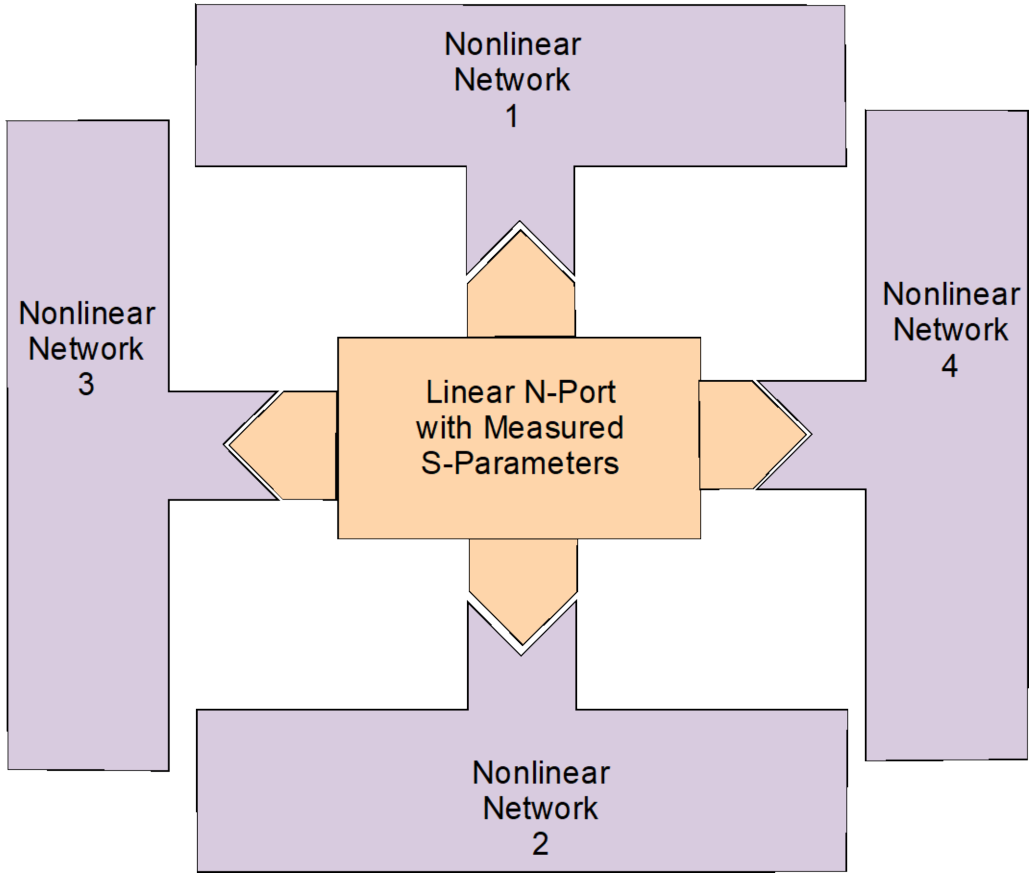
ECE 546
Lecture -14
Macromodeling

Spring 2026

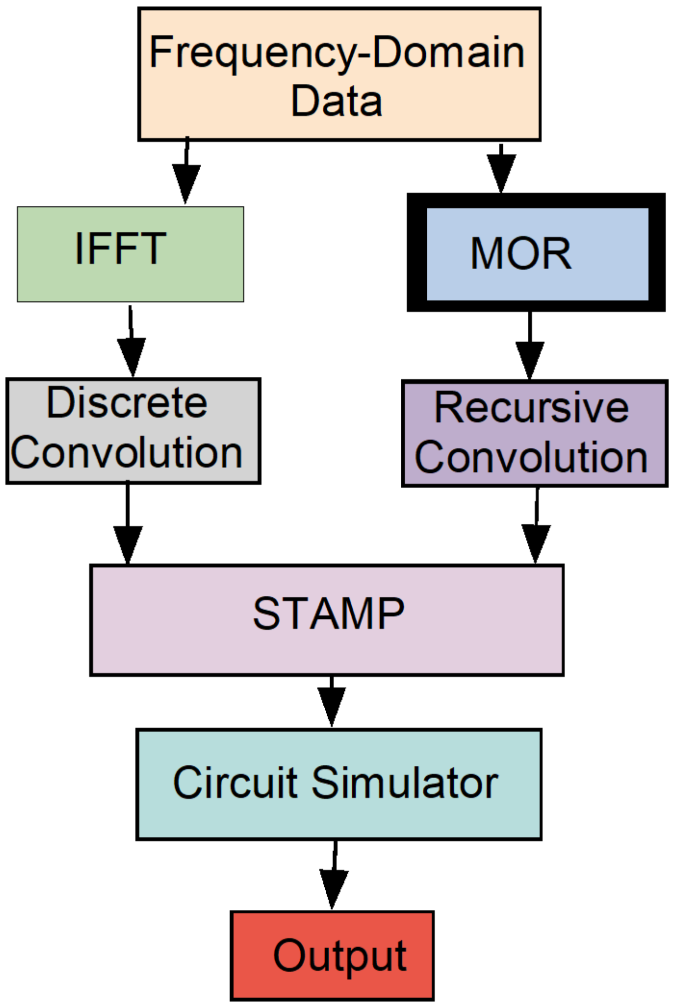
Jose E. Schutt-Aine
Electrical & Computer Engineering
University of Illinois
jesa@illinois.edu

Blackbox Macromodeling



Objective: Perform time-domain simulation of composite network to determine timing waveforms, noise response or eye diagrams

Macromodel Implementation



Blackbox Synthesis

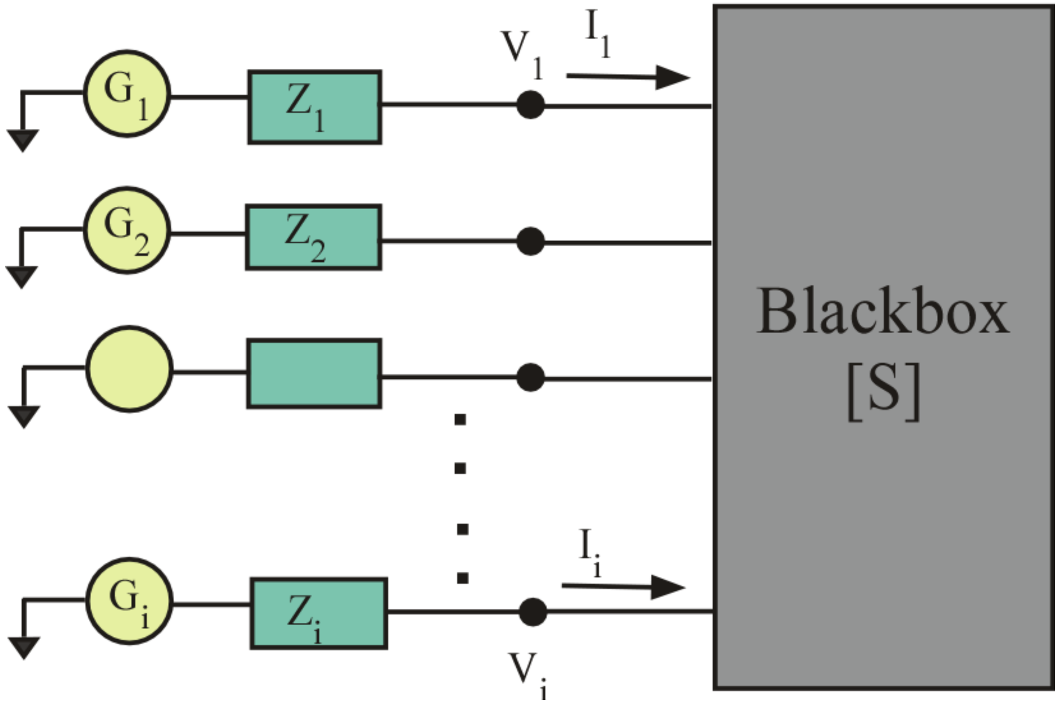
Motivations

- Only measurement data is available
- Actual circuit model is too complex

Methods

- Inverse-Transform & Convolution
  - IFFT from frequency domain data
  - Convolution in time domain
- Macromodel Approach
  - Curve fitting
  - Recursive convolution

Blackbox Synthesis



Terminations are described by a source vector $G(\omega)$ and an impedance matrix Z

Blackbox is described by its scattering parameter matrix S

Blackbox - Method 1

Scattering Parameters $B(\omega) = S(\omega)A(\omega)$ (1)

Terminal conditions $A(\omega) = \Gamma B(\omega) + TG(\omega)$ (2)

where $\Gamma = -\left[U + ZZ_o^{-1}\right]^{-1}\left[U - ZZ_o^{-1}\right]$

and $T = \left[U + ZZ_o^{-1}\right]^{-1}$

U is the unit matrix, Z is the termination impedance matrix, Z_o is the reference impedance matrix and $G(\omega)$ is the source vector.

Blackbox - Method 1

Combining (1) and (2) $A(\omega)=\left[U-\Gamma S(\omega)\right]^{-1}TG(\omega)$

and $B(\omega)=S(\omega)A(\omega)=S(\omega)\left[U-\Gamma S(\omega)\right]^{-1}TG(\omega)$

$$V(\omega)=A(\omega)+B(\omega)=\left[U+S(\omega)\right]\left[U-\Gamma S(\omega)\right]^{-1}TG(\omega)$$

$$I(\omega)=Z_o^{-1}\left[A(\omega)-B(\omega)\right]=Z_o^{-1}\left[U-S(\omega)\right]\left[U-\Gamma S(\omega)\right]^{-1}TG(\omega)$$

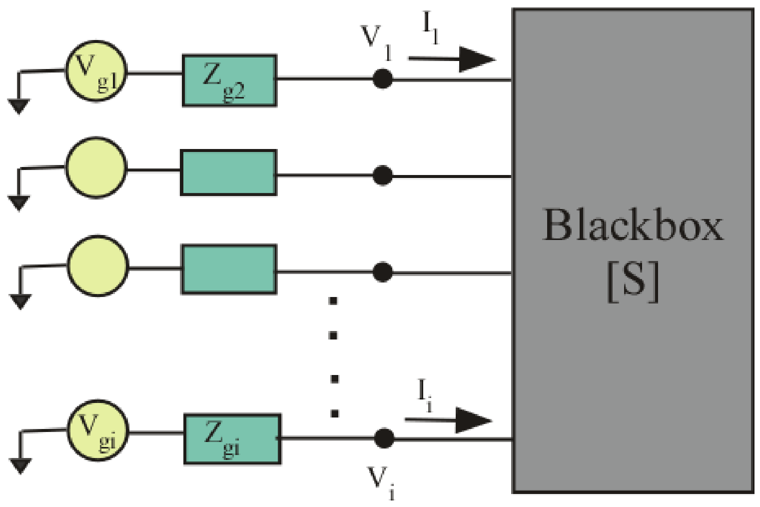
$$v(t)=IFFT\left\{V(\omega)\right\}$$

$$i(t)=IFFT\left\{I(\omega)\right\}$$

Method 1 - Limitations

- **No Frequency Dependence for Terminations**
  - Reactive terminations cannot be simulated
- **Only Linear Terminations**
  - Transistors and active nonlinear terminations cannot be described
- **Standalone**
  - This approach cannot be implemented in a simulator

Blackbox - Method 2



In frequency domain $B=SA$

In time domain $b(t) = s(t)*a(t)$

Convolution: $s(t)*a(t) = \int_{-\infty}^{\infty} s(t-\tau)a(\tau)d\tau$

Discrete Convolution

When time is discretized the convolution becomes

$$s(t)*a(t) = \sum_{\tau=1}^{t} s(t-\tau)a(\tau)\Delta\tau$$

Isolating *a(t)*

$$s(t)*a(t) = s(0)a(t)\Delta\tau + \sum_{\tau=1}^{t-1} s(t-\tau)a(\tau)\Delta\tau$$

Since *a(t)* is known for $t < t$, we have:

$$H(t) = \sum_{\tau=1}^{t-1} s(t-\tau)a(\tau)\Delta\tau \quad : \quad History$$

Terminal Conditions

Defining $s'(0) = s(0)\Delta\tau$, we finally obtain

$$b(t) = s'(0)a(t) + H(t)$$

$$a(t) = \Gamma(t)b(t) + T(t)g(t)$$

By combining these equations, the stamp can be derived

Stamp Equation Derivation

The solutions for the incident and reflected wave vectors are given by:

$$a(t) = \left[1 - \Gamma(t)s'(0)\right]^{-1}\left[T(t)g(t) + \Gamma(t)H(t)\right]$$

$$b(t) = s'(0)a(t) + H(t)$$

The voltage wave vectors can be related to the voltage and current vectors at the terminals

$$a(t) = \frac{1}{2}\left[v(t) + Z_o i(t)\right]$$

$$b(t) = \frac{1}{2}\left[v(t) - Z_o i(t)\right]$$

Stamp Equation Derivation

From which we get

$$\frac{1}{2}\left[v(t)-Z_o i(t)\right]=\frac{s'(0)}{2}\left[v(t)+Z_o i(t)\right]+H(t)$$

or

$$Z_o i(t)+s'(0)Z_o i(t)+2H(t)=\left[1-s'(0)\right]v(t)$$

or

$$\left[1+s'(0)\right]Z_o i(t)=\left[1-s'(0)\right]v(t)-2H(t)$$

which leads to

$$i(t)=Z_o^{-1}\left[1+s'(0)\right]^{-1}\left[1-s'(0)\right]v(t)-2Z_o^{-1}\left[1+s'(0)\right]^{-1}H(t)$$

Stamp Equation Derivation

$i(t)$ can be written to take the form

$$i(t) = Y_{stamp} v(t) - I_{stamp}$$

in which

$$Y_{stamp} = Z_o^{-1}\left[1 + s'(0)\right]^{-1}\left[1 - s'(0)\right]$$

and

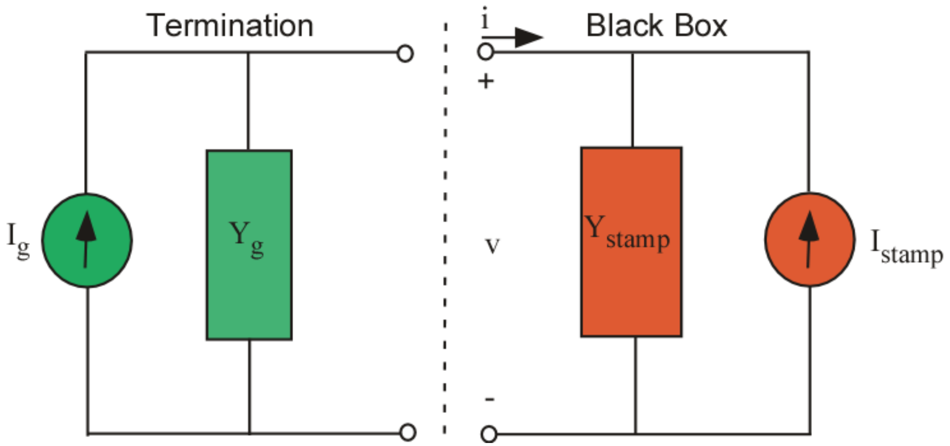
$$I_{stamp} = 2Z_o^{-1}\left[1 + s'(0)\right]^{-1} H(t)$$

Stamp Equations

$$i(t) = Y_{stamp} v(t) - I_{stamp}$$

$$Y_{stamp} = Z_o^{-1}\left[1 + s'(0)\right]^{-1}\left[1 - s'(0)\right]$$

$$I_{stamp} = 2Z_o^{-1}\left[1 + s'(0)\right]^{-1} H(t)$$



$$(Y_g + Y_{stamp}) v(t) = I_g + I_{stamp}$$

Frequency and Time Domains

1. For negative frequencies use conjugate relation $V(-\omega) = V^*(\omega)$

2. DC value: use lower frequency measurement

3. Rise time is determined by frequency range or bandwidth

4. Time step is determined by frequency range

5. Duration of simulation is determined by frequency step

Problems and Issues

- **Discretization:** (not a continuous spectrum)

- **Truncation:** frequency range is band limited

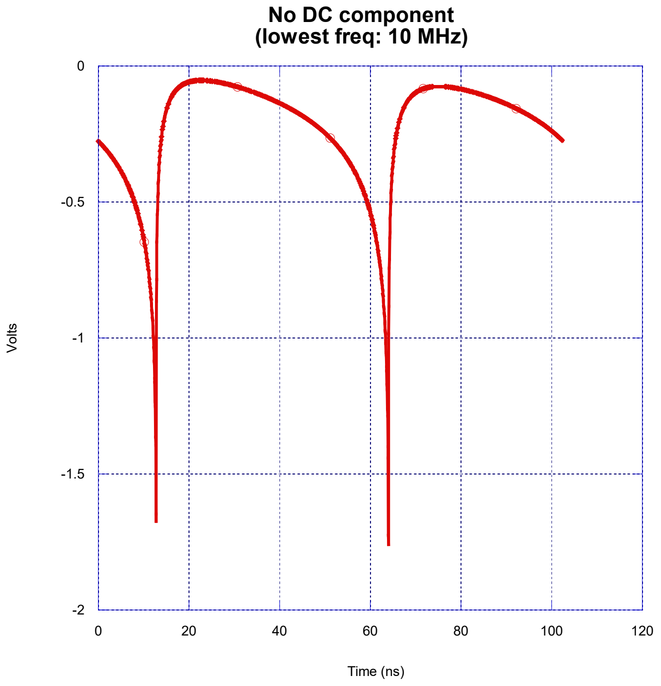
F: frequency range
N: number of points
$\Delta f = F/N$: frequency step
Δt = time step

