On Automated Generation of Behavioral Parameterized Macromodels

Part II: SPICE Equivalents and Applications

M. De Stefano, S. Grivet-Talocia, T. Bradde, A. Zanco

Dept. of Electronics and Telecommunications

Politecnico di Torino, Italy

European IBIS Summit, 25/05/2018, Brest, France

Outline

- **Uniform Stability Enforcement**
- **Model SPICE Extraction**
- **Parameter-dependent SPICE Synthesis**
- **Applications**

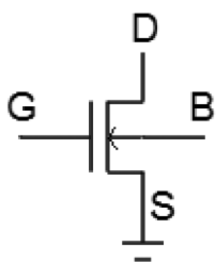
Bias-Dependent Components

$$\dot{\tilde{\boldsymbol{x}}}(\boldsymbol{t}) = \mathbf{A}(\boldsymbol{\vartheta})\tilde{\boldsymbol{x}}(\boldsymbol{t}) + \mathbf{B}(\boldsymbol{\vartheta})\tilde{\boldsymbol{u}}(\boldsymbol{t})$$
$$\tilde{\boldsymbol{y}}(\boldsymbol{t}) = \mathbf{C}(\boldsymbol{\vartheta})\tilde{\boldsymbol{x}}(\boldsymbol{t}) + \mathbf{D}(\boldsymbol{\vartheta})\tilde{\boldsymbol{u}}(\boldsymbol{t})$$

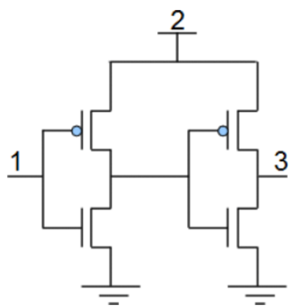
Linearized Systems → Bias-Voltage

From simple examples…

NMOS transistor



Two-Stage Buffer

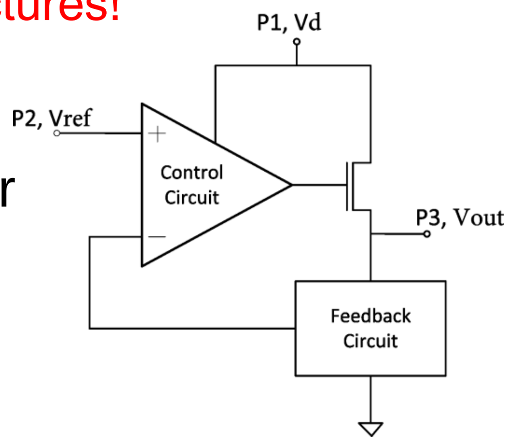


TRANSISTOR LEVEL

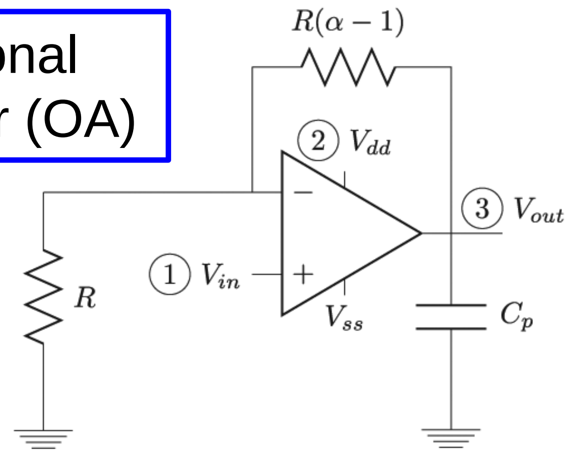
SLOW

… to complex structures!

Low Drop-Out Voltage Regulator (LDO)



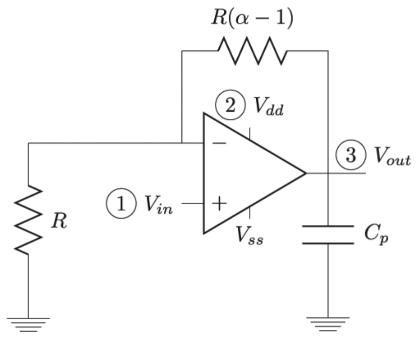
Operational Amplifier (OA)

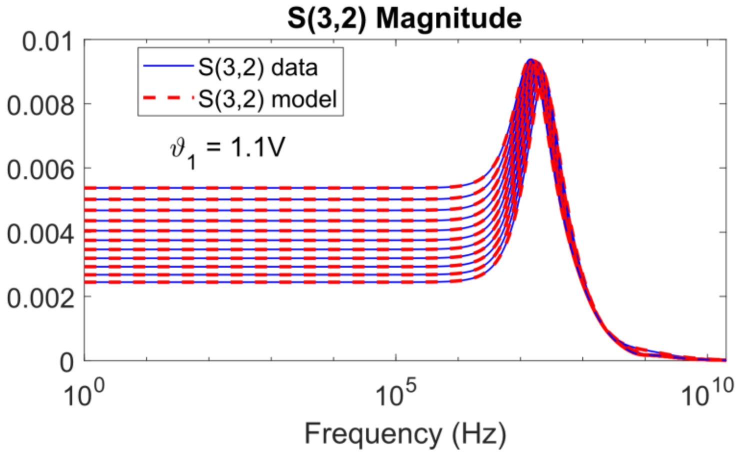


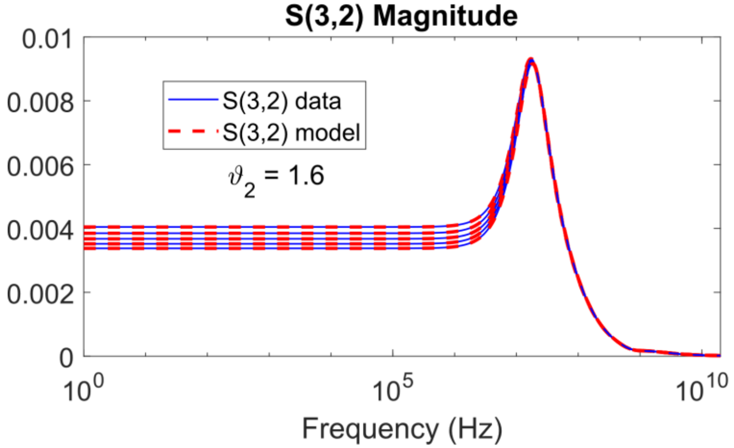
Direct SPICE Implementation : OA Example

Operational Amplifier (OA)

Two Design Parameters
1. Bias-Voltage (ϑ_1)
2. Gain (ϑ_2)



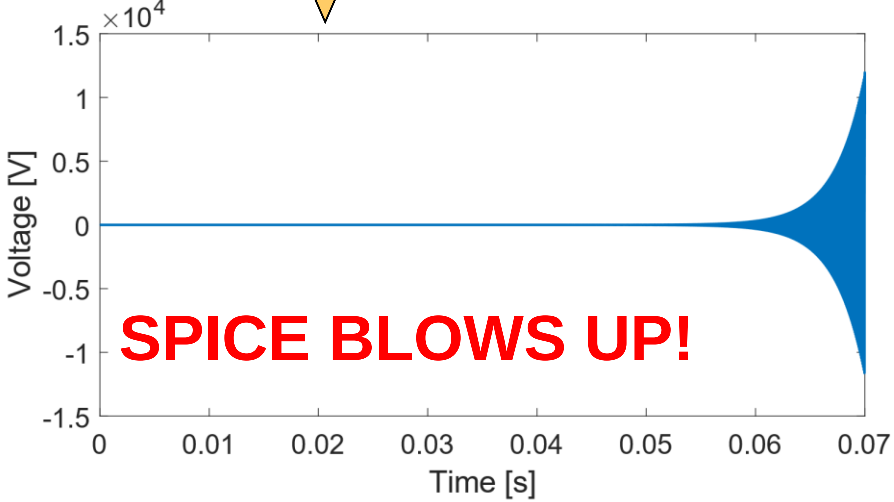




Model accuracy is good but…

OUT

Suppose to generate a SPICE netlist



Direct SPICE Implementation FAILS … Why?

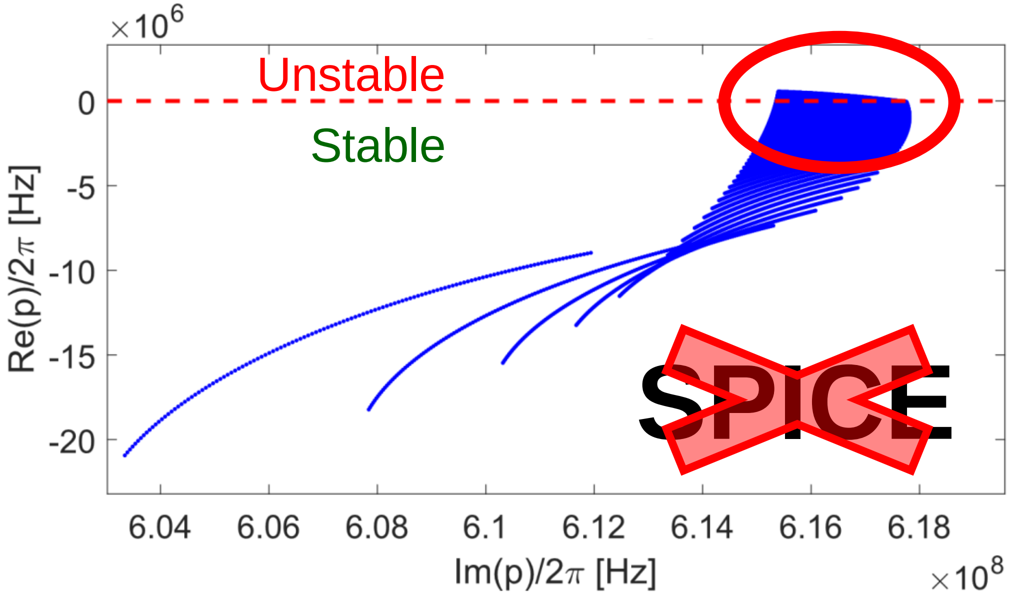
$$\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})} = \frac{\sum_n \sum_l \mathbf{R}_{n,l}\, \xi_l(\boldsymbol{\vartheta})\varphi_n(s)}{\sum_n \sum_l r_{n,l}\, \xi_l(\boldsymbol{\vartheta})\varphi_n(s)} = \sum_{n=1}^{N} \frac{R_n(\boldsymbol{\vartheta})}{s - p_n(\boldsymbol{\vartheta})} + H_\infty(\boldsymbol{\vartheta})$$

Stability ?

