Vector -Fitting Algorithm

(Frequency-Domain)

Dr. Behzad Nouri

2017

Carleton
UNIVERSITY

Department of Electronics

Carleton UNIVERSITY

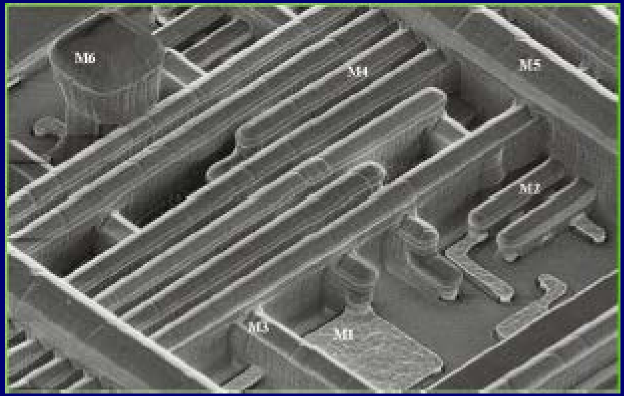
Problem Definition

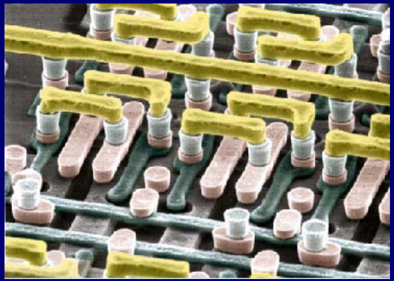
- At high frequencies some complex electrical devices may have no analytical models.

Example:

- 3D transmission lines
- vias, packages
- nonuniform transmission lines
- on-chip passive devices

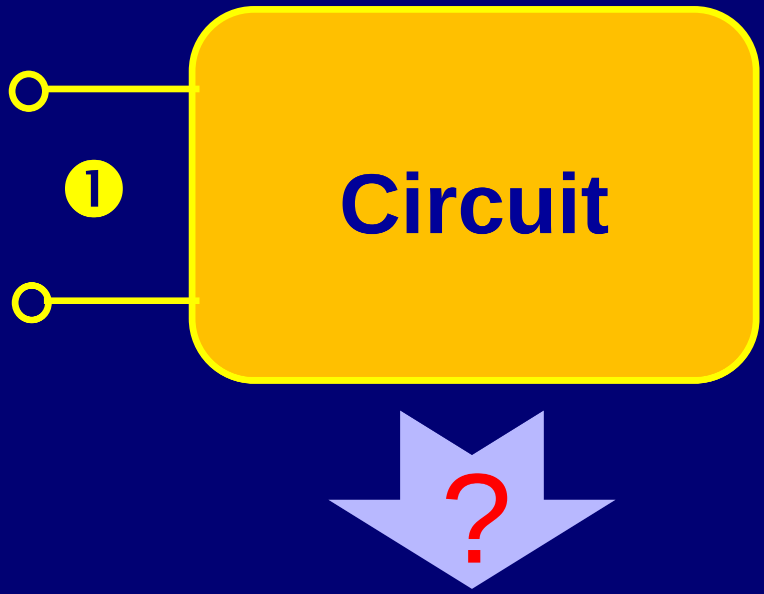
On-chip **interconnect** structures,
showing six separate metal layers (Al or Cu).





The dielectric material buffering the six-conductor layers has been (chemically) etched away.

Goal?



Realization of circuit

- Transfer Function (Y, Z, S, etc.)
- State-Space Formulation
- Circuit Realization (R,L,C, controlled-Sources, etc.)

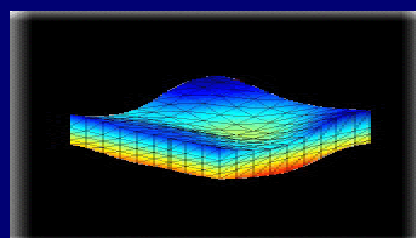
Data Acquisition

Devices/Subdesigns are characterized by tabulated data!

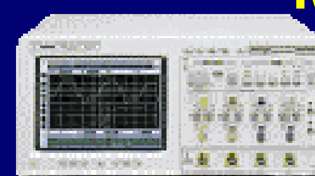
➊

Black Box

EM Simulation



Measurements



Simulated / measured
Tabulated data

Vector Fitting: Problem Definition

Given tabulated-data:

Freq.	Data
s_1	$f(s_1)$
s_2	$f(s_2)$
…	…
s_N	$f(s_N)$

Vector Fitting

Rational Transfer Function: F(s)=?

Circuit Elements

Or

State Space (ODE)

Spice Circuit Simulator

Common approach

$$f(s) = \frac{a_n s^n + a_{n-1} s^{n-1} + \ldots a_0}{b_n s^n + b_{n-1} s^{n-1} + \ldots 1}$$

$$\Downarrow$$

$$\begin{bmatrix} s_1^n & s_1^{n-1} & \ldots 1 & -f(s_1)s_1^n & -f(s_1)s_1^{n-1} & \ldots -f(s_1)s_1 \\ s_2^n & s_2^{n-1} & \ldots 1 & -f(s_2)s_2^n & -f(s_2)s_2^{n-1} & \ldots -f(s_2)s_2 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ s_i^n & s_i^{n-1} & \ldots 1 & -f(s_i)s_i^n & -f(s_i)s_i^{n-1} & \ldots -f(s_i)s_i \end{bmatrix} \begin{bmatrix} a_n \\ a_{n-1} \\ . \\ a_0 \\ b_n \\ b_{n-1} \\ . \\ b_1 \end{bmatrix} = \begin{bmatrix} f(s_1) \\ f(s_2) \\ . \\ . \\ f(s_i) \end{bmatrix}$$

Common approach

Such approaches which are based on a direct and primitive formulation can not handle the modern applications.

- Not able to handel the wide frequency band s (Practically ragning from DC to a few GHz)
- Easily becomes ill-conditioned
- Can not achieve Higher-Order Approximations

Equivalents Forms for Trans. Func.

