Model Reduction for VLSI Physical Verification

N.P. (Nick) van der Meijs

Delft University of Technology, The Netherlands

nick@cas.et.tudelft.nl
http://cas.et.tudelft.nl/space

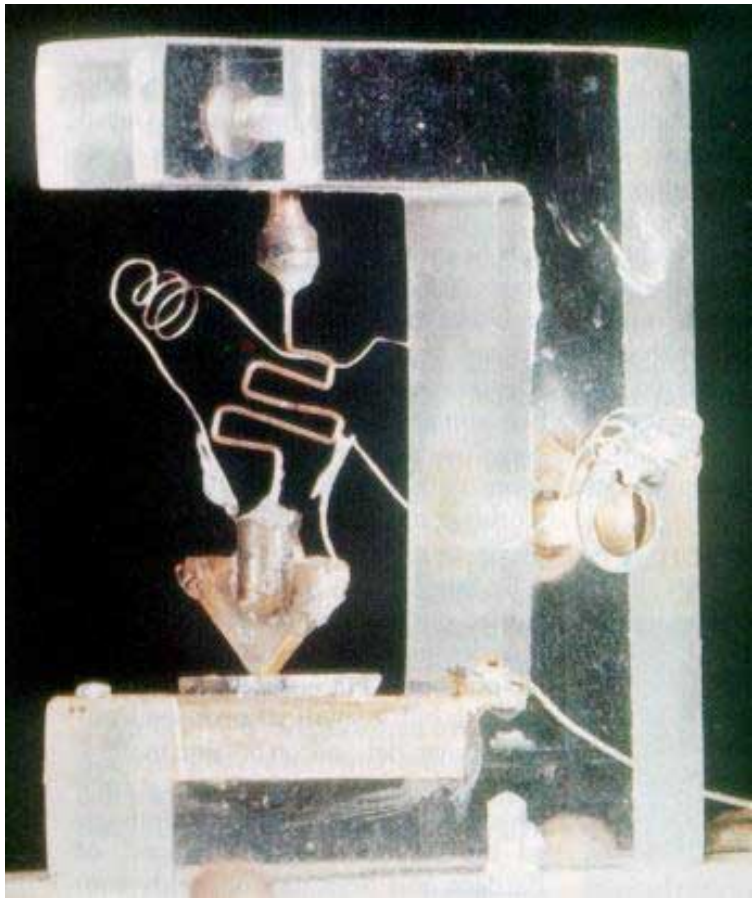
DIMES
Delft Institute of Microelectronics and Submicron Technology

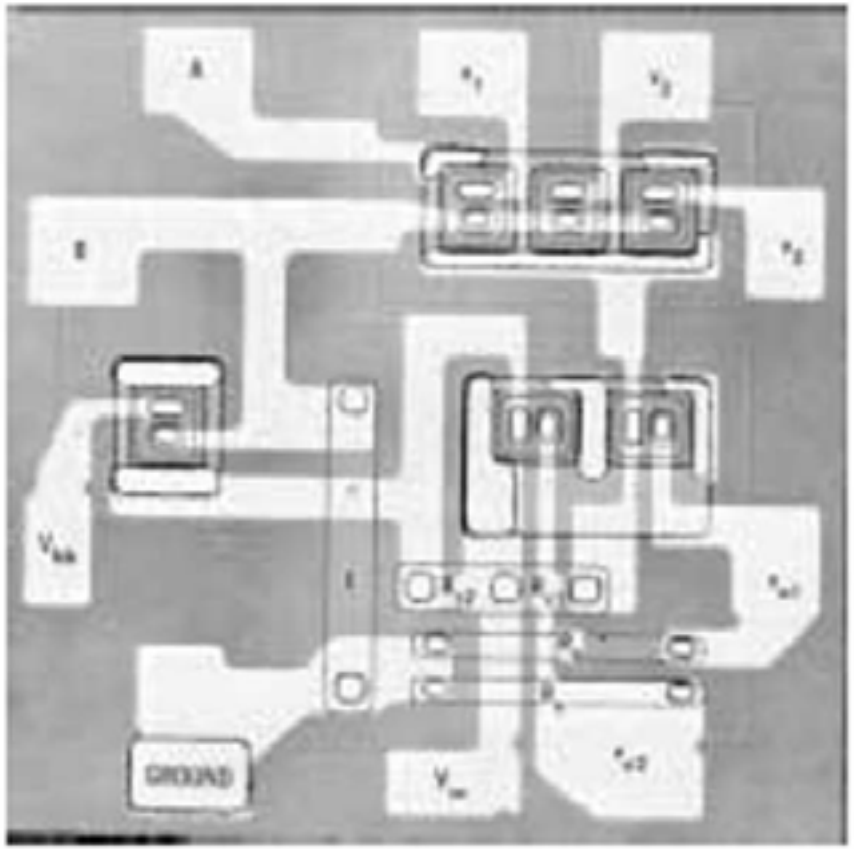
TU Delft
Delft University of Technology

Agenda

1. Micro-Electronics Background
   - Problem definition
2. General Model Order Reduction Issues
3. Interconnect Resistance and Capacitance
   - Gaussian Elimination Based Techniques
4. 3D BEM Capacitance Extraction
   - Matrix Extension Based Techniques
5. Inductance Extraction
6. Conclusion

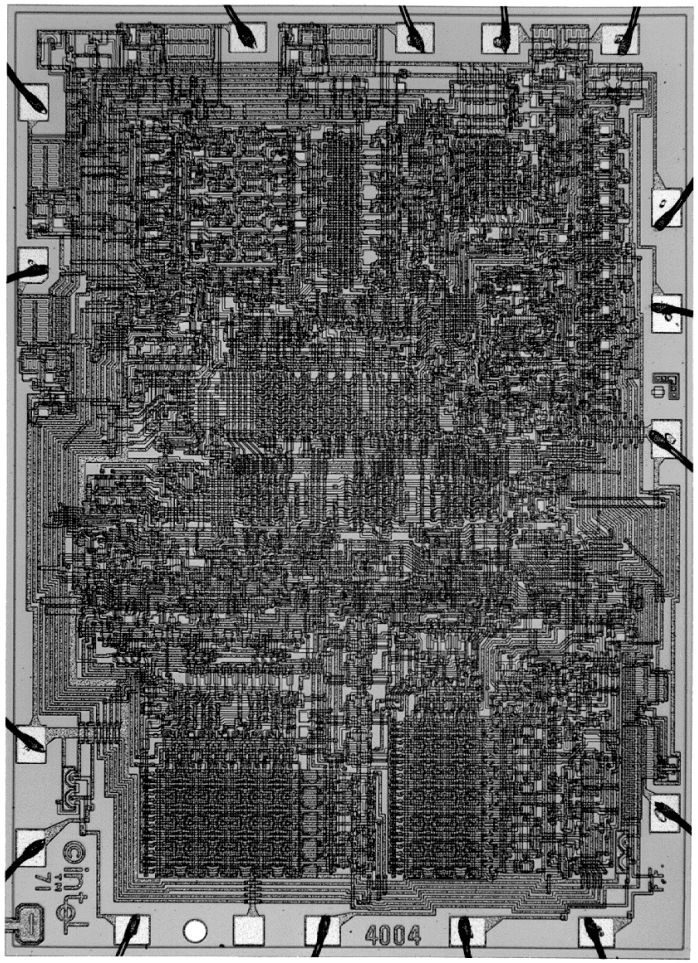
1 Micro-Electronics Background





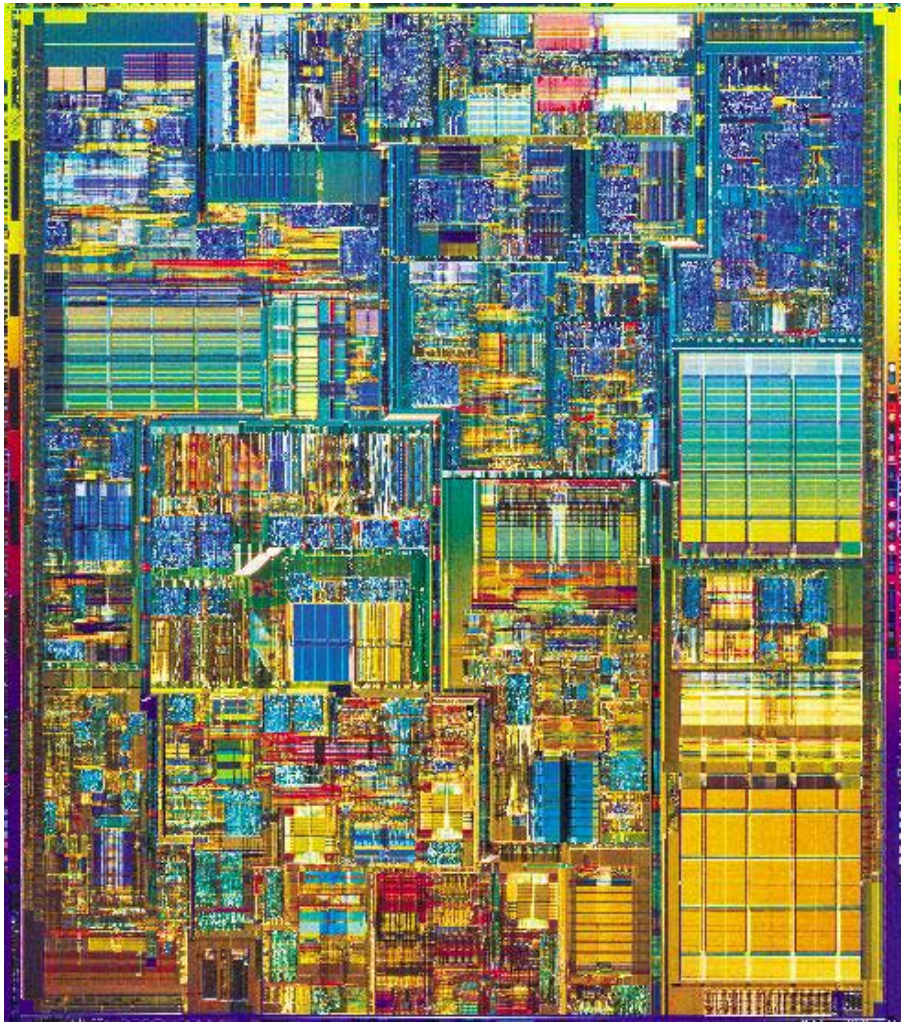
Intel 4004 Micro Processor

- 1971
- 10 μ details
- 2300 components
- 64 kHz speed

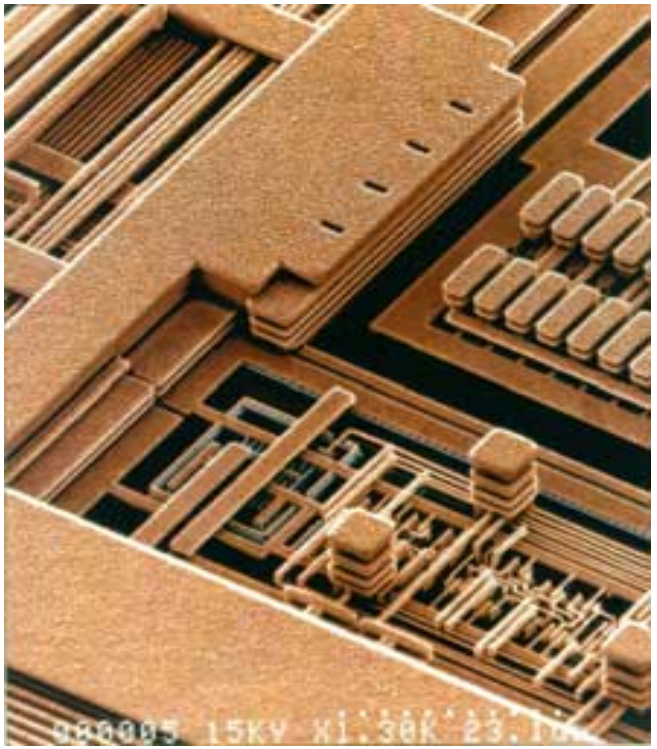


Intel Pentium IV

- 2001
- 0.18 μ details
- 42 million components
- 2 Ghz speed
- +/- 2 km interconnect