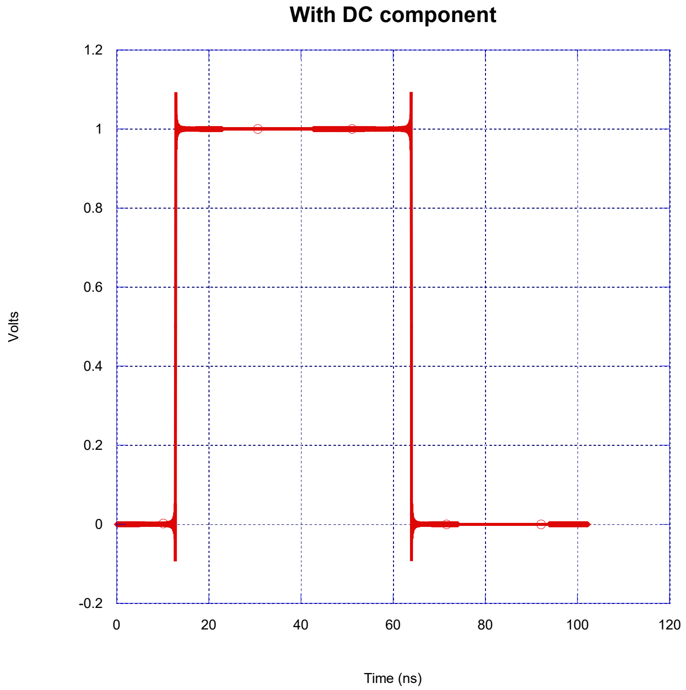
Problems and Issues

Problems & Limitations (in frequency domain)	Consequences (in time domain)	Solution
Discretization	Time-domain response will repeat itself periodically (Fourier series) Aliasing effects	Take small frequency steps. Minimum sampling rate must be the Nyquist rate
Truncation in Frequency	Time-domain response will have finite time resolution (Gibbs effect)	Take maximum frequency as high as possible
No negative frequency values	Time-domain response will be complex	Define negative-frequency values and use $V(-f)=V^*(f)$ which forces $v(t)$ to be real
No DC value	Offset in time-domain response, ringing in base line	Use measurement at the lowest frequency as the DC value

Effect of Low-Frequency Data

Calculating inverse Fourier Transform of: $V(f) = \dfrac{2\sin(2\pi ft)}{2\pi ft}$

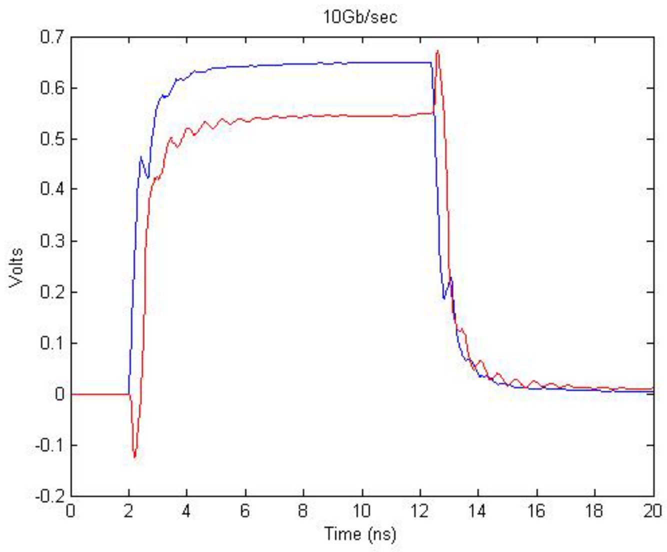




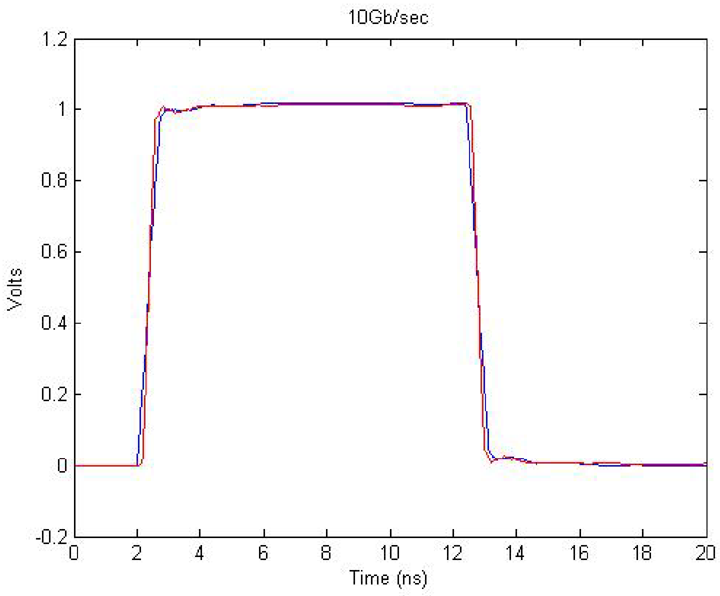
Left: IFFT of a sinc pulse sampled from 10 MHz to 10 GHz. Right: IFFT of the same sinc pulse with frequency data ranging from 0-10 GHz. In both cases 1000 points are used

Effects of DC Data

No DC Data Point

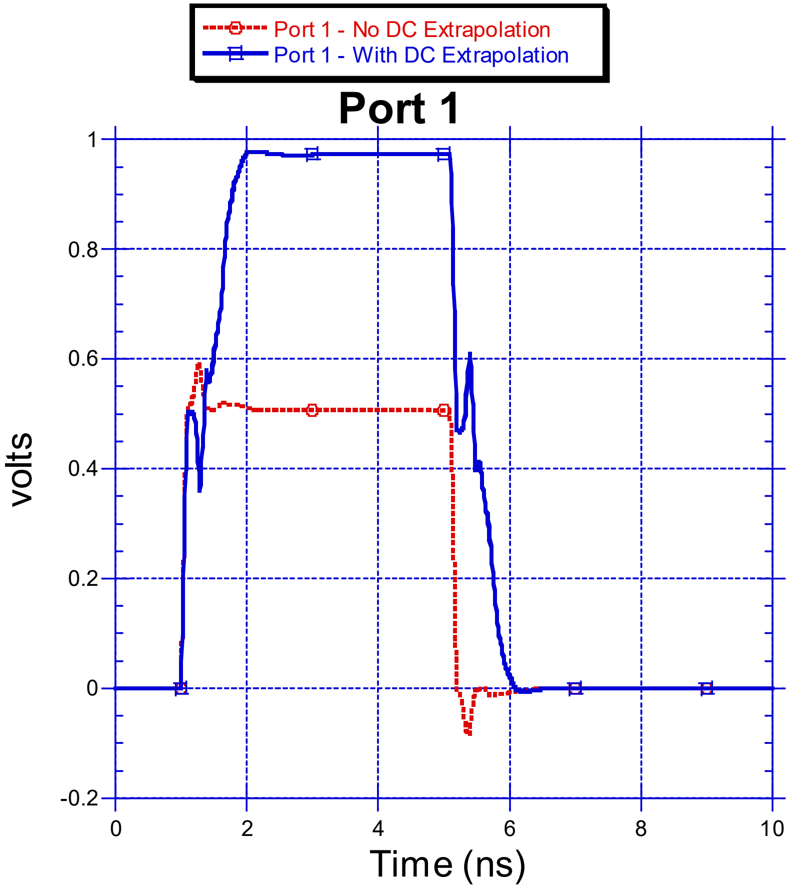


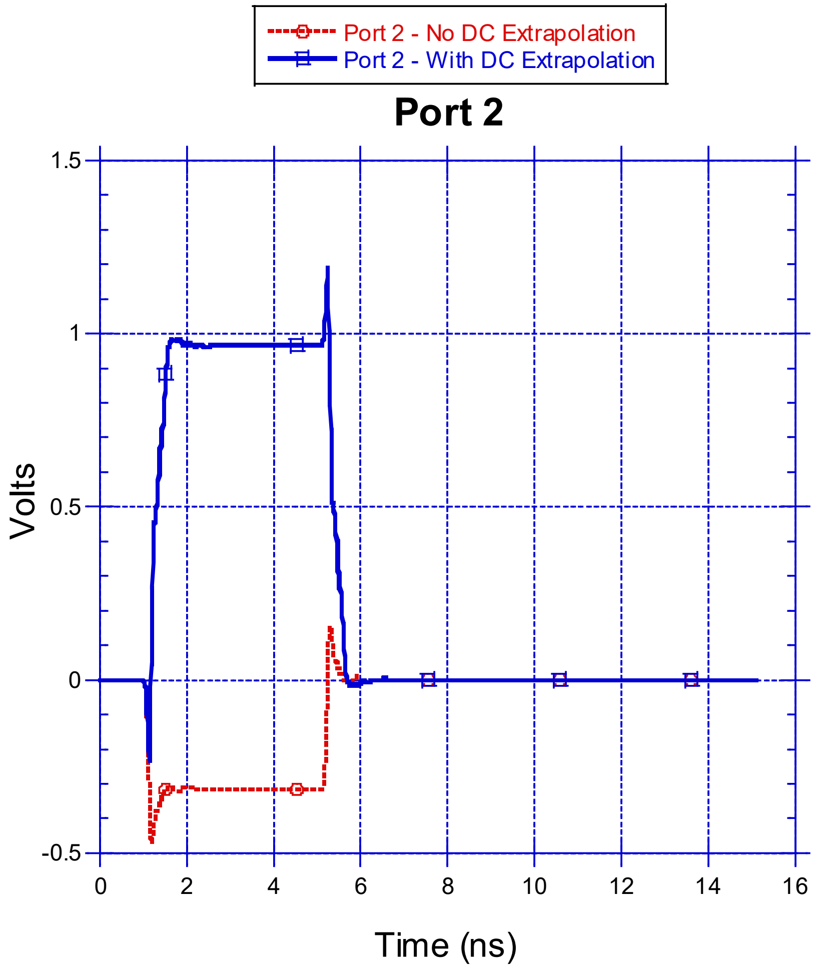
With DC Data Point



If low-frequency data points are not available, extrapolation must be performed down to DC.

Effect of Low-Frequency Data





Convolution Limitations

Frequency-Domain Formulation $$Y(\omega) = H(\omega)X(\omega)$$

Time-Domain Formulation $$y(t) = h(t) * x(t)$$

Convolution $$y(t) = h(t) * y(t) = \int_0^t h(t-\tau)y(\tau)d\tau$$

Discrete Convolution $$h(t) * x(t) = \sum_{\tau=1}^{t} h(t-\tau)x(\tau)\Delta\tau$$

$$H(t) = \sum_{\tau=1}^{t-1} h(t-\tau)x(\tau)\Delta\tau \quad : \quad History$$

Computing History is computationally expensive ➔ Use FD rational approximation and TD recursive convolution

Complex Plane

– An arbitrary network's transfer function can be described in terms of its s-domain representation

– s is a complex number $s = \sigma + j\omega$

– The impedance (or admittance) or transfer function of networks can be described in the s domain as

$$T(s) = \frac{a_m s^m + a_{m-1} s^{m-1} + \ldots + a_1 s + a_0}{s^n + b_{n-1} s^{n-1} + \ldots + b_1 s + b_0}$$

Transfer Functions

$$T(s) = \frac{a_m s^m + a_{m-1}s^{m-1} + \ldots + a_1 s + a_0}{s^n + b_{n-1}s^{n-1} + \ldots + b_1 s + b_0}$$

The coefficients *a* and *b* are real and the order *m* of the numerator is smaller than or equal to the order *n* of the denominator

A stable system is one that does not generate signal on its own.

For a stable network, the roots of the denominator should have negative real parts

Transfer Functions

The transfer function can also be written in the form

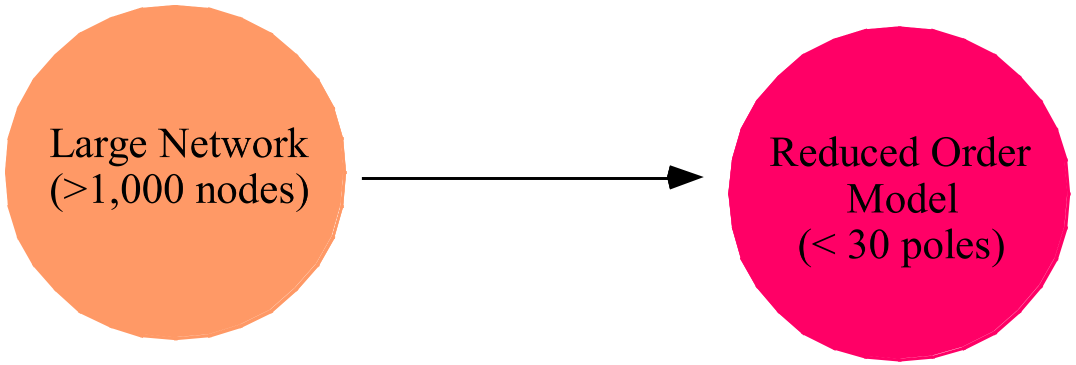
$$T(s) = a_m \frac{(s - Z_1)(s - Z_2)...(s - Z_m)}{(s - P_1)(s - P_2)...(s - P_m)}$$

$Z_1, Z_2, ...Z_m$ are the **zeros** of the transfer function

$P_1, P_2, ...P_m$ are the **poles** of the transfer function

For a stable network, the poles should lie on the left half of the complex plane

Model Order Reduction



SPICE

$Y(t)\, v(t) = i(t)$

$$\tilde{Y}(\omega) = \left[A_1 + \sum_{i=1}^{L} \frac{a_{1i}}{1 + j\omega/\omega_{c1i}} \right]$$

$Y(\omega)\, V(\omega) = I(\omega)$

↓

Order Reduction

↓

$Y(\omega) \simeq \tilde{Y}(\omega)$

↓

Recursive Convolution

↓

$\tilde{Y}(t)\, v(t) = i(t)$

Model Order Reduction

Objective: Approximate frequency-domain transfer function to take the form:

$$H(\omega) = \left[A_1 + \sum_{i=1}^{L} \frac{a_{1i}}{1 + j\omega / \omega_{c1i}} \right]$$

Methods

- AWE – Pade
- Pade via Lanczos (Krylov methods)
- Rational Function
- Chebyshev-Rational function
- Vector Fitting Method

Model Order Reduction (MOR)

Question: Why use a rational function approximation?

Answer: because the frequency-domain relation

$$Y(\omega) = H(\omega)X(\omega) = \left[d + \sum_{k=1}^{L} \frac{c_k}{1 + j\omega / \omega_{ck}} \right] X(\omega)$$

will lead to a time-domain *recursive* convolution:

$$y(t) = dx(t-T) + \sum_{k=1}^{L} y_{pk}(t)$$

where

$$y_{pk}(t) = a_k x(t-T)\left(1 - e^{-\omega_{ck}T}\right) + e^{-\omega_{ck}T} y_{pk}(t-T)$$

which is very fast!

Understanding Recursive Convolution

$$Y(\omega) = H(\omega)X(\omega) = \left[d + \sum_{k=1}^{L} \frac{c_k}{1 + j\omega / \omega_{ck}} \right] X(\omega)$$

Given that for each term

$$\frac{c_k}{1 + j\omega / \omega_{ck}} \leftrightarrow \omega_{ck} c_k e^{-\omega_{ck} t}$$

and

$$X(\omega) \leftrightarrow x(t)$$

We would need to evaluate

$$y_k(t) = \omega_{ck} c_k e^{-\omega_{ck} t} * x(t)$$

Recursive Convolution

We wish to evaluate $y(t) = Ae^{-\alpha t} * x(t)$ (1)

This means $$y(t) = \int_0^t Ae^{-\alpha\tau} x(t-\tau)d\tau \quad (2)$$

We can observe that: $$y(t-h) = \int_0^{t-h} Ae^{-\alpha\tau} x(t-h-\tau)d\tau \quad (3)$$

We next express (2) as

$$y(t) = \underbrace{\int_0^h Ae^{-\alpha\tau} x(t-\tau)d\tau}_{I_1} + \underbrace{\int_h^t Ae^{-\alpha\tau} x(t-\tau)d\tau}_{I_2} \quad (4)$$

Recursive Convolution

In the second integral, set $\tau = \tau' + h$ which implies that $\tau' = \tau - h$

$$I_2 = \int_0^{t-h} Ae^{-\alpha(\tau'+h)} x(t-\tau'-h)\,d\tau' \qquad (5)$$

$$I_2 = e^{-\alpha h} \int_0^{t-h} Ae^{-\alpha\tau'} x(t-\tau'-h)\,d\tau' = e^{-\alpha h} y(t-h) \qquad (6)$$

Thus

$$y(t) = \underbrace{\int_0^{h} Ae^{-\alpha\tau} x(t-\tau)\,d\tau}_{I_1} + e^{-\alpha h} y(t-h) \qquad (7)$$

Recursive Convolution

We next evaluate the first integral

$$I_1 = Ax(t-h)\int_0^h e^{-\alpha\tau} d\tau \qquad (8)$$

in which we assume a step invariant (constant) behavior of the input function. This can be evaluated to yield:

$$I_1 = \frac{Ax(t-h)}{\alpha}\left(1-e^{-\alpha\tau}\right) \qquad (9)$$

so that

$$y(t) = \frac{Ax(t-h)}{\alpha}\left(1-e^{-\alpha h}\right) + e^{-\alpha h} y(t-h) \qquad (11)$$

Which is the **recursive convolution** formula

Model Order Reduction

Transfer function is approximated as

$$H(\omega) = d + \sum_{k=1}^{L} \frac{c_k}{1 + j\omega / \omega_{ck}}$$

In order to convert data into rational function form, we need a curve fitting scheme ➔ Use Vector Fitting

History of Vector Fitting (VF)

- 1998 - Original VF formulated by Bjorn Gustavsen and Adam Semlyen*
- 2003 - Time-domain VF (TDVF) by S. Grivet-Talocia.
- 2005 - Orthonormal VF (OVF) by Dirk Deschrijver, Tom Dhaene, et al.
- 2006 - Relaxed VF by Bjorn Gustavsen.
- 2006 - VF re-formulated as Sanathanan-Koerner (SK) iteration by W. Hendrickx, Dirk Deschrijver and Tom Dhaene, et al.

