From Field Solvers to Parameterized Reduced Order Models

Luca Daniel, M.I.T.

luca@mit.edu

Thanks to:

Jacob White, M.I.T.

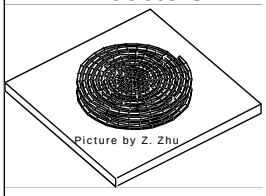
Joel Phillips, Cadence Berkeley Labs,

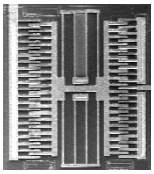
Zhenhai Zhu, IBM T.J.Watson

Electronic Systems on a Integrated Circuit (IC) or on a Multi-Chip Module (MCM)

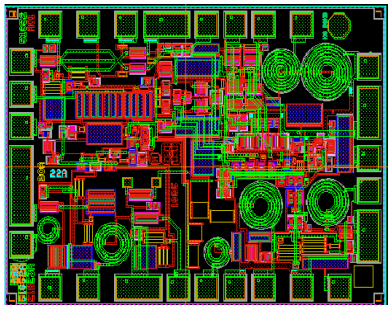
isQED

RF Inductors

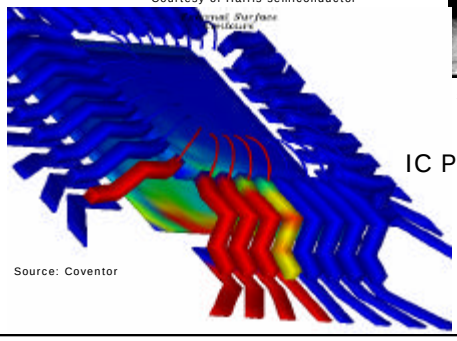




MEM resonators



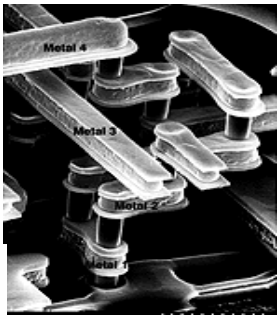
Courtesy of Harris semiconductor



Source: Coventor

IC Package

On-Chip Interconnect and Substrate



Source: Rabaey, Chandrakasan, Nicolic

2

Modern electronic systems consist of several circuit components for instance digital circuits, analog RF or mixed signal circuits, RF inductors, Micro-Electro-Mechanical resonators.

These components are assembled over a semiconductive substrate or over a package (Multi-Chip-Module) and live inside a very complicated network of wires.

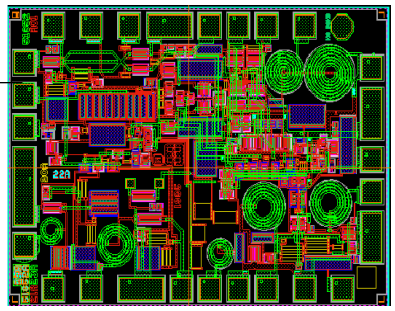
Electronic Systems on a Integrated Circuit (IC) or on a Multi-Chip Module (MCM)

isQED

RF Inductors

$$\nabla \times E = -m \frac{dH}{dt}$$

$$\nabla \times H = e \frac{dE}{dt}$$



Courtesy of Harris semiconductor

On-Chip Interconnect and Substrate

$$\nabla \times E = -m \frac{dH}{dt}$$

$$\nabla \times H = e \frac{dE}{dt}$$

$$EI \frac{\partial^4 u}{\partial x^4} - S \frac{\partial^2 u}{\partial x^2} = F_{elec} + \int_0^w (p - p_a) dy - r \frac{\partial^2 u}{\partial t^2}$$

$$\nabla \cdot ((1 + 6K) u^3 p \nabla p) = 12m \frac{\partial (pu)}{\partial t}$$