Chip Interconnect



IC Technology—Comparison

Chip: 4 cm^2

Netherlands: 40,000 km^2 (approximately)

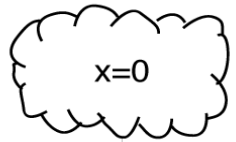
Scale: 2 cm / 200 km = 1:10,000,000

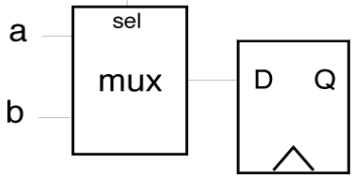
A chip compares to Netherlands full of roads:

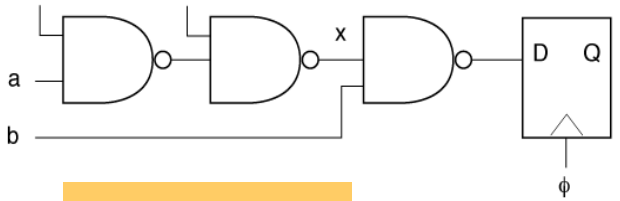
- 1.8 meters wide
- 1.8 meters apart
- 6 layers

Generic Chip Design Flow

```
if x = '0' then
      reg1 <= a;
else
      reg1 <= b;
end if;
```







detailed specs

architectural simulation

floorplan

register - transfer design

logic design

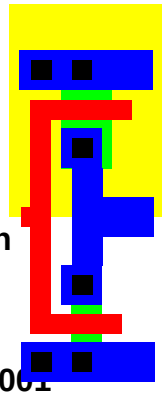
circuit design

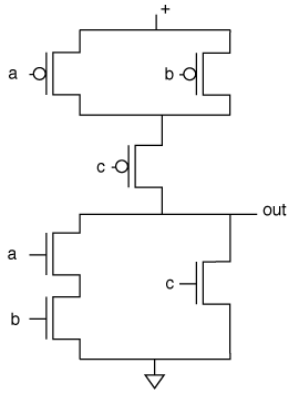
layout

functional / performance verification

tapeout

testability

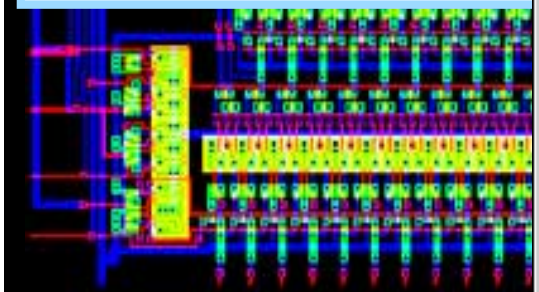


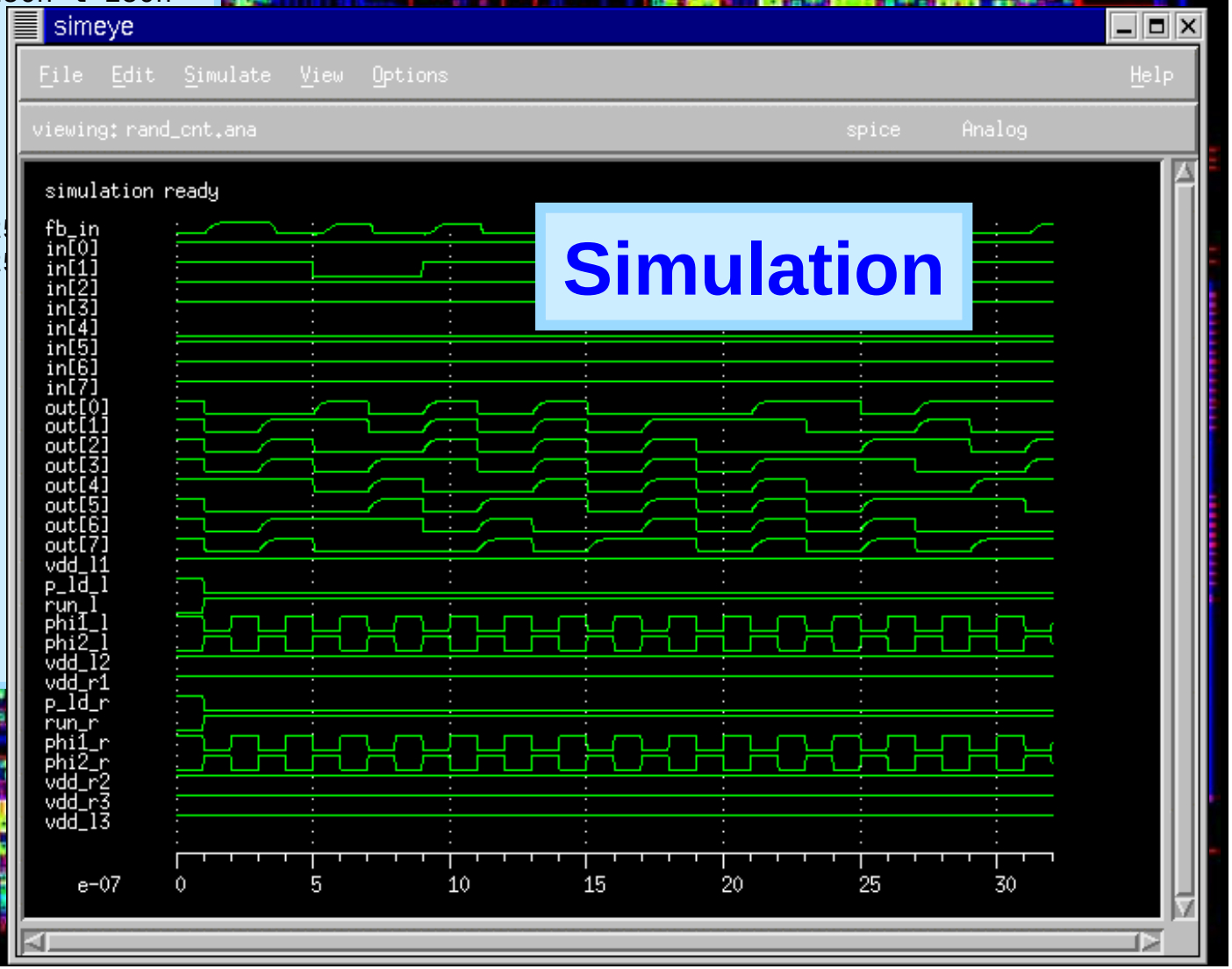


Physical Verification

Layout-to-Circuit Extraction

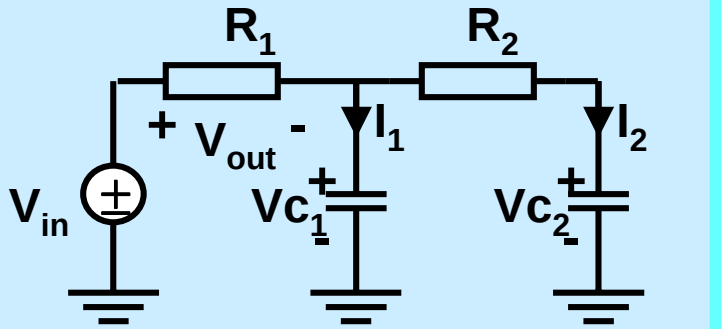
```
m1 11 12 13 14 penh_0 w=250n l=250n
m2 15 16 17 14 penh_0 w=250n l=250n
r1 15 4 329.7622
c1 15 18 1.096202f
r2 11 4 307.2622
c2 11 18 1.028856f
c3 4 18 2.004567f
c4 14 18 16.71091f
m3 19 20 21 22 nenh_0 w=2
m4 23 24 25 22 nenh_0 w=2
r3 13 8 1.709575k
c5 13 18 1.435885f
r4 19 26 157.2081
r5 19 8 3.320258k
c6 19 18 394.353a
r6 26 8 3.233061k
c7 26 18 418.6736a
r7 16 24 213.8774
r8 16 8 7.274456k
c8 16 18 110.7094a
   ...
```

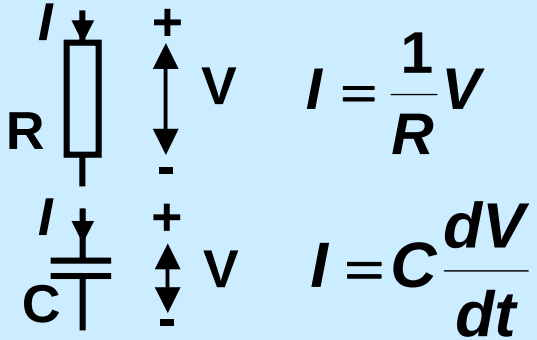




Simulation

State Space Formulation





$$I = \frac{1}{R} V \qquad I = C \frac{dV}{dt}$$

$$I_1 = C_1 \frac{dV_{C_1}}{dt} = \frac{1}{R_1}[V_{in} - V_{C_1}] + \frac{1}{R_2}[V_{C_2} - V_{C_1}]$$
$$I_2 = \ldots$$

$$\begin{bmatrix} \frac{d}{dt} V_{C_1} \\ \frac{d}{dt} V_{C_2} \end{bmatrix} = \begin{bmatrix} -\left(\frac{1}{R_1 C_1} + \frac{1}{R_2 C_1}\right) & \frac{1}{R_2 C_1} \\ \frac{1}{R_2 C_2} & -\frac{1}{R_2 C_2} \end{bmatrix} \begin{bmatrix} V_{C_1} \\ V_{C_2} \end{bmatrix} + \begin{bmatrix} \frac{1}{R_1 C_1} \\ 0 \end{bmatrix} V_{in}$$

$$out = \begin{bmatrix} -1 & 0 \end{bmatrix} \begin{bmatrix} V_{C_1} \\ V_{C_2} \end{bmatrix} + [1] V_{in}$$

$$\frac{d}{dt} x(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

- **Modeling**: determination of R,C network, or equivalently, ABCD
- **Simulation**: applying u(t)
- But: usually other formulation (MNA)
- Would include (non-linear) models for active devices