Model poles: $p_n(\boldsymbol{\vartheta})$ = zeros of $\mathrm{D}(s;\boldsymbol{\vartheta})$

Parameter-dependent

Model is UNSTABLE



Uniform stability

$\Re\{p_n(\boldsymbol{\vartheta})\} < 0, \forall \boldsymbol{\vartheta}$

We must focus on the model Denominator…

Model Stability and PR Denominator

$$\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})} = \frac{\sum_n \sum_l \mathbf{R}_{n,l}\, \xi_l(\boldsymbol{\vartheta})\varphi_n(s)}{\sum_n \sum_l r_{n,l}\, \xi_l(\boldsymbol{\vartheta})\varphi_n(s)} = \sum_{n=1}^{N} \frac{R_n(\boldsymbol{\vartheta})}{s - p_n(\boldsymbol{\vartheta})} + H_\infty(\boldsymbol{\vartheta})$$

Theorem: (Sufficient condition for Uniform Stability)

If $\mathrm{D}(s;\boldsymbol{\vartheta})$ is a **Positive-Real** function, then $\Re\{p_n(\boldsymbol{\vartheta})\} < 0, \forall \boldsymbol{\vartheta}$

See $\mathrm{D}(s;\boldsymbol{\vartheta})$ as a passive immittance function

1. $\mathrm{D}(s;\boldsymbol{\vartheta})$ regular for $\Re\{s\} > 0$
2. $\mathrm{D}^*(s;\boldsymbol{\vartheta}) = \mathrm{D}(s^*;\boldsymbol{\vartheta})$
3. $\Re\{\mathrm{D}(s;\boldsymbol{\vartheta})\} \geq 0$ for $\Re\{s\} > 0$

(1, 2) → guaranteed by model structure

(3) → can be checked only for $s = j\omega$

$$\Re\{\mathrm{D}(\mathrm{j}\omega;\boldsymbol{\vartheta})\} > 0 \quad \forall\omega, \forall\vartheta \in \Theta$$

Model Identification with Uniform Stability

PSK Scheme … with linear inequality constraints

$D_0 = 1$

for $\mu = 1,2,\ldots$

$$\min \left\| \frac{\mathbf{N}_\mu(s_k;\boldsymbol{\vartheta}_m) - \mathrm{D}_\mu(s_k;\boldsymbol{\vartheta}_m)\breve{\mathbf{H}}(s_k;\boldsymbol{\vartheta}_m)}{\mathrm{D}_{\mu-1}(s_k;\boldsymbol{\vartheta}_m)} \right\|$$

$$\text{s.t.} \quad \Re\{\mathrm{D}_\mu(s_i;\boldsymbol{\vartheta}_i)\} > 0$$

Which points??

end

Cast as

$$\min \|\boldsymbol{x}\|^2$$
$$\text{s.t. } \boldsymbol{Qx} < \boldsymbol{b}$$

Positive-real denominator guaranteed at each iteration

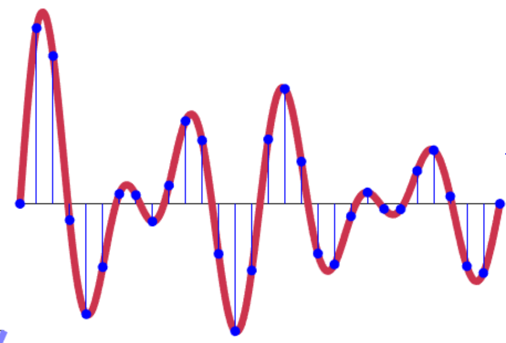
How to Realize the Constraints?

$$\Re\{\mathrm{D}_\mu(\mathrm{j}\omega_i;\boldsymbol{\vartheta}_i)\} > 0$$

Set of Points
at each μ-th iteration

Points from **two** datasets

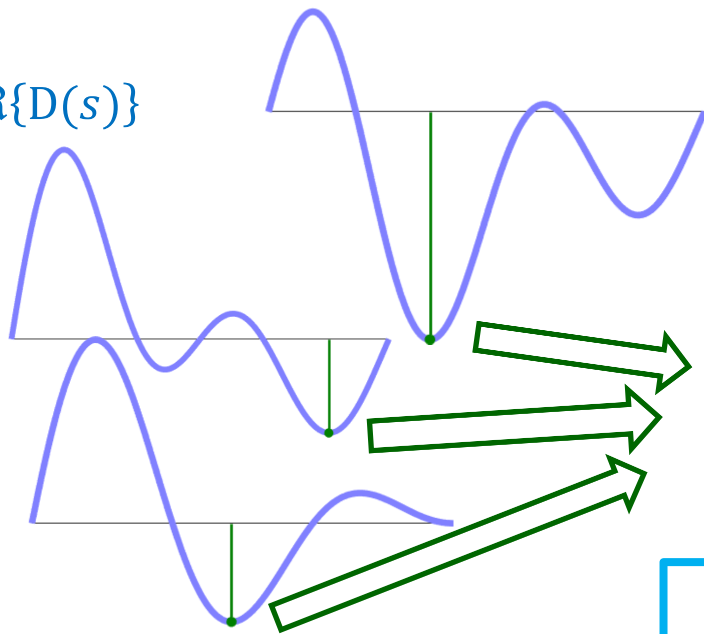
1. FIXED
2. ADAPTIVE



$(\mathrm{j}\omega_k;\boldsymbol{\vartheta}_m)$
$k = 1,\dots,K$
$m = 1,\dots,M$

$\Rightarrow \mathcal{F}$

$\Re\{\mathrm{D}(s)\}$



$\mathcal{W}^\mu$

Adaptive Sampling

- Local minima search based on passivity check
  - See Zanco *et al.* SPI2018

$$(\mathrm{j}\omega_i;\boldsymbol{\vartheta}_i) \in \mathcal{Z}^\mu = \mathcal{F} \cup \mathcal{W}^\mu$$

(Fast) Model Identification with Uniform Stability

$D_0 = 1$

for $\mu = 1,2, \ldots$

$$\min \left\| \frac{\mathbf{N}_\mu(s_k;\boldsymbol{\vartheta}_m) - \mathrm{D}_\mu(s_k;\boldsymbol{\vartheta}_m)\breve{\mathbf{H}}(s_k;\boldsymbol{\vartheta}_m)}{\mathrm{D}_{\mu-1}(s_k;\boldsymbol{\vartheta}_m)} \right\|$$

$$\text{s.t.}\quad \Re\{\mathrm{D}_\mu(s_i;\boldsymbol{\vartheta}_i)\} > 0$$

fixed and adaptive samples

end

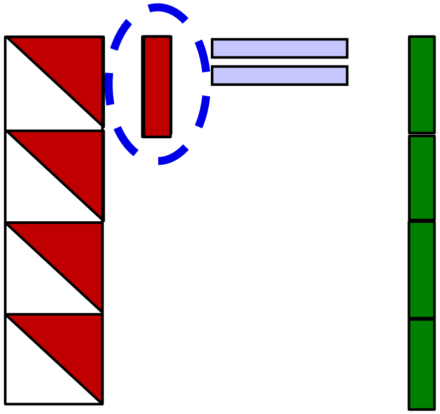
Cast as

$\min \|\boldsymbol{x}\|^2$

$\text{s.t.}\ \boldsymbol{Q}\boldsymbol{x} < \boldsymbol{b}$

Integrated with FPSK Scheme

Constrained LS problem to solve for the denominator unknowns

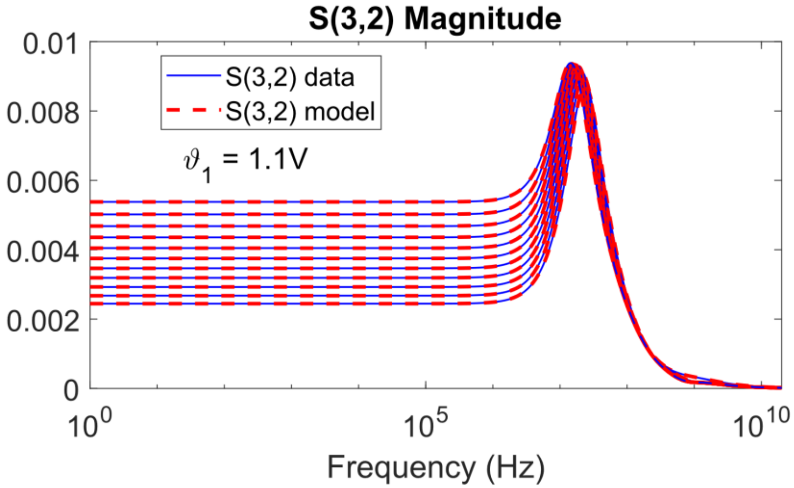


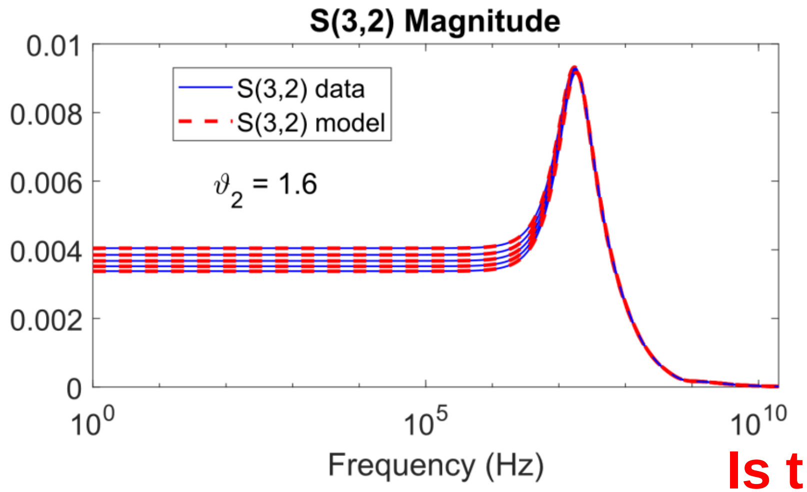
Numerical Results : OA Example

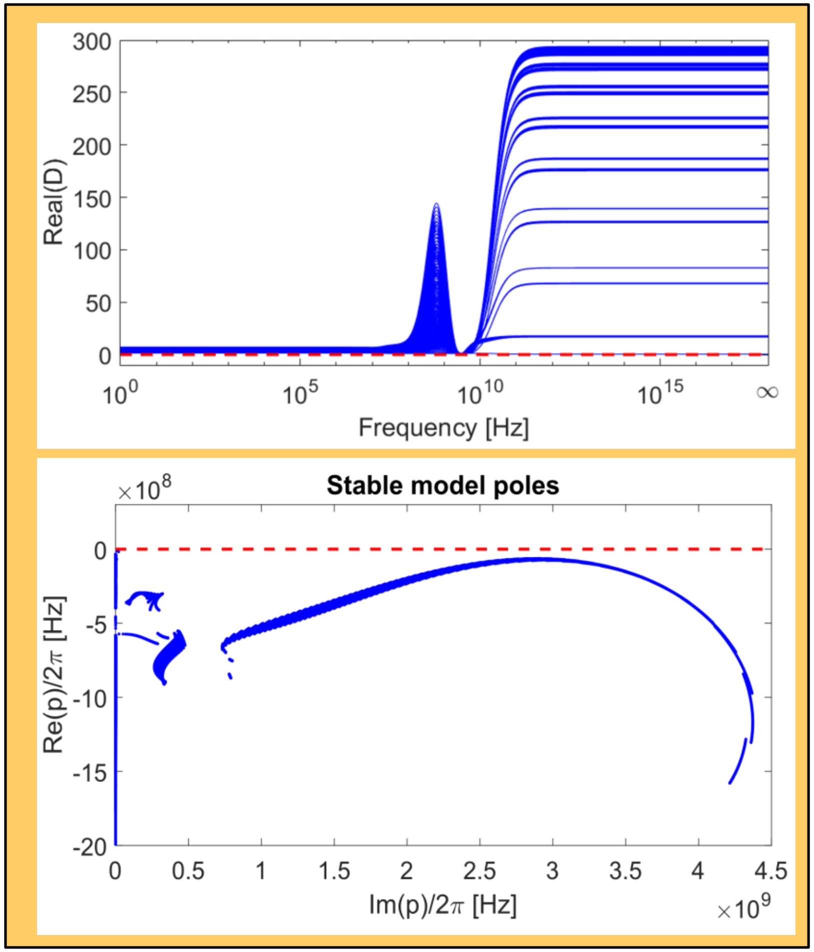
PR

Model accuracy is almost the same and…

… PR denominator → Model is stable!