(1) $$F(s)=\frac{a_{n-1}s^{n-1}+a_{n-2}s^{n-2}+\cdots+a_0}{b_n s^n+b_{n-1}s^{n-1}+\cdots+1}+d$$

Transfer Function (Proper form)

(2) $$F(s)=\frac{(\mathrm{s}-z_1)(\mathrm{s}-z_2)(\mathrm{s}-z_2^*)\times\cdots}{(\mathrm{s}-a_1)(\mathrm{s}-a_2)(\mathrm{s}-a_2^*)\times\cdots}=\frac{\prod_{i=1}^{n}(\mathrm{s}-z_i)}{\prod_{k=1}^{n}(\mathrm{s}-a_k)}$$

Poles / Zeros

(3) $$\boxed{F(s)=\frac{c_1}{\mathrm{s}-a_1}+\frac{c_2}{\mathrm{s}-a_2}+\frac{c_2^*}{\mathrm{s}-a_2^*}+\cdots+d}$$

Partial Fraction / Poles / Residue

New form for the Basis

real Residue

$$F(s)=\frac{c_1}{s-a_1}+\underbrace{\frac{c_2}{s-a_2}+\frac{c_2^*}{s-a_2^*}}_{Complex\text{ - }Conjugate\ pols}+\cdots+d$$

real Pole

$$\underbrace{\frac{c_2}{s-a_2}+\frac{c_2^*}{s-a_2^*}}_{Complex\text{ - }Conjugate\ pols}=\frac{c_2}{s-a_2}+\frac{c_2^*}{s-a_2^*}=\frac{\hat{c}_2+\hat{c}_3 j}{s-a_2}+\frac{\hat{c}_2-\hat{c}_3 j}{s-a_2^*}=$$

$$\frac{\hat{c}_2}{s-a_2}+\frac{\hat{c}_2}{s-\bar{a}_2}+\frac{\hat{c}_3 j}{s-a_2}-\frac{\hat{c}_3 j}{s-\bar{a}_2}=$$

$$\boxed{\hat{c}_2\left(\frac{1}{s-a_2}+\frac{1}{s-\bar{a}_2}\right)+\hat{c}_3\left(\frac{j}{s-a_2}-\frac{j}{s-\bar{a}_2}\right)}$$

New form for the Basis

Thereby, all the coefficients are real:

$$\hat{c}_1, \ \hat{c}_2, \ \hat{c}_3, \ \cdots, \ d \quad \in \mathbb{R}$$

$$F(s) = \underbrace{\hat{c}_1\left(\frac{1}{s-a_1}\right)}_{\text{Real Pole}} + \underbrace{\hat{c}_2\left(\frac{1}{s-a_2} + \frac{1}{s-a_2^*}\right) + \hat{c}_3\left(\frac{j}{s-a_2} - \frac{j}{s-a_2^*}\right)}_{\text{For the Pairs of Complex-Conjugate Poles}} + \cdots + d$$

Real Pole

For the Pairs of

Complex-Conjugate Poles

Select a Scaling Function

Data

$$F(s) = \delta(s)\ f(s)$$

Scaling Function

$$\sum_{i=1}^{n} \frac{\hat{c}_i}{s - a_i} + d \ = \delta(s)\ f(s)$$

Same Poles

$$\delta(s) = \sum_{i=1}^{n} \frac{\tilde{c}_i}{s - a_i} + 1$$

Select a Scaling Function

$$\sum_{i=1}^{n} \frac{\hat{c}_i}{s-a_i} + d \;=\; \delta(s)\; f(s)$$

Data

$$\sum_{i=1}^{n} \frac{\hat{c}_i}{s-a_i} + d = \left(\sum_{i=1}^{n} \frac{\tilde{c}_i}{s-a_i} + 1 \right) f(s)$$

Formulation: Real-Poles

We had: $$\sum_{i=1}^{n}\frac{\hat{c}_i}{s-a_i}+d=\left(\sum_{i=1}^{n}\frac{\tilde{c}_i}{s-a_i}+1\right)f(s)$$

$$\sum_{i=1}^{n}\frac{\hat{c}_i}{s-a_i}+d=\sum_{i=1}^{n}\frac{\tilde{c}_i}{s-a_i}f(s)+f(s)$$

$$\sum_{i=1}^{n}\frac{\hat{c}_i}{s-a_i}+d-\sum_{i=1}^{n}\frac{\tilde{c}_i}{s-a_i}f(s)=f(s)$$

$$\left(\frac{\hat{c}_1}{s-a_1}+\cdots+\frac{\hat{c}_n}{s-a_n}+d\right)-\left(\frac{f(s)\,\tilde{c}_i}{s-a_i}+\cdots+\frac{f(s)\,\tilde{c}_n}{s-a_n}\right)=f(s)$$

VF-Formulation: Real Poles

$$\frac{\hat{c}_1}{s_k - a_1} + \cdots + \frac{\hat{c}_n}{s_k - a_n} + d - \frac{f(s_k)\,\tilde{c}_1}{s_k - a_1} - \cdots - \frac{f(s_k)\,\tilde{c}_n}{s_k - a_n} = f(s_k)$$

$$\left\{\begin{array}{l}
\dfrac{\hat{c}_1}{s_1 - a_1} + \cdots + \dfrac{\hat{c}_n}{s_1 - a_n} + d - \dfrac{f(s_1)\,\tilde{c}_1}{s_1 - a_1} - \cdots - \dfrac{f(s_1)\,\tilde{c}_n}{s_1 - a_n} = f(s_1) \\[2ex]
\dfrac{\hat{c}_1}{s_2 - a_1} + \cdots + \dfrac{\hat{c}_n}{s_2 - a_n} + d - \dfrac{f(s_2)\,\tilde{c}_1}{s_2 - a_1} - \cdots - \dfrac{f(s_2)\,\tilde{c}_n}{s_2 - a_n} = f(s_2) \\[2ex]
\qquad\qquad\qquad\bullet \quad \bullet \quad \bullet \\[2ex]
\dfrac{\hat{c}_1}{s_2 - a_1} + \cdots + \dfrac{\hat{c}_n}{s_N - a_n} + d - \dfrac{f(s_N)\,\tilde{c}_1}{s_N - a_1} - \cdots - \dfrac{f(s_N)\,\tilde{c}_n}{s_N - a_n} = f(s_N)
\end{array}\right.$$

VF-Formulation: Real Poles

$$\begin{bmatrix} \frac{1}{s_1 - a_1} & \cdots & \frac{1}{s_1 - a_n} & 1 & \frac{-f(s_1)}{s_1 - a_1} & \cdots & \frac{-f(s_1)}{s_1 - a_n} \\ \frac{1}{s_2 - a_1} & \cdots & \frac{1}{s_2 - a_n} & 1 & \frac{-f(s_2)}{s_2 - a_1} & \cdots & \frac{-f(s_2)}{s_2 - a_n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1}{s_N - a_1} & \cdots & \frac{1}{s_N - a_n} & 1 & \frac{-f(s_N)}{s_N - a_1} & \cdots & \frac{-f(s_N)}{s_N - a_n} \end{bmatrix} \begin{bmatrix} \hat{c}_1 \\ \vdots \\ \hat{c}_n \\ d \\ \tilde{c}_1 \\ \vdots \\ \tilde{c}_n \end{bmatrix} = \begin{bmatrix} f(s_1) \\ f(s_2) \\ \vdots \\ f(s_N) \end{bmatrix}$$

- Equations are shown for real poles!
- They can be easily adapted to include complex poles too! (How?)

