

CSE 245: Computer Aided Circuit Simulation and Verification

Fall 2004, Oct 05

Lecture 3:

Dynamic Linear System II

Outline (last lecture)

□ Time Domain Analysis

- State Equations
- RLC Network Analysis by Taylor Expansion
- Impulse Response in time domain

□ Frequency Domain Analysis

- From time domain to Frequency domain
- Correspondence between time domain and frequency domain
- Serial expansion of $(s\mathbf{I}-\mathbf{A})^{-1}$

Outline (Cont')

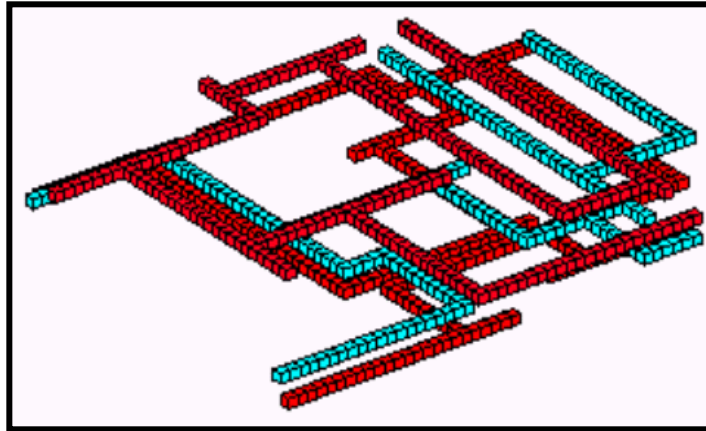
□ Model Order Reduction

- Moments
- Passivity, Stability and Realizability

□ Symbolic Analysis

- Y-Delta Transformation
- BDD Analysis

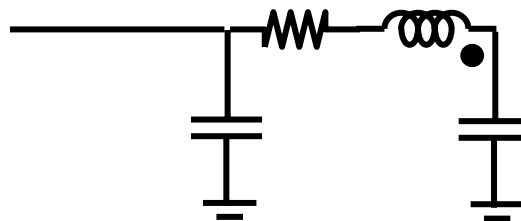
Parasitic Extraction



thousands of wires
e.g. critical path
e.g. gnd/vdd grid

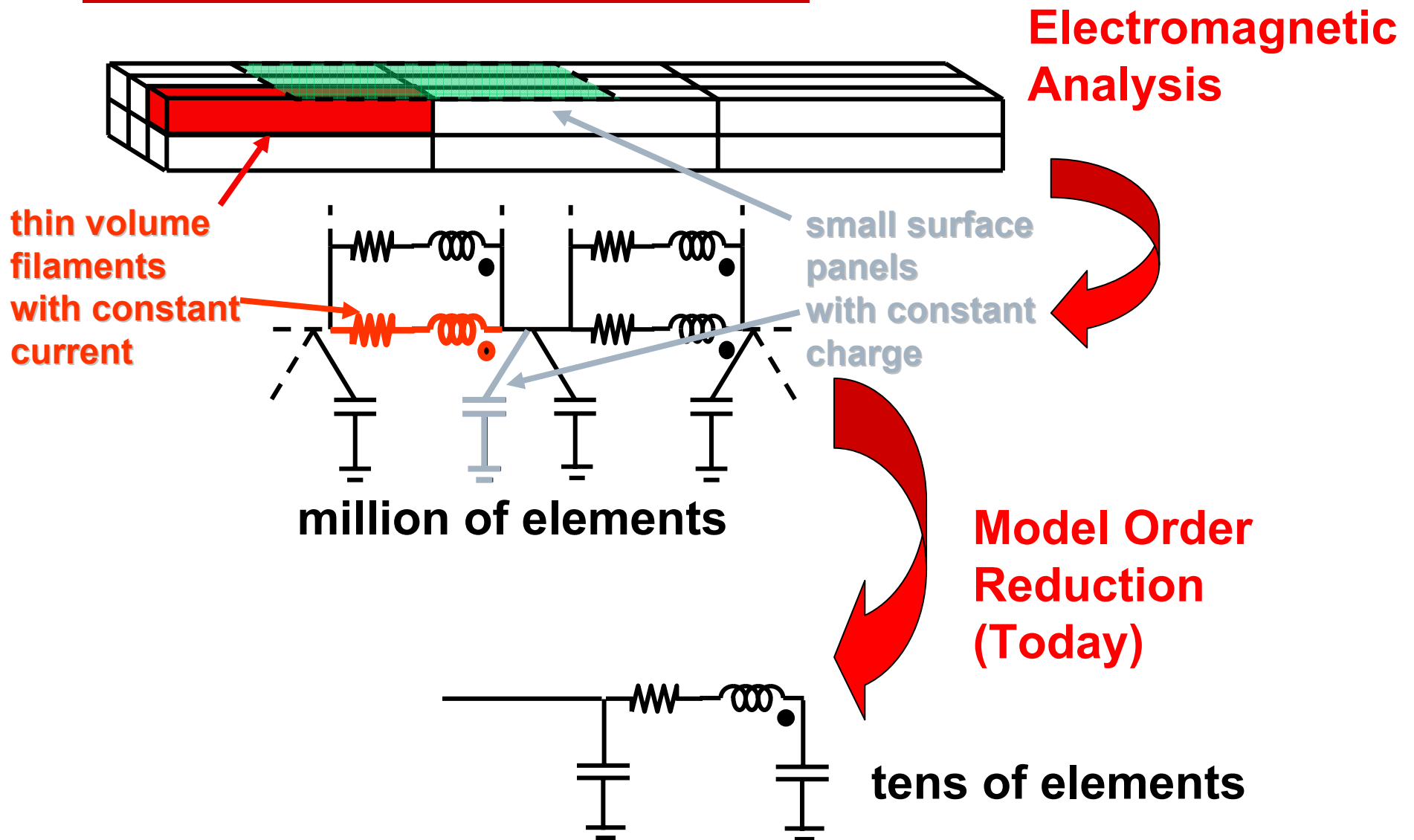
- identify some ports
- produce equivalent circuit that models response of wires at those ports