Is the model stability always guaranteed?

Final Stability Enforcement

Perturb denominator coefficients

$$\widehat{\mathrm{D}}(s;\boldsymbol{\vartheta}) = \sum_n \sum_l \left(r_{n,l} + \Delta r_{n,l}\right) \xi_l(\boldsymbol{\vartheta})\varphi_n(s)$$

$\Delta r_{n,l}$ → $\mathbf{x}$ decision variables

Iterate

Find local minima

Solve constrained least-squares problem

$$\min_{\Delta r_{n,l}} \left\| \underbrace{\widehat{\mathrm{D}}(j\omega;\boldsymbol{\vartheta}) - \mathrm{D}(j\omega;\boldsymbol{\vartheta})}_{\Delta\mathrm{D}(j\omega;\boldsymbol{\vartheta})} \right\|^2$$

$$s.t.\ \Re\{\widehat{\mathrm{D}}(\underbrace{j\omega_i;\boldsymbol{\vartheta}_i}_{\mathcal{Z}^\mu})\} > 0$$

Until PR

Cast as

$\min\|\boldsymbol{x}\|^2$

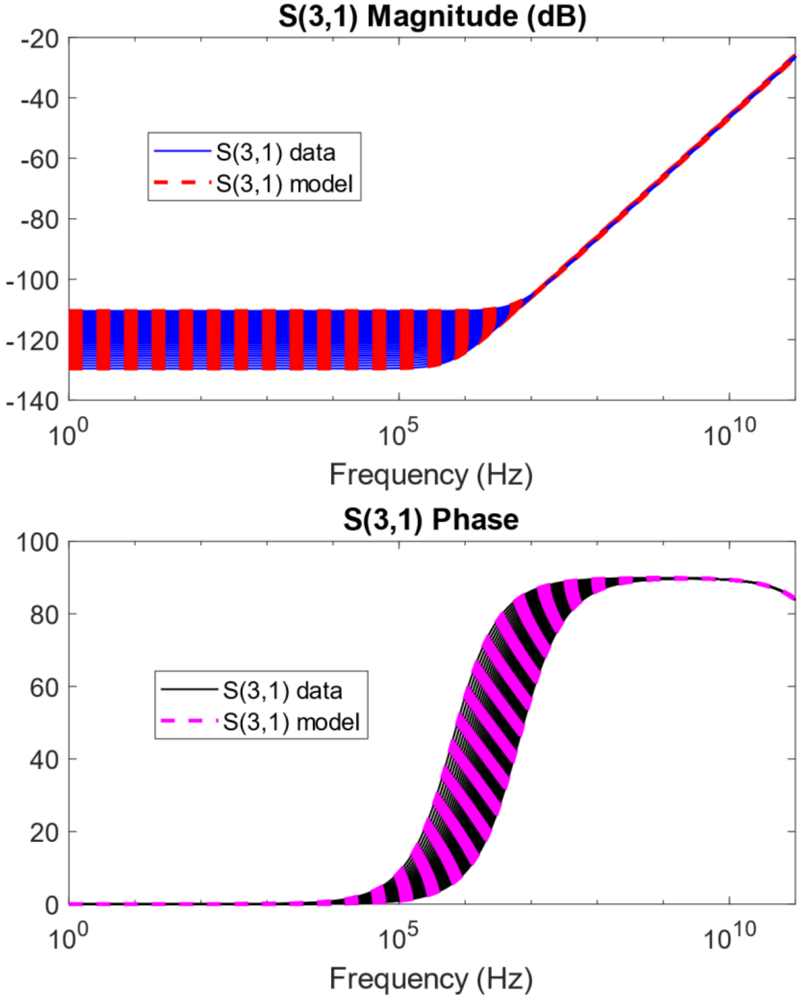
$\mathrm{s.t.}\ \boldsymbol{Q}\boldsymbol{x} < \boldsymbol{b}$

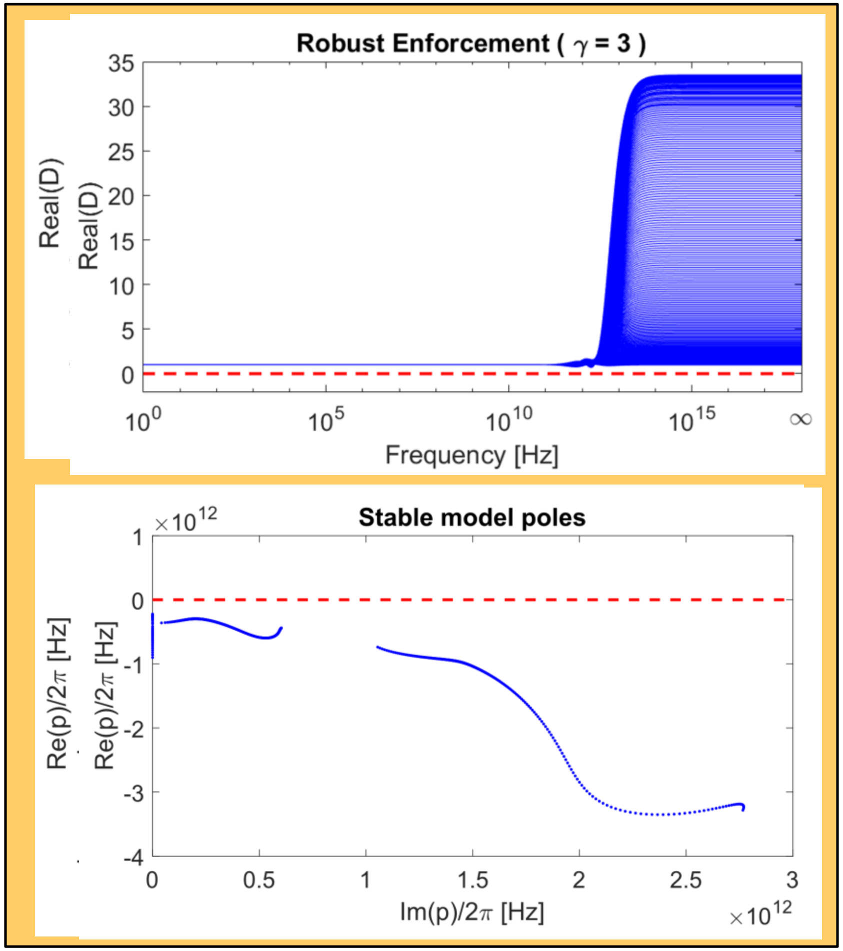
→ Optimize numerator coefficients

Numerical Results : NMOS Example

PR **Robust**

Model accuracy is good and … PR denominator → Model is stable!





(Admittance) SPICE Synthesis: Poles Realization

$$\mathbf{H}(s) = \frac{\mathbf{N}(s)}{\mathrm{D}(s)} = \frac{\sum_n \mathbf{R}_n \varphi_n(s)}{\sum_n r_n \varphi_n(s)}$$

$$\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})} = \frac{\sum_n \mathbf{R}_n(\boldsymbol{\vartheta}) \varphi_n(s)}{\sum_n r_n(\boldsymbol{\vartheta}) \varphi_n(s)}$$

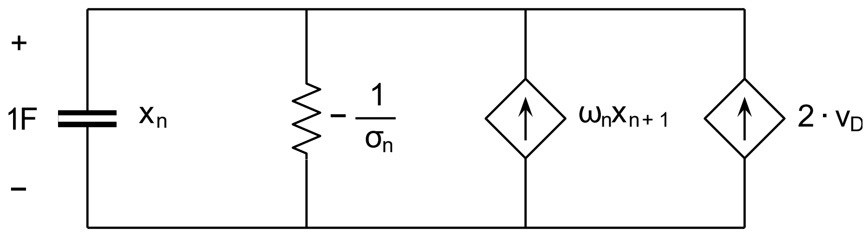
Single Admittance Sub-blocks

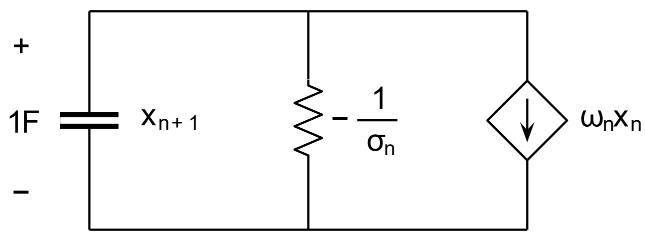
$$\mathrm{D}(s) = \sum_{n=1}^{N} \frac{r_n}{s - q_n} + r_0$$

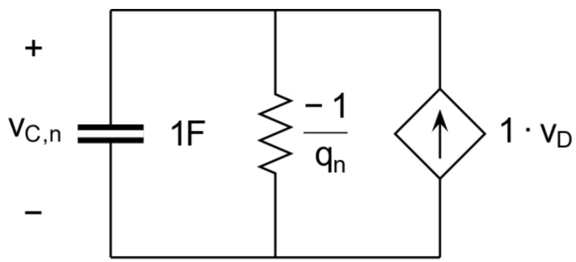
Sparse Synthesis

RC cells for model poles

Common between Numerator and Denominator Sub-blocks







Complex Poles $q_n = \sigma_n \pm j\omega_n$

Real Poles

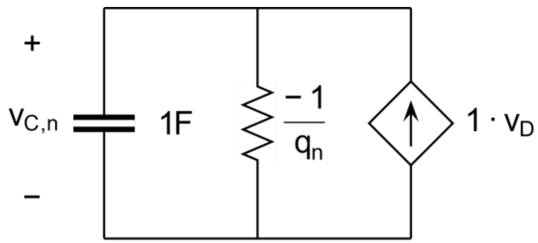
(Admittance) SPICE Synthesis: Interface Circuit

$$\mathbf{H}(s) = \frac{\mathbf{N}(s)}{\mathrm{D}(s)} = \frac{\sum_n \mathbf{R}_n \varphi_n(s)}{\sum_n r_n \varphi_n(s)}$$

$$\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})} = \frac{\sum_n \mathbf{R}_n(\boldsymbol{\vartheta}) \varphi_n(s)}{\sum_n r_n(\boldsymbol{\vartheta}) \varphi_n(s)}$$