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

Vector Fitting (VF)

Vector fitting algorithm

$$\begin{bmatrix} \sigma(s) f(s) \\ \sigma(s) \end{bmatrix} \approx \begin{bmatrix} \sum_{n=1}^{N} \frac{c_n}{s - \tilde{a}_n} + d + sh \\ \sum_{n=1}^{N} \frac{\tilde{c}_n}{s - \tilde{a}_n} + 1 \end{bmatrix}$$

$$\left(\sum_{n=1}^{N} \frac{c_n}{s - \tilde{a}_n} + d + sh \right) - \left(\sum_{n=1}^{N} \frac{\tilde{c}_n}{s - \tilde{a}_n} \right) f(s) \approx f(s).$$

Solve for $c_n, \tilde{c}_n, d, h$

Can show* that the zeros of $\sigma(s)$ are the poles of $f(s)$ for the next iteration

Avoid ill-conditioned matrix

Guarantee stability

Converge, accurate

With Good Initial Poles

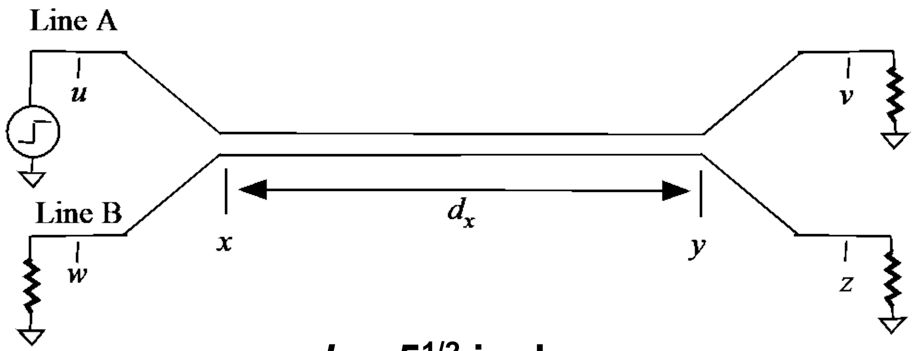
* B. Gustavsen and A. Semlyen, "Rational approximation of frequency responses by vector fitting," IEEE Trans. Power Del., vol. 14, no. 3, pp 1052–1061, Jul. 1999

Examples

1.- DISC: Transmission line with discontinuities

Length = 7 inches

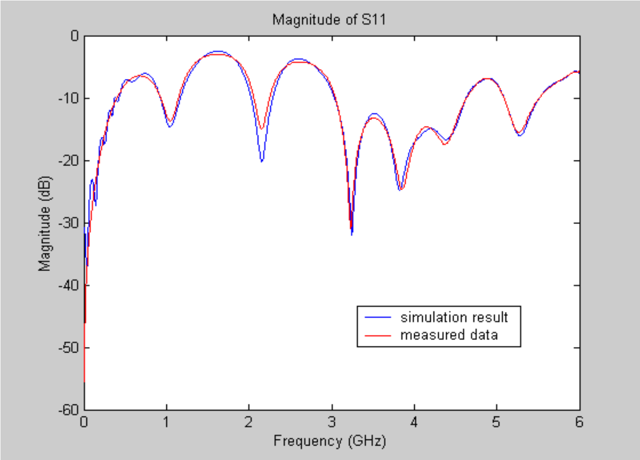
2.- COUP: Coupled transmission line2

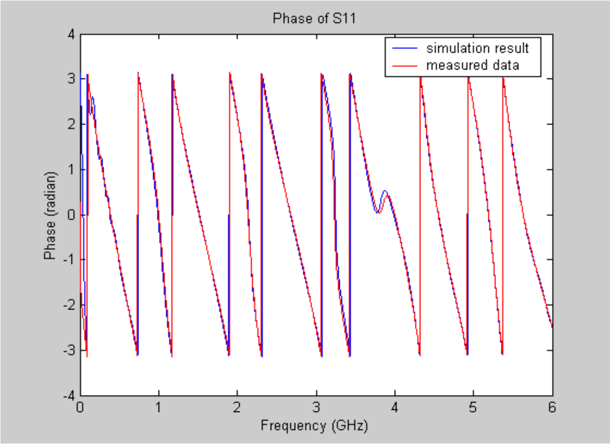


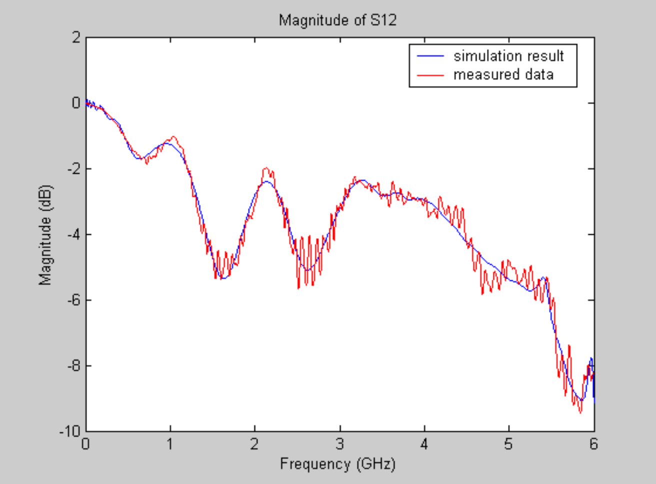
d_x = $5^{1/2}$ inches

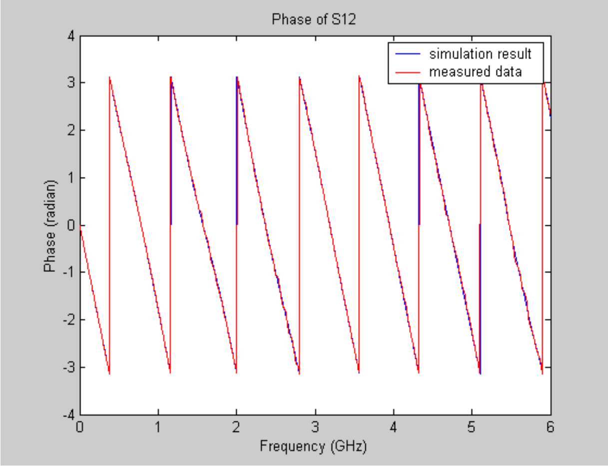
Frequency sweep: 300 KHz – 6 GHz

DISC: Approximation Results





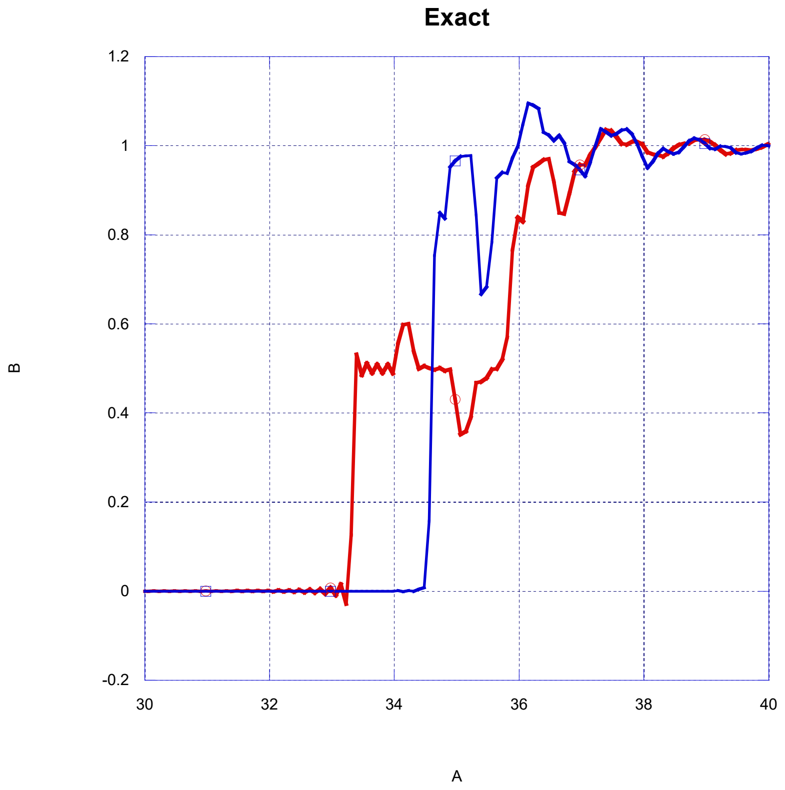


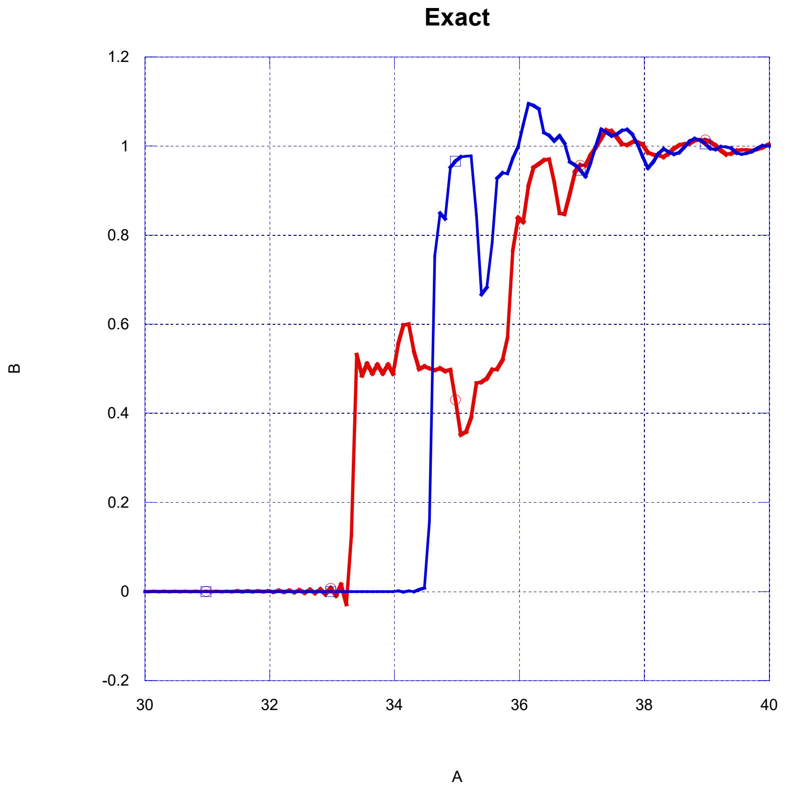


DISC: Approximation order 90

DISC: Simulations

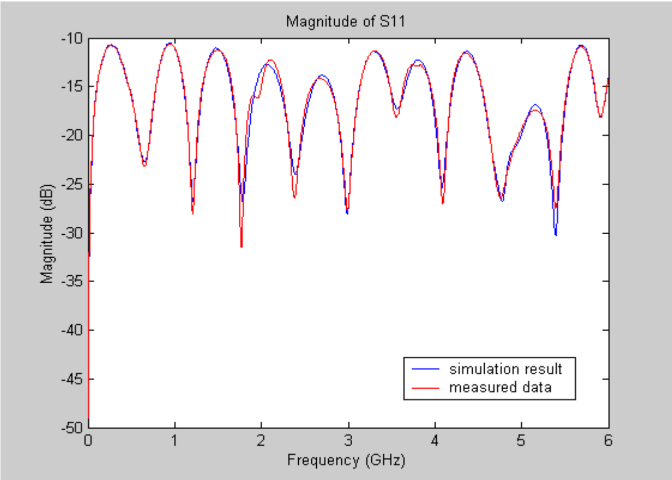
Microstrip line with discontinuities
Data from 300 KHz to 6 GHz

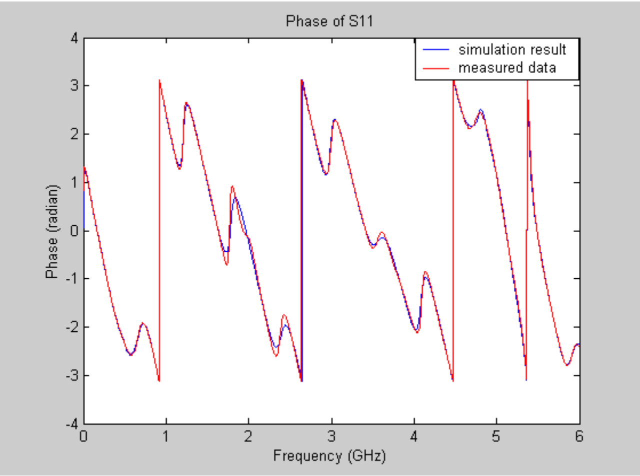


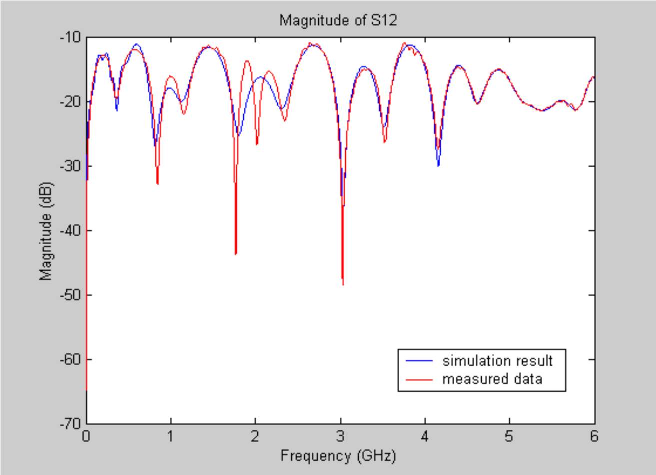


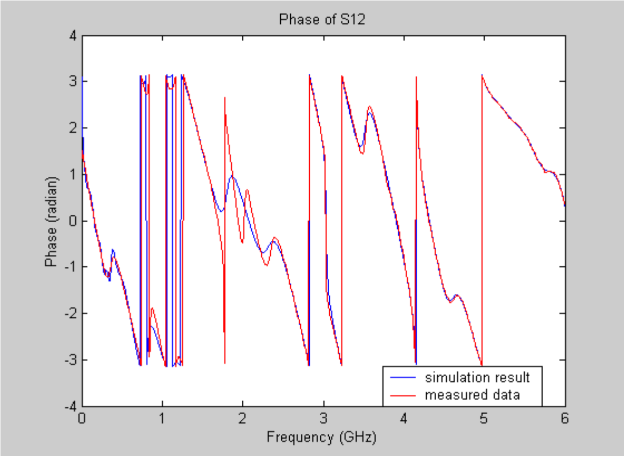
Observation: Good agreement

COUP: Approximation Results





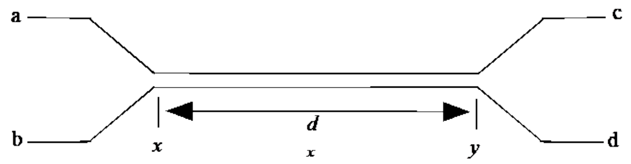


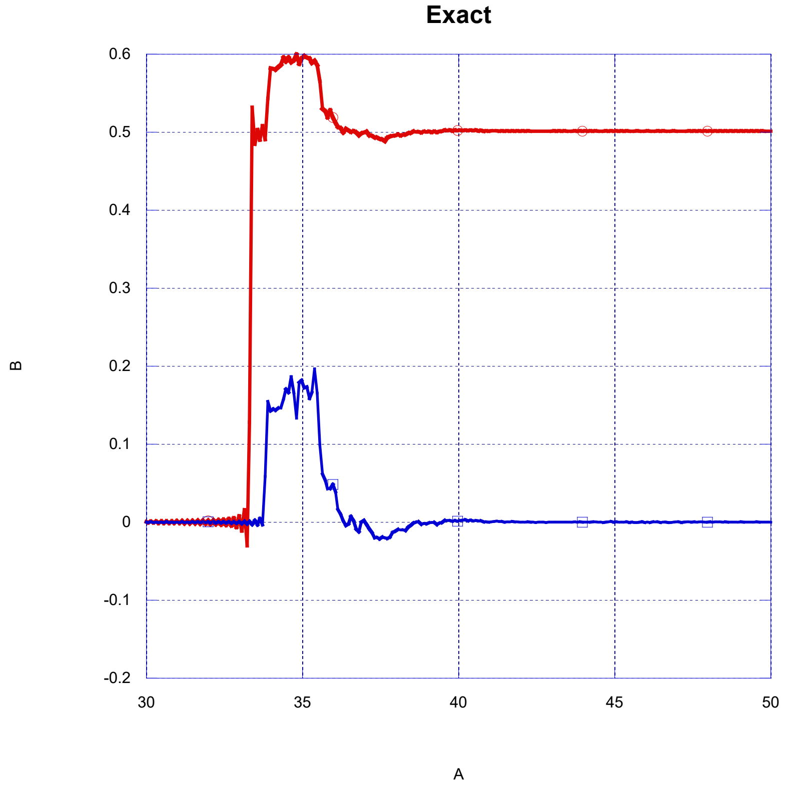


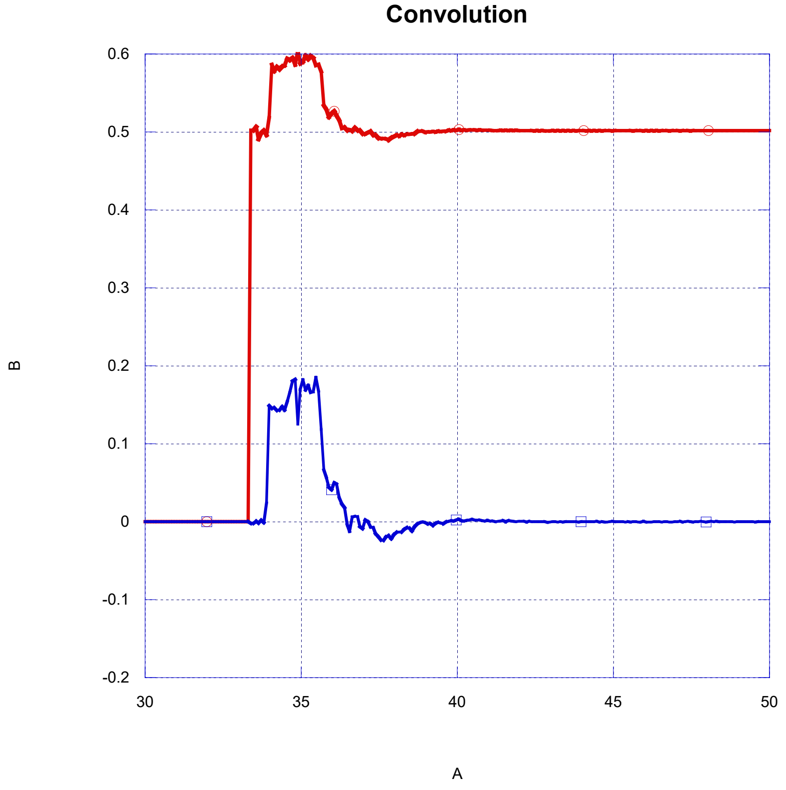
COUP: Approximation order 75 – Before Passivity Enforcement