Parasitic Extraction

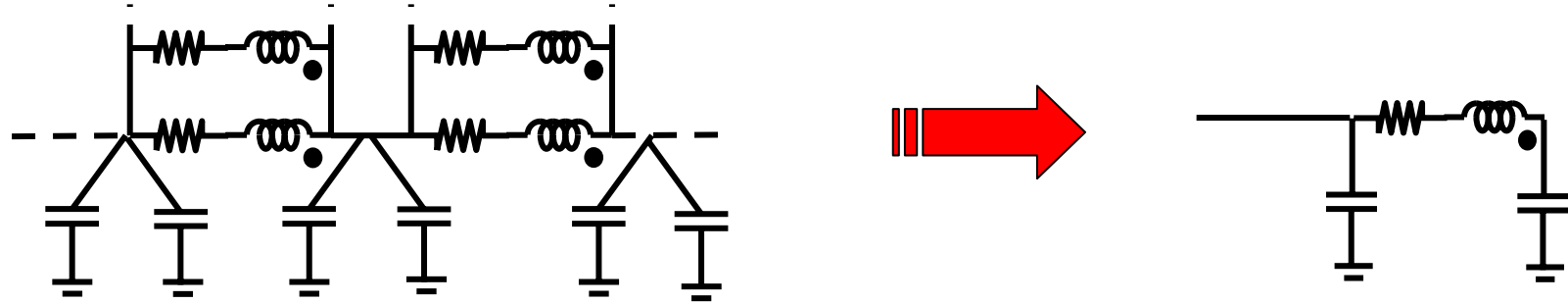


tens of circuit
elements for
spice simulation

Parasitic Extraction (the two steps)



Why build reduced models?



□ Compression for Efficiency

- It is possible to represent the system under study “precisely” with millions of elements
- But the simulation is too slow with the complicated representation

□ Abstraction

- I do not care at all about the precise representation
- In fact I would rather those details were not even there. I may not be able to create or manipulate the precise representation at all.