Single Admittance Sub-blocks

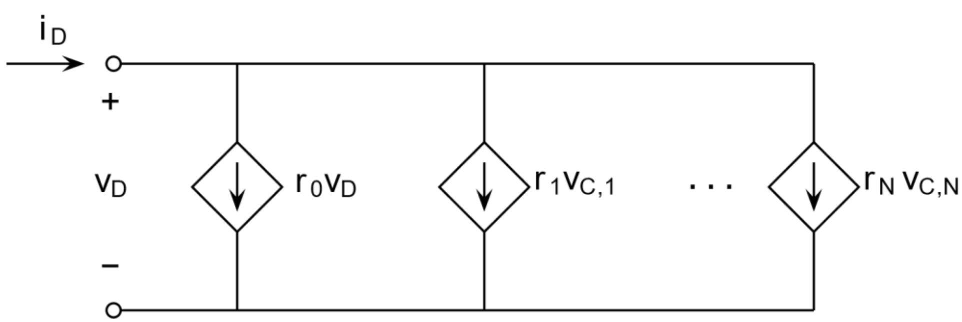
$$\mathrm{D}(s) = \sum_{n=1}^{N} \frac{r_n}{s - q_n} + r_0$$



Sparse Synthesis

RC cells for model poles

Model Residues (Coefficients)



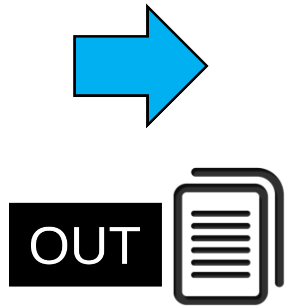
Admittance Interface Circuit

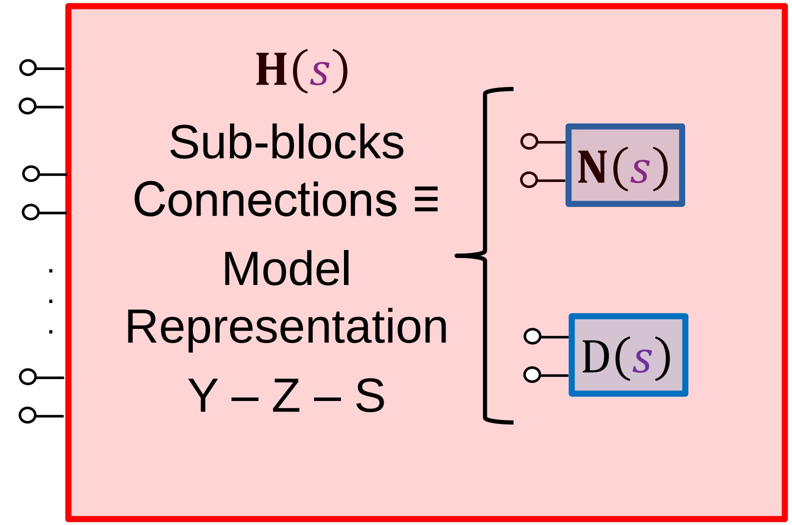
Same procedure for the **(Multiport) Numerator**

Model SPICE Extraction

$$\mathbf{H}(s) = \frac{\mathbf{N}(s)}{\mathrm{D}(s)} = \frac{\sum_n \mathbf{R}_n \varphi_n(s)}{\sum_n r_n \varphi_n(s)}$$

Single Admittance
Sub-blocks





3 SPICE **Output netlists**

1. Numerator file
2. Denominator file
3. Model file

Few kB of memory each

Total number of (elementary) **circuit elements** scales as

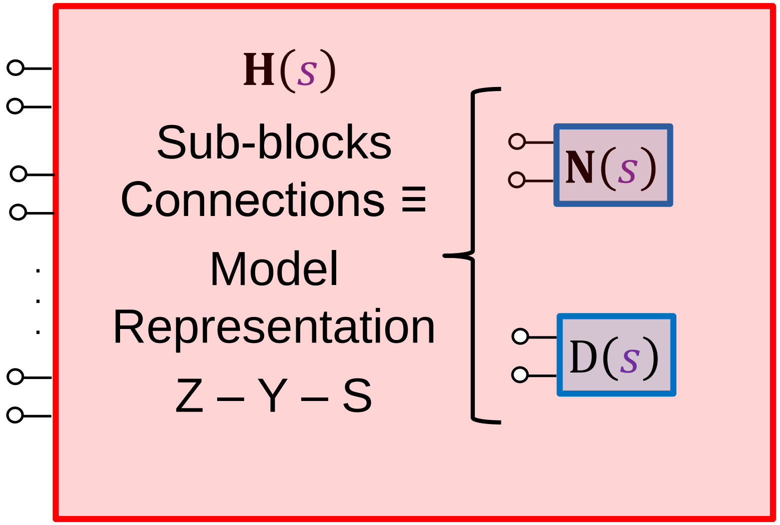
$$\mathbf{O}(NP^2)$$

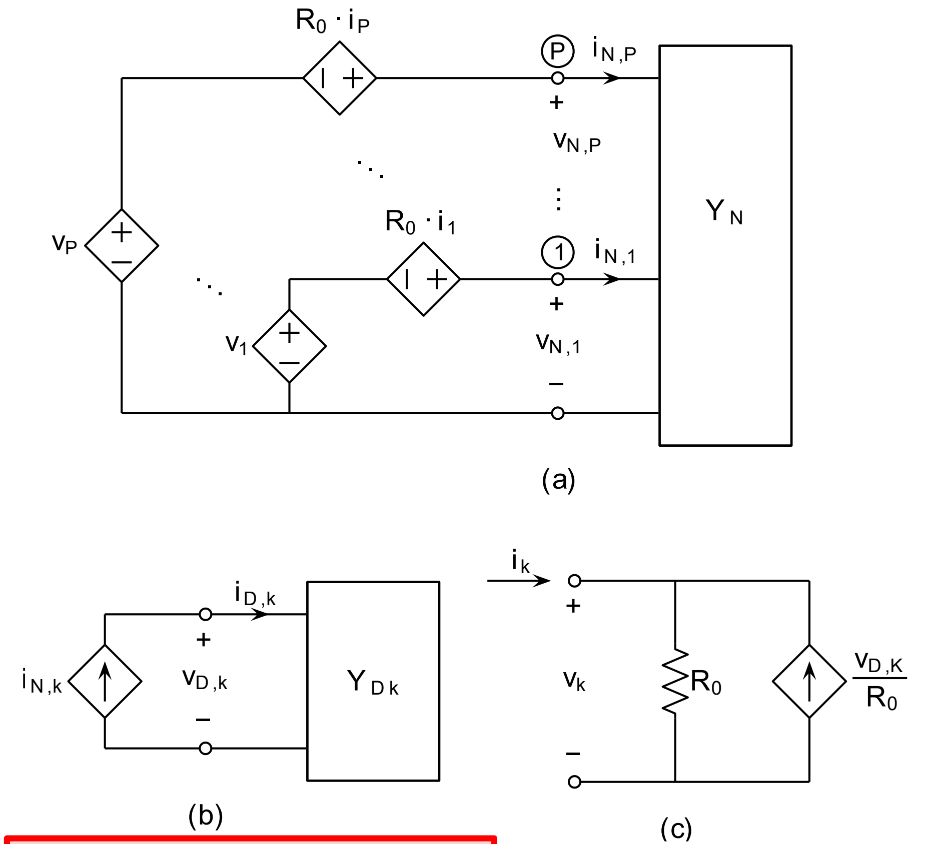
- N : Model poles
- P : Ports

Model SPICE Extraction

$$\mathbf{H}(s) = \frac{\mathbf{N}(s)}{\mathrm{D}(s)} = \frac{\sum_n \mathbf{R}_n \varphi_n(s)}{\sum_n r_n \varphi_n(s)}$$

Single Admittance Sub-blocks



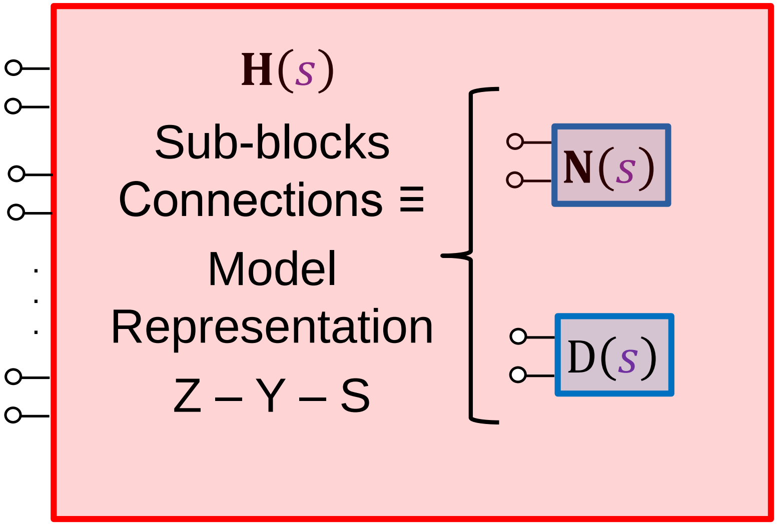


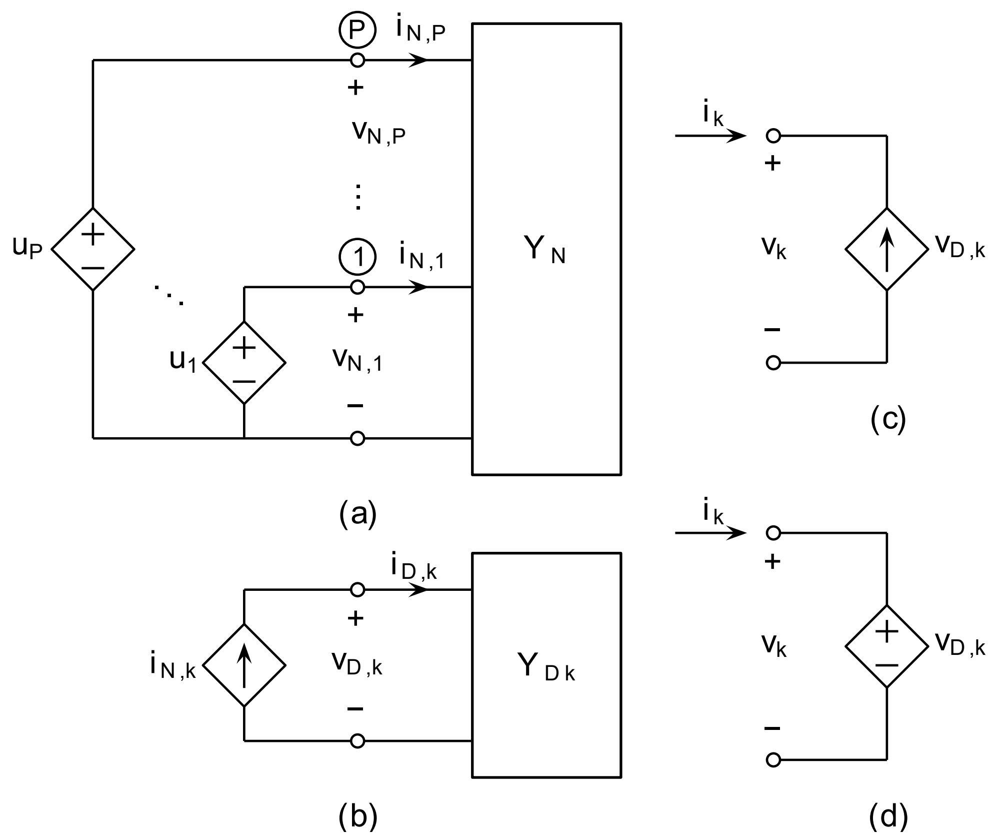
Scattering (S)