COUP: Simulations

Port 1: a – Port 2: d
Data from 300 KHz to 6 GHz

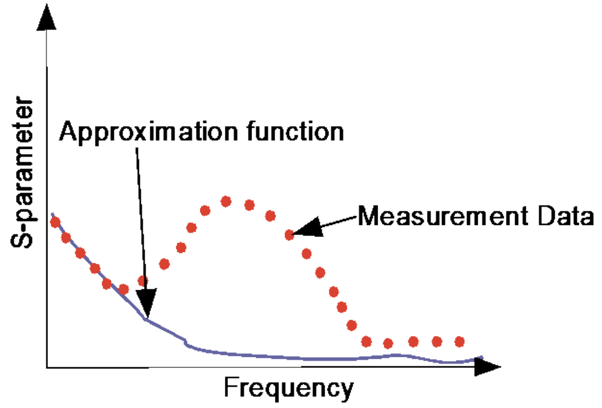


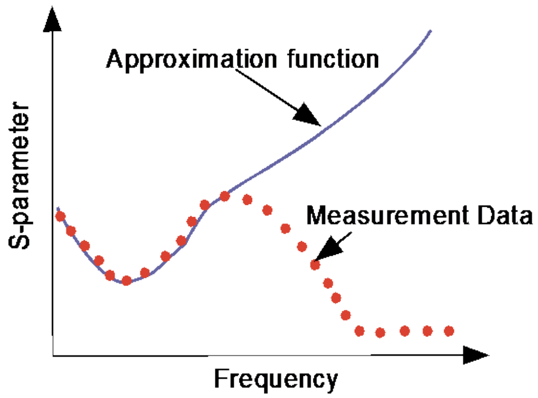


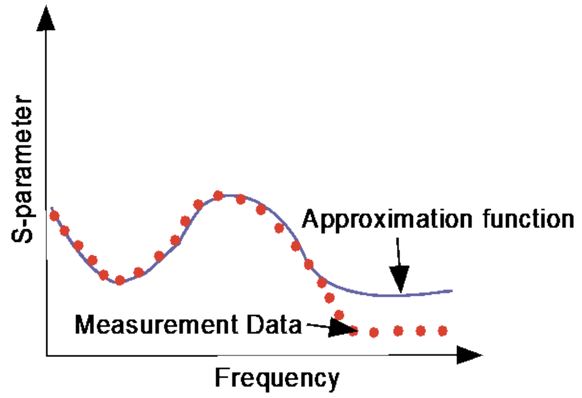


Observation: Good agreement

Orders of Approximation



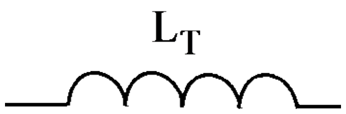


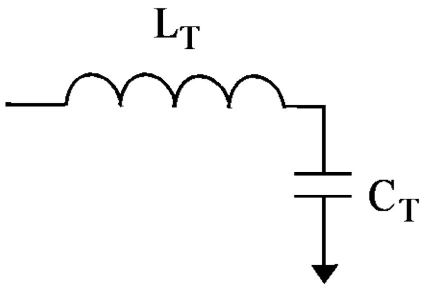


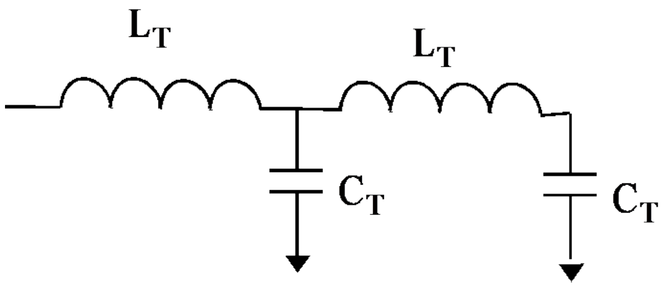
Low order

Medium order

Higher order







MOR Attributes

- **Accurate**:- over wide frequency range.
- **Stable**:- All poles must be in the left-hand side in s-plane or inside in the unit-circle in z-plane.
- **Causal**:- Hilbert transform needs to be satisfied.
- **Passive**:- H(s) is analytic

$$h[n] = h_e[n] + h_o[n] \Leftrightarrow H(j\omega) = H_R(j\omega) + jH_I(j\omega)$$

$$H^*(s) = H(s^*),$$

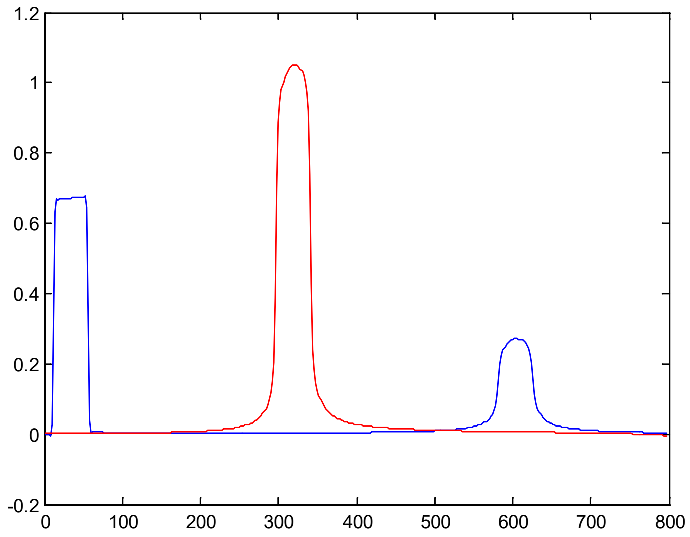
$$z^{*T}[H^T(s^*) + H(s)]z \geq 0, \quad \mathbb{R}[s] > 0$$,for Y or Z-parameters.

$$I - H^T(s^*)H(s) \geq 0, \quad \mathbb{R}[s] > 0$$,for S-parameters.

MOR Problems

- **Bandwidth**
  - Low-frequency data must be added
- **Passivity**
  - Passivity enforcement
- **High Order of Approximation**
  - Orders > 800 for some serial links
  - Delay need to be extracted

Causality Violations



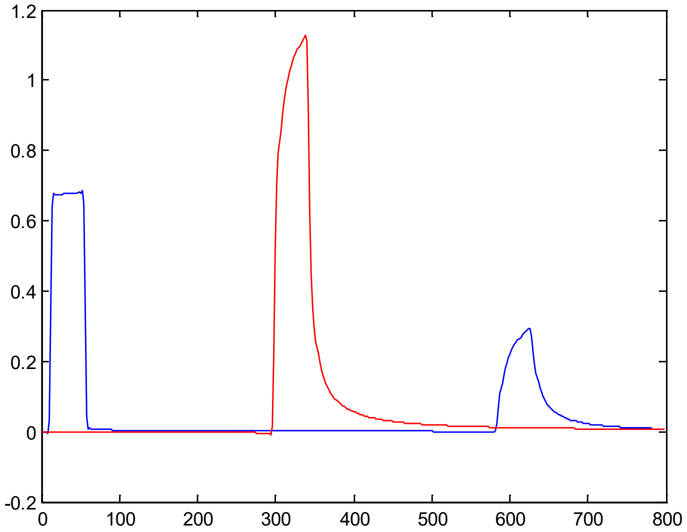
NON-CAUSAL ←

$$Z(f) = R_o\sqrt{f} + jL\omega$$

Near (blue) and Far (red) end responses of lossy TL

CAUSAL →

$$Z(f) = R_o\sqrt{f} + jR_o\sqrt{f} + jL\omega$$



ECE·ILLINOIS
Electrical and Computer Engineering
University of Illinois at Urbana-Champaign

Fourier Transform Pairs

$a_{re}(t)$: real part of even time-domain function
$a_{ie}(t)$: imaginary part of even time-domain function
$a_{ro}(t)$: real part of odd time-domain function
$a_{io}(t)$: imaginary part of odd time-domain function

$$a(t) = a_{re}(t) + ja_{ie}(t) + a_{ro}(t) + ja_{io}(t)$$

In the frequency domain accounting for all the components, we can write:

$A_{RE}(\omega)$: real part of even function in the frequency domain
$A_{IE}(\omega)$: imaginary part of even function in the frequency domain
$A_{RO}(\omega)$: real part of odd function in the frequency domain
$A_{IO}(\omega)$: imaginary part of odd function in the frequency domain

$$A(\omega) = A_{RE}(\omega) + jA_{IE}(\omega) + A_{RO}(\omega) + jA_{IO}(\omega)$$

Fourier Transform Pairs

We also have the Fourier-transform-pair relationships:

$$\text{Time Domain}: \quad a(t) = a_{re}(t) + ja_{ie}(t) + a_{ro}(t) + ja_{io}(t)$$

$$\begin{matrix} \updownarrow & \updownarrow & \updownarrow & \nwarrow \nearrow \\ & & & \swarrow \searrow \end{matrix}$$

$$\text{Freq Domain}: \quad A(\omega) = A_{RE}(\omega) + jA_{IE}(\omega) + A_{RO}(\omega) + jA_{IO}(\omega)$$

$$B(\omega) = S(\omega)\left[A_{RE}(\omega) + jA_{IE}(\omega) + A_{RO}(\omega) + jA_{IO}(\omega)\right]$$

In the time domain, this corresponds to:

$$b(t) = s(t) * \left[\left(a_{re}(t) + a_{ro}(t)\right) + j\left(a_{ie}(t) + a_{io}(t)\right)\right]$$

Fourier Transform Pairs

We now impose the restriction that in the time domain, the function must be real. As a result,

$$a_{ie}(t) = a_{io}(t) = 0 \quad \text{which implies that:} \quad A_{IE}(\omega) = A_{RO}(\omega) = 0$$

The Fourier-transform pair relationship then becomes:

$$\begin{array}{lccccc} \text{Time Domain}: & a(t) = & a_{re}(t) & + & a_{ro}(t) \\ & \updownarrow & \updownarrow & & \updownarrow \\ \text{Freq Domain}: & A(\omega) = & A_{RE}(\omega) & + & jA_{IO}(\omega) \end{array}$$

The frequency-domain relations reduce to:

$$B(\omega) = S(\omega)\left[A_{RE}(\omega) + jA_{IO}(\omega)\right]$$

Fourier Transform Pairs

In summary, the general relationship is:

$$\begin{array}{lccccccccc}
\text{Time Domain}: & b(t) = & b_{re}(t) & + & jb_{ie}(t) & + & b_{ro}(t) & + & jb_{io}(t) \\
 & \updownarrow & \updownarrow & & \updownarrow & & \nwarrow\ \nearrow & & \\
 & & & & & & \swarrow\ \searrow & & \\
\text{Freq Domain}: & B(\omega) = & B_{RE}(\omega) & + & jB_{IE}(\omega) & + & B_{RO}(\omega) & + & jB_{IO}(\omega)
\end{array}$$

But for a real system:

$$\begin{array}{lccccccccc}
\text{Time Domain}: & b(t) = & b_{re}(t) & + & jb_{ie}(t) & + & b_{ro}(t) & + & jb_{io}(t) \\
 & \updownarrow & \updownarrow & & \updownarrow & & \nwarrow\ \nearrow & & \\
 & & & & & & \swarrow\ \searrow & & \\
\text{Freq Domain}: & B(\omega) = & B_{RE}(\omega) & + & jB_{IE}(\omega) & + & B_{RO}(\omega) & + & jB_{IO}(\omega)
\end{array}$$

Fourier Transform Pairs

So, in summary

$$\text{Time Domain: } \quad b(t) = b_e(t) + b_o(t)$$

$$\updownarrow \qquad \updownarrow \qquad \updownarrow$$

$$\text{Freq Domain: } \quad B(\omega) = B_R(\omega) + jB_I(\omega)$$

The real part of the frequency-domain transfer function is associated with the even part of the time-domain response

The imaginary part of the frequency-domain transfer function is associated with the odd part of the time-domain response

Causality Principle

Consider a function *h(t)*

$$h(t) = 0, \quad t < 0$$

Every function can be considered as the sum of an even function and an odd function

$$h(t) = h_e(t) + h_o(t)$$

$$h_e(t) = \frac{1}{2}\left[h(t) + h(-t)\right]$$ **Even function**

$$h_o(t) = \frac{1}{2}\left[h(t) - h(-t)\right]$$ **Odd function**

$$h_o(t) = \begin{cases} h_e(t), & t > 0 \\ -h_e(t), & t < 0 \end{cases}$$

$$h_o(t) = \text{sgn}(t) h_e(t)$$

Hilbert Transform

$$h(t) = h_e(t) + \text{sgn}(t) h_e(t)$$

In frequency domain this becomes

$$H(f) = H_e(f) + \frac{1}{j\pi f} * H_e(f)$$

$$H(f) = H_e(f) - j\hat{H}_e(f)$$

➔Imaginary part of transfer function is related to the real part through the Hilbert transform

$\hat{H}_e(f)$ is the Hilbert transform of $H_e(f)$

$$\hat{x}(t) = x(t) * \frac{1}{\pi t} = \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{x(\tau)}{t - \tau} d\tau$$

Discrete Hilbert Transform

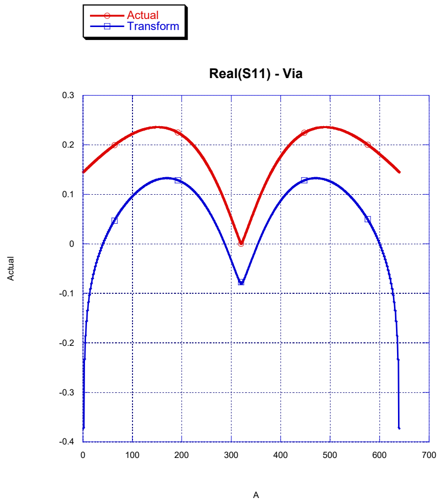
➔Imaginary part of transfer function cab be recovered from the real part through the Hilbert transform

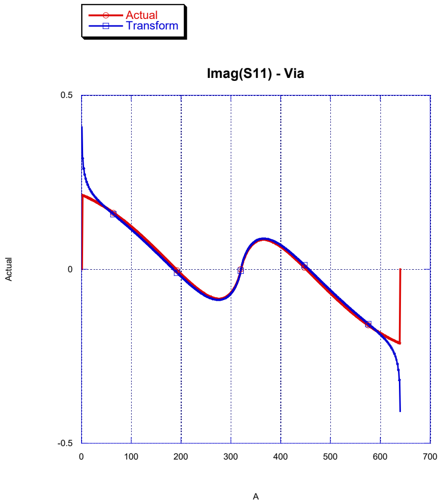
➔If frequency-domain data is discrete, use discrete Hilbert Transform (DHT)*

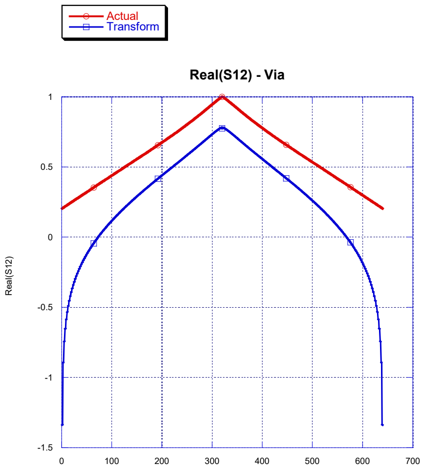
$$H(f_n) = \hat{f}_k = \begin{cases} \dfrac{2}{\pi} \displaystyle\sum_{n \text{ odd}} \dfrac{f_n}{k-n}, & k \text{ even} \\ \dfrac{2}{\pi} \displaystyle\sum_{n \text{ even}} \dfrac{f_n}{k-n}, & k \text{ odd} \end{cases}$$

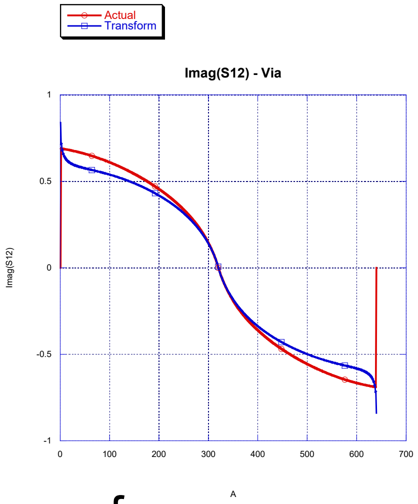
*S. C. Kak, "The Discrete Hilbert Transform", Proceedings of the IEEE, pp. 585-586, April 1970.

