remaining problem: how to drastically reduce the number of resistors

SIAM J. SCI. STAT. COMPUT.
Vol. 8, No. 3, May 1987

002

A NOTE ON DOWNDATING THE CHOLESKY FACTORIZATION*

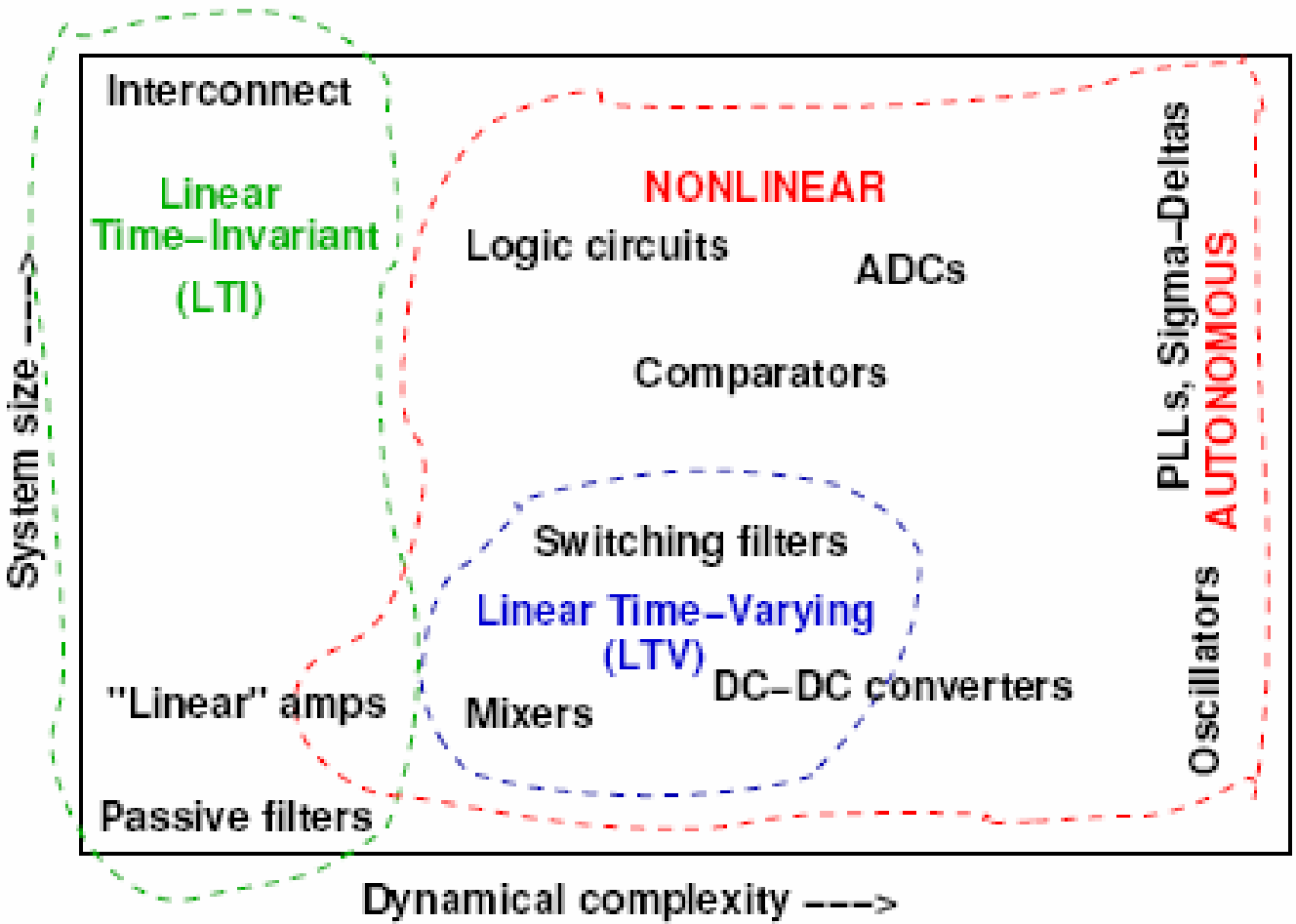
A. W. BOJANCZYK†¶, R. P. BRENT†, P. VAN DOOREN‡ AND F. R. DE HOOG§

Abstract. We analyse and compare three algorithms for "downdating" the Cholesky factorization of a positive definite matrix. Although the algorithms are closely related, their numerical properties differ. Two algorithms are stable in a certain "mixed" sense while the other is unstable. In addition to comparing the numerical properties of the algorithms, we compare their computational complexity and their suitability for implementation on parallel or vector computers.