Challenges for reduction algorithms

□ Accuracy

- Must be controllable and predictable

□ Efficiency

- Algorithms should be scalable to handle large systems

□ Numerical robustness

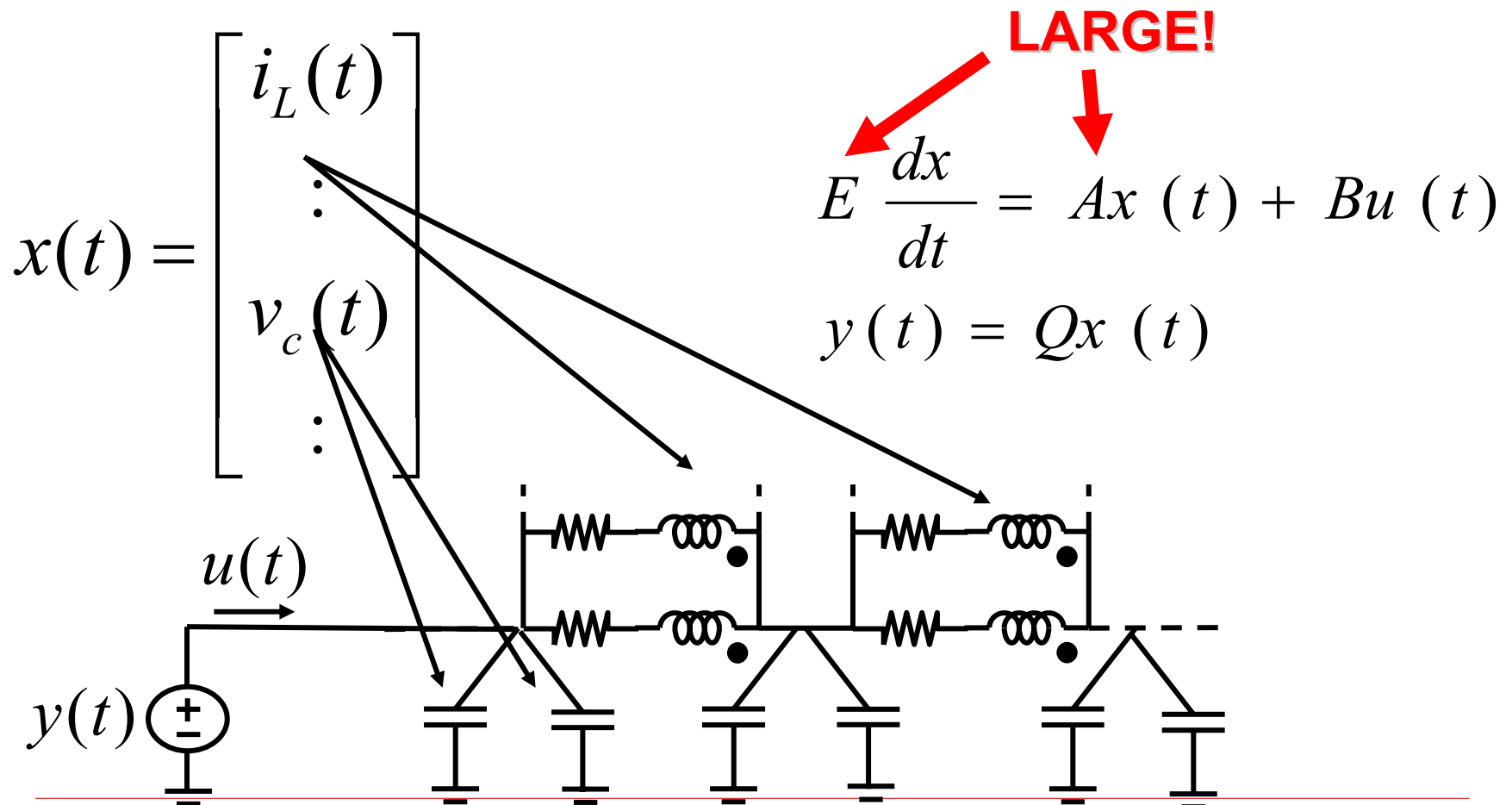
- Algorithms should work reliably for all reasonable inputs & accuracy requests

□ Models must work in simulation

- Composability : Combination of two good models is a good model

State Equation

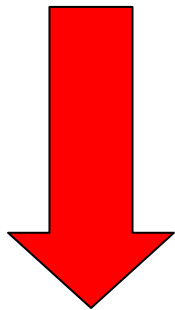
□ So in general....



A canonical form for model order reduction

$$E \frac{dx}{dt} = Ax(t) + Bu(t)$$

$$y(t) = Qx(t)$$



$$E \leftarrow -A^{-1}E$$

$$B \leftarrow -A^{-1}B$$

$$E \frac{dx}{dt} = x(t) + Bu(t)$$

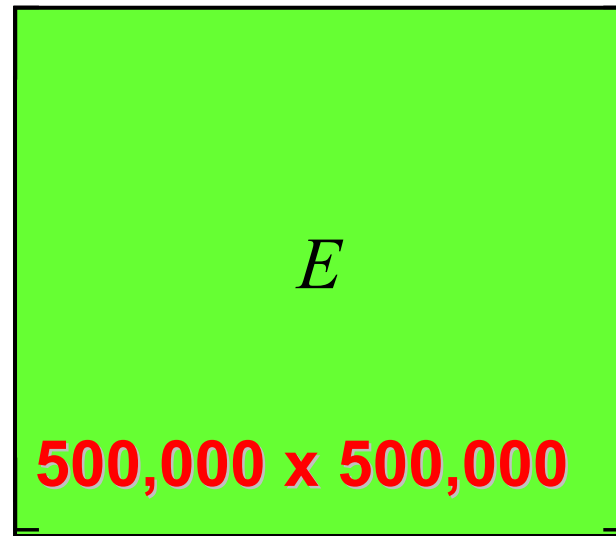
$$y(t) = Qx(t)$$

Assuming A is non-singular we can cast the dynamical linear system into one canonical form for model order reduction

Note: not necessarily always the best, but the simplest for educational purposes

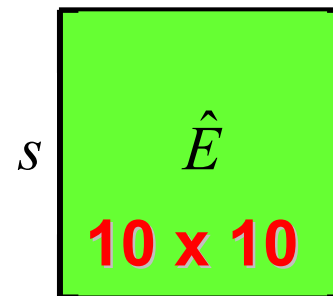
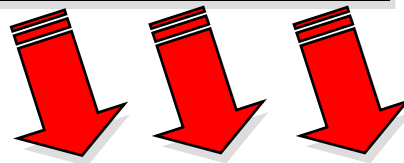
smaller yet accurate model

- Given a large linear system model:



$$\frac{d}{dt} \begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} x \end{bmatrix} + \begin{bmatrix} b \end{bmatrix} u$$

- Construct a linear system model with:
 - smaller complexity
 - same fidelity
 - small reduction cost



$$y = c^T x$$

$$\begin{bmatrix} \hat{x} \end{bmatrix} = \begin{bmatrix} x \end{bmatrix} + \begin{bmatrix} \hat{b} \end{bmatrix} u$$

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

Eigenvalue Based Reduction

Pole-Residue Form

$$H(s) = \sum_{i=1}^n \frac{R_i}{s - p_i}$$
$$h(t) = \sum_{i=1}^n R_i e^{p_i t}$$

Pole-Zero Form

$$H(s) = \frac{\prod_{i=1}^{n-1} (s - \zeta_i)}{\prod_{i=1}^n (s - p_i)}$$

□ Ideas for reducing order:

- Drop terms with small residues
- Drop terms with large $\text{Re } p$ (“fast” modes)
- Remove pole/zero near-cancellations
- Cluster poles that are “together”

□ How to compute poles and residues?