Vector Fitting Algorithm

Step❶: Use an initial guess of poles $\bar{a}_n$ And form the following matrix equation!

$$\underbrace{\begin{bmatrix} \frac{1}{s_1-\bar{a}_1} & \cdots & \frac{1}{s_1-\bar{a}_n} & 1 & \frac{-f(s_1)}{s_1-\bar{a}_1} & \cdots & \frac{-f(s_1)}{s_1-\bar{a}_n} \\ \frac{1}{s_2-\bar{a}_1} & \cdots & \frac{1}{s_2-\bar{a}_n} & 1 & \frac{-f(s_2)}{s_2-\bar{a}_1} & \cdots & \frac{-f(s_2)}{s_2-\bar{a}_n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1}{s_N-\bar{a}_1} & \cdots & \frac{1}{s_N-\bar{a}_n} & 1 & \frac{-f(s_N)}{s_N-\bar{a}_1} & \cdots & \frac{-f(s_N)}{s_N-\bar{a}_n} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \hat{c}_1 \\ \vdots \\ \hat{c}_n \\ d \\ \tilde{c}_1 \\ \vdots \\ \tilde{c}_n \end{bmatrix}}_{x} \underbrace{=}_{=} \underbrace{\begin{bmatrix} f(s_1) \\ f(s_2) \\ \vdots \\ f(s_N) \end{bmatrix}}_{B}$$

Vector Fitting Algorithm

Form: $$\underbrace{\begin{bmatrix} \Re eal(\mathbf{A}) \\ \Im mg(\mathbf{A}) \end{bmatrix}}_{\widetilde{\mathbf{A}}} \mathbf{x} = \underbrace{\begin{bmatrix} \Re eal(\mathbf{B}) \\ \Im mg(\mathbf{B}) \end{bmatrix}}_{\widetilde{\mathbf{B}}}$$

This is an over-determined linear problem!

The solution vector X includes the residues for both f(s) and δ(s)!

$$\mathbf{x} = \begin{bmatrix} \hat{c}_1 & \cdots & \hat{c}_n & d & \underbrace{\tilde{c}_1 & \cdots & \tilde{c}_n} \end{bmatrix}^T$$

We found the residues of δ(s)!

Step❷: **First least square solution to Find X**

Vector Fitting Algorithm

Solution vector: $$\mathbf{x} = \begin{bmatrix} \hat{c}_1 & \cdots & \hat{c}_n & d & \underbrace{\tilde{c}_1 & \cdots & \tilde{c}_n} \end{bmatrix}^T$$

We found the residues of δ(s)!

(For the proof see Apendix-1)

$$\delta(s) = \frac{\prod_{i=1}^{n} \left(s - \hat{z}_i \right)}{\prod_{i=1}^{n} \left(s - a_i \right)}$$

Zeros of the δ(s) are the (new) improved poles!

Step❸: **Compute the zeros of scaling function!**

Vector Fitting Algorithm

We use these (new) improved poles to start the next iteration!

Step❹: Repeat the steps ❷ and ❸ until poles converge!

$$\text{Error in Poles} = \max_{i=1}^{n}\left(\left|a_i^{(m)} - a_i^{(m-1)}\right|\right) < \text{Threshold}$$

Final Poles Obtained: ➡ $\underline{a}_i$

Vector Fitting Algorithm

Step ❺: For the final Poles run the second round of least square to find residues!

$$\begin{bmatrix} \dfrac{1}{s_1 - \underline{a}_1} & \cdots & \dfrac{1}{s_1 - \underline{a}_n} & 1 \\ \dfrac{1}{s_2 - \underline{a}_1} & \cdots & \dfrac{1}{s_2 - \underline{a}_n} & 1 \\ \vdots & \cdots & \vdots & \vdots \\ \dfrac{1}{s_N - \underline{a}_1} & \cdots & \dfrac{1}{s_N - \underline{a}_n} & 1 \end{bmatrix} \begin{bmatrix} \hat{c}_1 \\ \vdots \\ \hat{c}_n \\ d \end{bmatrix} = \begin{bmatrix} f(s_1) \\ f(s_2) \\ \vdots \\ f(s_N) \end{bmatrix}$$

The results are Poles and Residues of the f(s)!

Computation of Zeros of δ(s): Real case

For scaling function δ(s), given

$$\text{Poles:} \quad \{a_1, \quad a_2, \quad \dots \quad a_n\}$$

$$\text{Residues:} \quad \{\tilde{c}_1, \quad \tilde{c}_2, \quad \dots \quad \tilde{c}_n\}$$

?

$$\text{Zeros:} \quad \{\tilde{z}_1 = ?, \quad \tilde{z}_2 = ?, \quad \dots \quad \tilde{z}_n = ?\}$$

$$\delta(s) = \sum_{i=1}^{n} \frac{\tilde{c}_i}{s - a_i} + 1$$

?

$$\delta(s) = \frac{\prod_{i=1}^{n} (s - \tilde{z}_i)}{\prod_{i=1}^{n} (s - a_i)}$$

(See Apendix-1)

Computation of Zeros of δ(s): Real Case

Having: Poles: $a_1, a_2, \ldots, a_n$ Residues : $\tilde{c}_1, \tilde{c}_2, \ldots, \tilde{c}_n, 1$

$$\mathbf{A} = \begin{bmatrix} a_1 & & & \\ & a_2 & & \\ & & \ddots & \\ & & & a_n \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

$$\mathbf{C}^T = \begin{bmatrix} \tilde{c}_1 & \tilde{c}_2 & \cdots & \tilde{c}_n \end{bmatrix}$$

$$\boxed{\text{zeros of Transfer Func:} \quad \mathrm{z} = eig(\mathrm{A} - \mathrm{BC}^T)}$$

(Using eigenvalue decomposition “eig()” function in Matlab)

Computation of Zeros of δ(s): Complex Case

$$\delta(s) = \cdots + \tilde{c}_i \left(\frac{1}{s - a_i} + \frac{1}{s - a_i^*} \right) + \tilde{c}_{i+1} \left(\frac{j}{s - a_i} - \frac{j}{s - a_i^*} \right) + \cdots + 1$$

$$a_i = \alpha_i + j\beta_i \qquad a_i^* = \alpha_i - j\beta_i$$

Corresponding to the complex and its conjugate:

$$A_i = \begin{bmatrix} \alpha_i & \beta_i \\ -\beta_i & \alpha_i \end{bmatrix} \qquad d_i = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \qquad c_i^T = \begin{bmatrix} \tilde{c}_i & \tilde{c}_{i+1} \end{bmatrix}$$

State space Realization

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y} = \mathbf{C}^T\mathbf{x} + \mathbf{D}\mathbf{u}(t) \end{cases}$$

State-Space equation
(ODE)

$$\mathbf{A} = [?], \quad \mathbf{B} = [?], \quad \mathbf{C} = [?], \quad \mathbf{D} = [?]$$

State space Realization: Real Case

Having: Poles: $a_1, a_2, \ldots, a_n$ Residues: $\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_n,\ d$

Can Find: $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$

$$\mathbf{A} = \begin{bmatrix} a_1 & & & \\ & a_2 & & \\ & & \ddots & \\ & & & a_n \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

$$\mathbf{C}^T = \begin{bmatrix} \hat{c}_1 & \hat{c}_2 & \cdots & \hat{c}_n \end{bmatrix} \qquad \mathbf{D} = d$$