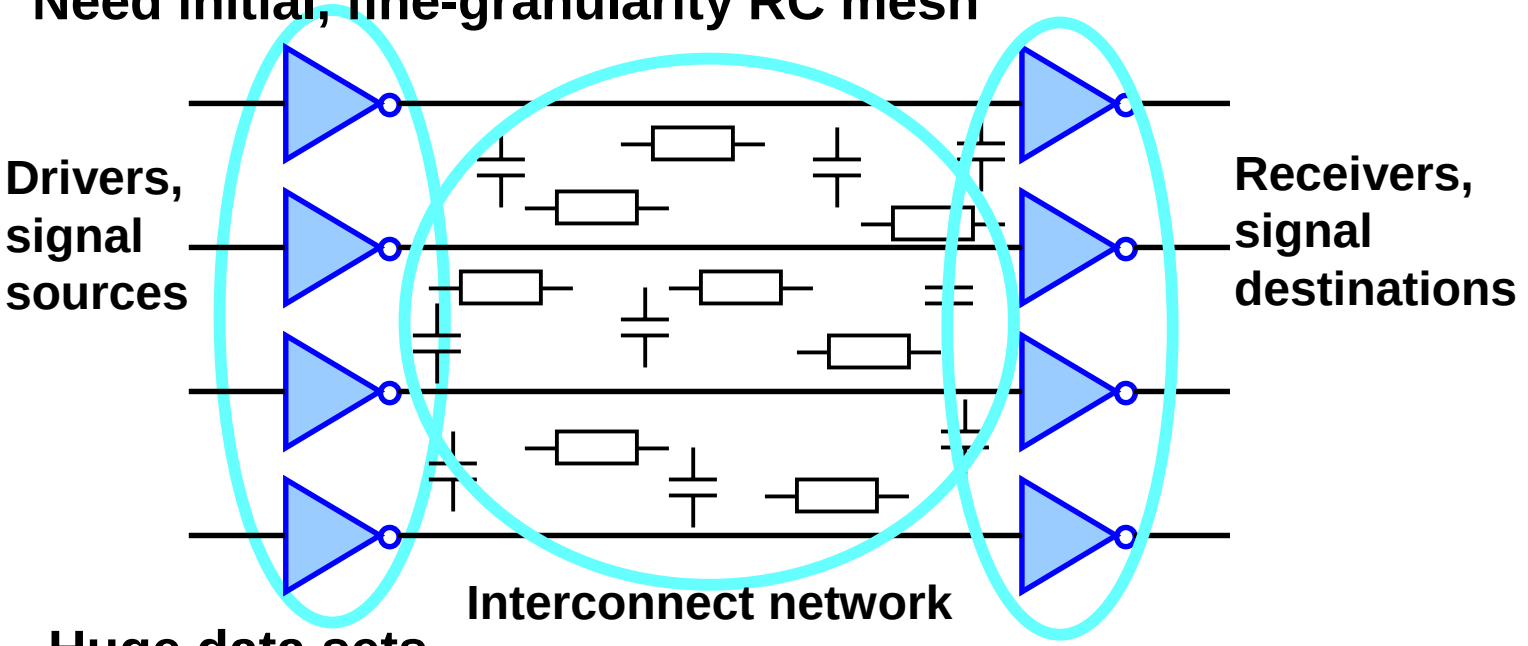
2 General Model Order Reduction Issues

James Clerk Maxwell (1831 - 1879)

Gustav Kirchhoff (1824 - 1887)

Consistent Modeling, Distributed Effects

- Chip Timing dominated by interconnect R, C
- R and C distributed along interconnect ⇨ Telegraphers eqs.
- Chip designers prefer Kirchhoff over Maxwell: PDE ⇨ ODE
- Need initial, fine-granularity RC mesh



- Huge data sets
- Impossible to analyze directly

Need model order reduction

Model Order Reduction

- Approximate huge system matrix by small one
- Frequency domain approach (usually)
- Using e.g. Krylov subspace techniques
- Active area of R&D
- Issues: stability, passivity, accuracy, CPU time
- Special simulator needed (or issue of realizability)
- Doesn't solve dataset problem

Need early model reduction

Model Order Reduction Phases

A Priori

- Before computation of the model
- Compute only part of the model
- Consistency of result
- Can reduce computational effort and/or model complexity

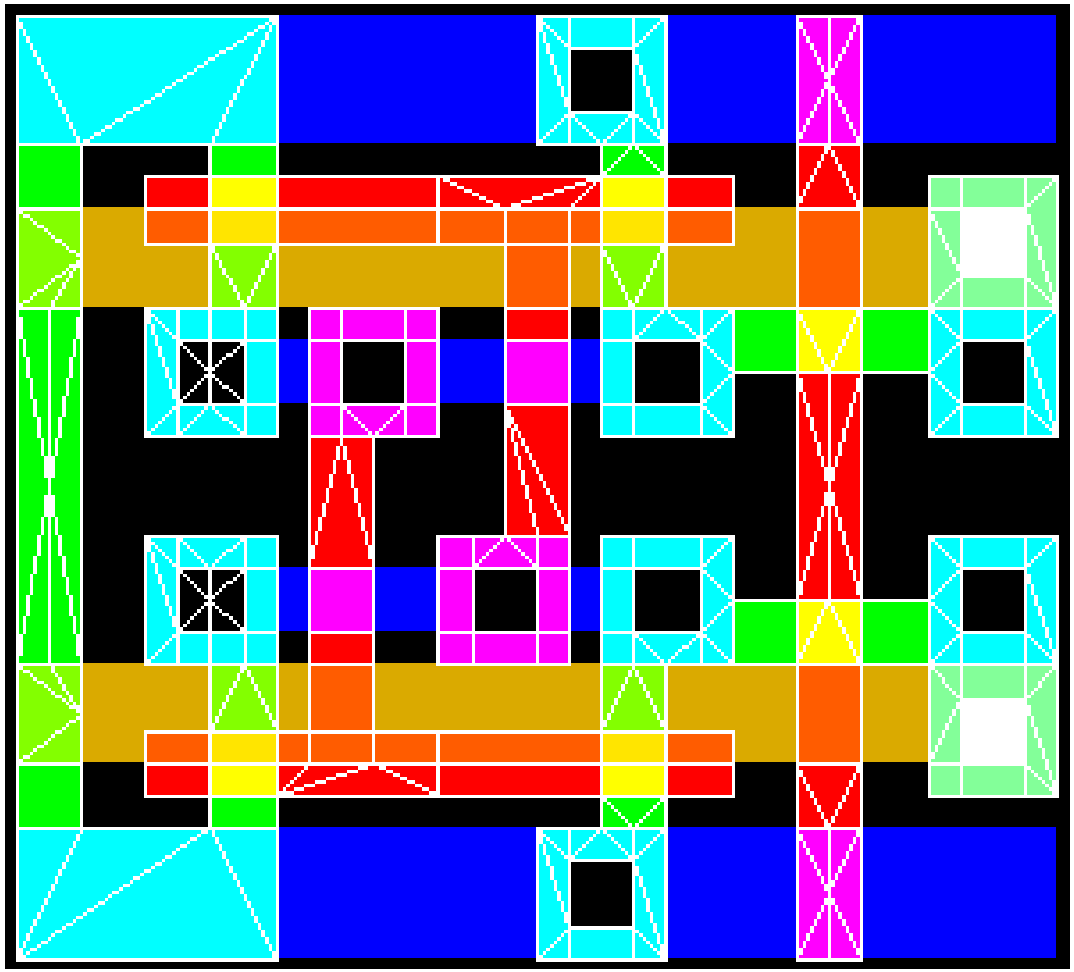
On the fly

- While computing the model
- Merge computational procedures
- Reduced model complexity at certain computational cost

A Posteriori

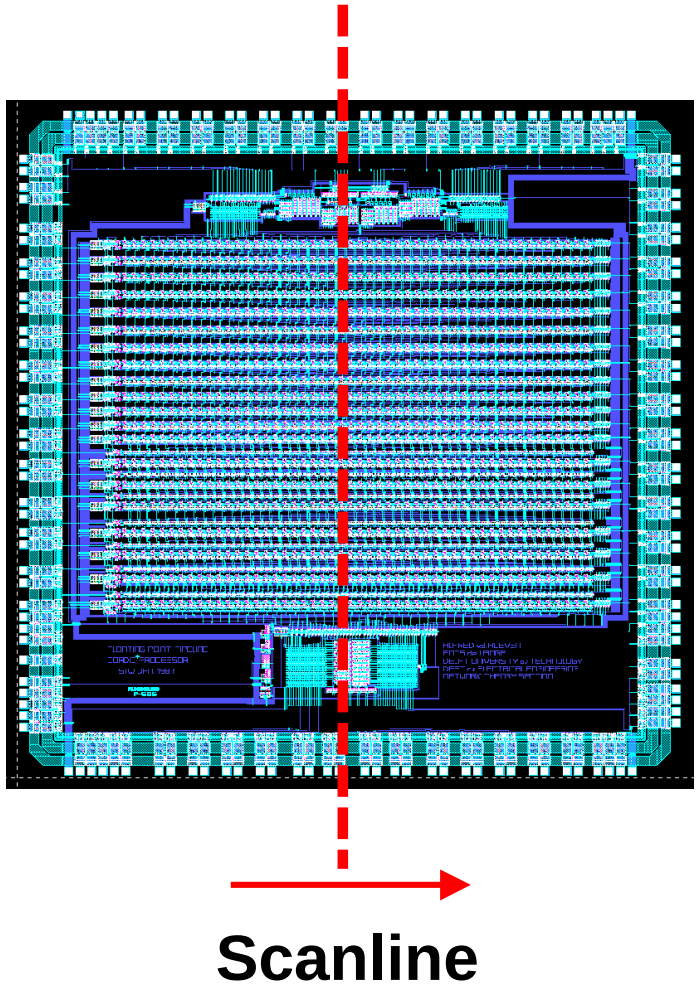
- After model has been computed
- Important domain of R&D
- Greatest opportunities for model reduction
- Sometimes large computational cost

3 2D FEM Resistance Extraction



Fundamental Approach

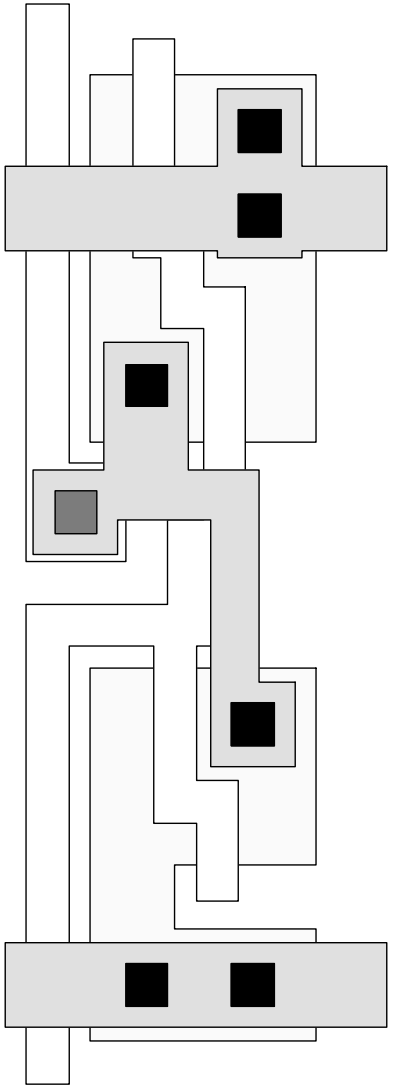
Based on Scanline Algorithm

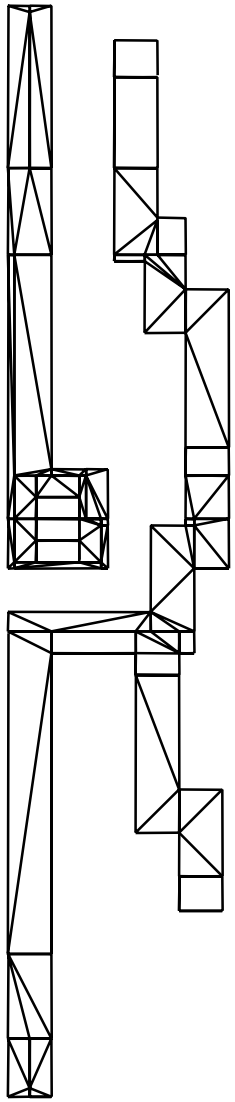


- Operations take place in a narrow band sliding over layout from left to right
- Layout data read in A.L.A.P.
- Circuit data written out A.S.A.P.
- Sublinear memory complexity
- Near-linear time complexity