Computing Poles & Residues

Transfer Function: $H(s) = Q(sI - A)^{-1} B$

$$A = X\Lambda X^{-1} \quad \text{where} \quad \Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & \vdots \\ \vdots & \cdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \lambda_k \end{bmatrix} \quad X = [X_1, X_2, \dots, X_k]$$

$\lambda_1, \lambda_2, \dots, \lambda_k$ eigenvalues $X_1, X_2, \dots, X_k$ eigenvectors

$$\begin{aligned} H(s) &= Q(sXX^{-1} - X\Lambda X^{-1})B = \\ &= QX(sI - \Lambda)^{-1} X^{-1}B = \end{aligned}$$

$$= \sum_{i=1}^k \frac{(QX)_i (XB^{-1})_i}{s - \lambda_i} \quad \begin{array}{l} \longleftarrow R_i \text{ residues} \\ \longleftarrow p_i \text{ Poles, eigenvalue of } A \end{array}$$

Eigenvalue Based Reduction

□ Advantages

- Conceptually familiar
- Simple physical interpretation : retains dominant system modes/poles

□ Drawbacks

- Relatively expensive : have to find the poles first
- Relatively inefficient. For a given model size, many other approaches can provide better accuracy

□ Rule of thumb

- Anything that can be done by manipulating pole/eigenvalues/eigenvectors can probably be done better with more sophisticated analysis, at the same or smaller cost.

Moment Matching

Taylor Series Expansion of the original transfer function around $s=0$

$$H(s) = H(0) + \left. \frac{dH}{ds} \right|_{s=0} s + \frac{1}{2!} \left. \frac{d^2 H}{ds^2} \right|_{s=0} s^2 + \frac{1}{3!} \left. \frac{d^3 H}{ds^3} \right|_{s=0} s^3 + \dots$$

$$H(s) = Q(sI - A)^{-1} B$$

$$\begin{aligned} H(s) &= Q[(I - sA^{-1})(-A)]^{-1} B \\ &= -QA^{-1}(I - sA^{-1})^{-1} B = -QA^{-1}(I + sA^{-1} + s^2 A^{-2} + \dots)B \end{aligned}$$

$$\begin{aligned} H(s) &= -QA^{-1}B - QA^{-2}Bs - QA^{-3}Bs^2 - QA^{-4}Bs^3 - \dots \\ &= m_0 + m_1s + m_2s^2 + m_3s^3 + \dots \end{aligned}$$

**The Taylor coef. = frequency domain moments =
= derivatives of the transfer function (up to a constant)**

Moments of the circuit response

$$\begin{aligned} H(s) &= \int_0^{\infty} e^{-st} h(t) dt \\ &= \left[\int_0^{\infty} h(t) dt \right] - \left[\int_0^{\infty} t h(t) dt \right] s + \frac{1}{2!} \left[\int_0^{\infty} t^2 h(t) dt \right] s^2 + \cdots + \frac{(-1)^{2q-1}}{(2q-1)!} \left[\int_0^{\infty} t^{2q-1} h(t) dt \right] s^{2q-1} + O(s^{2q}) \end{aligned}$$

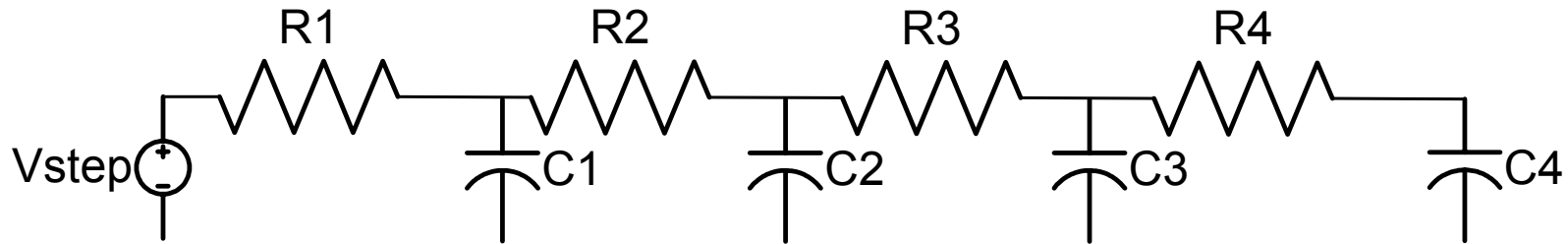
Compare:

$$H(s) = m_0 + m_1 s + m_2 s^2 + \cdots + m_{2q-1} s^{2q-1} + O(s^{2q})$$

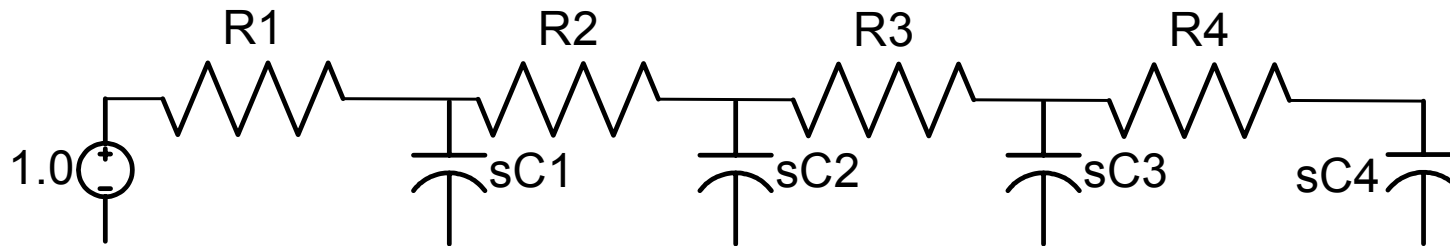
Kth moments:
$$m_k = \frac{(-1)^k}{k!} \int_0^{\infty} t^k h(t) dt$$

Moments of RC Trees (1)

□ RC Circuit

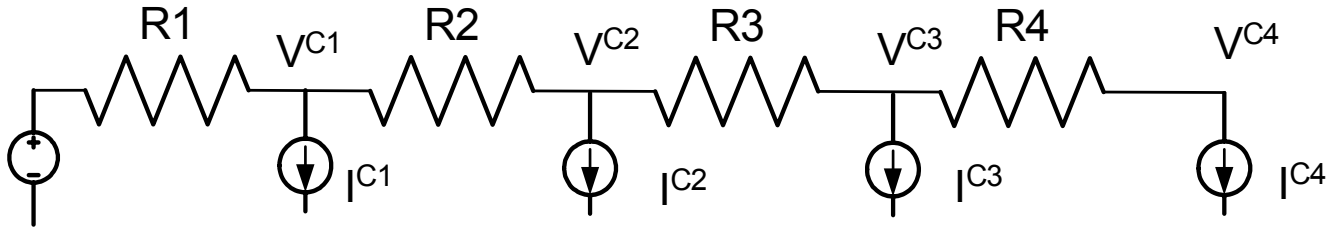


□ s-Domain representation of RC Circuit



Moments of RC Trees (2)

□ Equivalent Circuit



$$V^{c1} = m_0^{c1} + m_1^{c1}s + m_2^{c1}s^2 + m_3^{c1}s^3 + \dots$$

$$V^{c2} = m_0^{c2} + m_1^{c2}s + m_2^{c2}s^2 + m_3^{c2}s^3 + \dots$$

$$V^{c3} = m_0^{c3} + m_1^{c3}s + m_2^{c3}s^2 + m_3^{c3}s^3 + \dots$$

$$V^{c4} = m_0^{c4} + m_1^{c4}s + m_2^{c4}s^2 + m_3^{c4}s^3 + \dots$$

$$I^{c1} = sC_1(m_0^{c1} + m_1^{c1}s + m_2^{c1}s^2 + m_3^{c1}s^3 + \dots)$$

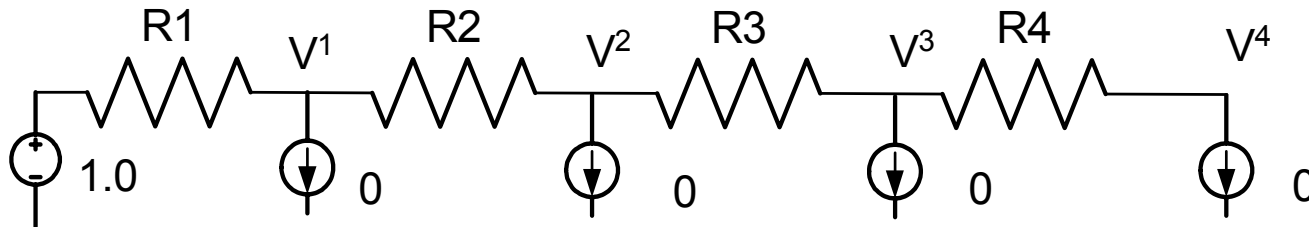
$$I^{c2} = sC_2(m_0^{c2} + m_1^{c2}s + m_2^{c2}s^2 + m_3^{c2}s^3 + \dots)$$

$$I^{c3} = sC_3(m_0^{c3} + m_1^{c3}s + m_2^{c3}s^2 + m_3^{c3}s^3 + \dots)$$

$$I^{c4} = sC_4(m_0^{c4} + m_1^{c4}s + m_2^{c4}s^2 + m_3^{c4}s^3 + \dots)$$

Moments of RC Trees (3)