HT for Via: 1 MHz – 20 GHz





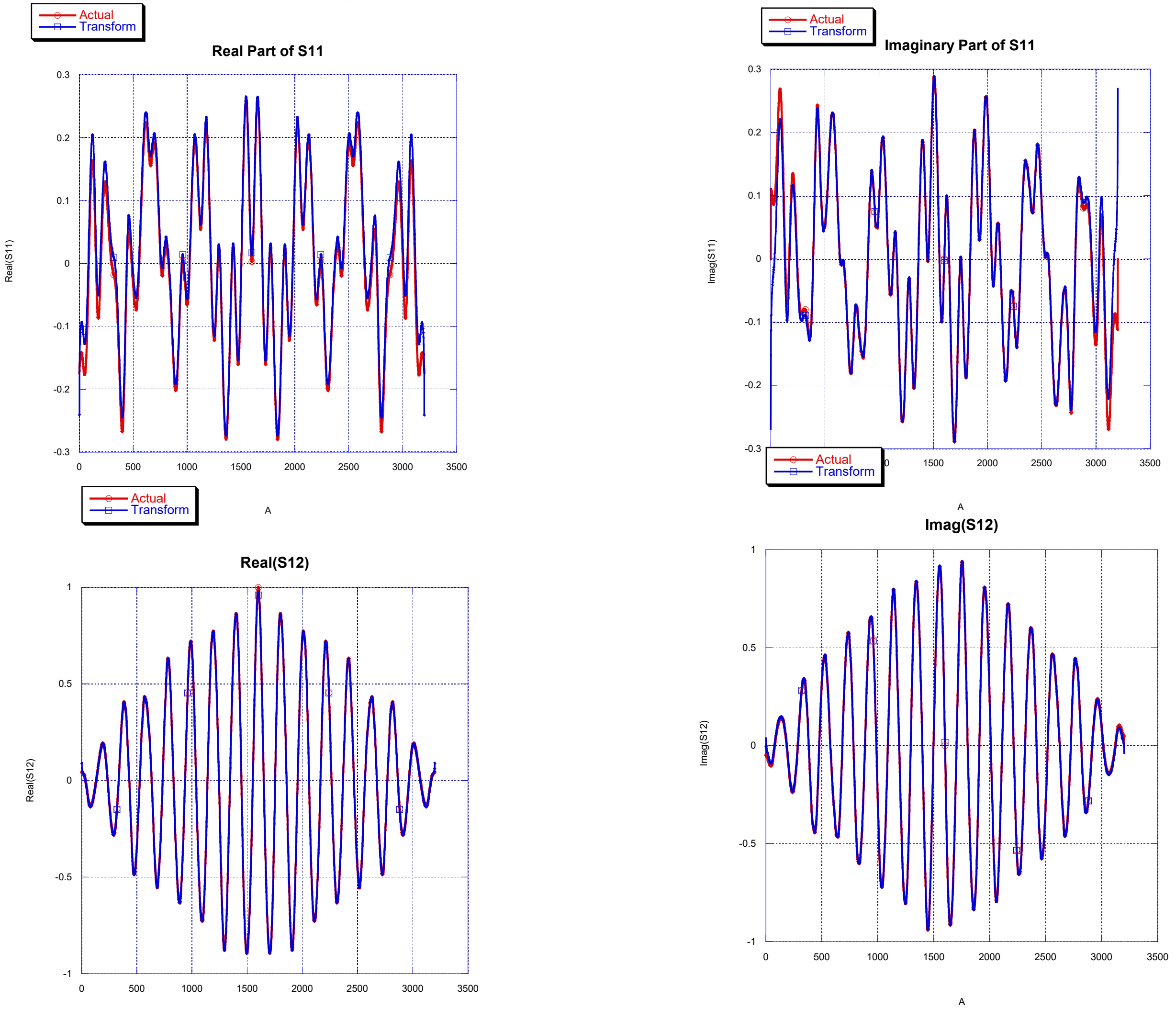




Actual is red, **HT is blue**

Observation: Poor agreement (because frequency range is limited)