State space Realization: Complex Case

Let's have:

$$\text{Poles:}\quad \ldots, \underbrace{a_{i-1}}_{\text{Real}}, \underbrace{a_i}_{\text{Complex}}, \underbrace{a_i^*}_{\text{Conjugate}}, \underbrace{a_{i+2}}_{\text{Real}}, \ldots$$

$$\text{Residues}: \quad \ldots, \hat{c}_{i-1}, \hat{c}_i, \hat{c}_{i+1}, \hat{c}_{i+2}, \ldots, d$$

$$\mathbf{A} = \begin{bmatrix} \ddots & & & & \\ & a_{i-1} & & & \\ & & \begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix} & & \\ & & & a_{i-2} & \\ & & & & \ddots \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} \vdots \\ 1 \\ \begin{pmatrix} 2 \\ 0 \end{pmatrix} \\ 1 \\ \vdots \end{bmatrix}$$

$$\mathbf{C}^T = \begin{bmatrix} \cdots & \hat{c}_{i-1} & \begin{pmatrix} \hat{c}_i & \hat{c}_{i+1} \end{pmatrix} & \hat{c}_{i+2} & \cdots \end{bmatrix} \qquad \mathbf{D} = d$$

Example

Let the initial guess for the poles be

$$\bar{a}_1 \text{ is Real, } (\bar{a}_1 < 0)$$

$$\bar{a}_2 = \alpha + j\beta, \quad \text{and} \quad \bar{a}_2^* = \alpha - j\beta,$$

$$\text{where} \quad \alpha, \beta \in \text{Real}, \; (\alpha < 0)$$

(i) Show the format of transfer function F(x) we are intended to find.

(ii) Show required format for the scalar function $\delta(s)$ to be used.

(iii) Show the Vector Fitting equation.

(iv) Form the Vector Fitting Matrix equation.

(v) How to solve the equations to ensure real residues.

(vi) How to find zeros of the scaling function are obtained.

Example

- Show the format of transfer function F(x) we are intended to find.

Using the same initial poles, $\bar{a}_1$, $\bar{a}_2 = \alpha + j\beta$, $\bar{a}_2^* = \alpha - j\beta$,

$$F(s) = \hat{c}_1\left(\frac{1}{s-a_1}\right) + \hat{c}_2\left(\frac{1}{s-a_2} + \frac{1}{s-a_2^*}\right) + \hat{c}_3\left(\frac{j}{s-a_2} - \frac{j}{s-a_2^*}\right) + d$$

Example

- Show required format for the scalar function $\delta(s)$ to be used.

Using the same initial poles, $\bar{a}_1, \ \bar{a}_2 = \alpha + j\beta, \ \bar{a}_2^* = \alpha - j\beta,$

$$\delta(s) = \tilde{c}_1\left(\frac{1}{s-a_1}\right) + \tilde{c}_2\left(\frac{1}{s-a_2} + \frac{1}{s-a_2^*}\right) + \tilde{c}_3\left(\frac{j}{s-a_2} - \frac{j}{s-a_2^*}\right) + 1$$

Example

- Show the Vector Fitting equation at frequency s_i .

$$\bar{a}_1,\ \ \bar{a}_2 = \alpha + j\beta,\ \ \ \bar{a}_2^* = \alpha - j\beta,$$

$$F(s_i) = \delta(s_i) f(s_i)$$

$$\Downarrow$$

$$\hat{c}_1\left(\frac{1}{s_i - a_1}\right) + \hat{c}_2\left(\frac{1}{s_i - a_2} + \frac{1}{s_i - a_2^*}\right) + \hat{c}_3\left(\frac{j}{s_i - a_2} - \frac{j}{s_i - a_2^*}\right) + d -$$

$$\tilde{c}_1\left(\frac{f(s_i)}{s_i - a_1}\right) - \tilde{c}_2\left(\frac{f(s_i)}{s_i - a_2} + \frac{f(s_i)}{s_i - a_2^*}\right) - \tilde{c}_3\left(\frac{f(s_i)\,j}{s_i - a_2} - \frac{f(s_i)\,j}{s_i - a_2^*}\right) = f(s_i)$$

Example

- Form the Vector Fitting matrix equation at all given frequencies.

$$\bar{a}_1,\quad \bar{a}_2 = \alpha + j\beta,\quad \bar{a}_2^{*} = \alpha - j\beta,$$

$$\underbrace{\begin{bmatrix}
\frac{1}{s_1-\bar{a}_1} & \left(\frac{1}{s_1-\bar{a}_2}+\frac{1}{s_1-\bar{a}_2^{*}}\right) & \left(\frac{j}{s_1-\bar{a}_2}-\frac{j}{s_1-\bar{a}_2^{*}}\right) & 1 & \frac{-f(s_1)}{s_1-\bar{a}_1} & -\left(\frac{1}{s_1-\bar{a}_2}+\frac{1}{s_1-\bar{a}_2^{*}}\right)f(s_1) & -\left(\frac{j}{s_1-\bar{a}_2}-\frac{j}{s_1-\bar{a}_2^{*}}\right)f(s_1) \\
\frac{1}{s_2-\bar{a}_1} & \left(\frac{1}{s_2-\bar{a}_2}+\frac{1}{s_2-\bar{a}_2^{*}}\right) & \left(\frac{j}{s_2-\bar{a}_2}-\frac{j}{s_2-\bar{a}_2^{*}}\right) & 1 & \frac{-f(s_2)}{s_2-\bar{a}_1} & -\left(\frac{1}{s_2-\bar{a}_2}+\frac{1}{s_2-\bar{a}_2^{*}}\right)f(s_2) & -\left(\frac{j}{s_2-\bar{a}_2}-\frac{j}{s_2-\bar{a}_2^{*}}\right)f(s_2) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{1}{s_N-\bar{a}_1} & \left(\frac{1}{s_N-\bar{a}_2}+\frac{1}{s_N-\bar{a}_2^{*}}\right) & \left(\frac{j}{s_N-\bar{a}_2}-\frac{j}{s_N-\bar{a}_2^{*}}\right) & 1 & \frac{-f(s_N)}{s_N-\bar{a}_1} & -\left(\frac{1}{s_N-\bar{a}_2}+\frac{1}{s_N-\bar{a}_2^{*}}\right)f(s_1) & -\left(\frac{j}{s_N-\bar{a}_2}-\frac{j}{s_N-\bar{a}_2^{*}}\right)f(s_N)
\end{bmatrix}}_{A}
\underbrace{\begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \\ \hat{c}_3 \\ d \\ \tilde{c}_1 \\ \tilde{c}_2 \\ \tilde{c}_3 \end{bmatrix}}_{X}
=
\underbrace{\begin{bmatrix} f(s_1) \\ f(s_2) \\ \vdots \\ f(s_N) \end{bmatrix}}_{B}$$

Example

(i) How to solve the equations to ensure real residues.

$$Ax=B$$

$$\underbrace{\begin{bmatrix} \Re eal(\mathbf{A}) \\ \Im mg(\mathbf{A}) \end{bmatrix}}_{\widetilde{\mathbf{A}}} \mathbf{x} = \underbrace{\begin{bmatrix} \Re eal(\mathbf{B}) \\ \Im mg(\mathbf{B}) \end{bmatrix}}_{\widetilde{\mathbf{B}}}$$

$$\mathbf{x} = \widetilde{\mathbf{A}} \backslash \widetilde{\mathbf{B}}$$ (Matlab)

Example

- How to find zeros of the scaling function are obtained.