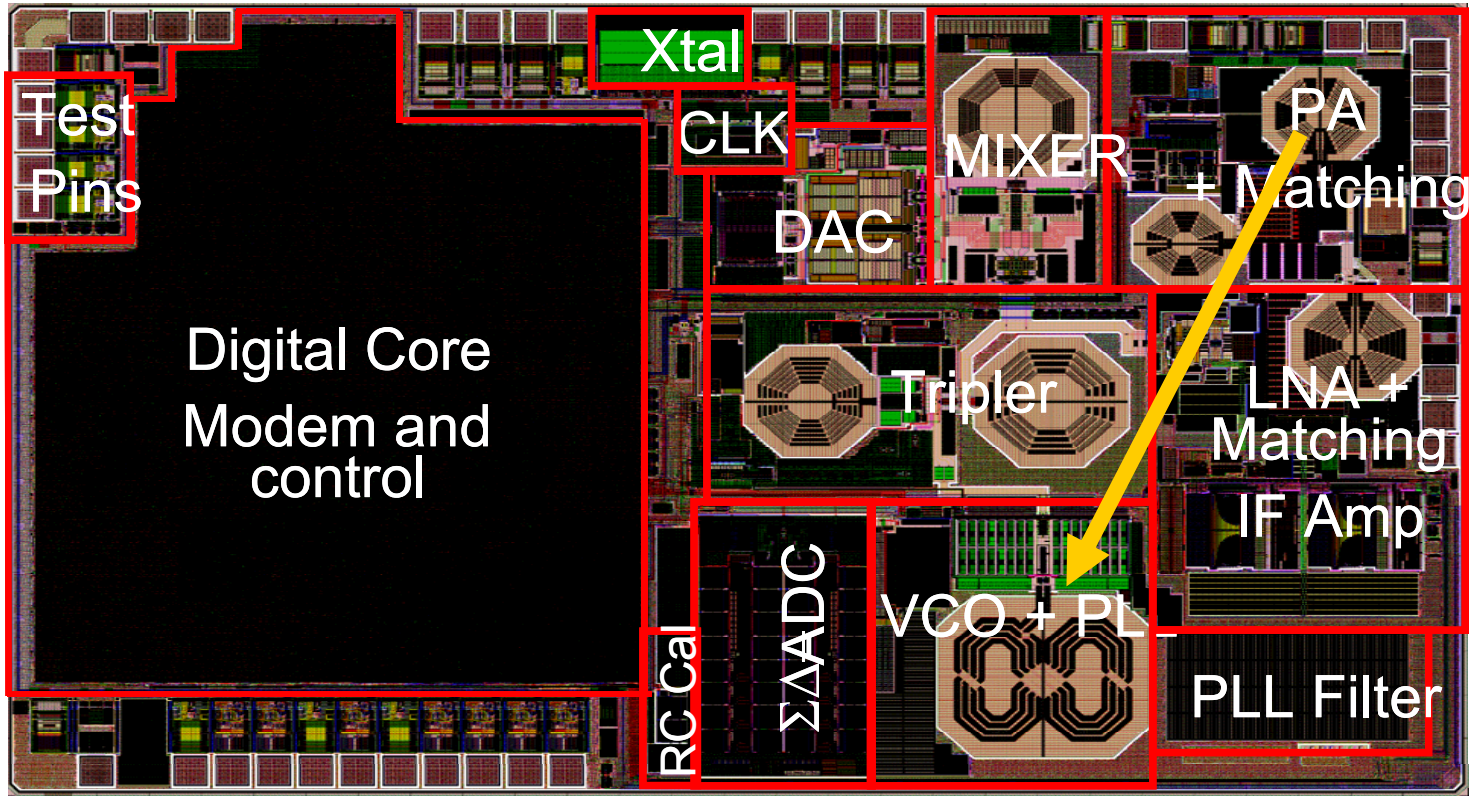
Topic 3: Nonlinear MOR

- One of the most popular methods is proper orthogonal decomposition (POD)
- It generates a matrix of snapshots (in time), then calculates the correlation matrix and its singular value decomposition (SVD)
- The vectors corresponding to the largest singular values are used to form a basis for solutions
- Drawback of POD: there is no model, only a basis for the solution space
- Researchers are also developing nonlinear balancing methods, but so far these can only be used for systems of a very limited size (<10)
  - E. Verriest, “Time variant balancing and nonlinear balanced realizations”
  - J. Scherpen, “SVD analysis and balanced realizations for nonlinear systems”

Classification of circuits



Fast and accurate simulation of oscillators



Challenges:

- **Linear modeling of oscillators does not provide accurate solutions and in most cases is not able to capture subtle nonlinear dynamics of oscillators (injection locking, jitter, etc)**