Model SPICE Extraction

$$\mathbf{H}(s) = \frac{\mathbf{N}(s)}{\mathrm{D}(s)} = \frac{\sum_n \mathbf{R}_n \varphi_n(s)}{\sum_n r_n \varphi_n(s)}$$

Single Admittance Sub-blocks





Impedance (Z)

Admittance (Y)

NEW

Model SPICE Extraction

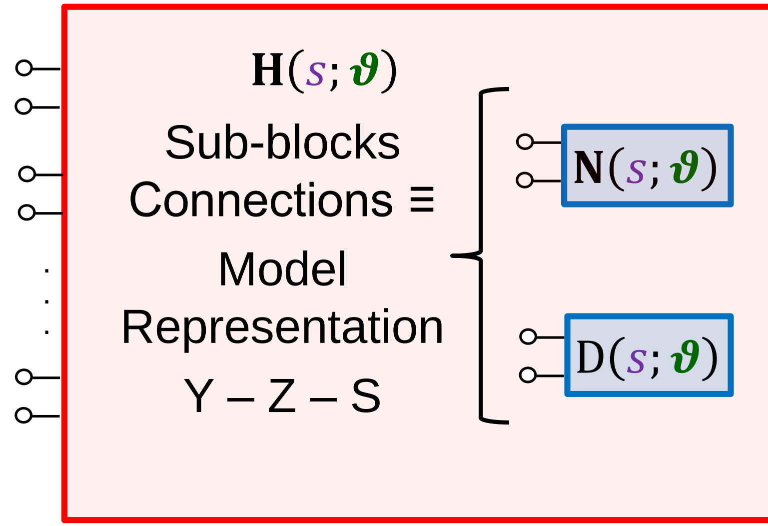
$$D(s;(\boldsymbol{\vartheta})) = \sum_{n=1}^{N} \frac{r_n(\boldsymbol{\vartheta})}{s - q_n} + r_0(\boldsymbol{\vartheta})$$

Parameter-dependent Netlist

$$\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{D(s;\boldsymbol{\vartheta})} = \frac{\sum_n \mathbf{R}_n(\boldsymbol{\vartheta})\varphi_n(s)}{\sum_n r_n(\boldsymbol{\vartheta})\varphi_n(s)}$$

Single Admittance Sub-blocks

OUT



Same pole sub-circuits of non-parametrized model

Parameter-dependent components in **admittance** sub-circuits

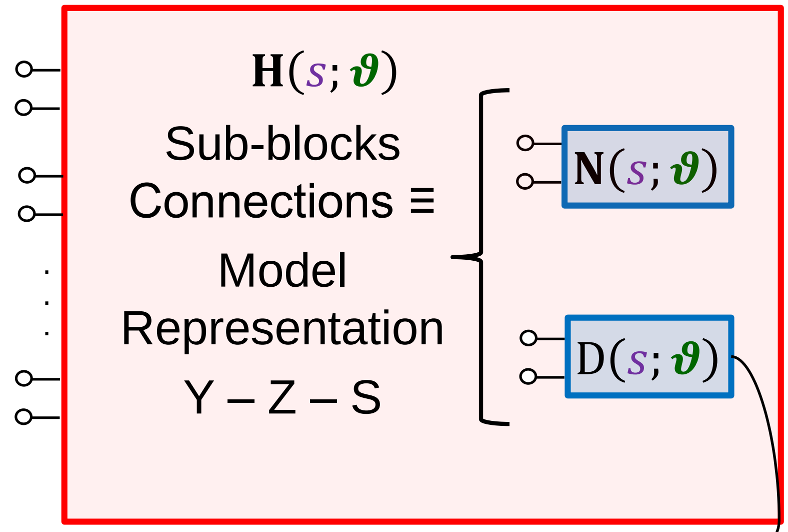
Electrical Interface Nodes

HOW TO PROVIDE THE PARAMETER?

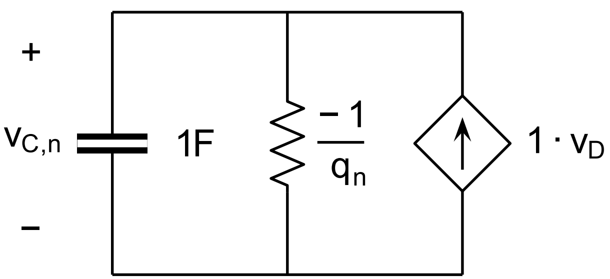
Parameter Call

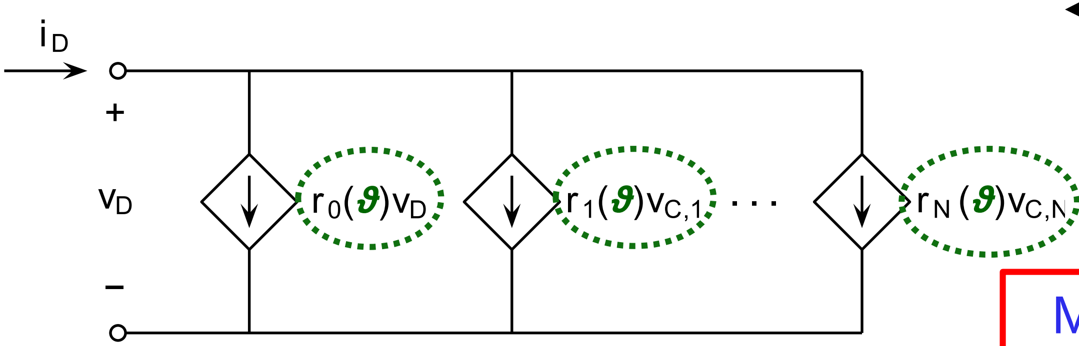
NEW Parameter-Dependent SPICE Models

$$\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})} = \frac{\sum_n \mathbf{R}_n(\boldsymbol{\vartheta})\varphi_n(s)}{\sum_n r_n(\boldsymbol{\vartheta})\varphi_n(s)}$$



Parameter-dependent Netlist





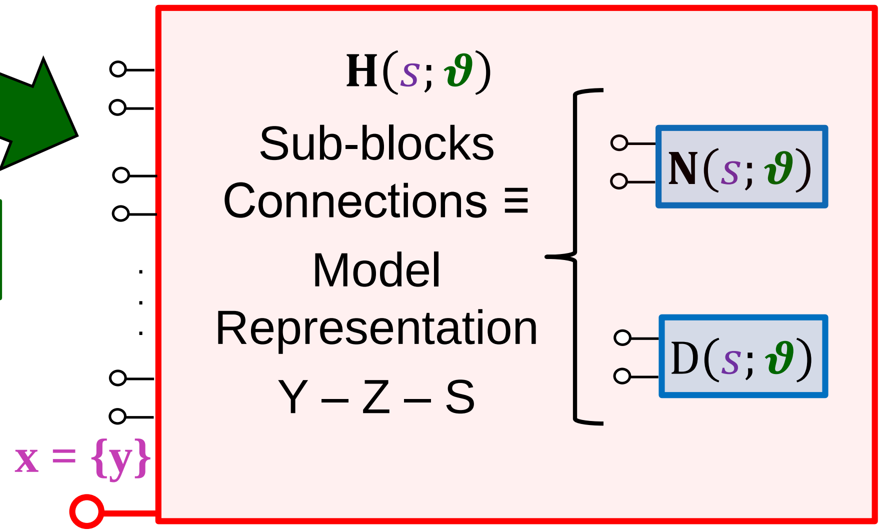
$$\mathrm{D}(s;\boldsymbol{\vartheta}) = \sum_{n=1}^{N} \frac{r_n(\boldsymbol{\vartheta})}{s - q_n} + r_0(\boldsymbol{\vartheta})$$

Multivariate SPICE extraction available

NEW

Parameter-Dependent SPICE Models

$$\mathbf{H}(s;\boldsymbol{\vartheta}) = \frac{\mathbf{N}(s;\boldsymbol{\vartheta})}{\mathrm{D}(s;\boldsymbol{\vartheta})} = \frac{\sum_n \mathbf{R}_n(\boldsymbol{\vartheta})\varphi_n(s)}{\sum_n r_n(\boldsymbol{\vartheta})\varphi_n(s)}$$



Parameter-dependent Netlist

Electrical Interface Nodes

HOW TO PROVIDE THE PARAMETER?

Parameter Call

3 alternative interface circuits

1. Global Par → Global variable
2. Independent Par → Input Variable
3. Control Pin → Additional Input Pin

Option 1: external parameter

```
* Testing netlist
x1 1 0 myModel x={y}
i1 0 1 1


* BEGIN: parameterized macromodel
.subckt myModel n1 n2 params: x=1

[…stuff…]

.ends
* END: parameterized macromodel

* Run a parameter sweep
.step param y -1 1 0.1
.op
.print v(1)
.end
```

Option 2: extra input pin

```
* Testing netlist
x1 1 0 2 myModel
i1 0 1 1
vControl 2 0 {y}

* BEGIN: parameterized macromodel
.subckt myModel n1 n2 controlPin

[…stuff…]

.ends
* END: parameterized macromodel

* Run a parameter sweep
.step param y -1 1 0.1
.op
.print v(1)
.end
```