Having poles and residues of the scaling function as:

$$\bar{a}_1, \ \bar{a}_2 = \alpha + j\beta, \quad \bar{a}_2^* = \alpha - j\beta, \quad \text{and} \quad \tilde{c}_1, \tilde{c}_2, \tilde{c}_3$$

$$\mathbf{A} = \begin{bmatrix} \bar{a}_1 & 0 & 0 \\ 0 & \alpha & \beta \\ 0 & -\beta & \alpha \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \qquad \mathbf{D} = d$$

$$\mathbf{C}^T = [\tilde{c}_1 \quad \tilde{c}_2 \quad \tilde{c}_3]$$

$$\boxed{\text{zeros of scaling func:} \quad \mathrm{z} = eig(\mathrm{A} - \mathrm{BC}^T)}$$

Conversion of Macromodels to Equivalent Subcircuits:

Equivalent Subcircuits

Conversion of the a given state-space equation (ODE) representation of a subcircuit to its equivalent subcircuits can be accomplished in several ways.

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y} = \mathbf{C}^T\mathbf{x} + \mathbf{D}\mathbf{u}(t) \end{cases}$$

State-Space equation

(Ordinary Differential Equation)

Equivalent Subcircuits

As an illustrative example let's consider a simple two-port network which only has two states represented:

States

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$\begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

Port Currents

Port Voltages

$$\begin{aligned} \dot{x}_1 &= a_{11}x_1 + a_{12}x_2 + b_{11}v_1 + b_{12}v_2 \\ \dot{x}_2 &= a_{21}x_1 + a_{22}x_2 + b_{21}v_1 + b_{22}v_2 \\ i_1 &= c_{11}x_1 + c_{12}x_2 + d_{11}v_1 + d_{12}v_2 \\ i_2 &= c_{21}x_1 + c_{22}x_2 + d_{21}v_1 + d_{22}v_2. \end{aligned}$$

Equivalent Subcircuits

Each state in the macromodel requires a separate node in the equivalent circuit. Thus,

State-Eq.#1: $$-\dot{x}_1 + a_{11}x_1 + a_{12}x_2 + b_{11}v_1 + b_{12}v_2 = 0$$

$$\begin{cases} \text{Node\#1}: & x_1 = v_{n1} \\ \text{Node\#2}: & x_2 = v_{n2} \end{cases}$$

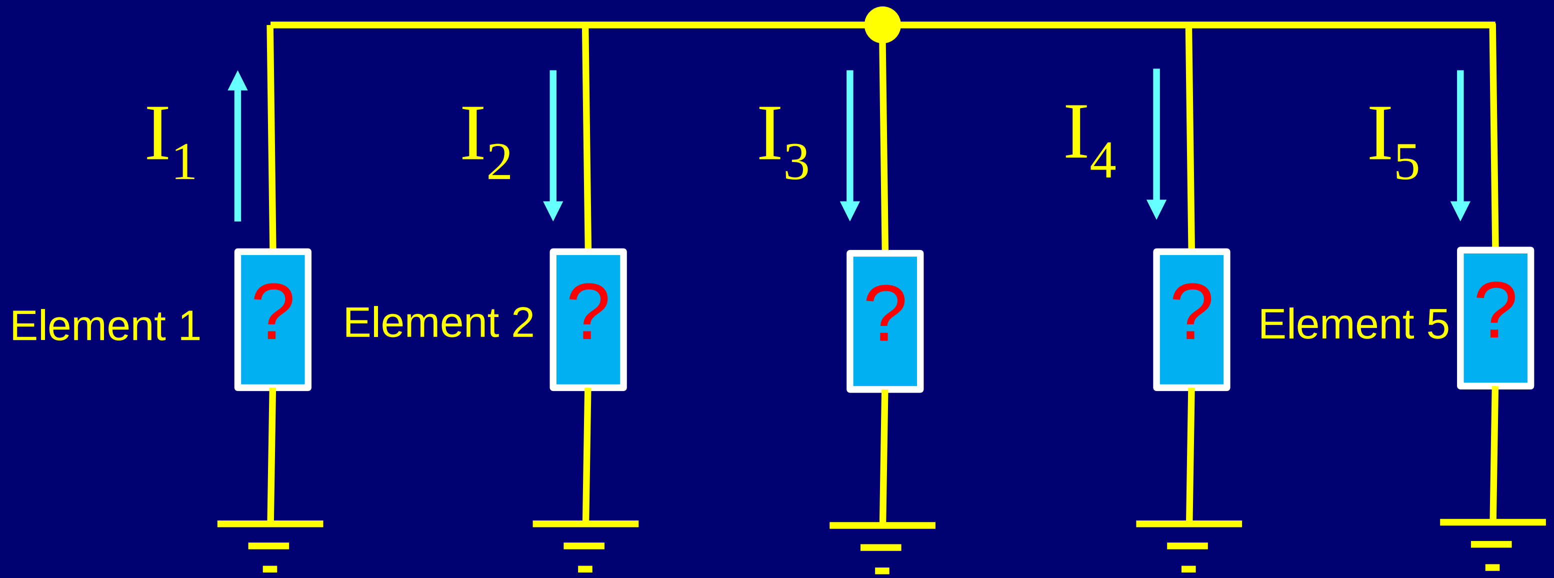
$$\Downarrow$$

Node#1: $$-\dot{v}_{n1} + a_{11}v_{n1} + a_{12}v_{n2} + b_{11}v_1 + b_{12}v_2 = 0$$

Equivalent Subcircuits

$$\textit{KCL @ Node} - 1: \quad \underbrace{-\dot{v}_{n1}}_{-I_1} + \underbrace{a_{11} v_{n1}}_{I_2} + \underbrace{a_{12} v_{n2}}_{I_3} + \underbrace{b_{11} v_1}_{I_4} + \underbrace{b_{12} v_2}_{I_5} = 0$$

$$x_1 = v_{n1}$$



Equivalent Subcircuits

KCL @ Node – 1 : $$\underbrace{-\dot{v}_{n1}}_{-I_1} + \underbrace{a_{11} v_{n1}}_{I_2} + \underbrace{a_{12} v_{n2}}_{I_3} + \underbrace{b_{11} v_1}_{I_4} + \underbrace{b_{12} v_2}_{I_5} = 0$$

Element#1: $$\begin{cases} I_1 = \dot{v}_{n1} \\ I_c = C\dot{v} \end{cases} \Rightarrow C = 1, \ \text{Capasitor}$$

Element#2: $$\begin{cases} I_2 = a_{11} v_{n1} \\ I_R = \dfrac{1}{R} v_R \end{cases} \Rightarrow R = \frac{1}{a_{11}}, \quad \text{Resistor}$$