- **Nonlinear models are necessary which are fast, accurate and generic**

Nonlinear modeling of perturbed oscillators

Process & Key Observations:

1. **PSS of oscillator**:
   $d/dt\,[\mathbf{q}(\mathbf{x}_{PSS})] + \mathbf{j}(\mathbf{x}_{PSS}) = \mathbf{0},\ T = T_{OSC}$
2. $\mathbf{u}_1(t) = d/dt(\mathbf{x}_{PSS})(t)$ ('right Floquet' eigenfunction)
3. $\mathbf{v}_1(t)$ solves a linearized adjoint system ('left Floquet' eigenfunction)

4. **Perturbed oscillator**: $d/dt\,[\mathbf{q}(\mathbf{x})] + \mathbf{j}(\mathbf{x}) = \mathbf{b}(t)$ has solution

   $\mathbf{x}(t) = \mathbf{x}_{PSS}(t+\alpha(t)) + \mathbf{x}_n(t)$, [$\alpha(t)$ phase noise]
5. $\alpha(t)$ satisfies a non-linear scalar differential equation

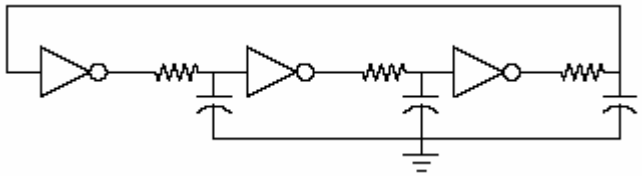
   $d/dt\,(\alpha)(t) = \mathbf{v}_1(t+\alpha(t)).\mathbf{b}(t), \quad \alpha(0)=0$

   Because of orthogonality only the component of $\mathbf{b}(t)$ along $\mathbf{u}_1(t+\alpha(t))$ is important