NEW

Parameter-Dependent SPICE Models

Option 1: external parameter

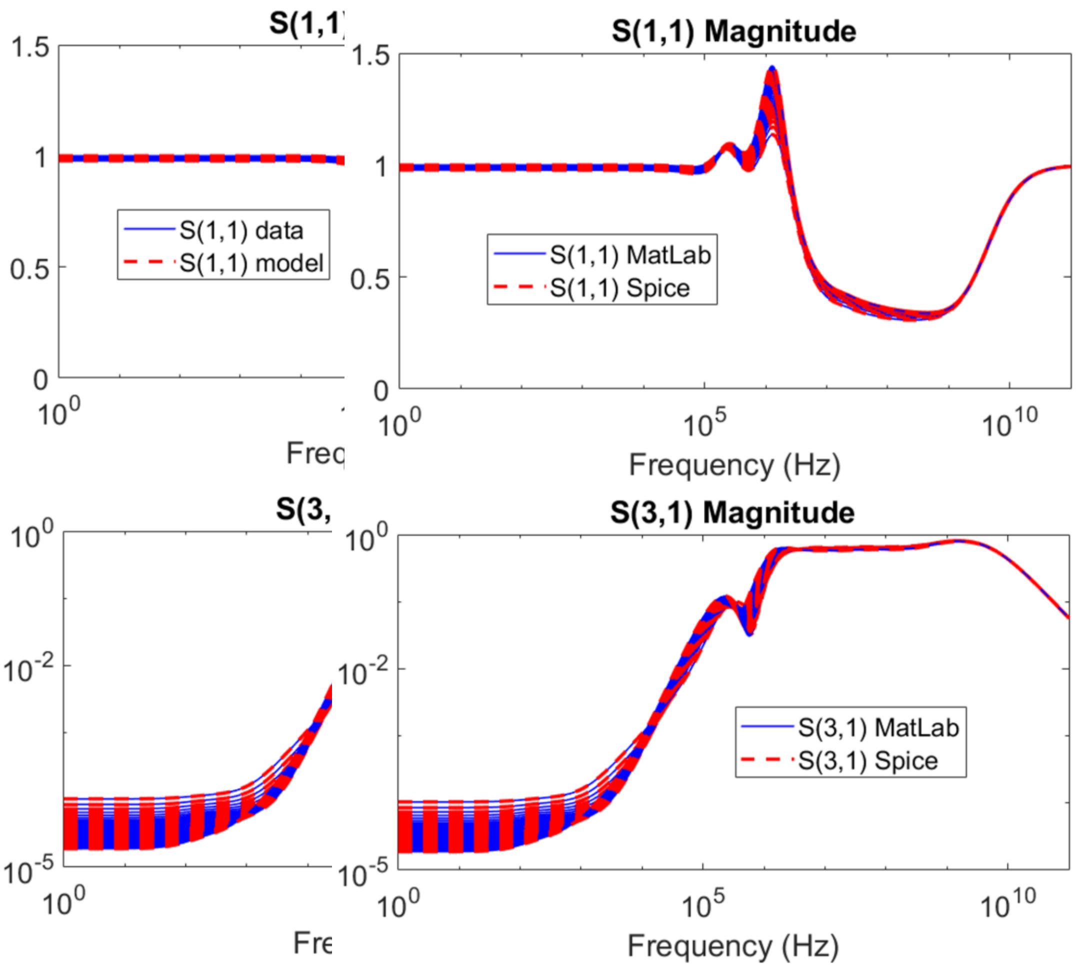
```
* Testing netlist
x1 1 0 myModel x={y}
i1 0 1 1


* BEGIN: parameterized macromodel
.subckt myModel n1 n2 params: x=1

[…stuff…]

.ends
* END: parameterized macromodel

* Run a parameter sweep
.step param y -1 1 0.1
.op
.print v(1)
.end
```

Option 2: extra input pin

```
* Testing netlist
x1 1 0 2 myModel
i1 0 1 1
vControl 2 0 {y}

* BEGIN: parameterized macromodel
.subckt myModel n1 n2 controlPin

[…stuff…]

.ends
* END: parameterized macromodel

* Run a parameter sweep
.step param y -1 1 0.1
.op
.print v(1)
.end
```

SPICE Validations and Numerical Results: LDO

Model is accurate and Stable

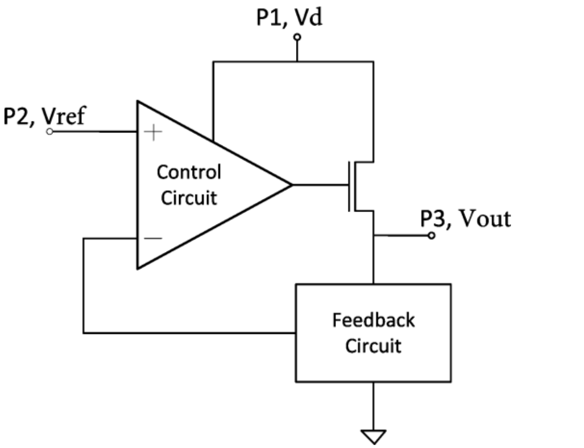


SPICE responses identical to the MATLAB-computed responses

Maximum error is less than 1e-12

Three circuit interfaces available

- 2 with electrical interface nodes only and parameter as variable
- Electrical interface nodes only + control pin

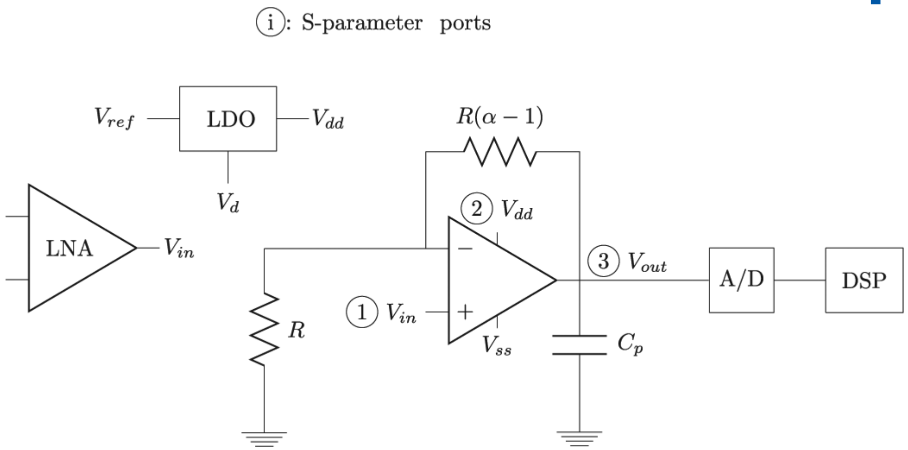


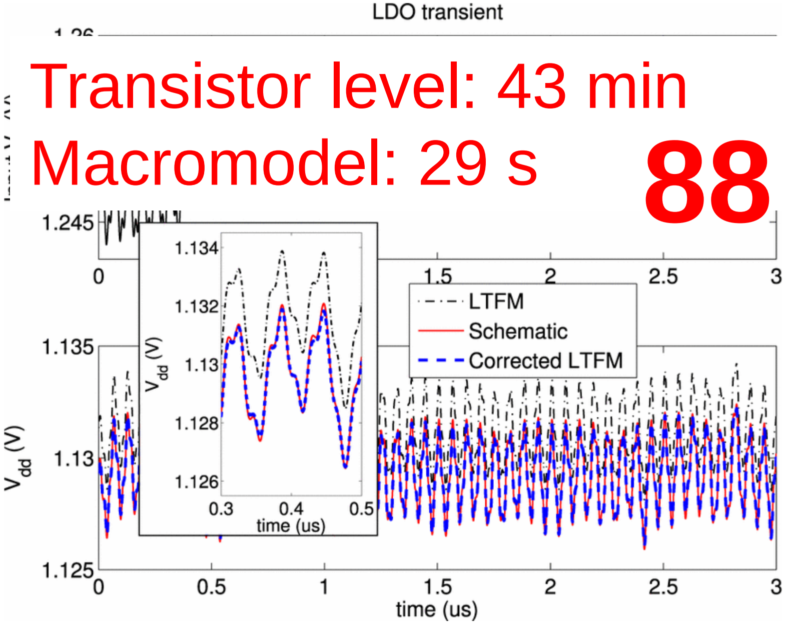
Example 1

LDO and OA

Work in Progress

Application of parameter-dependent DC correction

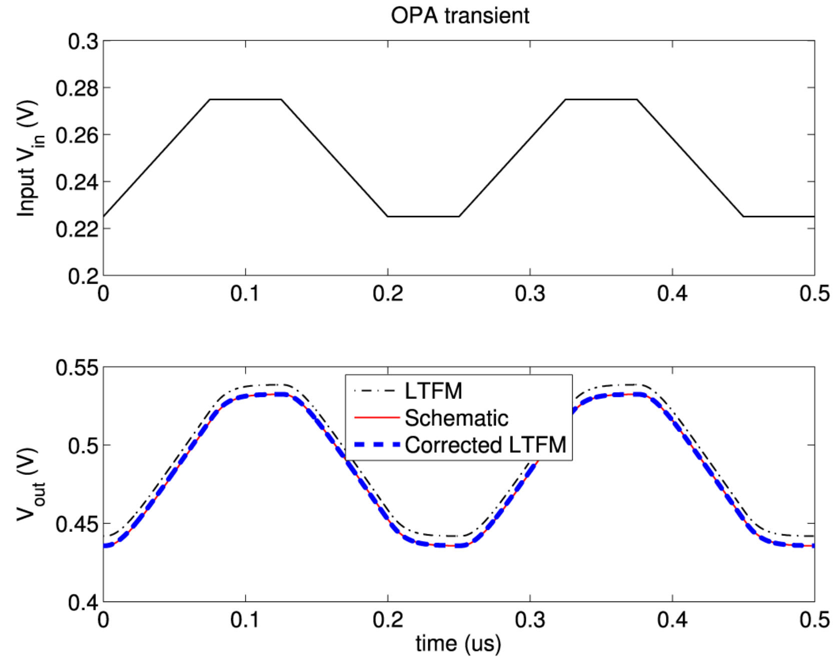




Transistor level: 43 min

Macromodel: 29 s

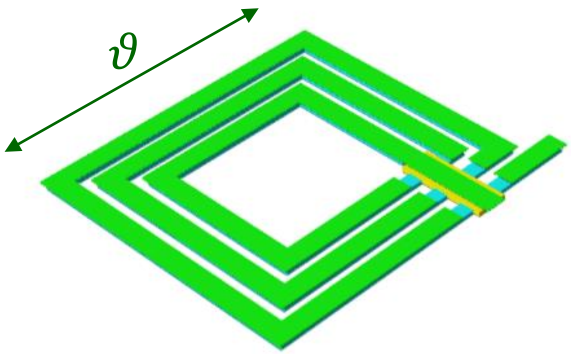
88 X



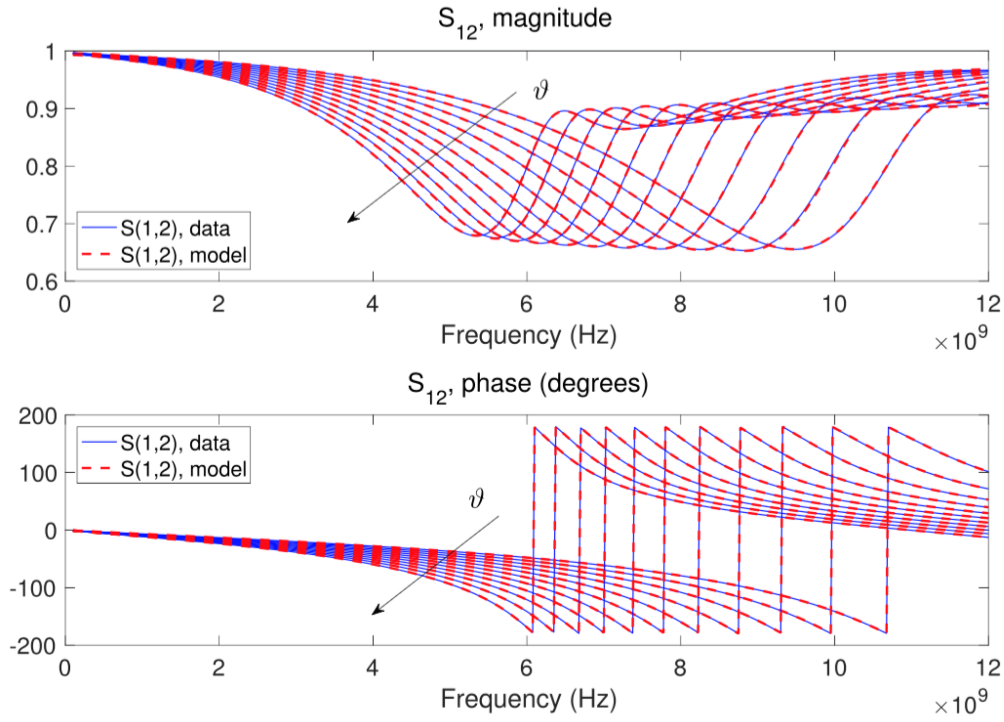
Example 2

Integrated inductor (square, 1.5 turns, multilayer)