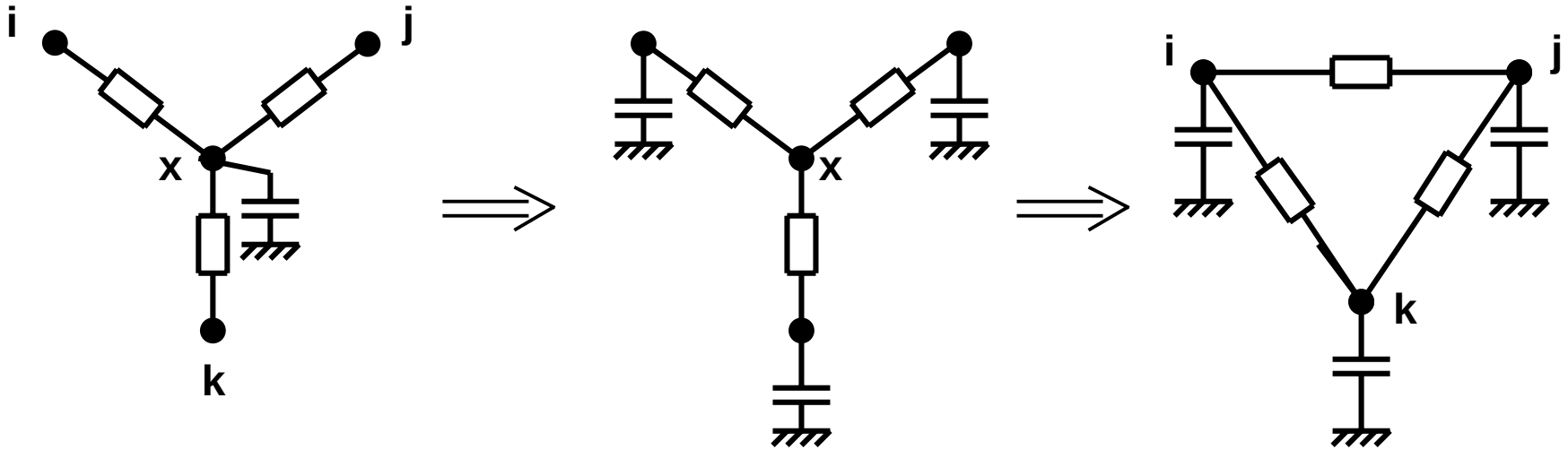
Interconnect Resistances

- FEM to determine resistances
- Solution only necessary at ‘boundary nodes’
- Model reduction via Gaussian elimination of internal nodes (matrix based)
- FE Mesh ⇔ resistance network
- Gaussian elimination ⇔ star-delta transformation: network based
- Node can be eliminated when all its neighbors are known





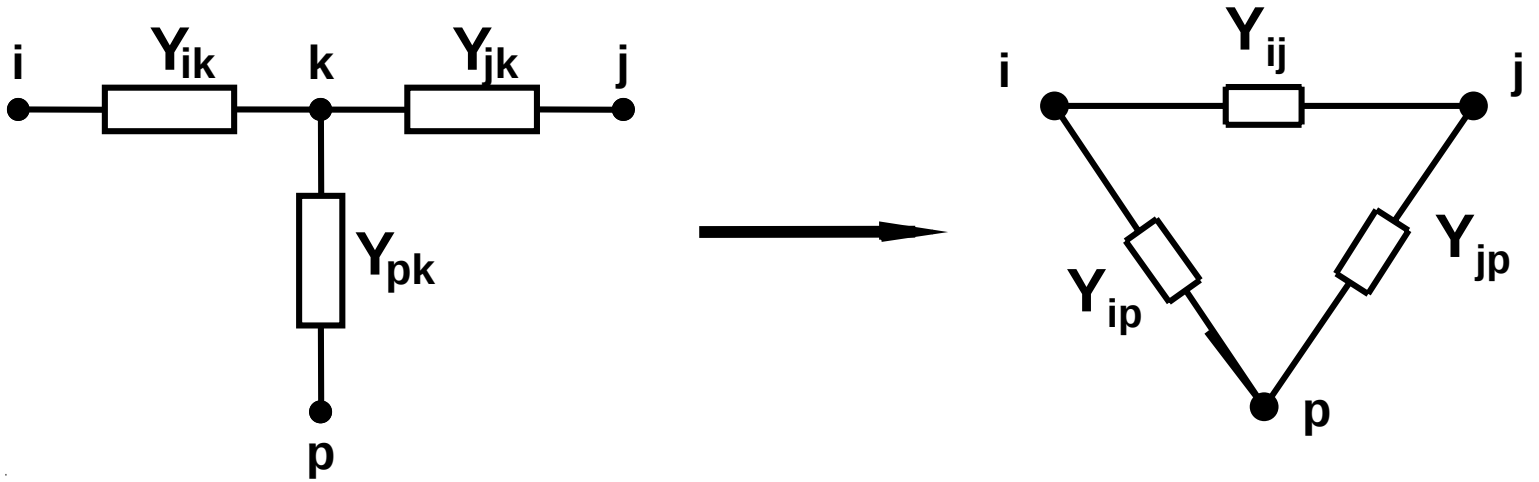
Gaussian Elimination for RC Networks



$$C_i := C_i + C_x \frac{G_{xi}}{\sum_{n \notin x} G_{xn}} \qquad G_{ij} := G_{ij} + \frac{G_{xi} G_{xj}}{\sum_{n \notin x} G_{xn}}$$

- Gaussian Elimination = Star-delta transformation
- Elmore delay (first moment of impulse response) is preserved
- Also for coupling capacitances
- On-the-fly elimination

Moment-Preserving Node Elimination



$$Y_{ij}(s) = M_{ij}^{(0)} + M_{ij}^{(1)} s + M_{ij}^{(2)} s^2 + \ldots = Y_{ij}(s) + \frac{Y_{ik}(s) Y_{jk}(s)}{\sum_{l=1, l \neq k}^{N} Y_{kl}(s)}$$

- $M_{ij}^{(0)}$ represents 1/resistance, and $M_{ij}^{(1)}$ capacitance between nodes i and j.
- using Gaussian elimination each $M_{ij}^{(q)}$ can be calculated exactly, recursively using $M_{ij}^{(q-1)}$.
- fully contains Elmore-delay preserving elimination.
- can e.g. be used for moment-matching (Padé).

Frequency Dependent Model Reduction

Selective Node Elimination (SNE)

- Star-delta transformation preserves first moment
- Error arises in second moment
- Combine 2nd moment with operating frequency ⇨ error weight
- Selective Internal Node Elimination, (SNE) procedure:
  1. Eliminate least critical node;
  2. Update weights of old neighbors;
  3. Repeat 1 & 2 until min weight exceeds threshold
- Reduced model is accurate up to specified frequency f_s
- No issues of passivity, stability, realizability
- Usually network small enough for simulation

Freq. Dependent Model Reduction (2)

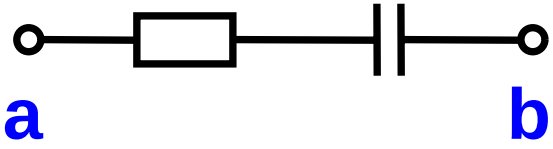
Selective Node Elimination (SNE)

$$\delta_{ij} = \frac{\omega_s^2 \left| M_{ij}^{(2)} \right|}{\sqrt{\left(\left| M_{ij}^{(0)} \right|^2 + \omega_s^2 \left| M_{ij}^{(1)} \right|^2 \right)}}, \qquad \omega_s = 2\pi f_s$$

$$\delta_k = \left\| \left(\delta_{ij} \middle| i, j \in N_k \right) \right\|, \qquad N_k = \left\{ n \middle| \text{connected to node } k \right\}$$

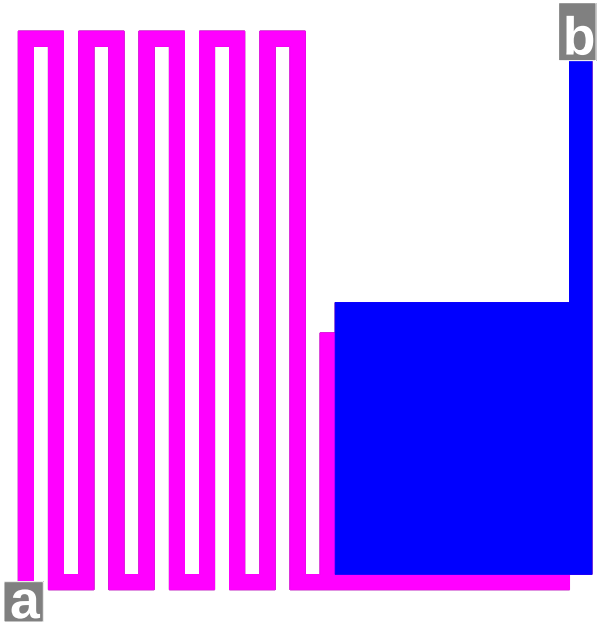
1. Calculate Node Error Weights δ_k of initial network, Error Weights are frequency dependent.
2. Eliminate node k having the lowest weight.
3. Update error weight of all nodes previously connected to k.
4. Repeat step 2 and 3 until all remaining errors exceed a specified tolerance.