□ DC equivalent circuit for m_0



$$V^1 = m_0^{c1}$$

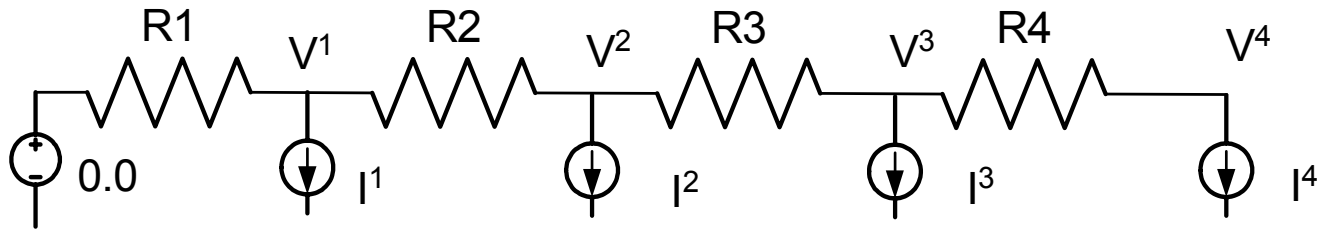
$$V^2 = m_0^{c2}$$

$$V^3 = m_0^{c3}$$

$$V^4 = m_0^{c4}$$

Moments of RC Trees (4)

- DC equivalent circuit for other moments



$$I^1 = C_1 m_q^{c1}$$

$$V^1 = m_{q+1}^{c1}$$

$$I^2 = C_2 m_q^{c2}$$

$$V^2 = m_{q+1}^{c2}$$

$$I^3 = C_3 m_q^{c3}$$

$$V^3 = m_{q+1}^{c3}$$

$$I^4 = C_4 m_q^{c4}$$

$$V^4 = m_{q+1}^{c4}$$

Moment matching: Pade Approximation

$$H(s) = m_0 + m_1s + m_2s^2 + \cdots + m_{2q-1}s^{2q-1} + O(s^{2q})$$

$$\hat{H}(s) = \frac{b_0 + b_1s + \cdots + b_{q-1}s^{q-1}}{1 + a_1s + \cdots + a_qs^q}$$

Choose the $2q$ rational function coefficients $a_1, a_2, \cdots, a_{q-1}, b_1, b_2, \cdots, b_{q-1}$,
So that the reduced rational function matches the first $2q$ moments of
The original transfer function

$$\frac{b_0 + b_1s + \cdots + b_{q-1}s^{q-1}}{1 + a_1s + \cdots + a_qs^q} = m_0 + m_1s + m_2s^2 + \cdots + m_{2q-1}s^{2q-1}$$

Moment matching: Pade' Approximation

- Step 1: calculate the first $2q$ moments of $H(s)$

$$m_0, \quad m_1, \quad m_2, \quad \cdots, \quad m_{2q-1}$$

$$m_k = -QA^{-k}B$$

- Step 2: calculate the $2q$ coeff. of the Pade' approx, matching the first $2q$ moments of $H(s)$

$$b_0, b_1, \dots, b_{q-1}, a_1, \dots, a_q$$

Calculation of Pade' coeff.

$$\frac{b_0 + b_1s + \cdots + b_{q-1}s^{q-1}}{1 + a_1s + \cdots + a_qs^q} = m_0 + m_1s + m_2s^2 + \cdots + m_{2q-1}s^{2q-1}$$

**For coeff. a's
solve the
following
linear system:**

$$\begin{bmatrix} m_0 & m_1 & m_2 & \cdots & m_{q-1} \\ m_1 & m_2 & & & \vdots \\ m_2 & & & & \\ \vdots & & & & \\ m_{q-1} & \cdots & & & m_{2q-2} \end{bmatrix} \begin{bmatrix} a_q \\ a_{q-1} \\ a_{q-2} \\ \vdots \\ a_1 \end{bmatrix} = - \begin{bmatrix} m_q \\ m_{q+1} \\ m_{q+2} \\ \vdots \\ m_{2q-1} \end{bmatrix}$$

$$b_0 = m_0$$

**For coeff. b's
simply
calculate:**

$$b_1 = m_1 + a_1m_0$$

$$\vdots$$

$$b_{q-1} = m_{q-1} + a_1m_{q-2} + \cdots + a_{q-1}m_0$$

Pade matrix can be very ill-conditioned

$$m_k = -QA^{-k}B \quad m_k \cong \lambda^k m_0$$

- matrix powers converge to the eigenvector corresponding to the largest eigenvalue.

$$\begin{bmatrix} m_0 & m_1 & m_2 & \cdots & m_{q-1} \\ m_1 & m_2 & & & \vdots \\ m_2 & & & & \\ \vdots & & & & \\ m_{q-1} & \cdots & & & m_{2q-2} \end{bmatrix} \begin{bmatrix} a_q \\ a_{q-1} \\ a_{q-2} \\ \vdots \\ a_1 \end{bmatrix} = - \begin{bmatrix} m_q \\ m_{q+1} \\ m_{q+2} \\ \vdots \\ m_{2q-1} \end{bmatrix}$$

**Columns become linearly dependent for large q
the problem is numerically very ill-conditioned!**

Numerical Issues

- ❑ Converges to an eigenvector corresponding to an eigenvalue of matrix $\mathbf{A}$ with largest absolute value
- ❑ Matrix will be ill-conditioned
- ❑ Increasing number of poles dose not improve the accuracy
- ❑ Unable to estimate the accuracy of the approximating model