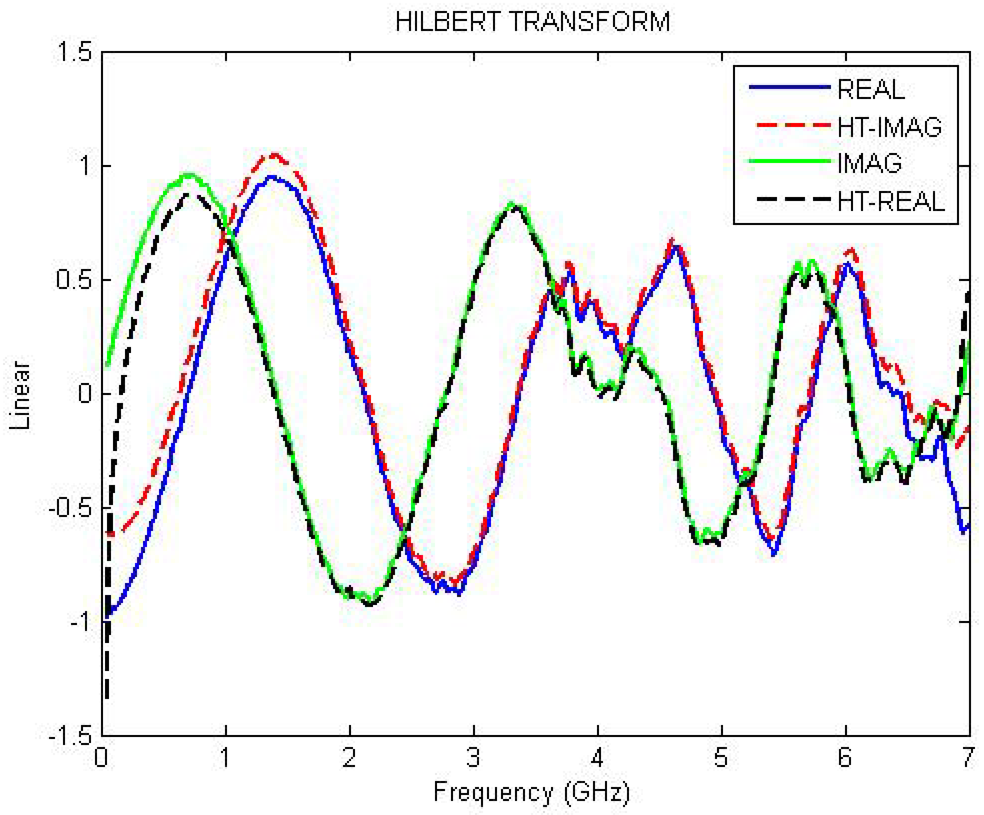
Example: 300 KHz – 6 GHz



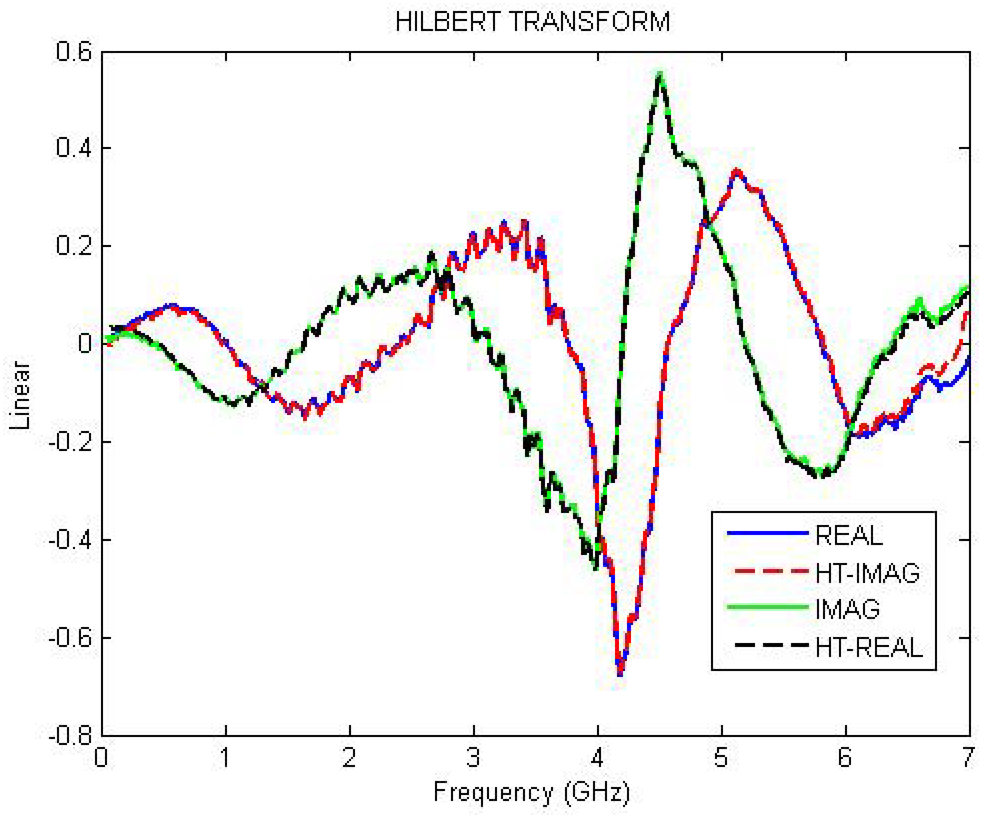
Actual is red, **HT is blue**

Observation: Good agreement

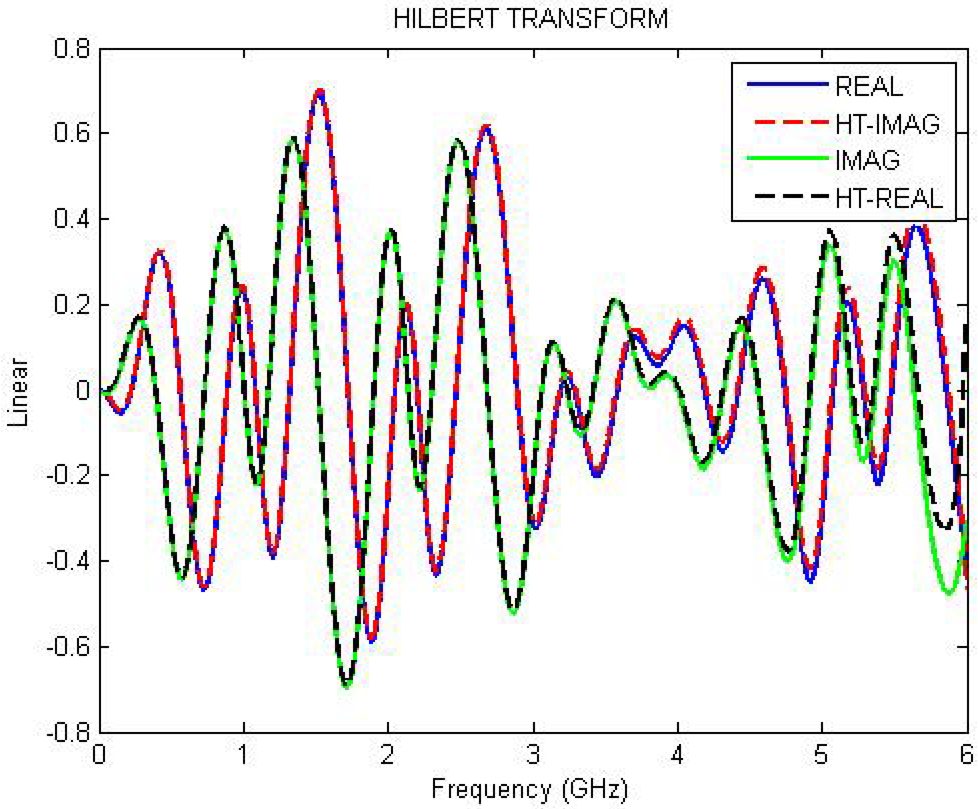
Microstrip Line S11



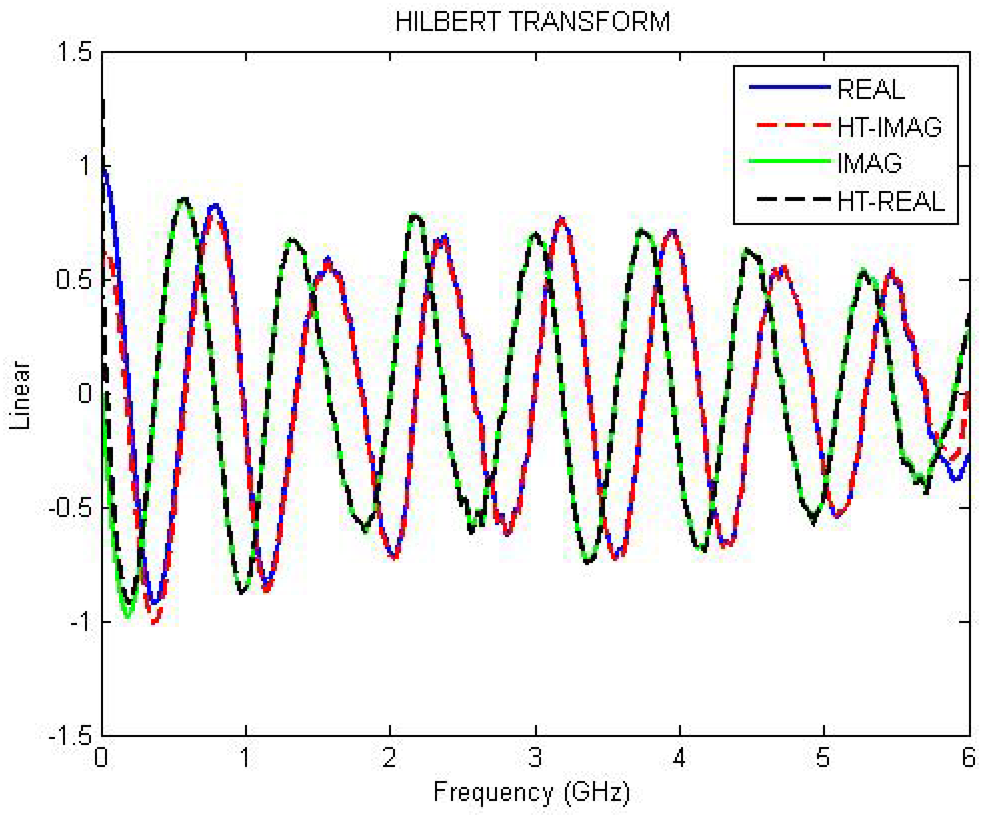
Microstrip Line S21



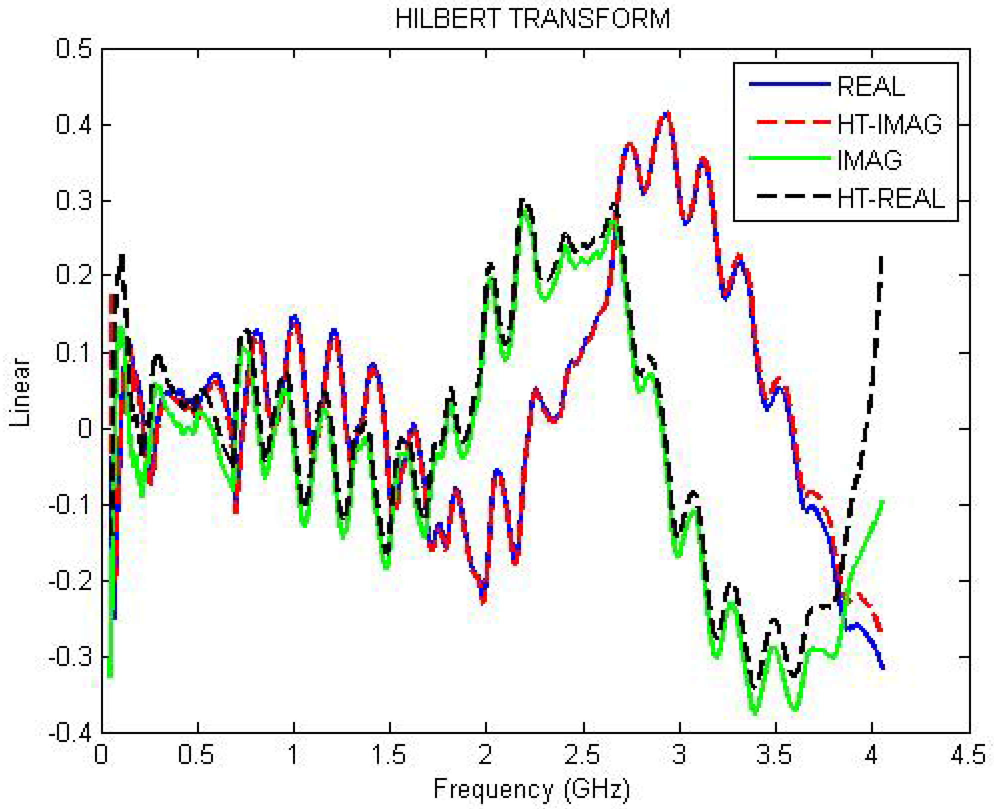
Discontinuity S11



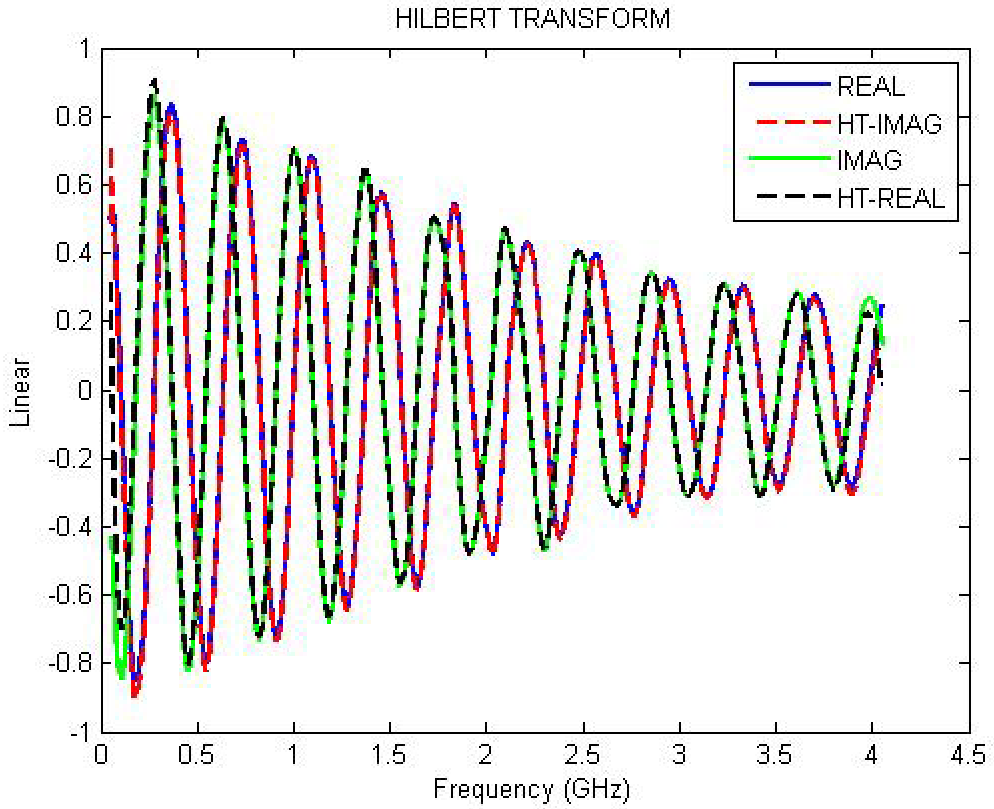
Discontinuity S21



Backplane S11



Backplane S21



HT of Minimum Phase System

$$|H_{ij}(s)| = |M_{ij}(s)||P_{ij}(s)||e^{-s\tau_{ij}}|$$

$$H(j\omega) = H_{\min}(j\omega) \times H_a(j\omega)$$

$$|P_{ij}(j\omega)| = |e^{-j\omega\tau_{ij}}| = 1$$

$$H_a(j\omega) = H_b(j\omega) \times e^{-j\omega\tau}$$

$$s = j\omega$$

$$|H_{ij}(j\omega)| = |M_{ij}(j\omega)| H_a(j\omega) = H_b(j\omega) \times e^{-j\omega\tau}$$

The phase of a minimum phase system can be completely determined by its magnitude via the Hilbert transform

$$\arg[M_{ij}(\omega)] = \frac{2\omega}{\pi} \int_0^{\infty} \frac{U(\xi) - U(\omega)}{(\xi + \omega)(\xi - \omega)} d\xi$$

$$U(\omega) = \ln|M_{ij}(\omega)| = \ln|H_{ij}(\omega)|$$

HT of Minimum Phase System

$$H(j\omega) = H_{\min}(j\omega) \times H_a(j\omega)$$

$$H_a(j\omega) = H_b(j\omega) \times e^{-j\omega\tau}$$

$$\left|H_b(j\omega)\right| = \left|e^{-j\omega\tau}\right| = 1$$

$$\left|H(j\omega)\right| = \left|H_{\min}(j\omega)\right|$$

The phase of a minimum phase system can be completely determined by its magnitude via the Hilbert transform

$$\arg\left[H(j\omega)\right] = -\mathcal{H}\left\{\ln\left|H(j\omega)\right|\right\}$$

where $\mathcal{H}\{\bullet\}$ Is the Hilbert transform

Enforcing Causality in TL

The complex phase shift of a lossy transmission line

$$X = e^{-\gamma l} = e^{-\sqrt{(R+j\omega L)(G+j\omega C)}l}$$ is non causal

We assume that

$$e^{-j\phi(\omega)}e^{-\alpha(\omega)} = e^{+j\omega\sqrt{LC}l}e^{-\sqrt{(R+j\omega L)(G+j\omega C)}l}$$ is minimum phase non causal

$$HT\left\{\ln\left|e^{-j\phi(\omega)}e^{-\alpha(\omega)}\right|\right\} = HT\left\{\ln\left|e^{-\gamma l}\right|\right\} = -\phi^{'}(\omega)$$

$$e^{-j\phi^{'}(\omega)}e^{-\alpha(\omega)}$$ is minimum phase and causal

$$X^{'} = e^{-j\phi^{'}(\omega)}e^{-\alpha(\omega)}e^{-j\omega\sqrt{LC}l}$$ is the causal phase shift of the TL

In essence, we keep the magnitude of the propagation function of the TL but we calculate/correct for the phase via the Hilbert transform.

Passivity Enforcement Techniques

➔ Hamiltonian Perturbation Method [1]

➔ Residue Perturbation Method [2]

(1) S. Grivet-Talocia, "Passivity enforcement via perturbation of Hamiltonian matrices," *IEEE Trans. Circuits Syst. I*, vol. 51, no. 9, pp. 1755-1769, Sep. 2004.

(2) D. Saraswat, R. Achar, and M. Nakhla, "A fast algorithm and practical considerations for passive macromodeling of measured/simulated data," *IEEE Trans. Adv. Packag.*, vol. 27, no. 1, pp. 57–70, Feb. 2004.

Passivity Assessment

Can be done using S parameter Matrix

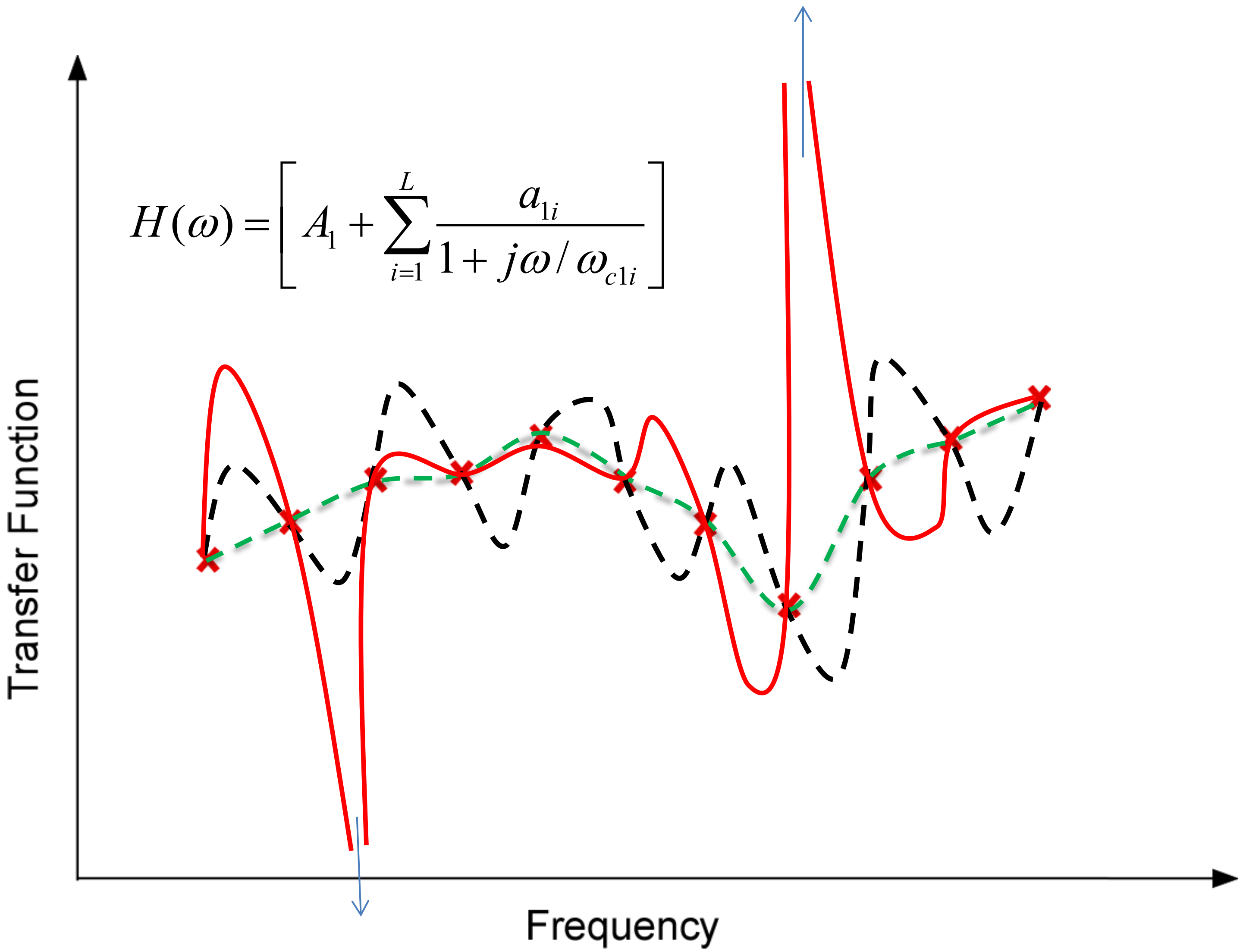
$$\boldsymbol{D} = \left(\boldsymbol{1} - \boldsymbol{S}^{*T}\boldsymbol{S}\right) = Dissipation\ Matrix$$

All the eigenvalues of the dissipation matrix must be greater than 0 at each sampled frequency points.

This assessment method is not very robust since it may miss local nonpassive frequency points between sampled points.

➔ **Use Hamiltonian from State Space Representation**

MOR and Passivity



State-Space Representation

The State space representation of the transfer function is given by

$$\dot{x}(t) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t) + Du(t)$$

The transfer function is given by

$$S(s) = C\left(sI - A\right)^{-1} B + D$$

Procedure

- **Approximate all N^2 scattering parameters using Vector Fitting**
- **Form Matrices A, B, C and D for each approximated scattering parameter**
- **Form A, B, C and D matrices for complete N-port**
- **Form Hamiltonian Matrix H**

Constructing A_{ii}

Matrix A_{ii} is formed by using the poles of S_{ii}. The poles are arranged in the diagonal.

$$
\mathbf{A}_{ii} = \begin{pmatrix} a_1^{(ii)} & b_1^{(ii)} & 0 & 0 \\ -b_1^{(ii)} & a_1^{(ii)} & 0 & 0 \\ 0 & 0 & \bullet & \bullet \\ 0 & 0 & \bullet & a_L^{(ii)} \end{pmatrix}
$$

Complex poles are arranged with their complex conjugates with the imaginary part placed as shown.

A_{ii} is an $L \times L$ matrix

Constructing C_{ij}

Vector C_{ij} is formed by using the residues of S_{ij}.

$$\mathbf{C_{ij}} = \begin{pmatrix} c_1^{(ij)} & c_2^{(ij)} & \bullet & c_N^{(ij)} \end{pmatrix}$$

where $c_k^{(ij)}$ is the kth residue resulting from the Lth order approximation of S_{ij}

C_{ij} is a vector of length L

Constructing B_{ii}

For each real pole, we have an entry with a 1

For each complex conjugate pole, pair we have two entries as: $\begin{pmatrix} 2 \\ 0 \end{pmatrix}$

$$B_{ii} = \begin{pmatrix} 1 \\ 1 \\ 2 \\ 0 \\ \bullet \\ 1 \\ \\ \end{pmatrix}$$

B_{ii} is a vector of length L

Constructing D_{ij}

D_{ij} **is a scalar which is the constant term from the Vector fitting approximation:**

$$S_{ij} \simeq d_{ij} + \sum_{k=1}^{L} \frac{c_k^{(ij)}}{s - a_k^{(ii)}}$$

$$\boldsymbol{D_{ij}} = d_{ij}$$

Constructing *A*

Matrix A for the complete N-port is formed by combining the A_{ii}'s in the diagonal.

$$\boldsymbol{A} = \begin{pmatrix} A_{11} & & & \\ & A_{22} & & \\ & & \bullet & \\ & & & A_{NN} \end{pmatrix}$$

A is a $NL \times NL$ matrix

Constructing *C*

Matrix *C* for the complete N-port is formed by combining the C_{ij}'s.

$$\boldsymbol{C} = \begin{pmatrix} C_{11} & C_{12} & \bullet & \\ C_{21} & C_{22} & & \\ \bullet & & \bullet & \\ & & & C_{NN} \end{pmatrix}$$

C is a $N \times NL$ matrix

Constructing *B*

Matrix B for the complete two-port is formed by combining the B_{ii}'s.

$$\boldsymbol{B} = \begin{pmatrix} B_{11} & 0 & 0 & \\ 0 & B_{22} & 0 & \\ \bullet & & \bullet & \\ 0 & & & B_{NN} \end{pmatrix}$$

B is a $NL \times N$ matrix

Hamiltonian

Construct Hamiltonian Matrix *M*

$$M = \begin{bmatrix} A - BR^{-1}D^TC & -BR^{-1}B^T \\ C^TS^{-1}C & -A^T + C^TDR^{-1}B^T \end{bmatrix}$$

$$R = \left(D^TD - I\right) \quad and \quad S = \left(DD^T - I\right)$$

The system is passive if *M* has no purely imaginary eigenvalues

If imaginary eigenvalues are found, they define the crossover frequencies ($j\omega$) at which the system switches from passive to non-passive (or vice versa)

➔ **gives frequency bands where passivity is violated**

Perturb Hamiltonian

Perturb the Hamiltonian Matrix ***M*** by perturbing the pole matrix ***A***

$$A \to A' = A + \Delta A$$

$$M + \Delta M = \begin{bmatrix} A + \Delta A - B\left(D + D^T\right)^{-1} C & B\left(D + D^T\right)^{-1} B^T \\ -C^T\left(D + D^T\right)^{-1} C & -\left(A + \Delta A\right)^T + C^T\left(D + D^T\right)^{-1} B^T \end{bmatrix}$$

$$\Delta M = \begin{bmatrix} \Delta A & 0 \\ 0 & -\left(\Delta A\right)^T \end{bmatrix}$$

This will lead to a change of the state matrix:

$$A \to A' = A + \Delta A$$

State-Space Representation

The State space representation of the transfer function in the time domain is given by

$$\dot{x}(t) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t) + Du(t)$$

The solution in discrete time is given by

$$x[k+1] = A_d x[k] + B_d u[k]$$

$$y[k] = C_d x[k] + D_d u[k]$$

State-Space Representation

where

$$\mathbf{A_d} = e^{AT} \qquad \mathbf{B_d} = \left(\int_0^T e^{A\tau} d\tau \right) \mathbf{B}$$

$$\mathbf{C_d} = \mathbf{C} \qquad \mathbf{D_d} = \mathbf{D}$$

which can be calculated in a straightforward manner

When ***y(t)***➔***b(t)*** is combined with the terminal conditions, the complete blackbox problem is solved.

State-Space Passive Solution

If M' is passive, then the state-space solution using A' will be passive.

$$\boldsymbol{A} \rightarrow \boldsymbol{A}' = \boldsymbol{A} + \boldsymbol{\Delta A}$$

The *passive* solution in discrete time is given by

$$\boldsymbol{x}[k+1] = \boldsymbol{A}_d' \boldsymbol{x}[k] + \boldsymbol{B}_d \boldsymbol{u}[k]$$

$$\boldsymbol{y}[k] = \boldsymbol{C}_d \boldsymbol{x}[k] + \boldsymbol{D}_d \boldsymbol{u}[k]$$

Size of Hamiltonian

$$M = \begin{bmatrix} A - BR^{-1}D^T C & -BR^{-1}B^T \\ C^T S^{-1} C & -A^T + C^T DR^{-1}B^T \end{bmatrix}$$

M has dimension $2NL$

For a 20-port circuit with VF order of 40, M will be of dimension $2\times 40 \times 20 = 1600$

The matrix M has dimensions 1600×1600

Too Large !

➔ **Eigen-analysis of this matrix is prohibitive**

Example

$$A = \begin{bmatrix} -10 & 0 & 0 \\ 0 & -1 & 100 \\ 0 & -100 & -1 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 1 & 0.1 \end{bmatrix}$$

$$D = \begin{bmatrix} 10^{-5} \end{bmatrix}$$

This macromodel is nonpassive between 99.923 and 100.11 radians

Example

$$A' = \begin{bmatrix} -10 & 0 & 0 \\ 0 & -1-0.005 & 100 \\ 0 & -100 & -1-0.005 \end{bmatrix}$$

The Hamiltonian M' associated with A' has no pure imaginary ➔ System is passive

Benchmarks*

Data file	No. of points	MOR with Vector Fitting					Fast Convolution
		Order	Time (s)				Time (s)
			VFIT‡	Passivity Enforcement	Recursive Convolution#	TOTAL	
Blackbox 1	501	10*	0.14	0.01NV	0.02	0.17	0.078
Blackbox 2	802	20*	0.41	5.47	0.03	5.91	0.110
Blackbox 3	802	40*	1.08	0.08NV	0.06	1.22	0.125
Blackbox 4	802	60*	2.25	1.89	0.09	4.23	0.125
		100	3.17	5.34	0.16	8.67	
Blackbox 5	2002	50*	4.97	0.09NV	0.28	5.34	0.328
Blackbox 6	802	100*	3.17	0.56NV	0.16	3.89	0.109
Blackbox 7	1601	100*	24.59	28.33	1.31	53.23	0.438
		120	31.16	27.64	1.58	60.38	
Blackbox 8	5096	220	250.08	25.77NV	10.05	285.90	2.687
Blackbox 9	1601	200*	58.47	91.63	2.59	152.69	0.469
		250	80.64	122.83	3.22	206.69	
		300	106.53	61.58NV	3.86	171.97	

* J. E. Schutt-Aine, P. Goh, Y. Mekonnen, Jilin Tan, F. Al-Hawari, Ping Liu; Wenliang Dai, "Comparative Study of Convolution and Order Reduction Techniques for Blackbox Macromodeling Using Scattering Parameters," IEEE Trans. Comp. Packaging. Manuf. Tech., vol. 1, pp. 1642-1650, October 2011.