SNE Example: Snake RC



Intended circuit

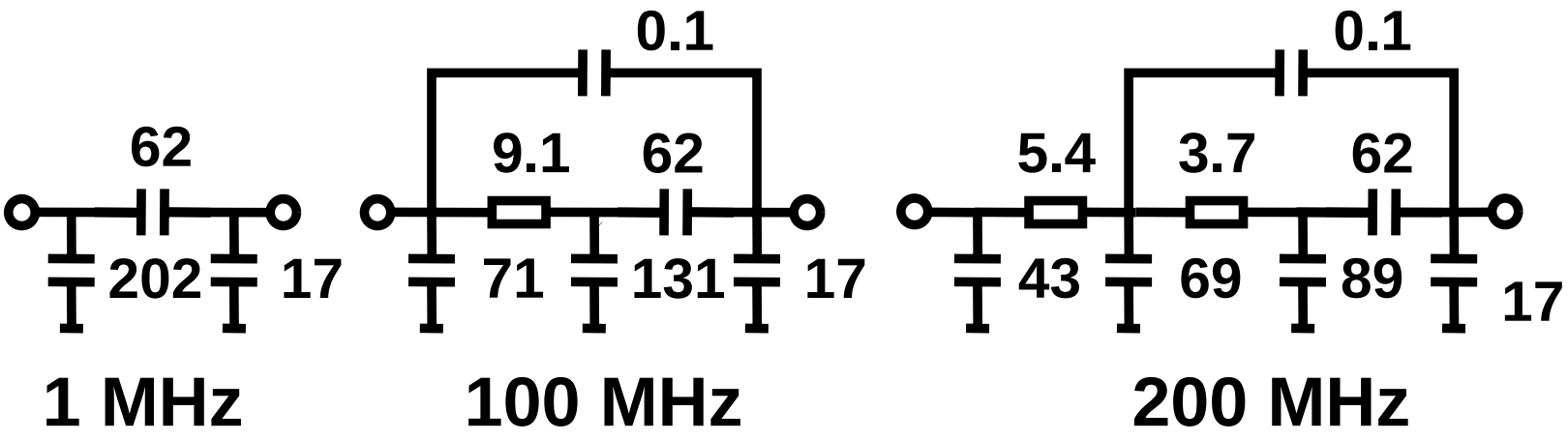
Layout



Initial 3D extraction ⇨

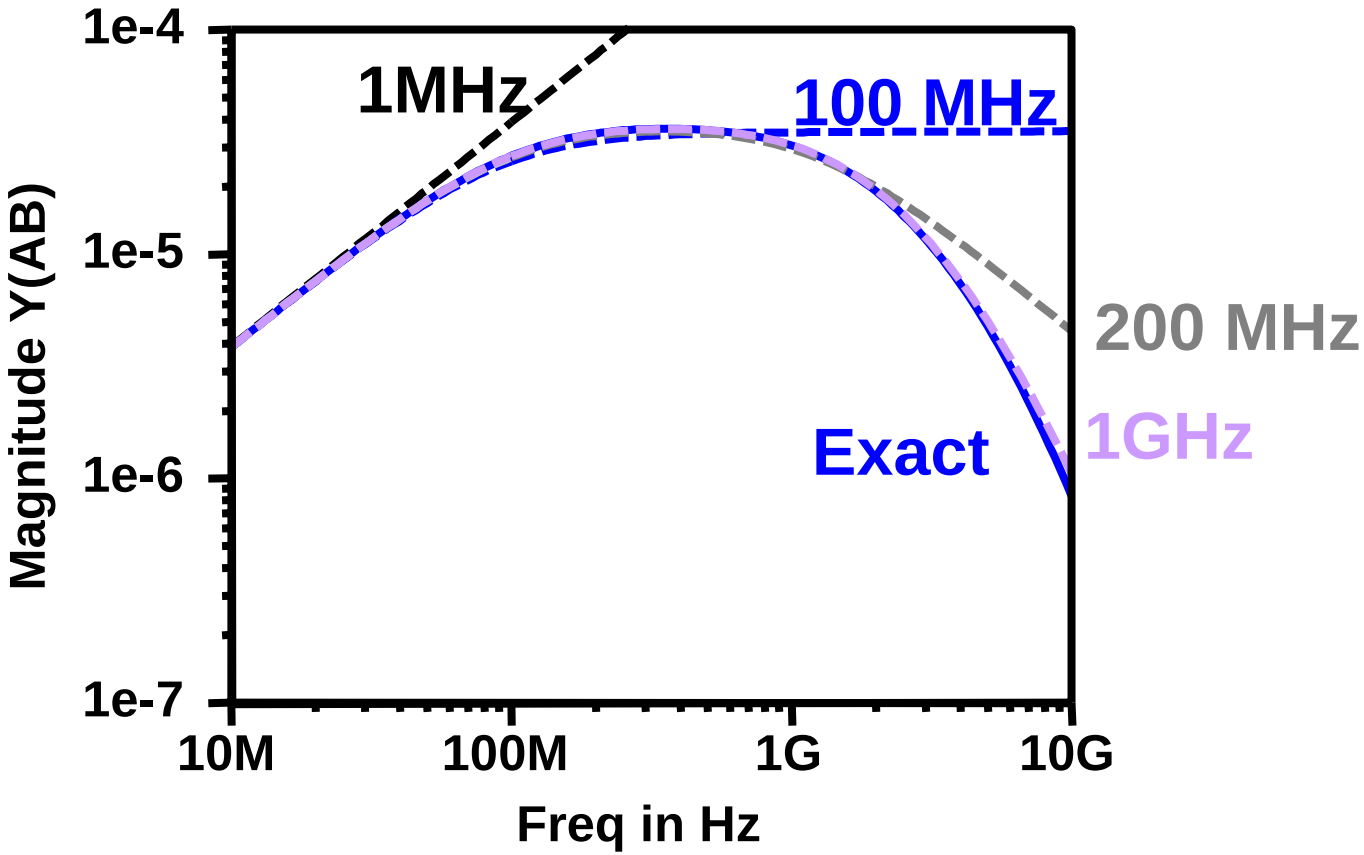
lumped RC mesh
109 nodes
164 resistors
720 capacitors

SNE circuit results for Snake RC

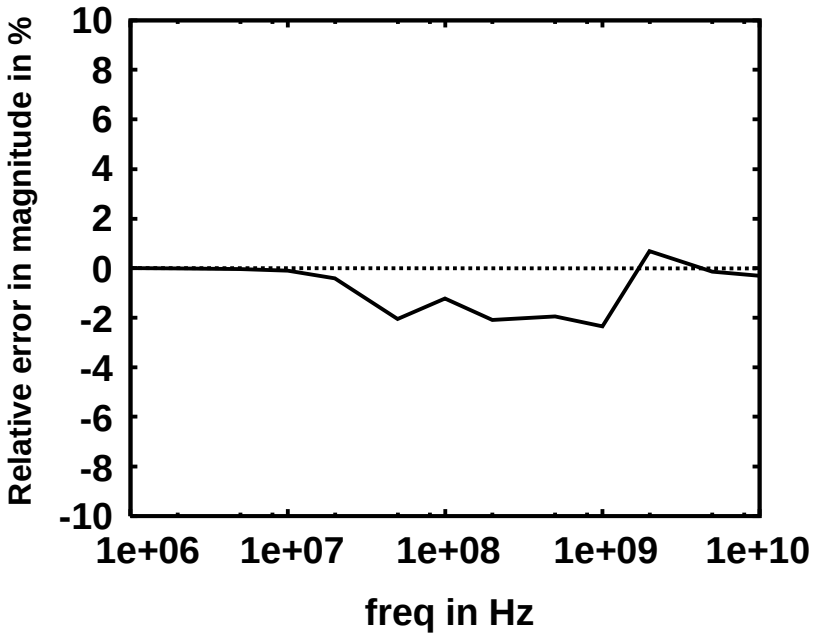


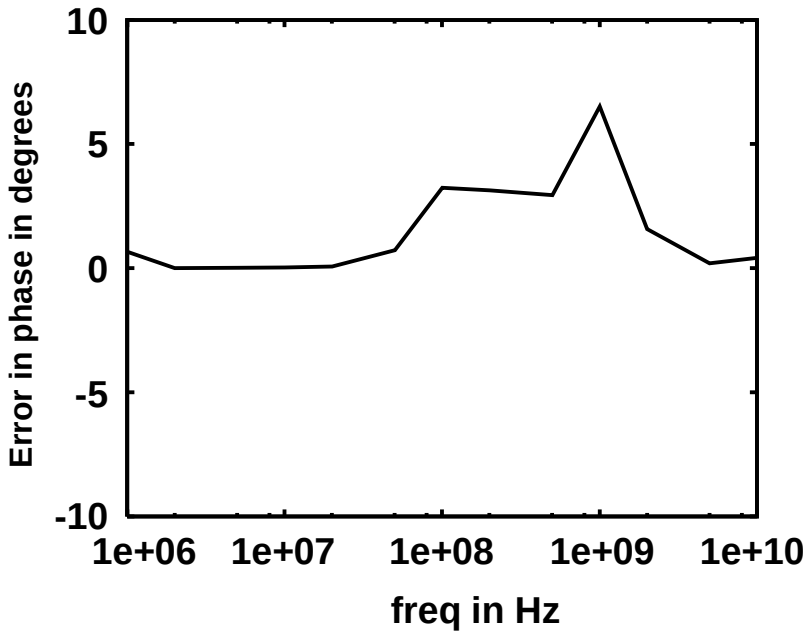
Exact	SNE			AWE
Poles	100 MHz	200 MHz	500 MHz	N=3
$8.8\ 10^{08}$	$7.6\ 10^{08}$	$8.4\ 10^{08}$	$8.7\ 10^{08}$	$8.8\ 10^{08}$
$9.9\ 10^{09}$		$8.4\ 10^{09}$	$8.9\ 10^{09}$	$1.0\ 10^{10}$
$2.4\ 10^{10}$			$2.1\ 10^{10}$	$2.1\ 10^{10}$

SNE simulation results for Snake RC



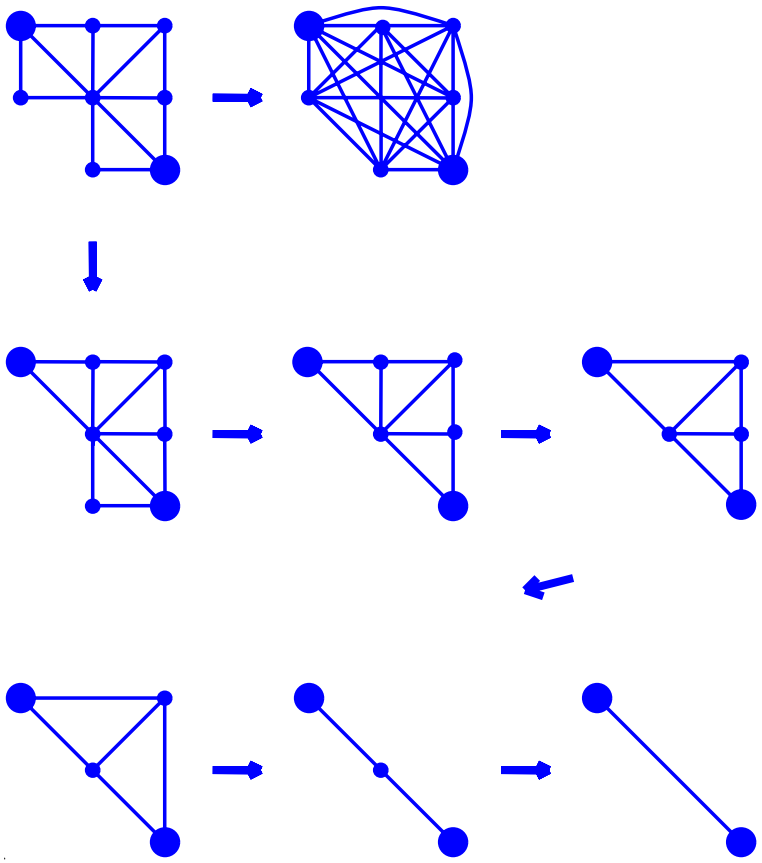
Example (3)





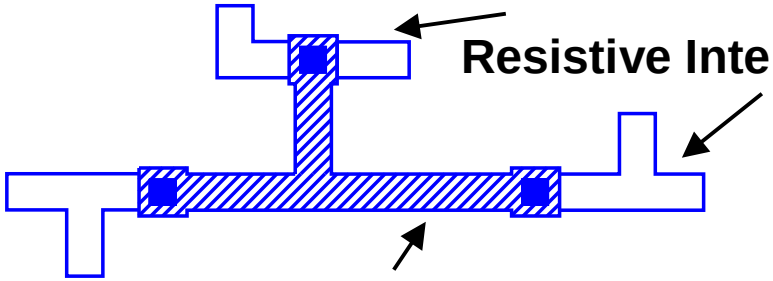
Difference in magnitude and phase between the exact and the extracted circuit as a function of f_s, at that frequency.