MEM resonators

$$\nabla \times E = -m \frac{dH}{dt}$$

$$\nabla \times H = e \frac{dE}{dt}$$

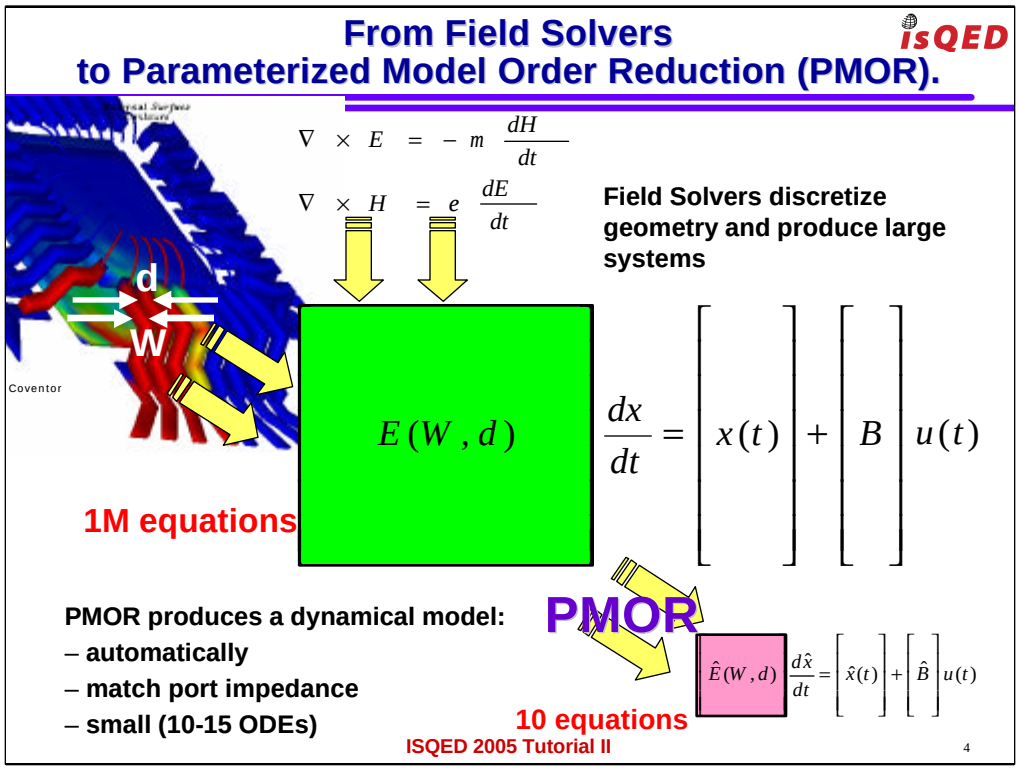
IC Package

ISQED 2005 Tutorial II

3

The designers of these Systems on Chip or Systems on Package are well aware that the performance of their systems depend critically on what they call “second order effects” (e.g. capacitive coupling, inductive coupling, electromagnetic fullwave coupling, skin effect, proximity effect, substrate noise, package resonances.)

These second order effects can be described accurately only starting from the underling partial differential equations (Maxwell, or Navier-stokes).



In the previous talk we have seen how the field solver based parasitic extraction tools can efficiently assemble a very accurate model describing the input out behavior of the system components.

The model typically consist of a set of ordinary differential equations whose coefficients could in general depend on layout parameters such as wire width W and wire separation d.

The task of the Parameterized Model Order Reduction is to produce a dynamical system model automatically, with same input out behavior but much smaller number of ODE (e.g. 10-15), and that can still be instantiated quickly for different values of the layout parameters W and d.

Outline

- Introduction
- **Parameterized Model Order Reduction Classification**
- From Field Solvers to Parameterized Models
- Case 1: Model Reduction with Geometrical Parameters
- Case 2: Model Reduction with Frequency Parameter
- Conclusions

Here is an outline for the remaining part of this talk.