Passive VF Simulation Code

- ➔ Performs VF with common poles
- ➔ Assessment via Hamiltonian
- ➔ Enforcement: Residue Perturbation Method
- ➔ Simulation: Recursive convolution

Number of Ports	Order	CPU-Time
4-Port	20	1.7 secs
6-port	32	3.69 secs
10-port	34	8.84 secs
20-port	34	33 secs
40	50	142 secs
80	12	255 secs

Passive VF Code - Examples

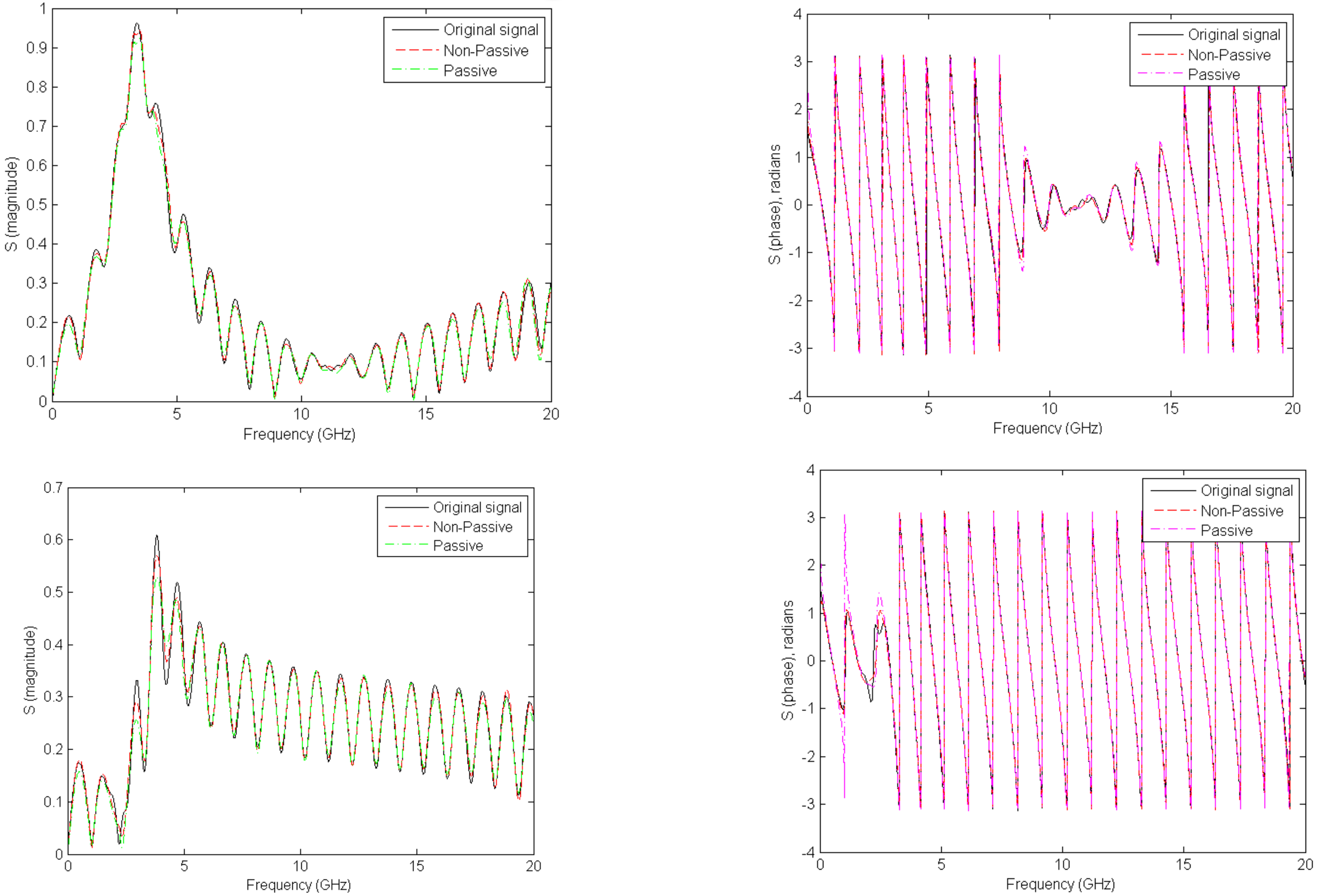
Example 1

4 ports
order = 60

Example 2

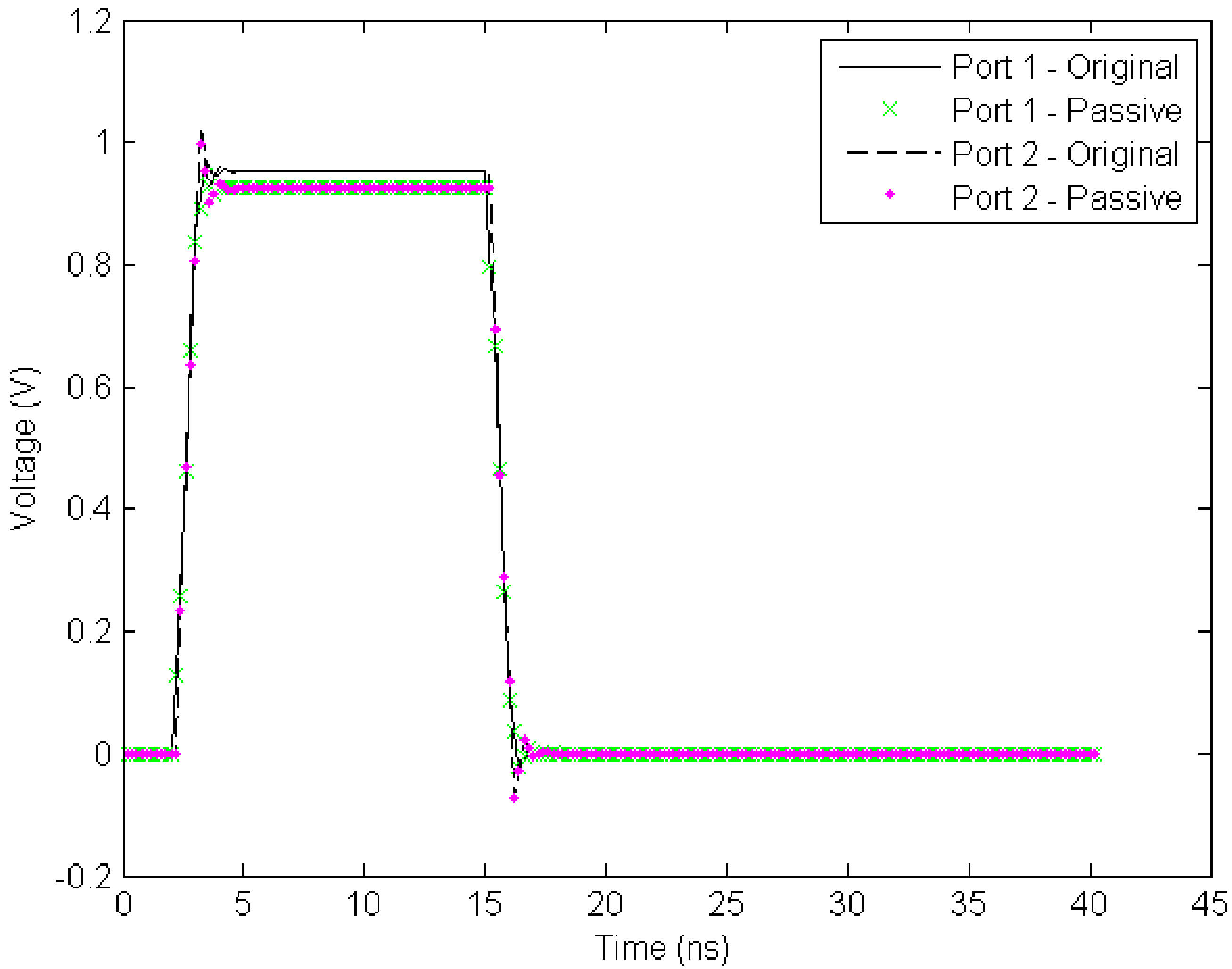
40 ports
order = 50

Passivity Enforced VF

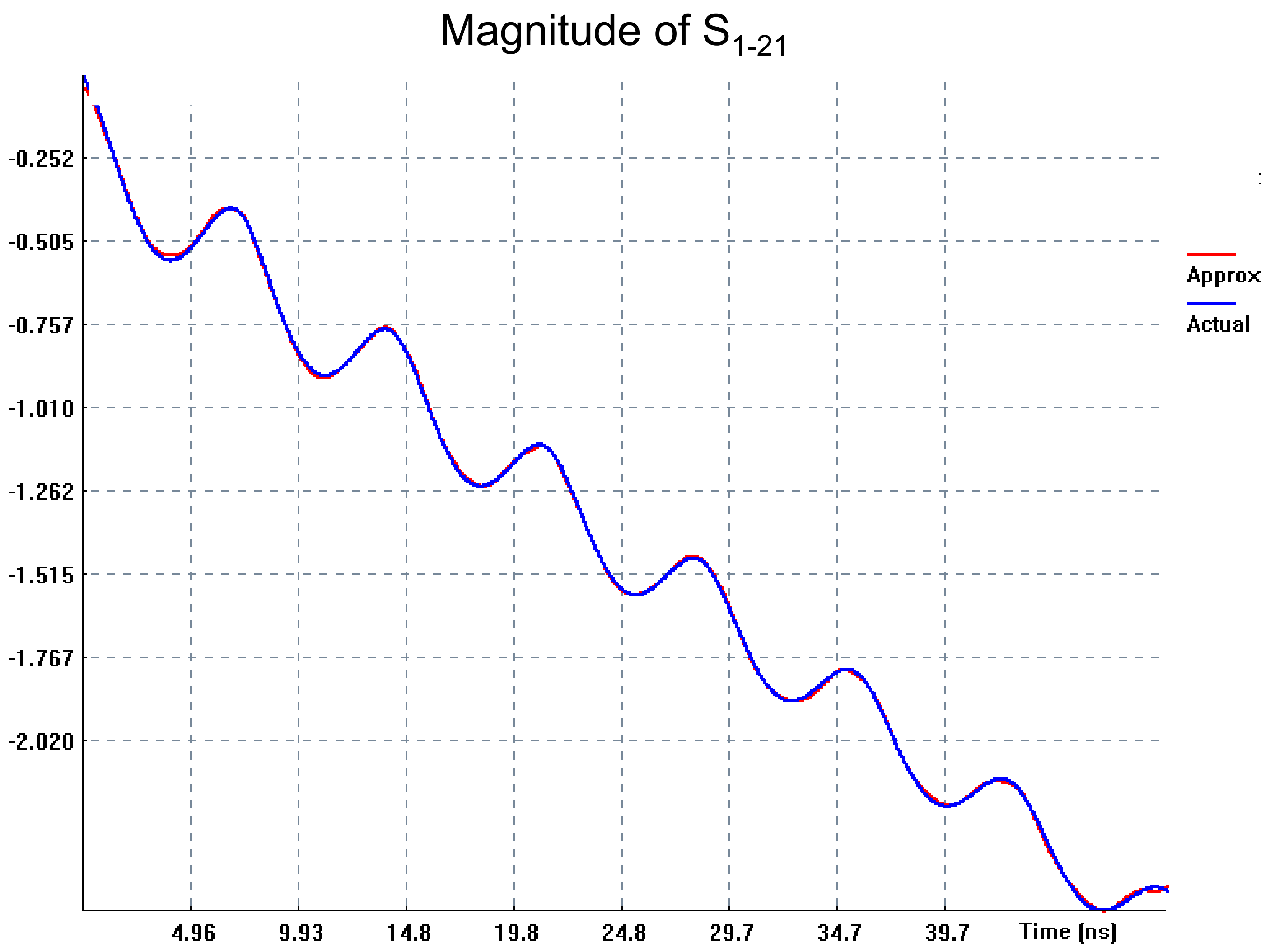


4 ports, 2039 data points - VFIT order = 60 (4 iterations ~6-7mins), Passivity enforcement: 58 Iterations (~1hour)