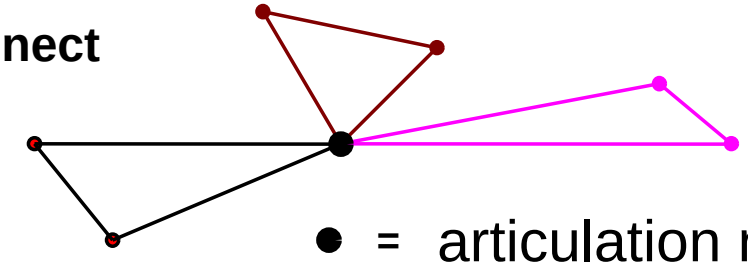
Delayed Frontal Solution



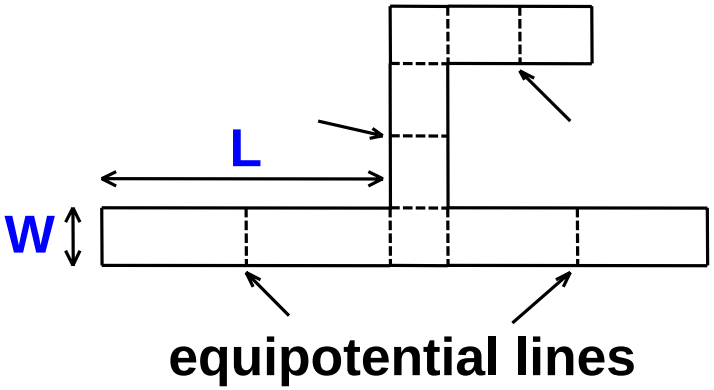
- Elimination of 1 node $O(d^2)$
- Minimize # of fill-ins by optimizing elimination order
- Minimum-degree order traditionally effective heuristic
- ‘Natural’ Scanline order not optimal
- Use priority queue (small buffer) to select nodes in minimum-degree order

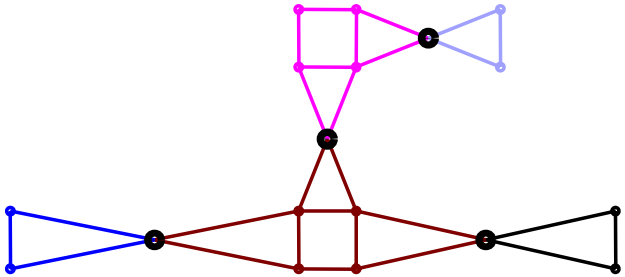
Articulation Nodes





● = articulation node

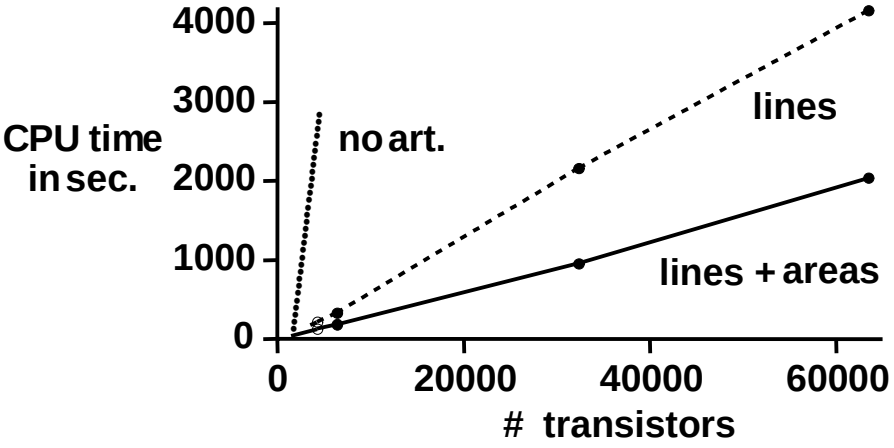


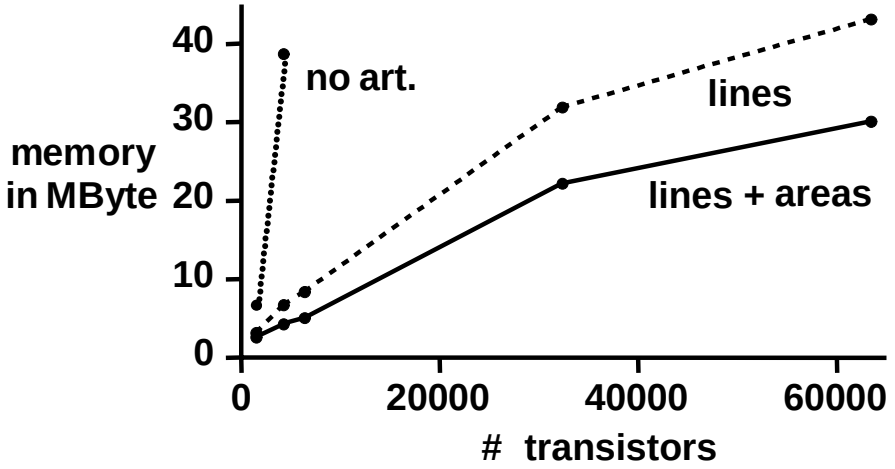


○ = articulation node

- **Articulation Nodes reduce number of fill-ins**

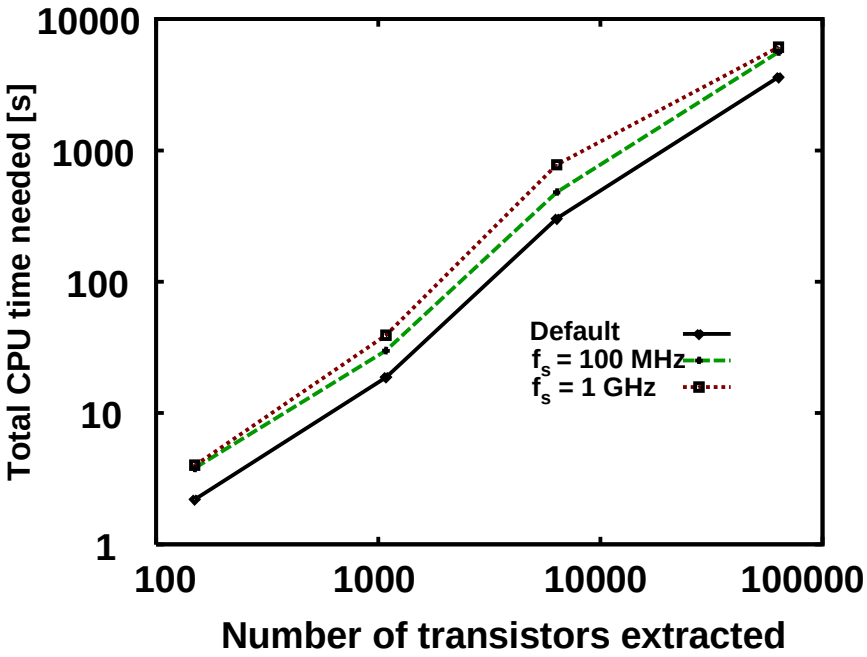
Improved FEM Performance

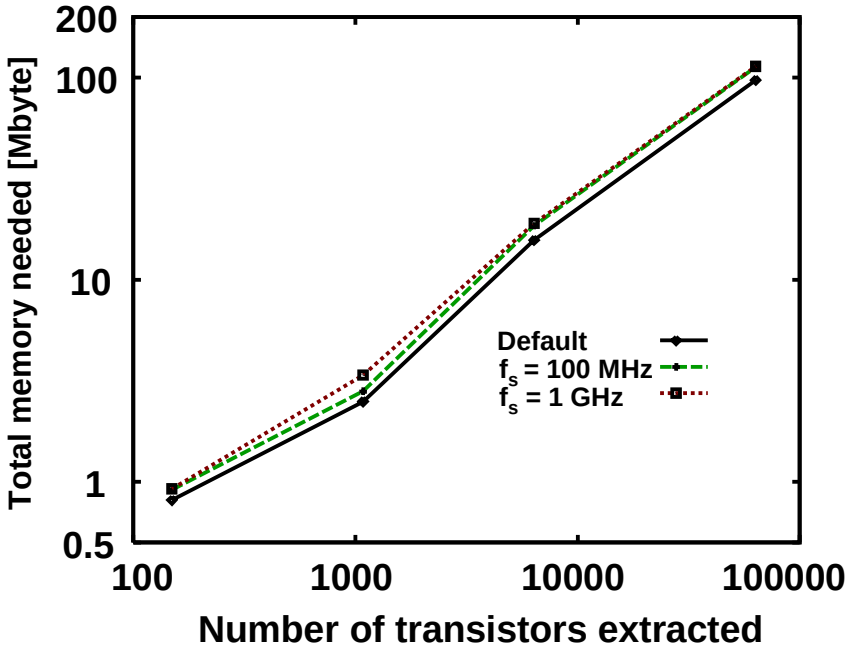




Delayed Frontal Solution and Articulation Nodes enable FEM resistance extraction for large circuits