➔Involves: Time integration, Floquet Theory, Poincaré maps,

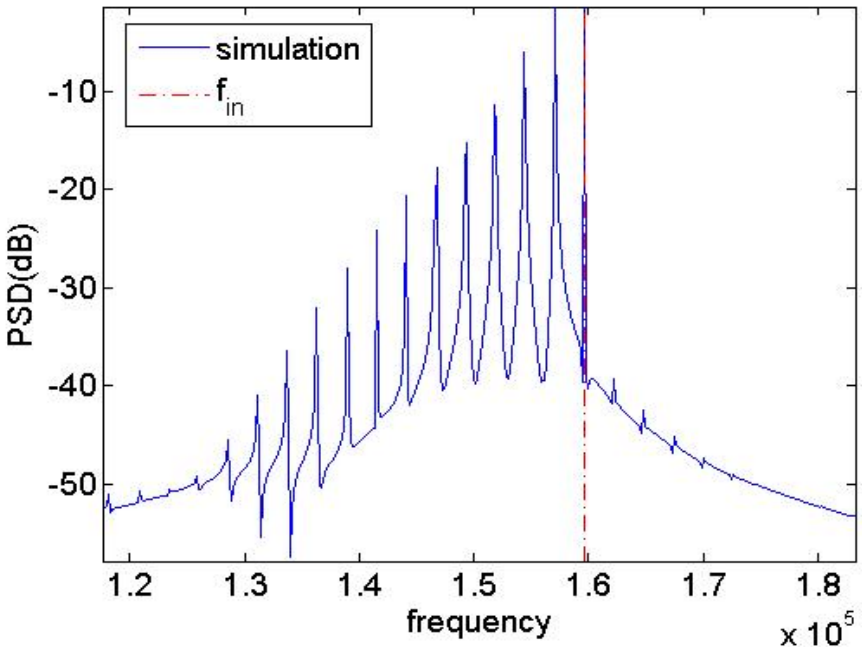
Nonlinear eigenvalue methods, Model Order Reduction!

Example- three stage ring oscillator

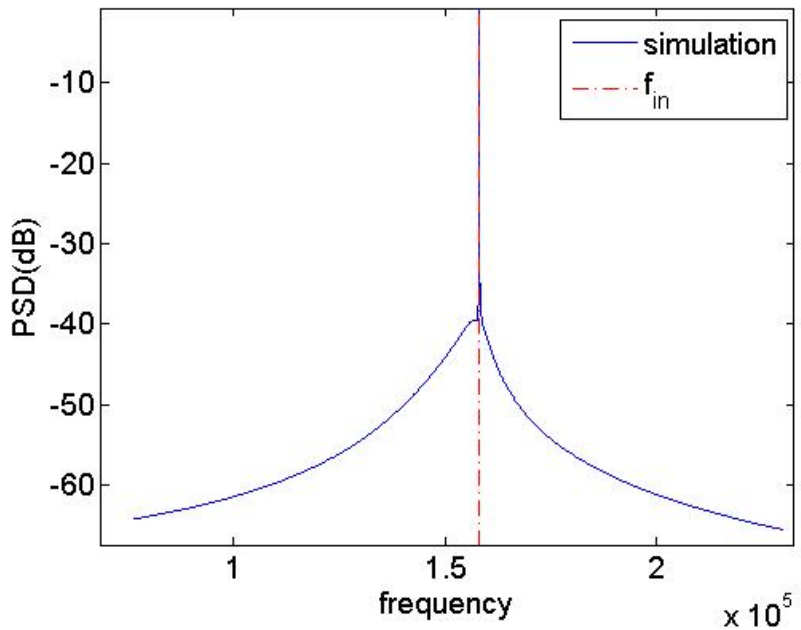


153kHz ring oscillator

Unlocked osc: $i_{inj}= 6 * 10^{-5} *\sin(1.04\omega_0 * t)$



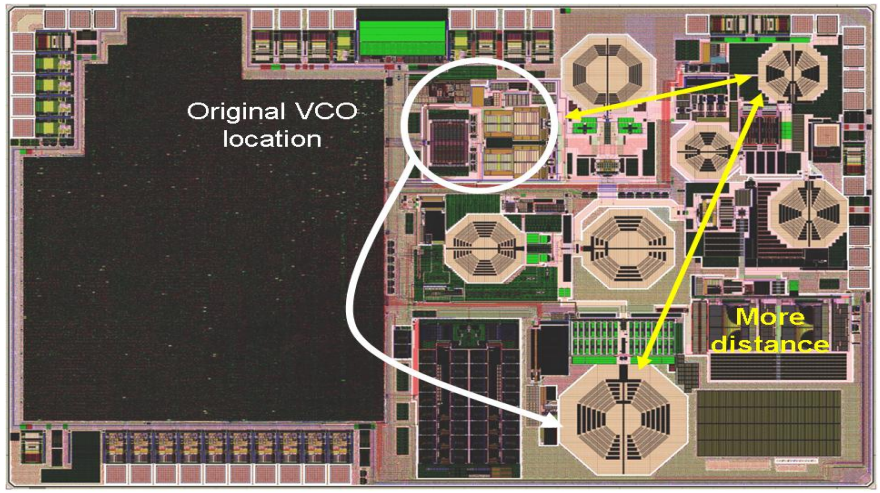
Locked osc: $i_{ini}=6 * 10^{-5} * \sin(1.03\omega_0 * t)$