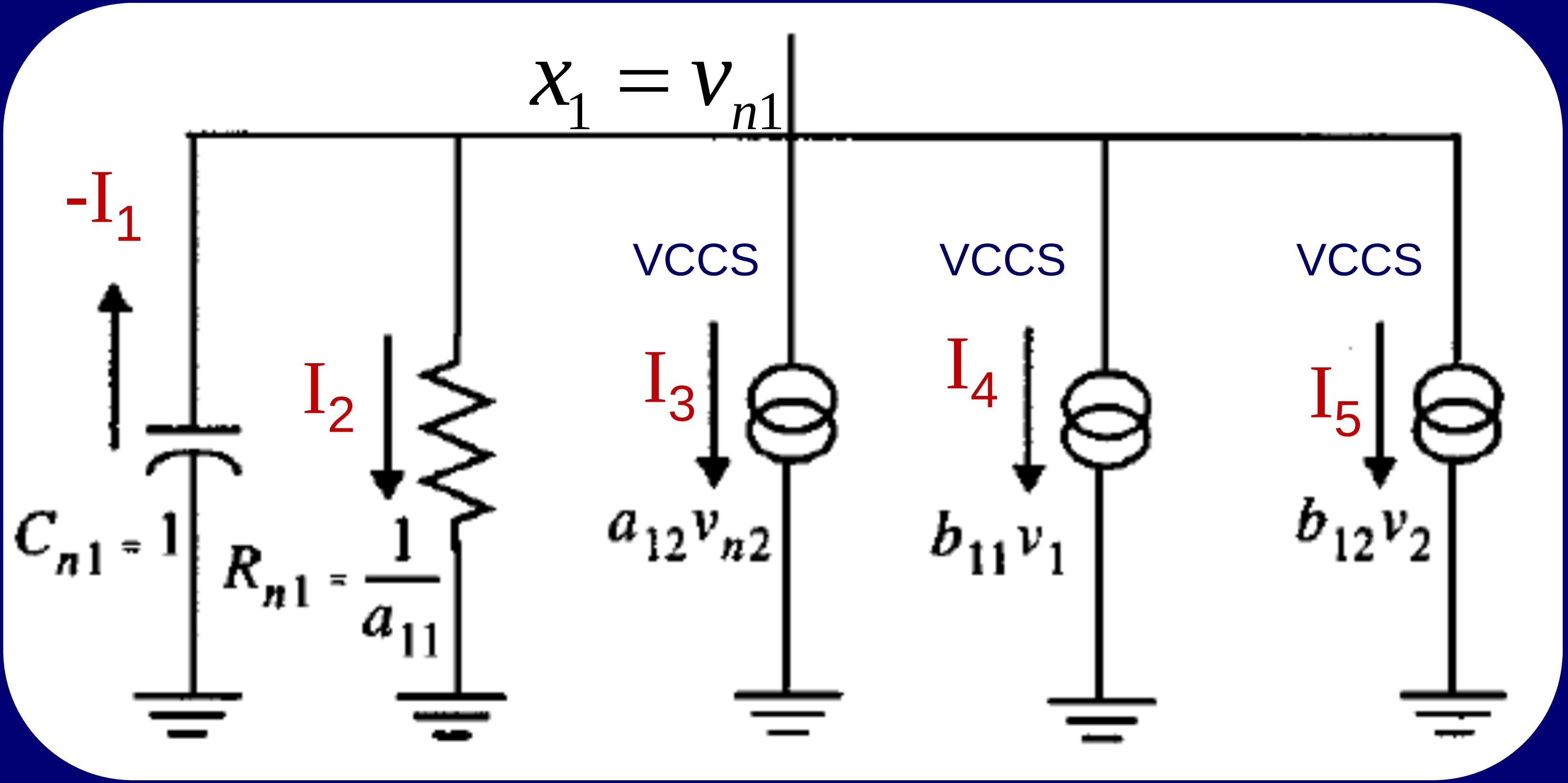
Element#3: $$\begin{cases} I_3 = a_{12} v_{n2} \\ I \ = k\, v_k \end{cases} \Rightarrow k = a_{12}, \quad \text{VCCS}$$

The difference between the case of Element#2 and Element#3 is that:

- for Element#2, V_{n1} is a voltage across the same branch. Hence, it is a resistor.
- for Element#3, V_{n2} is a voltage some where else in the circuit (not across the branch).

Equivalent Subcircuits

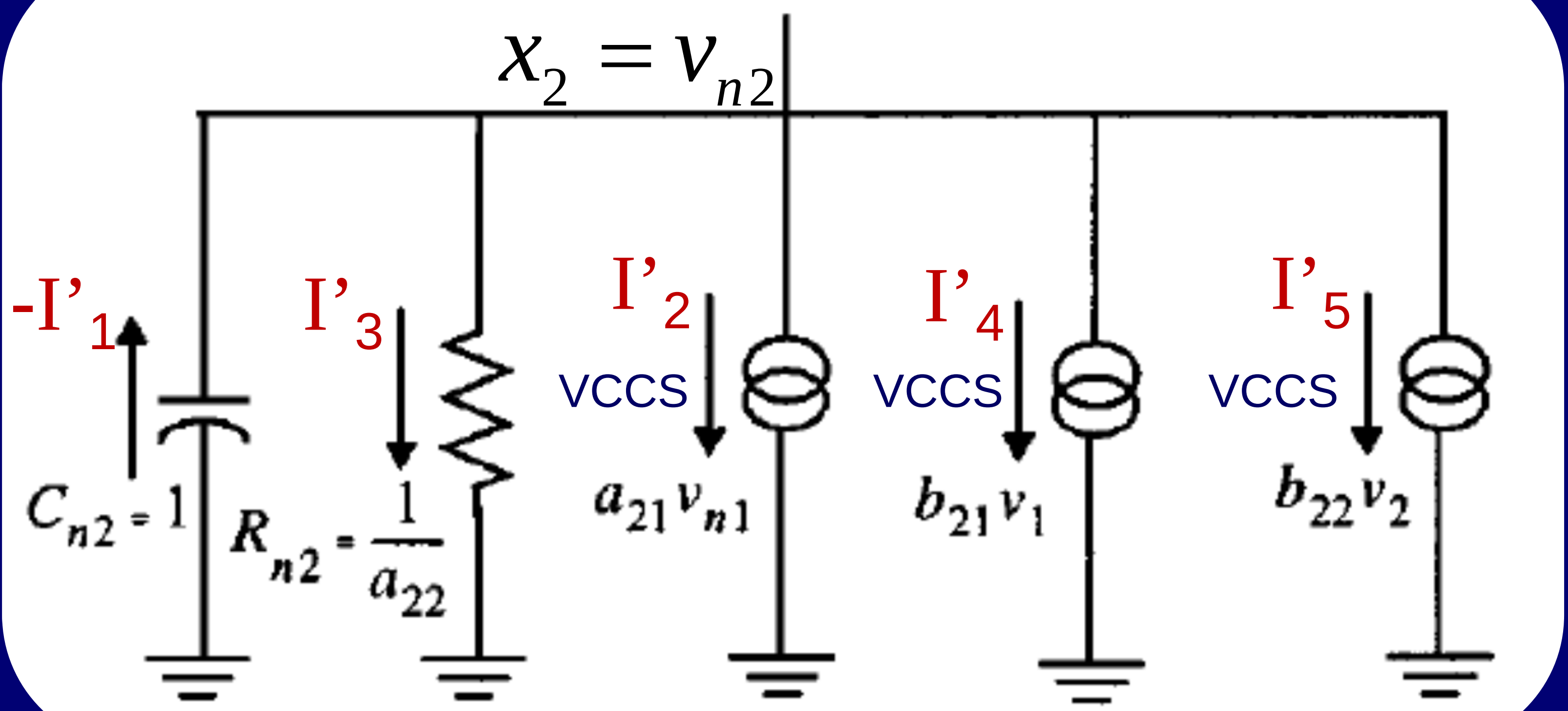
$$KCL\ @\ Node-1: \quad \underbrace{-\dot{v}_{n1}}_{-I_1} + \underbrace{a_{11}v_{n1}}_{I_2} + \underbrace{a_{12}v_{n2}}_{I_3} + \underbrace{b_{11}v_1}_{I_4} + \underbrace{b_{12}v_2}_{I_5} = 0$$