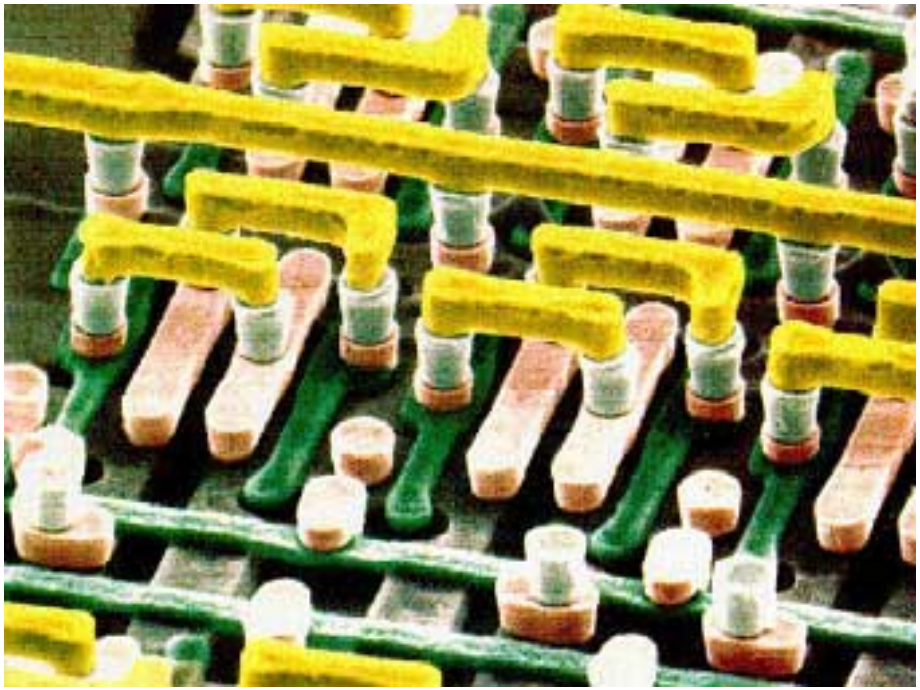
SNE Performance

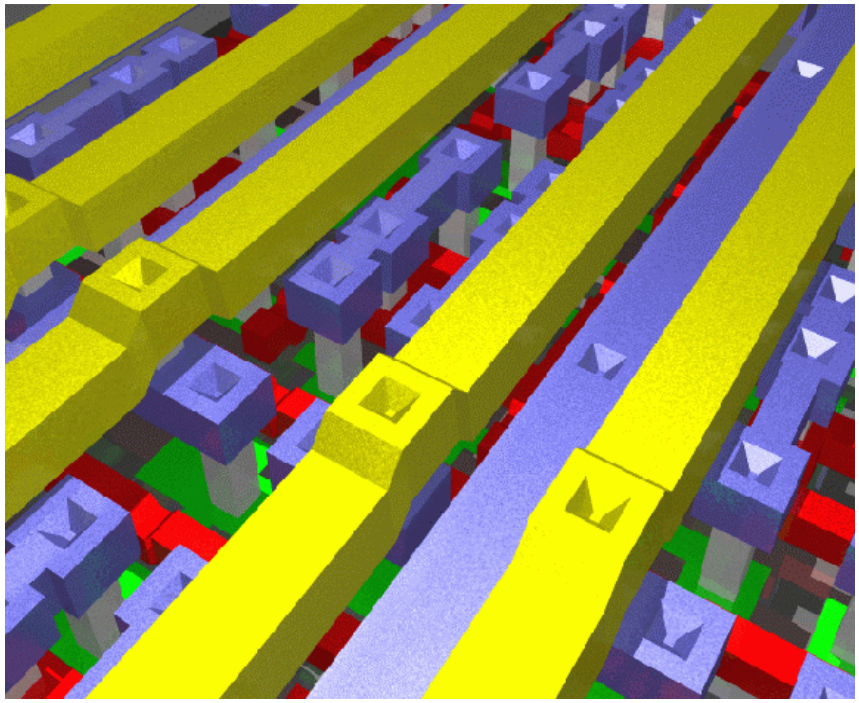




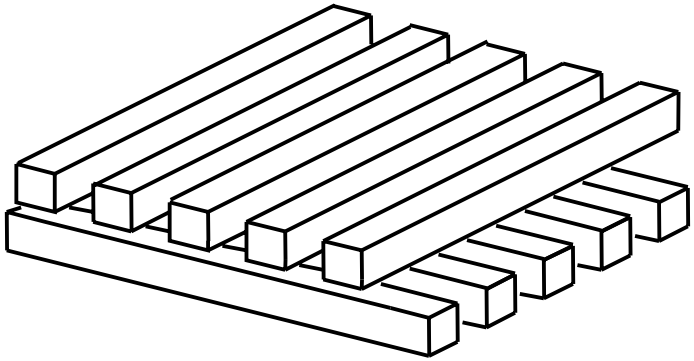
SNE Calculations at relatively small cost

4 3D BEM Capacitance Extraction





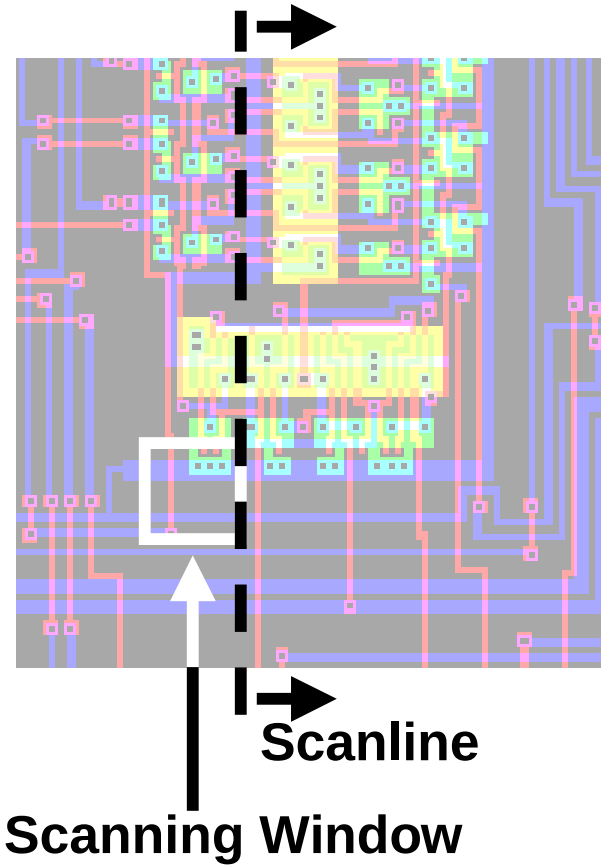
Boundary Element Method



- $\varphi(\vec{x}_0) = \int_{allcharge} G(\vec{x};\vec{x}_0)\xi(\vec{x})d\vec{x}$
- Green's function $G(\vec{x}_j;\vec{x}_i)$ potential at $\vec{x}_j$, due to point-charge at $\vec{x}_i$
- Stratified medium (with ground)
- Only 2D boundary discretization
- N panels, m conductors

- $G_{ij} = \frac{1}{A_j}\int_{\Gamma_j} G(\vec{x}_j;\vec{x}_i)d\Gamma_j$, full but small matrix
- $F \in \{0,1\}^{Nxm}$ relates elements to conductors: $C = F^T \bullet G^{-1} \bullet F$
- Solving above is usually performance bottleneck

Windowing Technique

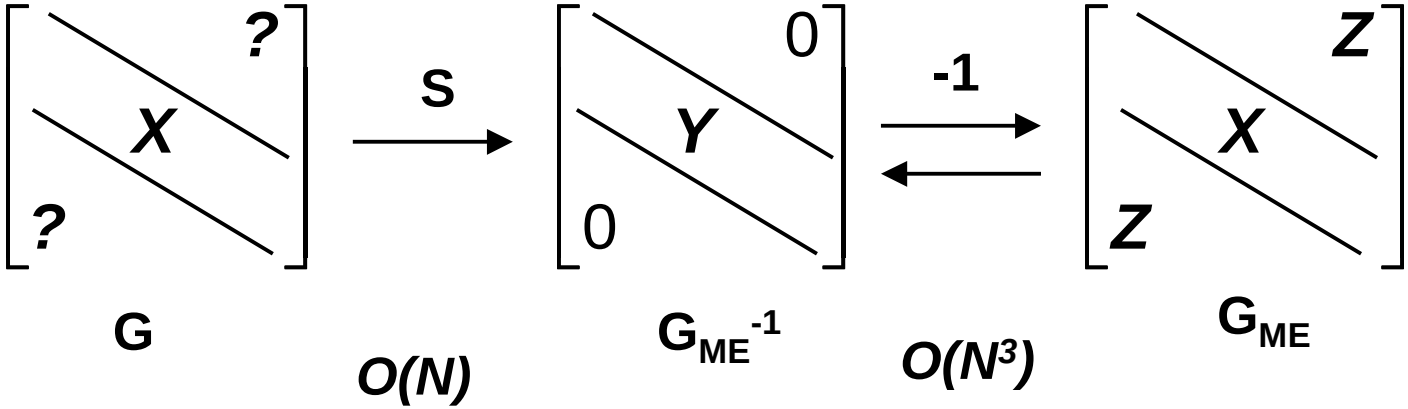


- Capacitive Coupling only over a limited distance
- Only compute Capacitances for features at distance ≤ w
- Moving window attached to scanline
- May not simply put elements of G to zero

⇒ Schur Algorithm

Maximum Entropy Approximation (Schur)

- Capacitance matrix: $C = F^T G^{-1} F$
- Need to invert G, but avoid $O(N^3)$ time complexity.

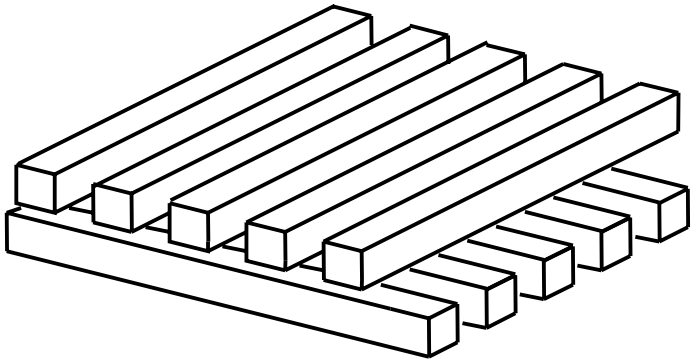


- Partially specified G $\Rightarrow$ reduced order C
- linear in number of panels N
- quadratic in width of band b
- b quadratic in interaction window size w
- complexity $O(Nw^4)$

Schur Modeling Properties

- Good approximation of exact inverse on controllable band around main diagonal
  - entries out of band are 0
  - no direct coupling between distant features
- Neglected coupling detail is accounted for in total capacitance
- Enables linear time complexity with constant memory usage
- A-priori model reduction
- Illustrated on next slide for test structure with 2x5 parallel lines

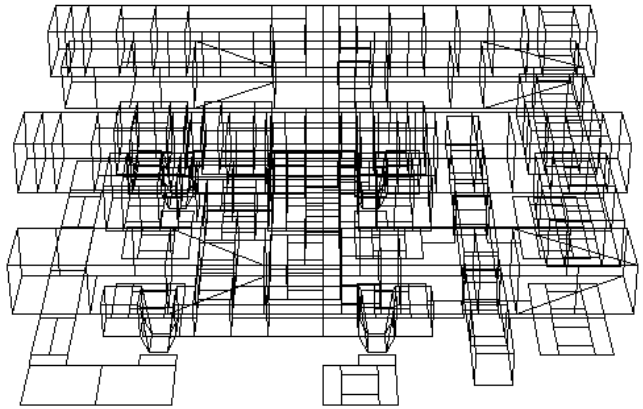
Schur Inversion



w (μ)	time (sec)	mem (Mb)	Capacitances (aF)					
			C_{1g}	C_{12}	C_{13}	C_{14}	C_{16}	C_{1s}
11	144.5	14.74	451.6	596.4	42.3	18.1	147.5	1814.5
5	83.2	4.14	461.1	594.8	41.2	17.4	146.7	1818.7
4	47.8	2.10	474.6	593.9	41.0	21.6	146.3	1822.0
3	26.8	0.78	497.6	592.4	48.1		145.8	1828.1
2	12.5	0.32	552.3	608.6			144.7	1845.6

3D Capacitance – SRAM Example

3D-Model constructed from layout and technology data by the Space layout to circuit extractor.