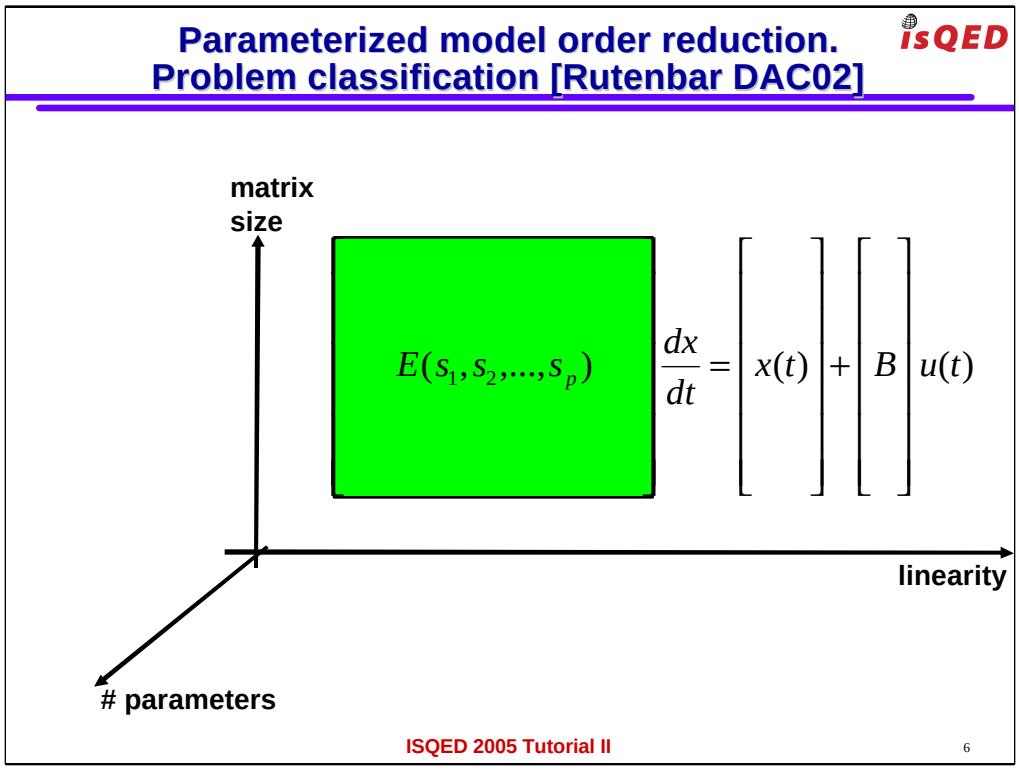
We will first try to classify the Parameterized Model Order Reduction (PMOR) problem.

Then we will see in a simple example how one can assemble a large dynamical linear system model from the output of a field solver

Finally we will present techniques for reducing the size of the model.

We will have to distinguish two important cases: the case where the system parameters are geometrical (e.g. wire width and separation)

and the case where the parameter is frequency.