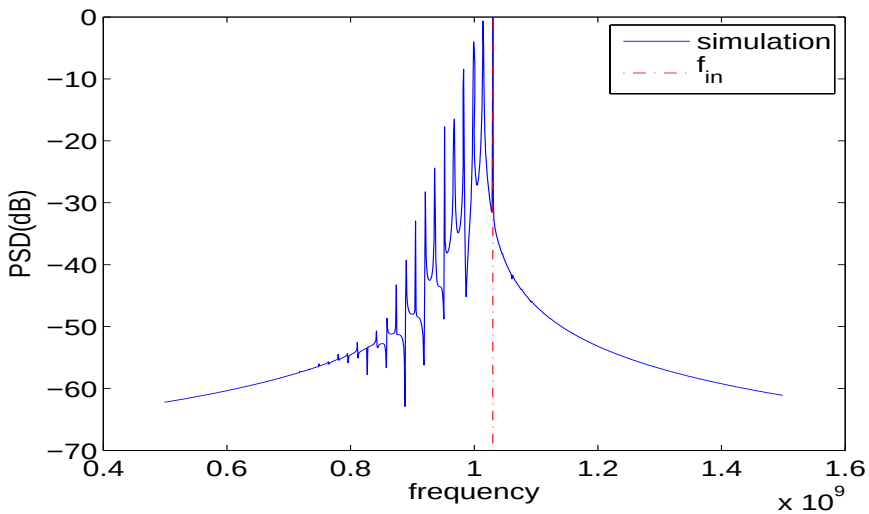


MOR: Fast and accurate modeling of VCO pulling

- VCO pulling due to PA and other blocks/oscillators needs to be analyzed before production
- Full system simulation is CPU intensive or infeasible
- Behavioural model order reduction gives fast and accurate insight in pulling/locking and coupling

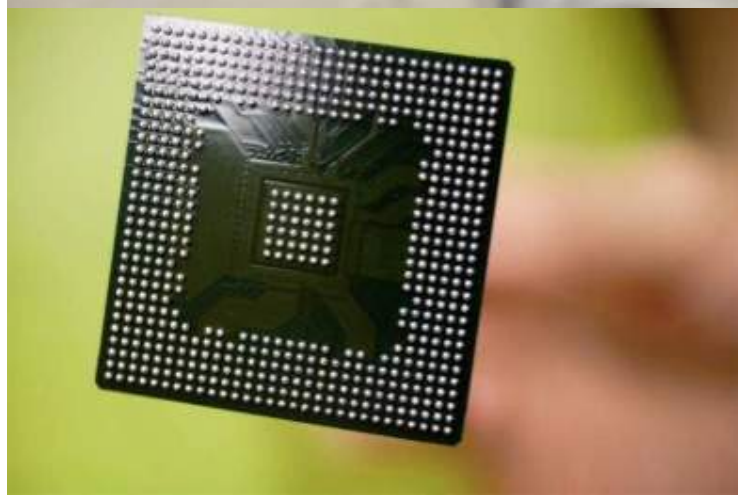
Full mathematical theory of locking/unlocking mechanisms and conditions lacking!





Outline

- Introduction and motivation
- Preliminaries
- Model order reduction basics
- Challenges in MOR
- Conclusions



Conclusions regarding MOR

- Model Order Reduction is a flourishing field of research, both within systems & control and in numerical mathematics
- Strong relation to numerical linear algebra
- Future developments need mathematical methods from a wide variety of fields (graph theory, Floquet theory, combinatorial optimization, differential-algebraic systems, stochastic system theory,…..)

Some recent books on MOR