Statistics (HP 9000/871) – 0.5µm technology

influence window:	3.0	(µm)
number of BE's	835	
number of matrices	3	
number of green's functions	287,093	
largest matrix dimension	625	
largest number entries/row	318	
total cpu-time	1:07	(min:sec)
total memory	8.96	(Mbyte)

3D Capacitances — Window Influences

Cpu time ([hours:min:sec]), memory usage ([MBytes]) — on an HP 9000/871 — and number of capacitances as a function of window size ([μ]).

window	SRAM (6 transistors) cpu time	memory	#C	SRAM 5x5 (150 transistors) cpu time	memory	#C
2	9	3.9	32	4:59	7.2	684
4	1:14	23.3	35	1:18:56	36.5	1046
6	1:45	38.7	36	5:46:08	60.0	1198

5 Interconnect Inductance

- Inductance more important
  - dI/dT noise
  - Crosstalk
  - Reflections
- Multi-million transistor chips as Microwave Circuits
- Microwave design approach (controlled current loops) infeasible
- Microwave modeling techniques inadequate
  - Based on identification of current loops
- Identification of current loops is difficult
  - Include parasitic R and C
  - Current distributes over many possible loops
  - Change with frequency

Partial Element Equivalent Circuits

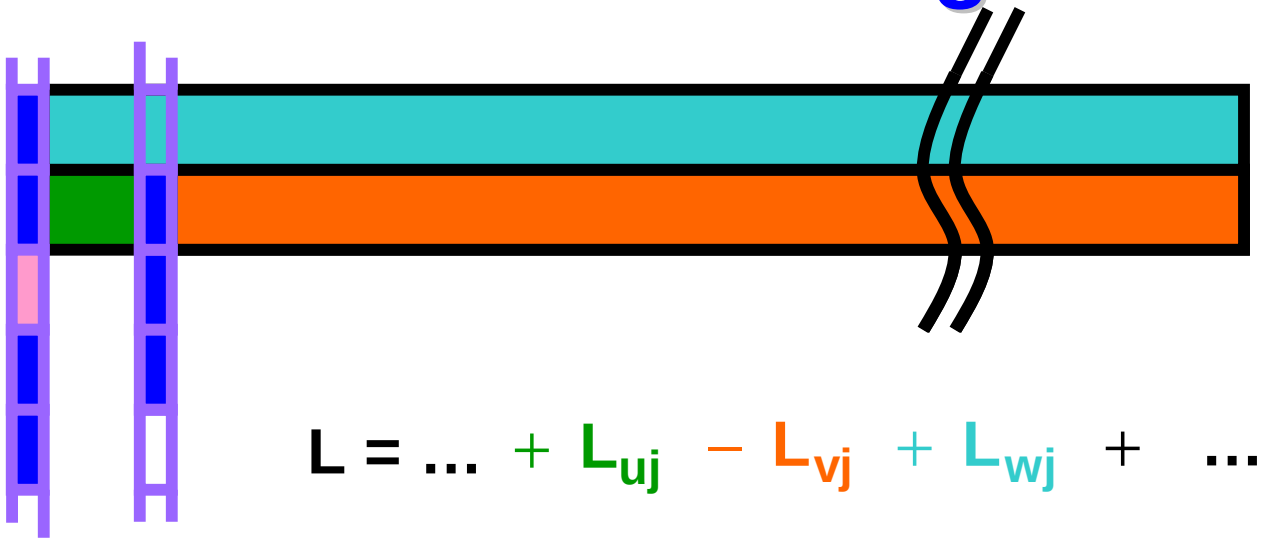
- PEEC [Ruehli]
  - Break loops into loop segments
  - Partial Inductance between each pair of segments
  - Effectively model all possible current loops
- PEEC delivers full comprehensive circuit model
  - Simple algebraic model reduction (w/o R and C)
  - Model reduction (AWE, PVL, …)
  - Circuit simulation
  - Large, dense matrix
- Reduce computation time

Virtual Screening Method

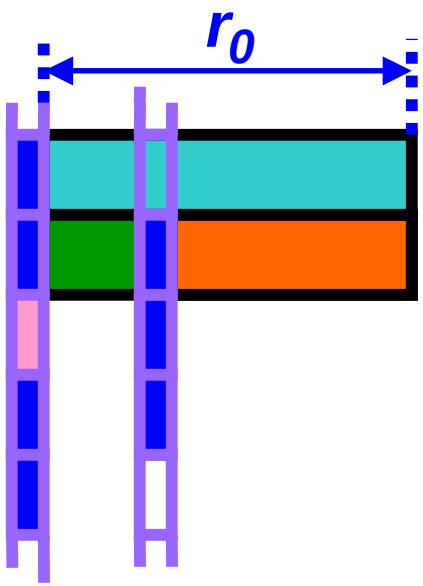
to obtain a

Sparse Partial Inductance Matrix

Virtual Screening Method

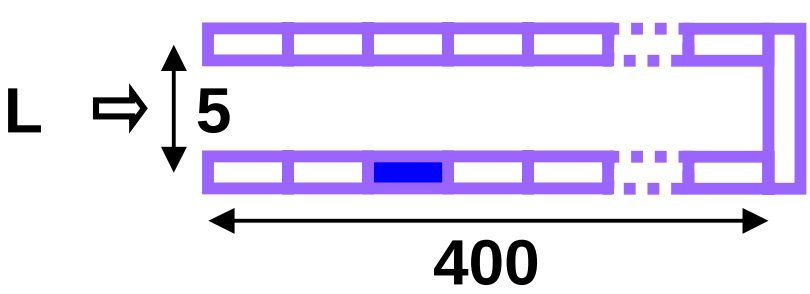


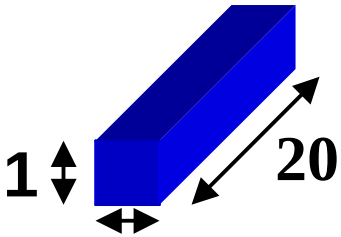
$$L = \ldots + L_{uj} - L_{vj} + L_{wj} + \ldots$$



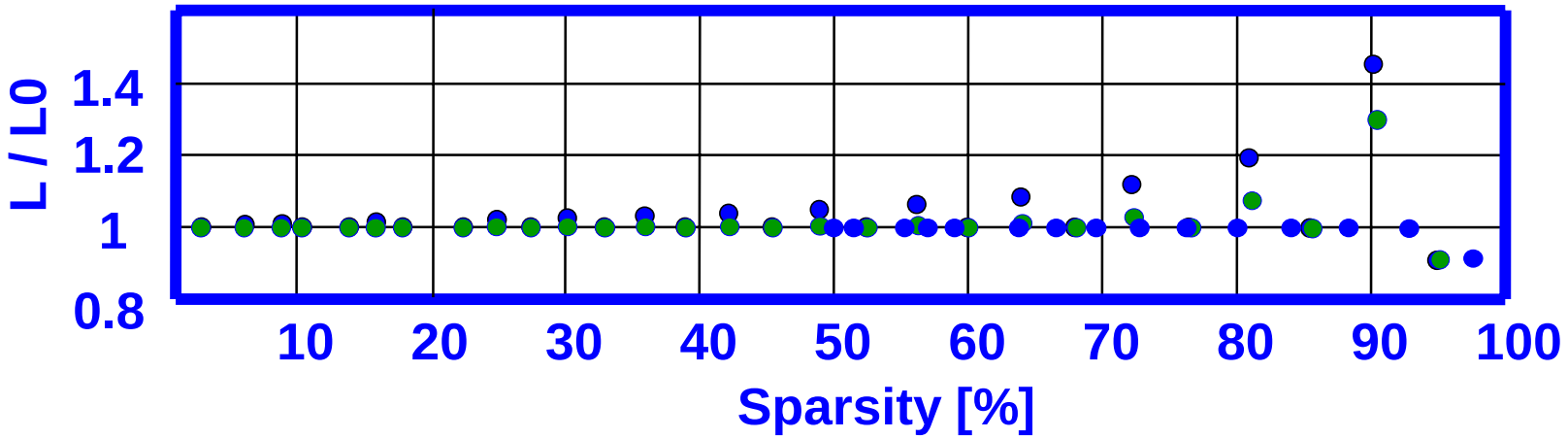
- Cut away **same flux area** for each pair of segments
- No sparsification: **exact results**
- (Strong) sparsification: **improved accuracy**

Results





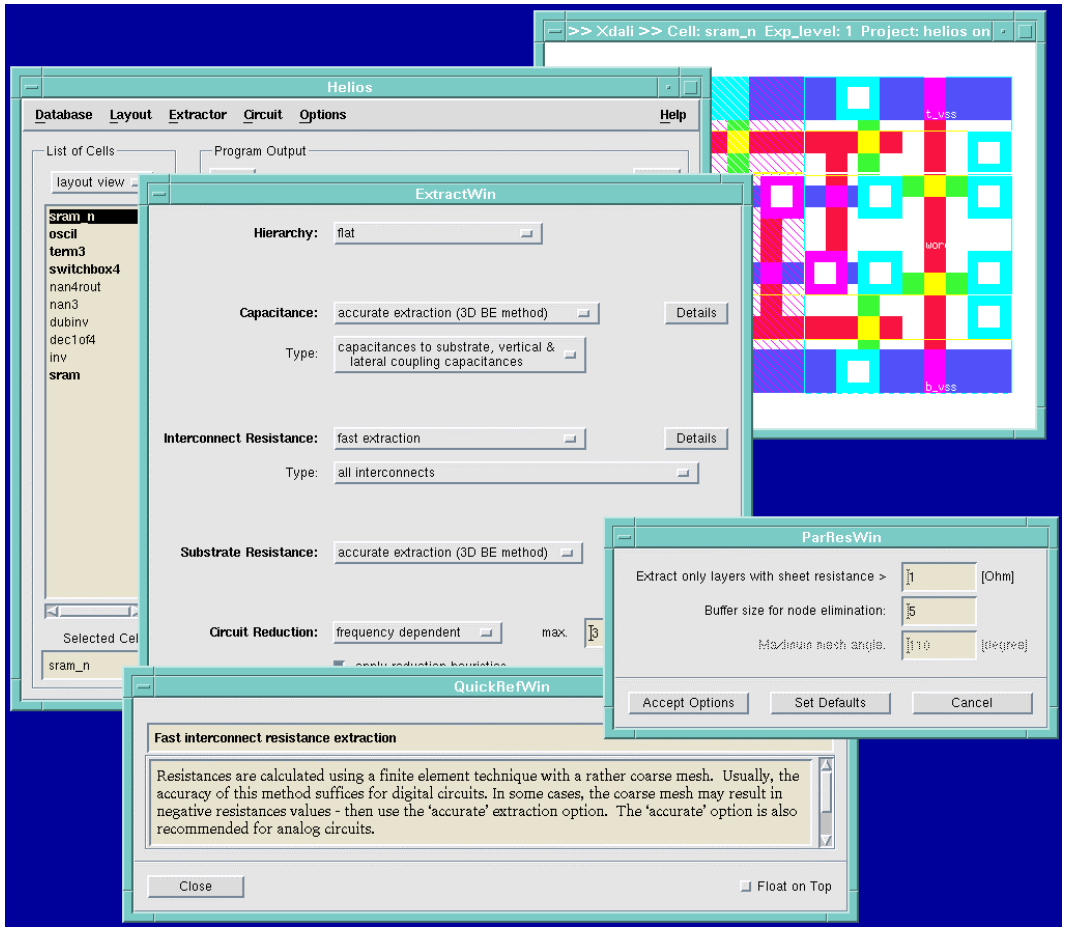
$\#L_{ij}=1600$



$r_0 = \infty$ $r_0 = 50$ $r_0 = 5$

6

Conclusion



- Physical verification of Integrated Circuits
- From geometry (chip layout) to reduced order model (netlist)
- Devices, interconnect (R, C), substrate
- Model Order Reduction as soon as possible
- Consistency of models remains a challenge
- Being used in practice

cas.et.tudelft.nl/space