Equivalent Subcircuits

Each state in the macromodel requires a separate node in the equivalent circuit. Thus, $x_1 = v_{n1},\ x_2 = v_{n2}$

KCL @ Node – 2 : $$\underbrace{-\dot{v}_{n2}}_{-I'_1} + \underbrace{a_{21} v_{n1}}_{I'_2} + \underbrace{a_{22} v_{n2}}_{I'_3} + \underbrace{b_{21} v_1}_{I'_4} + \underbrace{b_{22} v_2}_{I'_5} = 0$$

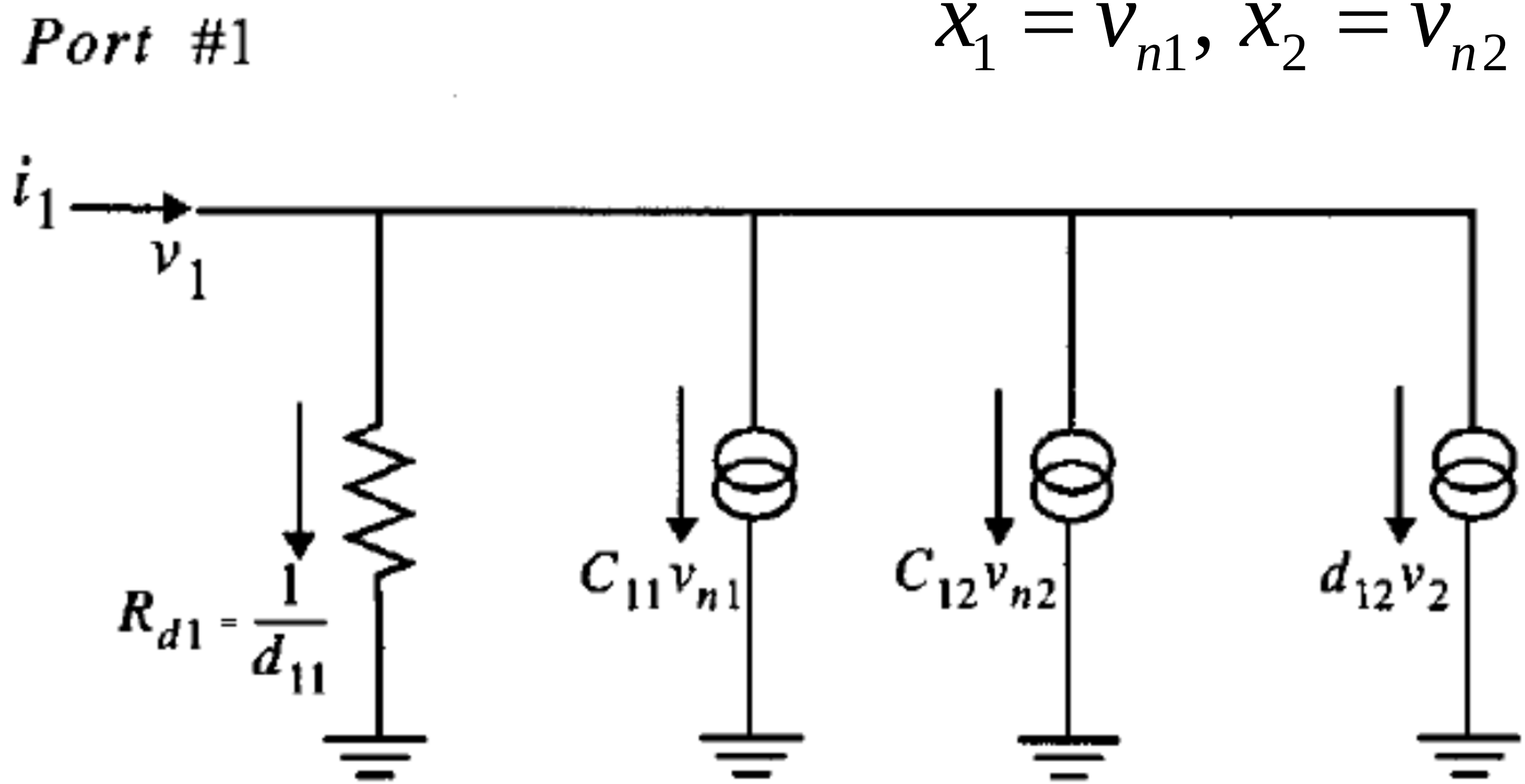


Equivalent Subcircuits

The first "output equation" is realized through equivalent circuits as:

$$i_1 = c_{11} v_{n1} + c_{12} v_{n2} + d_{11} v_1 + d_{12} v_2$$

$$x_1 = v_{n1},\ x_2 = v_{n2}$$

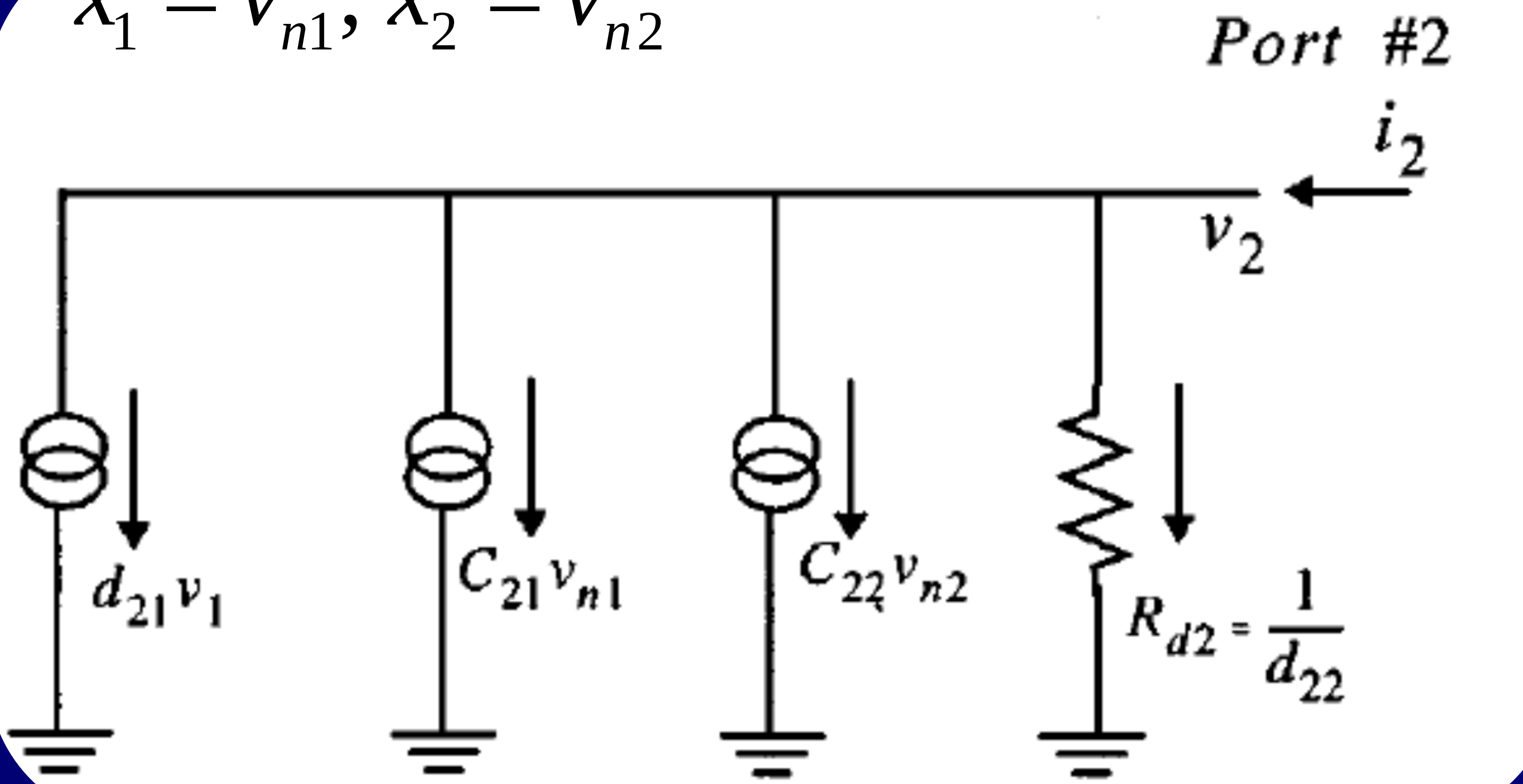


Equivalent Subcircuits

The first “output equation” is realized through equivalent circuits as:

$$i_2 = c_{21}v_{n1} + c_{22}v_{n2} + d_{21}v_1 + d_{22}v_2$$

$$x_1 = v_{n1},\ x_2 = v_{n2}$$



References

Reference:

B. Gustavsen and A. Semlyen, “Rational approximation of frequency domain responses by vector fitting,” IEEE Transactions on Power Delivery, vol. 14, no. 3, pp. 1052–1061, Jul. 1999

Vector Fitting Website: https://www.sintef.no/projectweb/vectfit/

Downloads: https://www.sintef.no/projectweb/vectfit/downloads/

Appendix-1: Forms of TF

$$f(s)=\frac{a_{n-1}s^{n-1}+\cdots+a_0}{b_n s^n+b_{n-1}s^{n-1}+\cdots+1}+d$$

Transfer Function
(Strictly proper form)

$$f(s)=\frac{(\mathrm{s}-z_1)(\mathrm{s}-z_2)(\mathrm{s}-z_2^*)\times\cdots}{(\mathrm{s}-a_1)(\mathrm{s}-a_2)(\mathrm{s}-a_2^*)\times\cdots}$$

Poles / Zeros
Model

Residue

$$f(s)=\frac{c_1}{\mathrm{s}-a_1}+\frac{c_2}{\mathrm{s}-a_2}+\frac{c_2^*}{\mathrm{s}-a_2^*}+\cdots+d$$

Poles

Partial Fraction

(Poles / Residue)

Appendix-1: New Basis …. (cnt'd)

real Residue

$$f(s)=\frac{c_1}{s-a_1}+\underbrace{\frac{c_2}{s-a_2}+\frac{c_2^*}{s-a_2^*}}_{\textit{Complex - Conjugate pols}}+\cdots+d$$

real Pole

$$\frac{c_2}{s-a_2}+\frac{c_2^*}{s-\bar{a}_2}=\frac{\alpha+\beta j}{s-a_2}+\frac{\alpha-\beta j}{s-\bar{a}_2}=$$

$$\frac{\alpha}{s-a_2}+\frac{\alpha}{s-\bar{a}_2}+\frac{\beta j}{s-a_2}-\frac{\beta j}{s-\bar{a}_2}=$$

$$\alpha\left(\frac{1}{s-a_2}+\frac{1}{s-\bar{a}_2}\right)+\beta\left(\frac{j}{s-a_2}-\frac{j}{s-\bar{a}_2}\right)$$

$$\boxed{f(s)=\hat{c}_1\left(\frac{1}{s-a_1}\right)+\hat{c}_2\left(\frac{1}{s-a_2}+\frac{1}{s-\bar{a}_2}\right)+\hat{c}_3\left(\frac{j}{s-a_2}-\frac{j}{s-\bar{a}_2}\right)+\cdots+d}$$

Apendix-2

Solution vector: $\mathbf{x} = \left[\hat{c}_1 \quad \cdots \quad \hat{c}_n \quad d \quad \underbrace{\tilde{c}_1 \quad \cdots \quad \tilde{c}_n}\right]^T$

We found the residues of δ(s)!

Zeros of the δ(s) are the improved poles of f(s)! (Why?)

Proof:

$$\sum_{i=1}^{n} \frac{\hat{c}_i}{s-a_i} + d = \delta(s) f(s) \quad \Rightarrow \quad \sum_{i=1}^{n} \frac{\hat{c}_i}{s-a_i} + d = \underbrace{\left(\sum_{i=1}^{n} \frac{\tilde{c}_i}{s-a_i} + 1 \right)}_{\delta(s)} f(s)$$

$$\frac{\prod_{i=1}^{n}(s-\hat{z}_i)}{\cancel{\prod_{i=1}^{n}(s-a_i)}} = \frac{\prod_{i=1}^{n}(s-\tilde{z}_i)}{\cancel{\prod_{i=1}^{n}(s-a_i)}} f(s) \quad \Rightarrow \quad \frac{\prod_{i=1}^{n}(s-\hat{z}_i)}{\prod_{i=1}^{n}(s-\tilde{z}_i)} = f(s)$$