Passivity

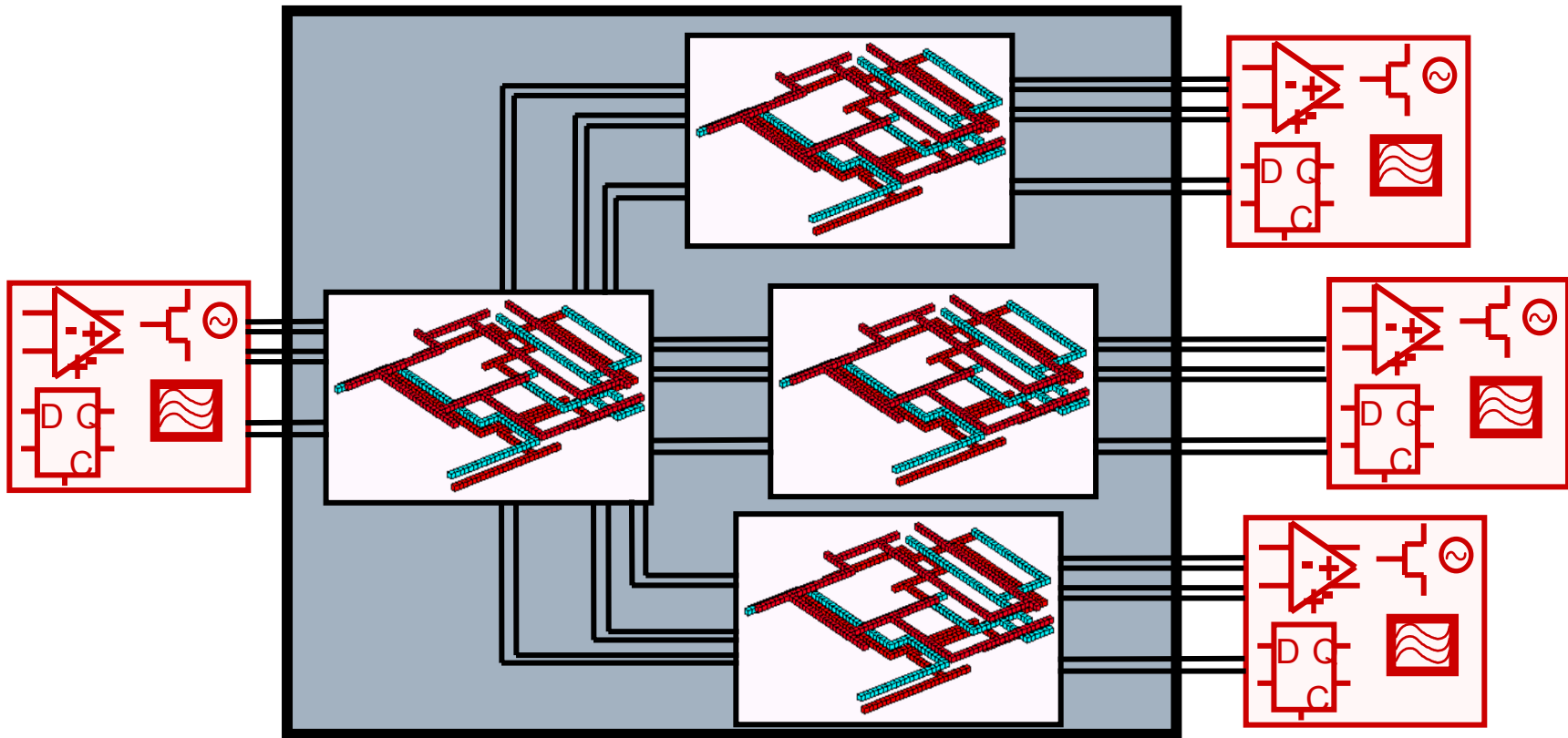
- Passive systems do not generate energy. We cannot extract out more energy than is stored. A **passive** system does not provide energy that is not in its storage elements.

$$\text{Energy} = \int_{-\infty}^t i(\tau)v(\tau) d\tau \geq 0$$

- If the reduced model is not passive it can generate energy from nothingness and the simulation will explode

Interconnecting Passive Systems

- The interconnection of stable models is not necessarily stable
- BUT the interconnection of passive models is a passive model:



Positive Real Functions

- A positive real function is a function internally stable with non-negative real part

$$i(s) = H(s)v(s)$$

$\mathbf{H}(s)$ is analytic for $\text{Re}(s) > 0$  (no unstable poles)

$\overline{\mathbf{H}(s)} = \mathbf{H}(\bar{s})$ for $\text{Re}(s) > 0$  (real response)

$\mathbf{H}(s) + \mathbf{H}(s)^H \geq 0$, for $\text{Re}(s) > 0$  (no negative resistors)

↙ Hermitian=conjugate and transposed

↘ It means its real part is a positive semidefinite matrix at all frequencies

Positive Realness & Passivity

- For systems with immittance (impedance or admittance) matrix representation, positive-realness of the transfer function is equivalent to **passivity**



$$\begin{bmatrix} v_1^{\text{in}}(s) \\ v_2^{\text{in}}(s) \end{bmatrix} = Z(s) \begin{bmatrix} I_1^{\text{out}}(s) \\ I_2^{\text{out}}(s) \end{bmatrix}$$

Necessary conditions for passivity

- The positive-real condition on the matrix rational function $H(S)$ implies that:
 - If $H(s)$ is positive-real also its inverse is positive real
 - If $H(s)$ is positive-real it has no poles in the RHP, and hence also no zeros there.
- Occasional misconception : “if the system function has no poles and no zeros in the RHP the system is passive”.
- It is *necessary* that a positive-real function have no poles or zeros in the RHP, but not *sufficient*.