Passive Time-Domain Simulation

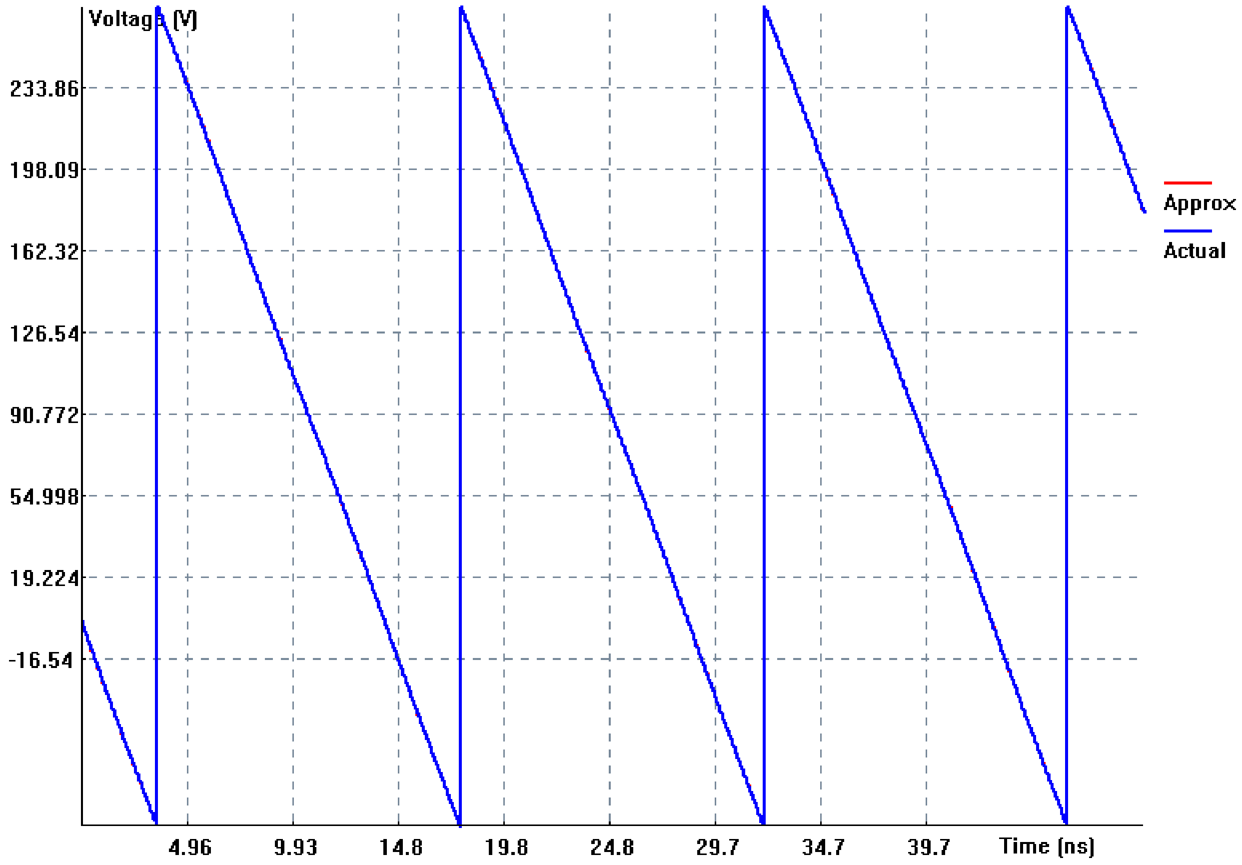


40-Port Passivity Enforced VF



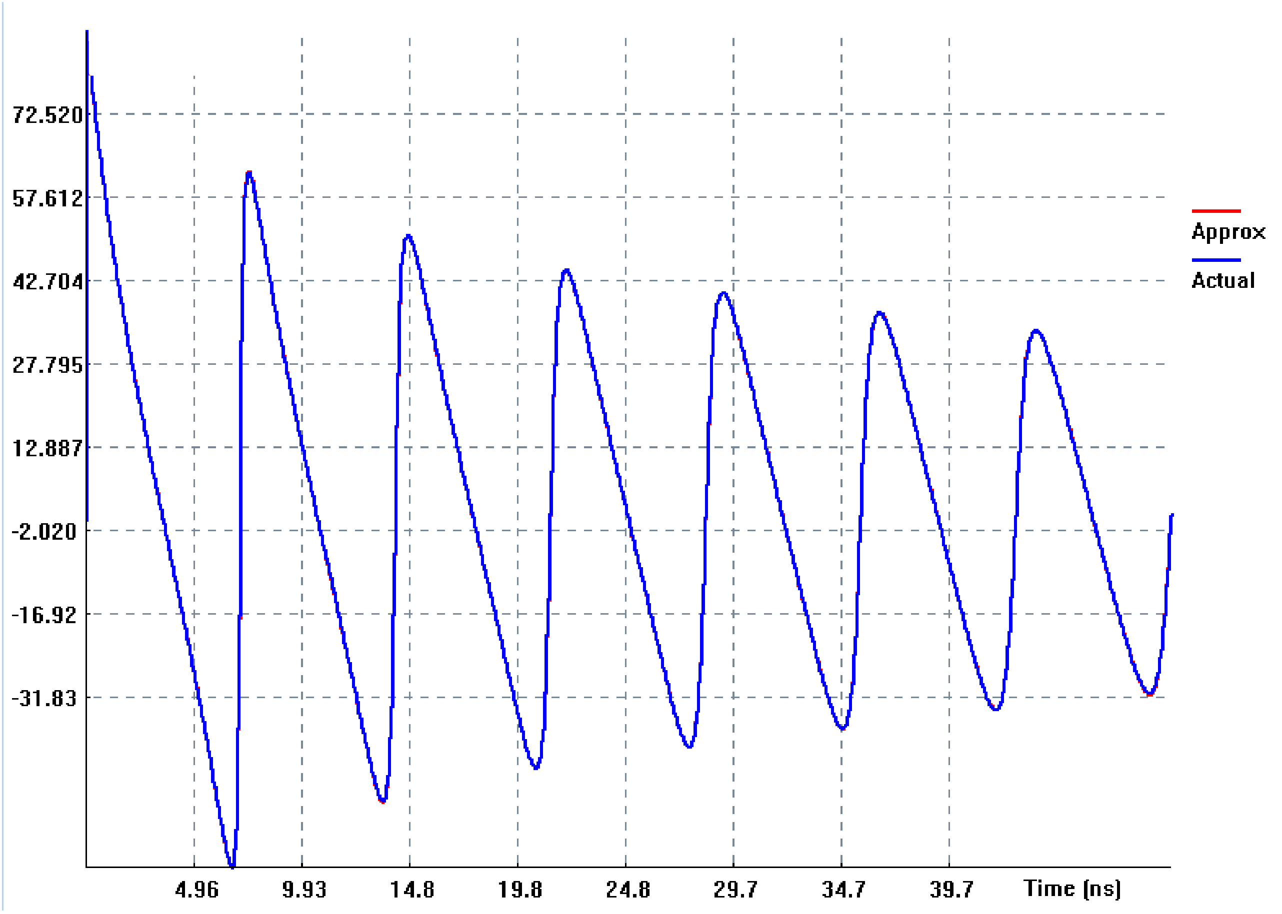
40-Port Passivity Enforced VF

Phase of S_{1-21}



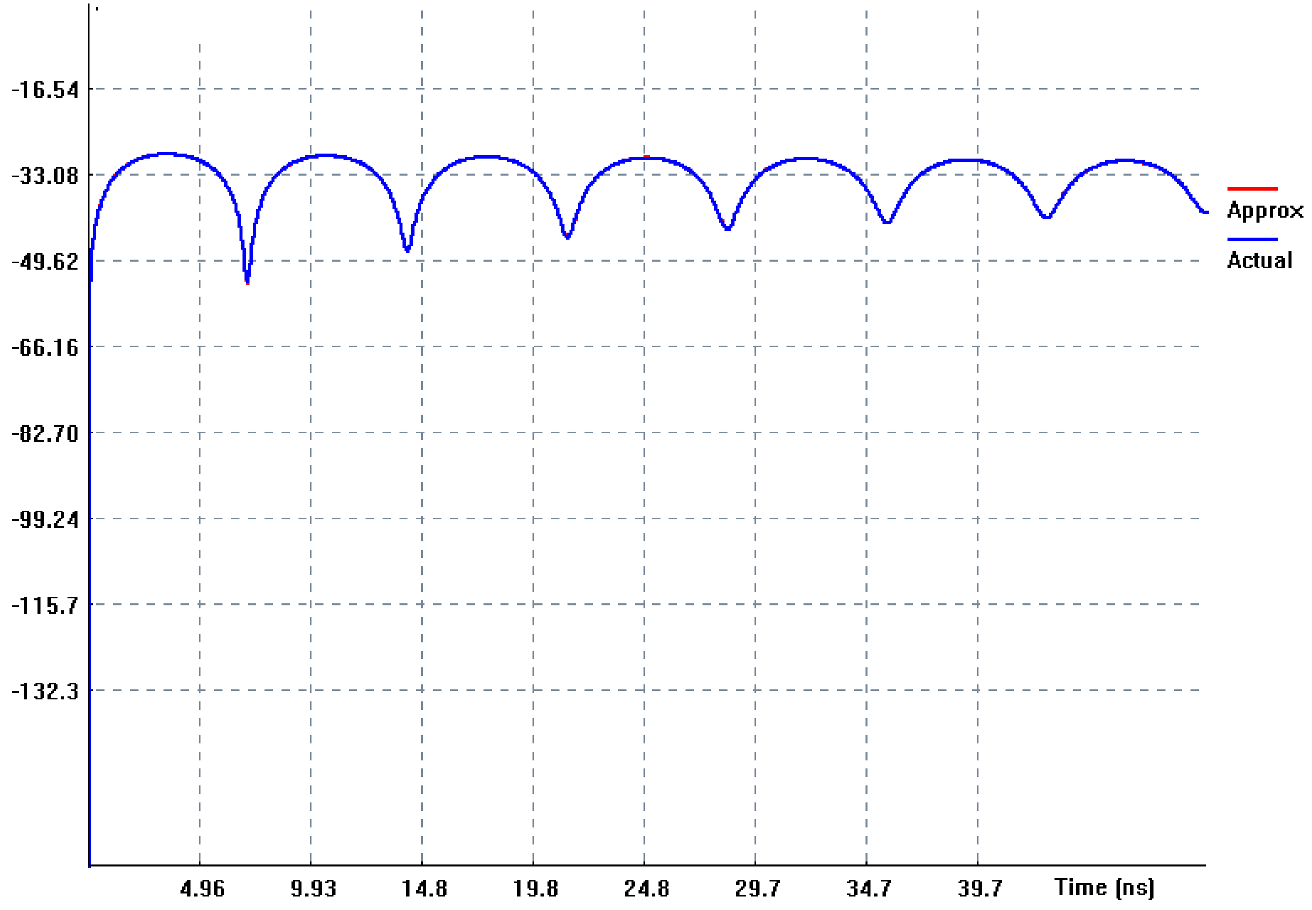
40-Port Passivity Enforced VF

Phase of S_{21}

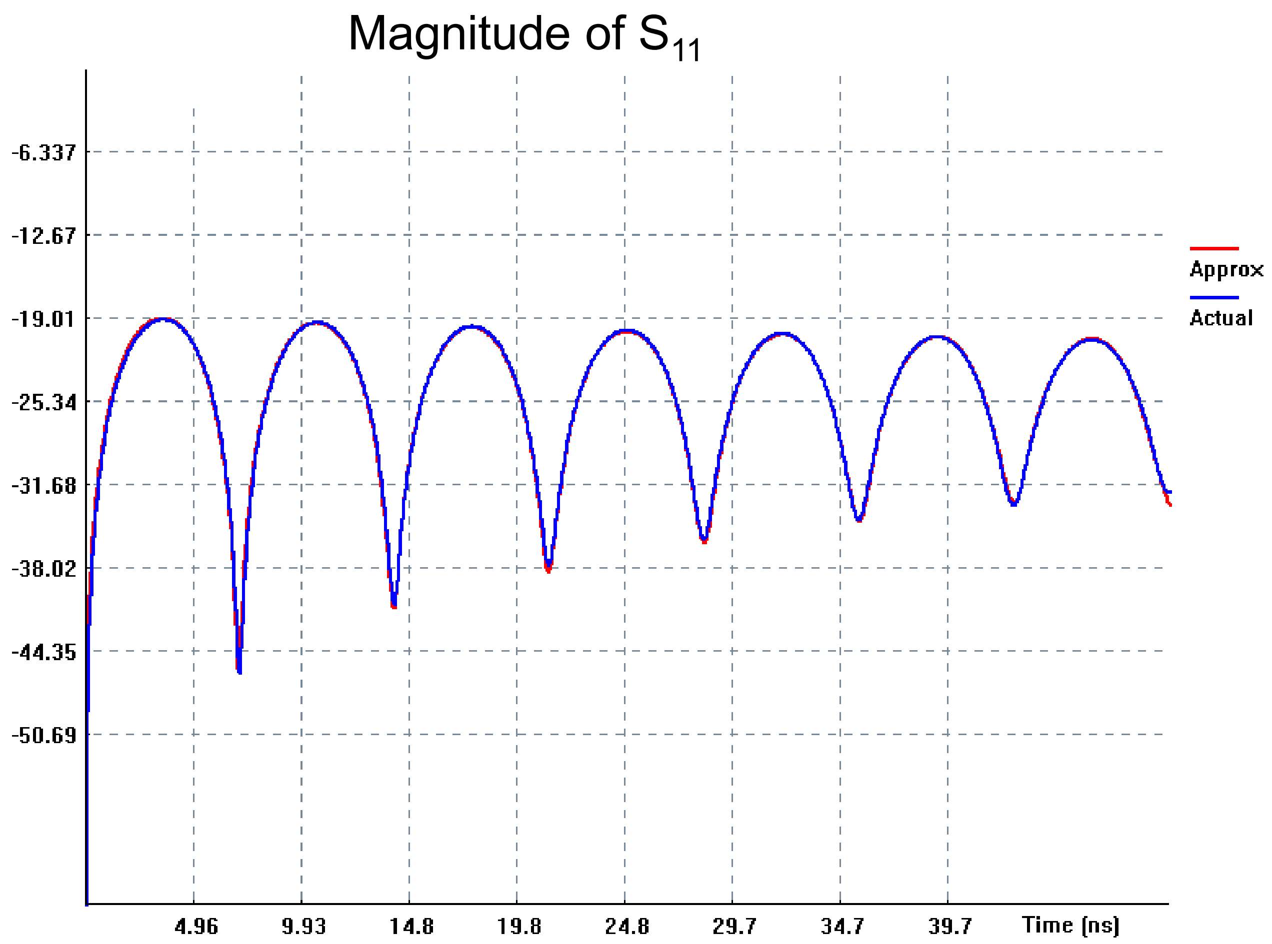


40-Port Passivity Enforced VF

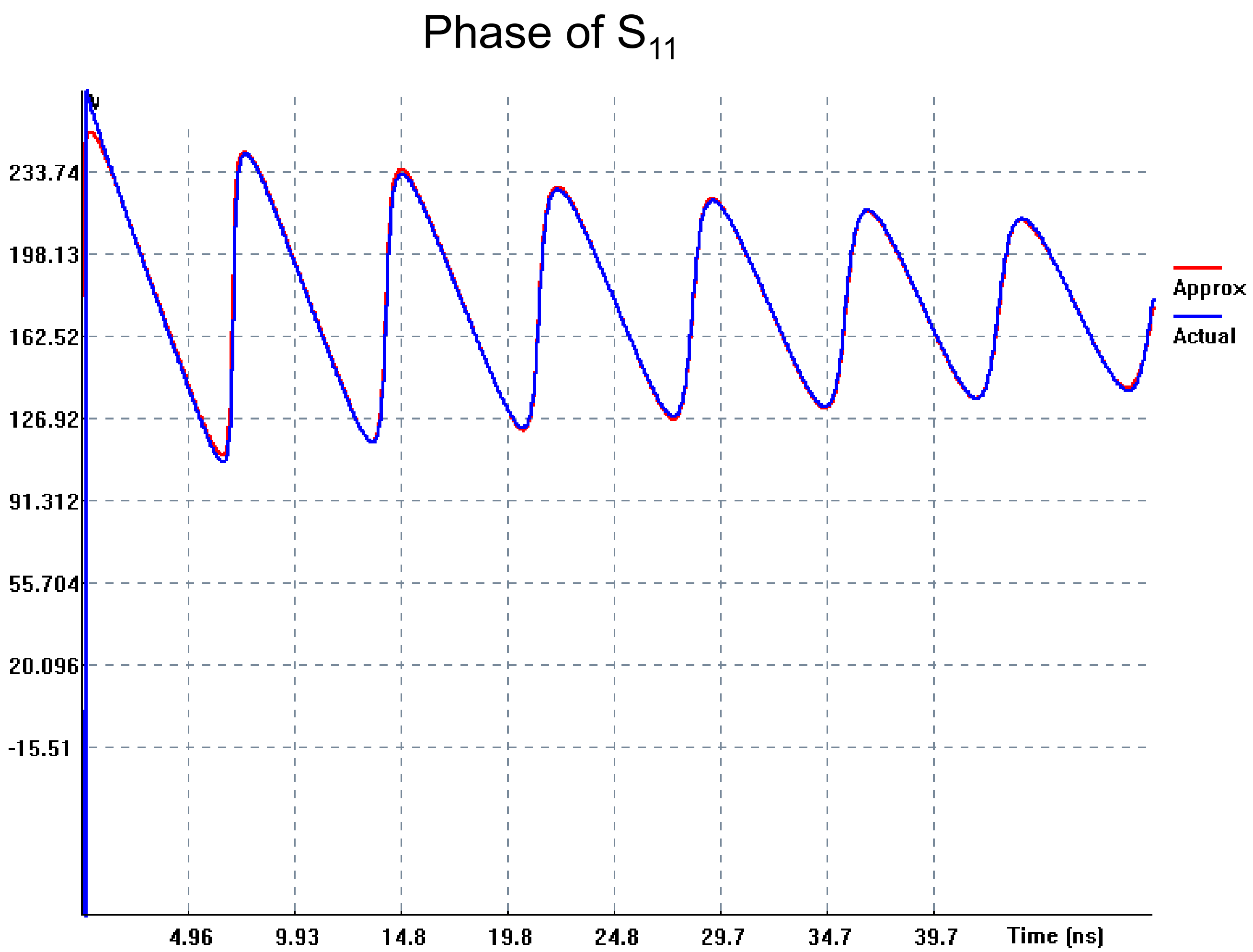
Magnitude of S_{21}



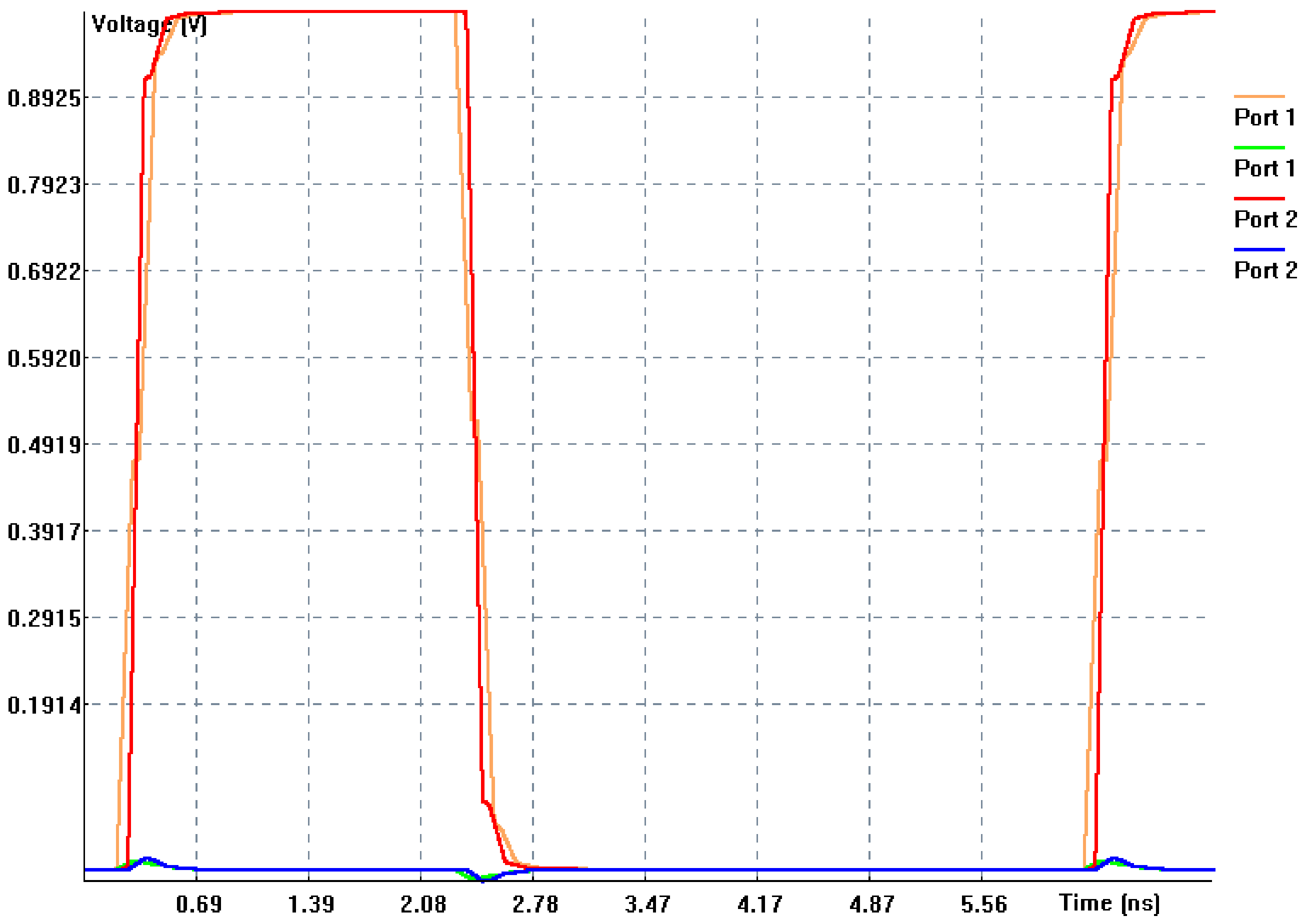
40-Port Passivity Enforced VF



40-Port Passivity Enforced VF



40-Port Time-Domain Simulation



40-Port Time-Domain Simulation