[Courtesy: Prof. Swaminathan, Georgia Tech, USA]

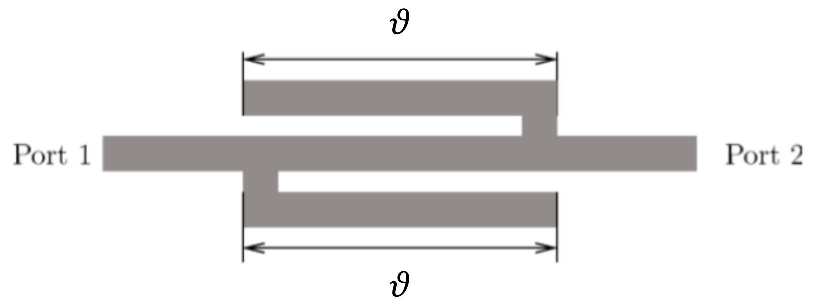


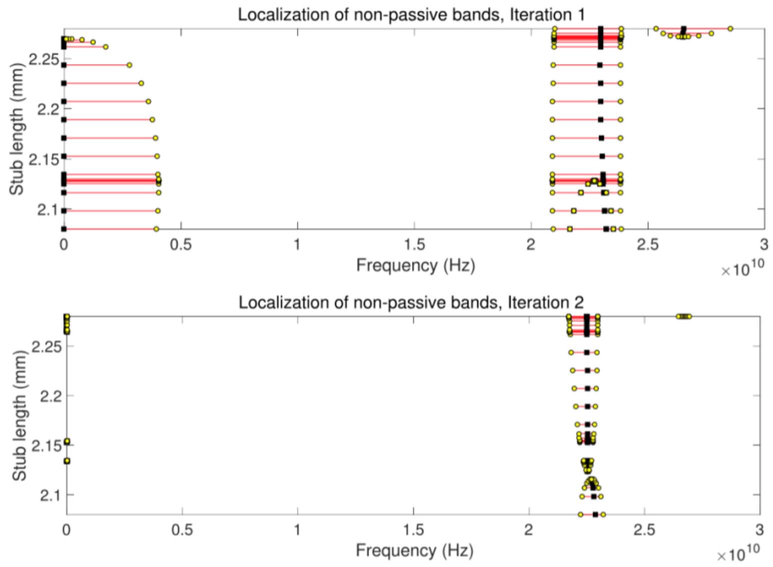
(drawing for illustration only)