Sufficient conditions for passivity

$$sEx = Ax + Bu$$

$$y = Qx$$

$$H(s) = Q(sE - A)^{-1}B$$

□ Sufficient conditions for passivity:

1) $Q = B^T$

2) $x(E + E^H)x \geq 0$, for all x

3) $x(A + A^H)x \geq 0$, for all x

□ Note that these are NOT necessary conditions
(common misconception)

Congruence Transformations Preserve Positive Semidefiniteness

□ Def. congruence transformation

$$\hat{E} = U^T E U$$

□ Property: a congruence transformation preserves the positive semidefiniteness of the matrix

$$x U^T E U x \geq 0, \quad y = U x \quad \text{for all } x$$

□ Note:

$$x U^T (E + E^H) U x \geq 0, \quad \text{for all } x$$

PRIMA (for preserving passivity)

$V=U$, $U^T U=I$, Arnoldi Krylov Projection Framework:

$$\begin{array}{lcl} sEx = Ax + bu & \Rightarrow & s \overbrace{U_q^T E U_q}^{\hat{E}} x = \overbrace{U_q^T A U_q}^{\hat{A}} x + \overbrace{U_q^T b}^{\hat{b}} u \\ y = b^T x & & y = b^T \underbrace{U_q}_{{\hat{b}}^T} x \end{array}$$

$$\text{span} \{ \vec{u}_1, \dots, \vec{u}_q \} = \kappa_q(A^{-1}E, b) = \text{span} \{ b, A^{-1}E, \dots, (A^{-1}E)^{q-1} b \}$$

$$U = V = \{ \vec{u}_1, \dots, \vec{u}_q \}$$

$$\underline{U^T U = I} \quad \longleftarrow \text{Use Arnoldi: Numerically very stable}$$

PRIMA preserves passivity

- PRIMA applies the projection framework to

$$sEx = Ax + Bu \qquad y = B^T x$$

- PRIMA preserves passivity because
 - uses Arnoldi so that $U=V$ and the projection becomes a congruence transformation
 - E and A produced by electromagnetic analysis are typically positive semidefinite while $A^{-1}E$ may not be.
 - input matrix must be equal to output matrix

Algebraic proof of moment matching for PRIMA expansion at $s=0$, $V=U$, orthonormal $U^T U=I$

$$H(s) = -b^T (A - sE)^{-1} b = -b^T (I - sA^{-1}E)^{-1} A^{-1} b = -\sum_{k=0}^{\infty} \left[b^T (A^{-1}E)^k A^{-1} b \right] s^k$$

$$\hat{H}(s) = -\hat{b}^T (\hat{A} - s\hat{E})^{-1} \hat{b} = -\hat{b}^T (I - s\hat{A}^{-1}\hat{E})^{-1} \hat{A}^{-1} \hat{b} = -\sum_{k=0}^{\infty} \left[\hat{b}^T (\hat{A}^{-1}\hat{E})^k \hat{A}^{-1} \hat{b} \right] s^k$$

$$\begin{aligned} \hat{b}^T (\hat{A}^{-1}\hat{E})^k \hat{A}^{-1} \hat{b} &= b^T U \left[(U^T A U)^{-1} U^T E U \right]^k (U^T A U)^{-1} U^T b = \\ &= b^T U (U^T A U)^{-1} U^T E U \cdots (U^T A U)^{-1} U^T E U (U^T A U)^{-1} U^T b = \\ &= b^T A^{-1} E \cdots A^{-1} E A^{-1} b = b^T [A^{-1} E]^k A^{-1} b \end{aligned}$$

Used Lemma: If U is orthonormal ($U^T U=I$) and b is a vector such that

$$A^{-1} b \in \text{colspan}(U) \quad \text{then} \quad (U^T A U)^{-1} U^T b = U^T A^{-1} b$$

Proof of lemma

$$A^{-1}b \in \text{colspan}(U) \quad \Rightarrow \quad (U^T A U)^{-1} U^T b = U^T A^{-1}b$$

Proof:

$$A^{-1}b \in \text{colspan}(U) \quad \Rightarrow \quad \exists g \text{ s.t. } A^{-1}b = Ug$$

$$\begin{aligned} (U^T A U)^{-1} U^T b &= \\ &= (U^T A U)^{-1} U^T A A^{-1}b = \\ &= (U^T A U)^{-1} U^T A U g = \\ &= g = U^T U g = U^T A^{-1}b \end{aligned}$$