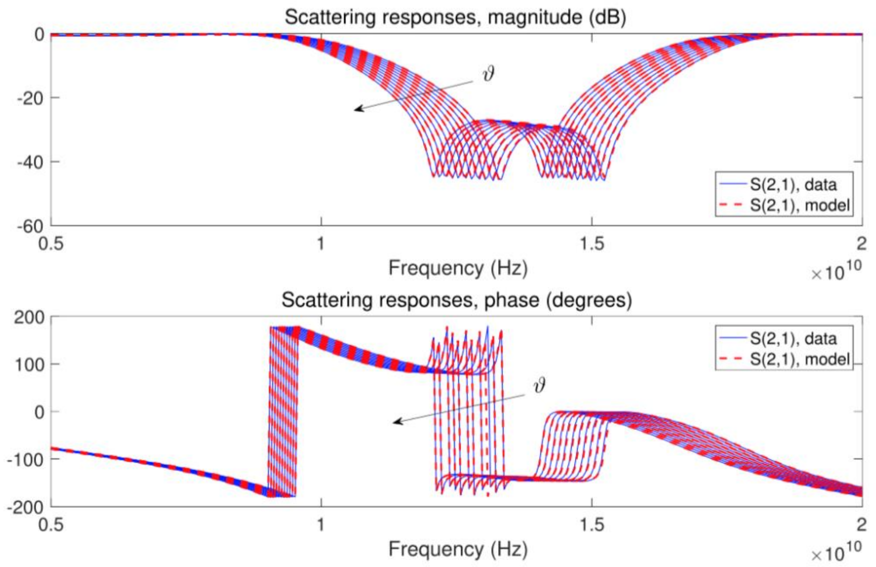


Example 3

Double-folded microstrip filter

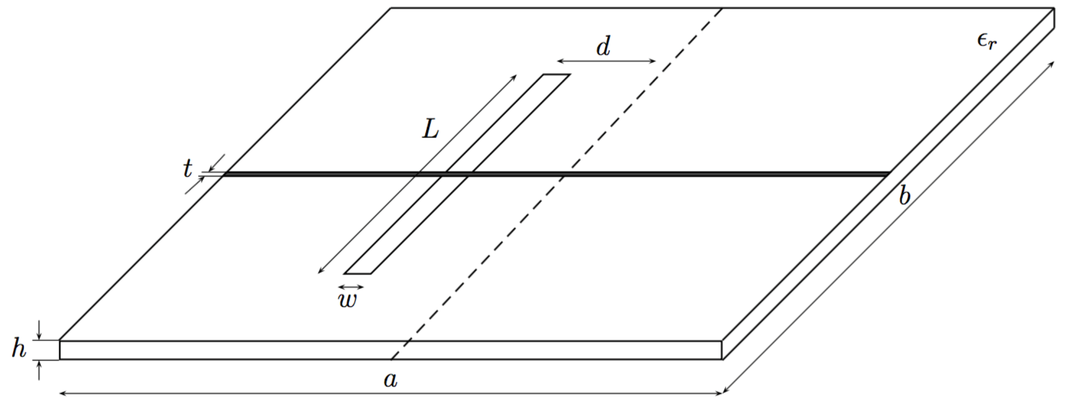


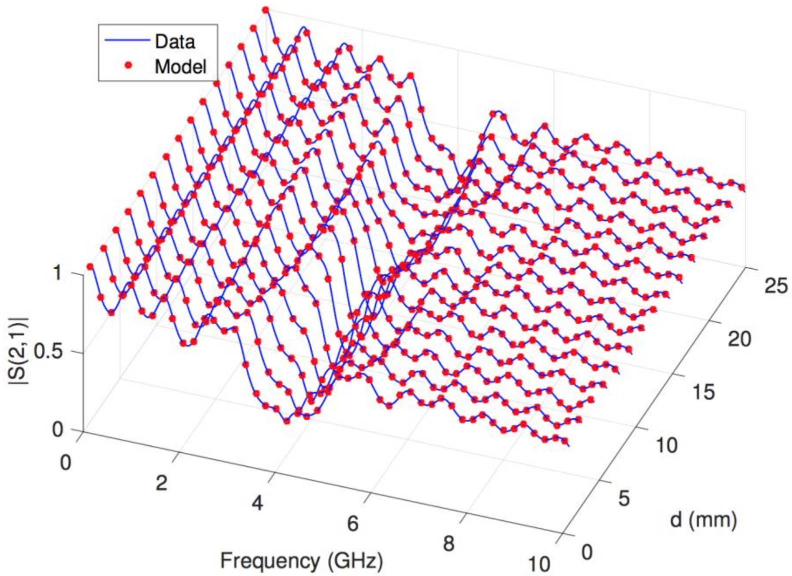


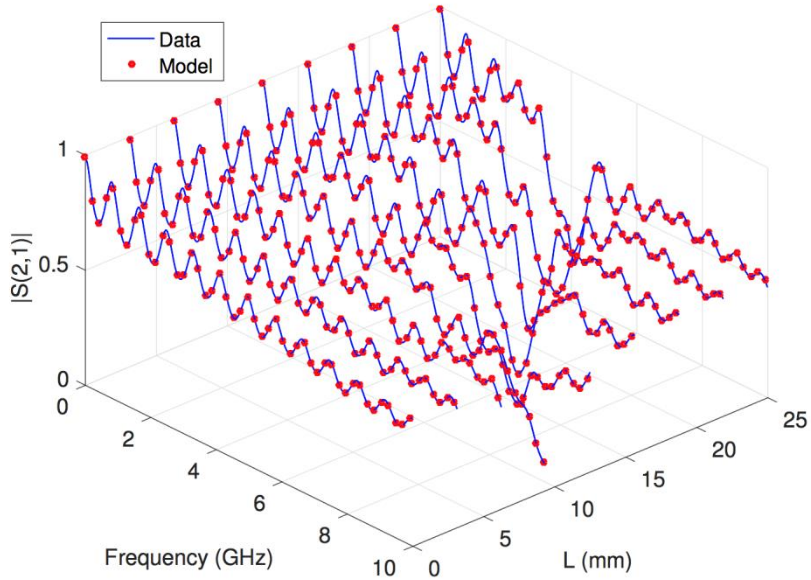


Example 4

PCB interconnect over a slotted reference plane

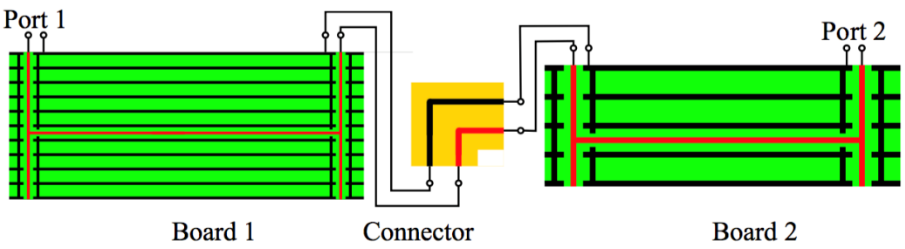




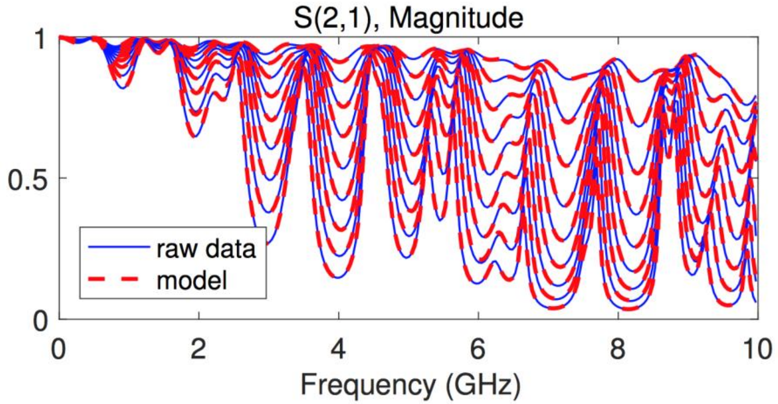


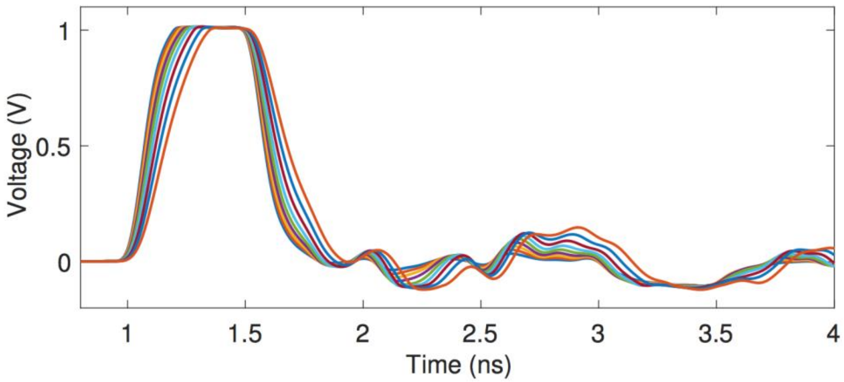
Example 5

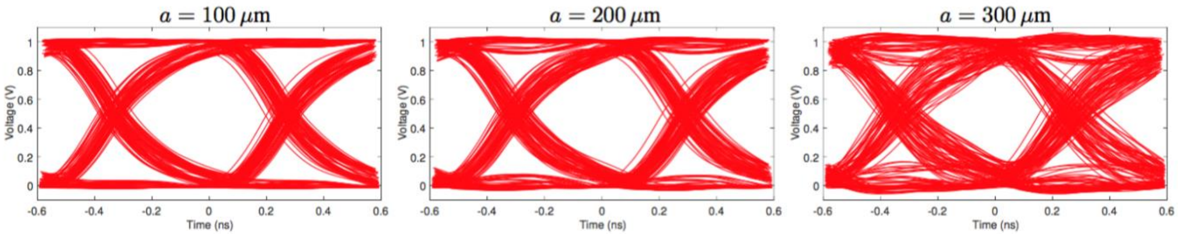
Multiboard PCB link parameter: via radius *a*



[Courtesy: J. B. Preibisch and C. Schuster, Technische Universität Hamburg-Harburg, Hamburg, Germany]



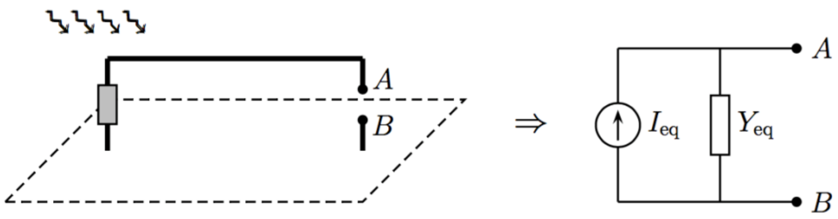


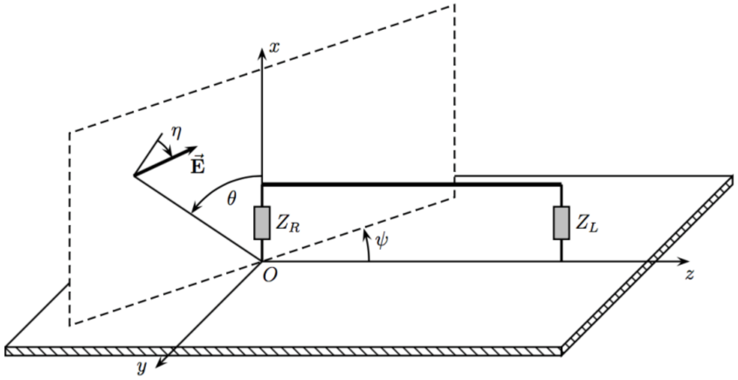


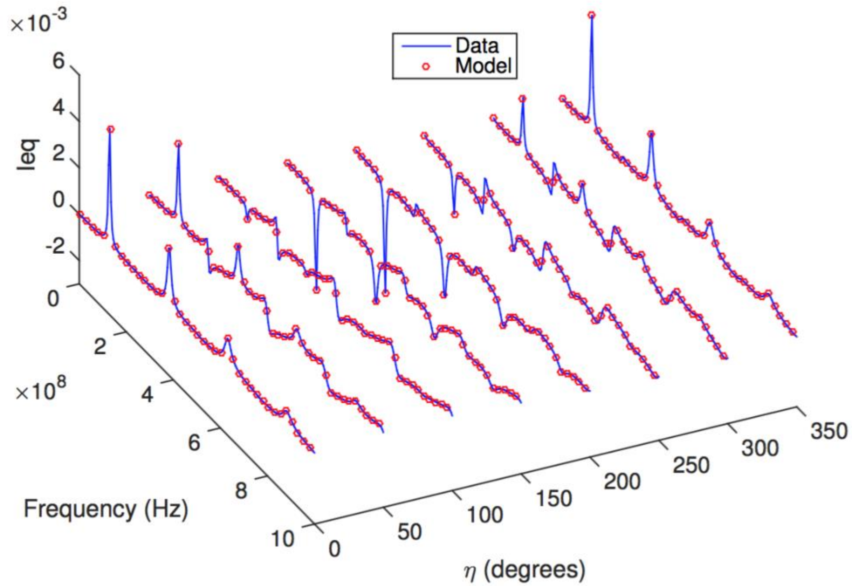
SPICE transient simulations with NL terminations

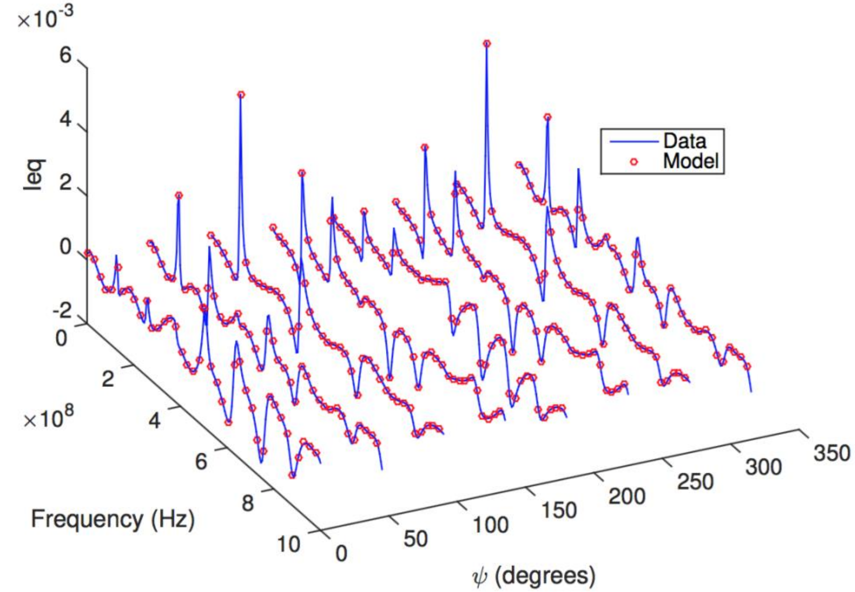
Example 6

Field-excited transmission line Data from full-wave solver



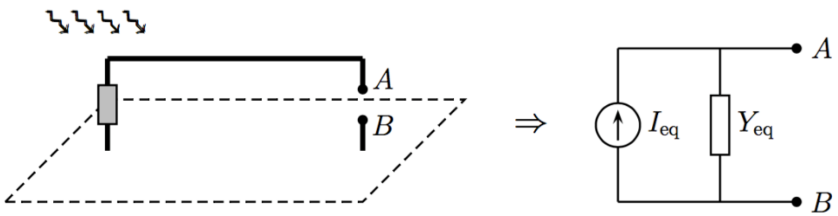


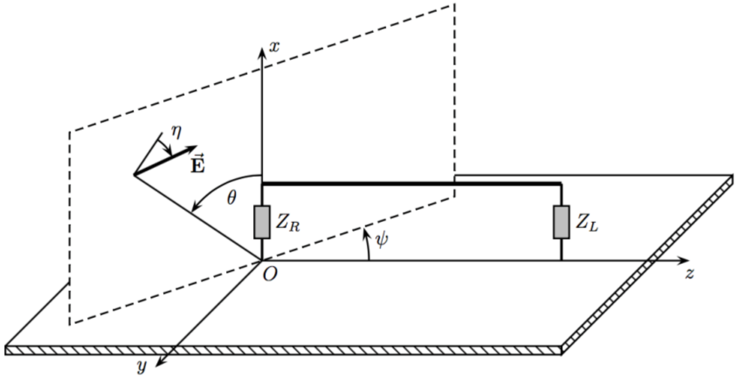




Example 6

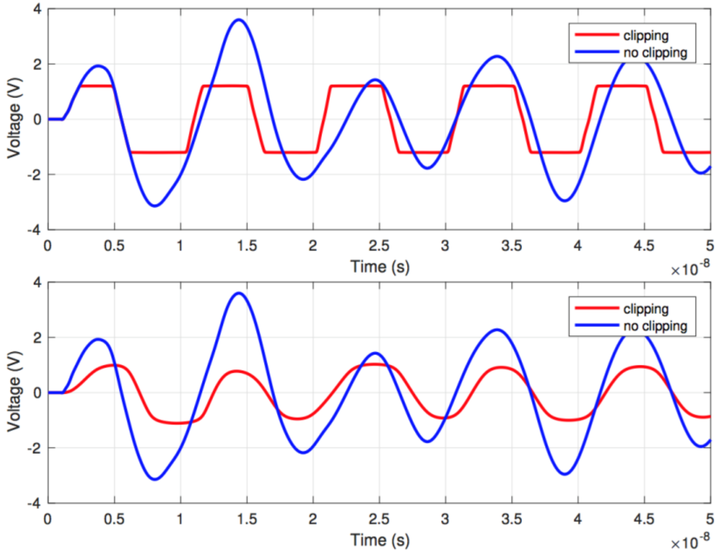
Field-excited transmission line Data from full-wave solver



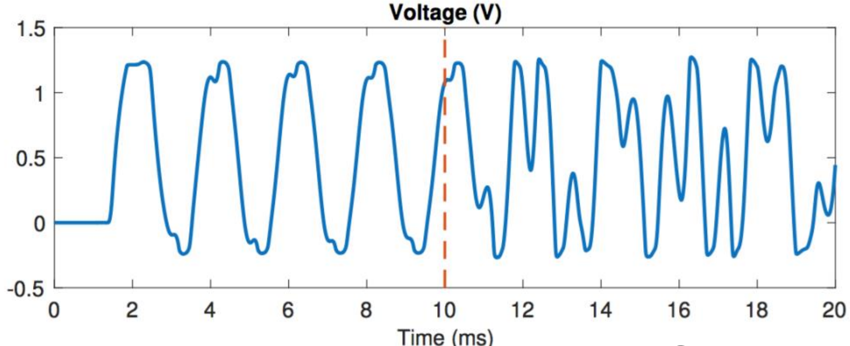


CW field with clipping diode

SPICE transient simulations



Clock signal + EMI field



EMI OFF EMI ON

Conclusions and Further Improvements

- Total number of test cases **25**
  - Maximum relative error : less than 1e-2
  - All models simulations are stable
- Speed-up (transient simulation) between model and transistor level $\sim 10-100$ X
- Intellectual property guaranteed by model construction
- Further Improvements
  - Extension to all SPICE languages (LTSpice)

Thank You All