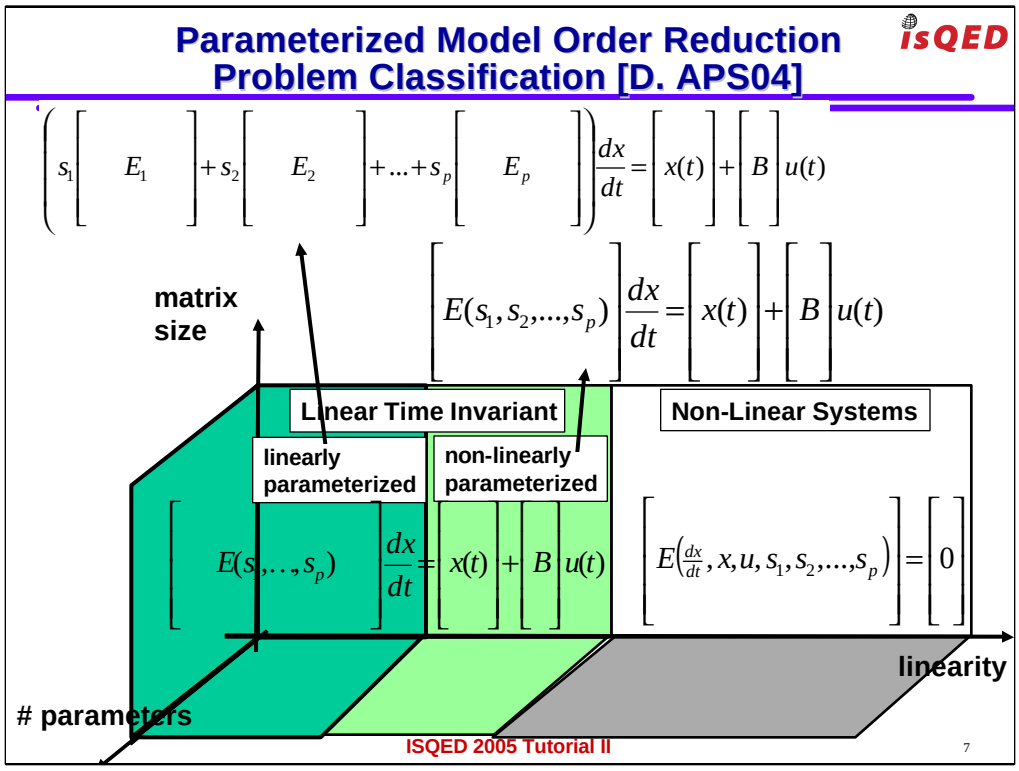


The level of difficulty of a parameterized model order reduction problem can be classified according to Rutenbar using 3 main axis:

the number of parameters

the number of equations (or size of the system)

and how linear those equations are



A linear system is a system for which

-if for instance I apply double the input I double the output

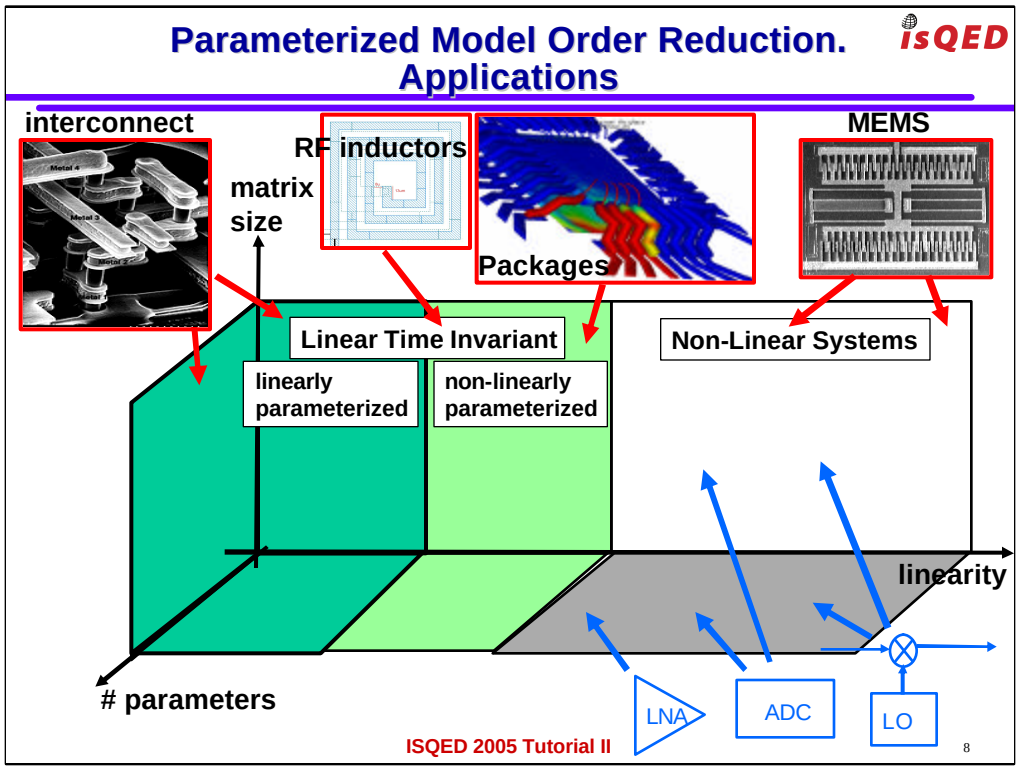
-if I sum two inputs the output is the some of their separate outputs.

However let me introduce a further distinction WITHIN the LINEAR systems.

The coefficients of the equations of a linear system could

-either depend linearly on the parameters

-or could depend in a nonlinear way on the parameters



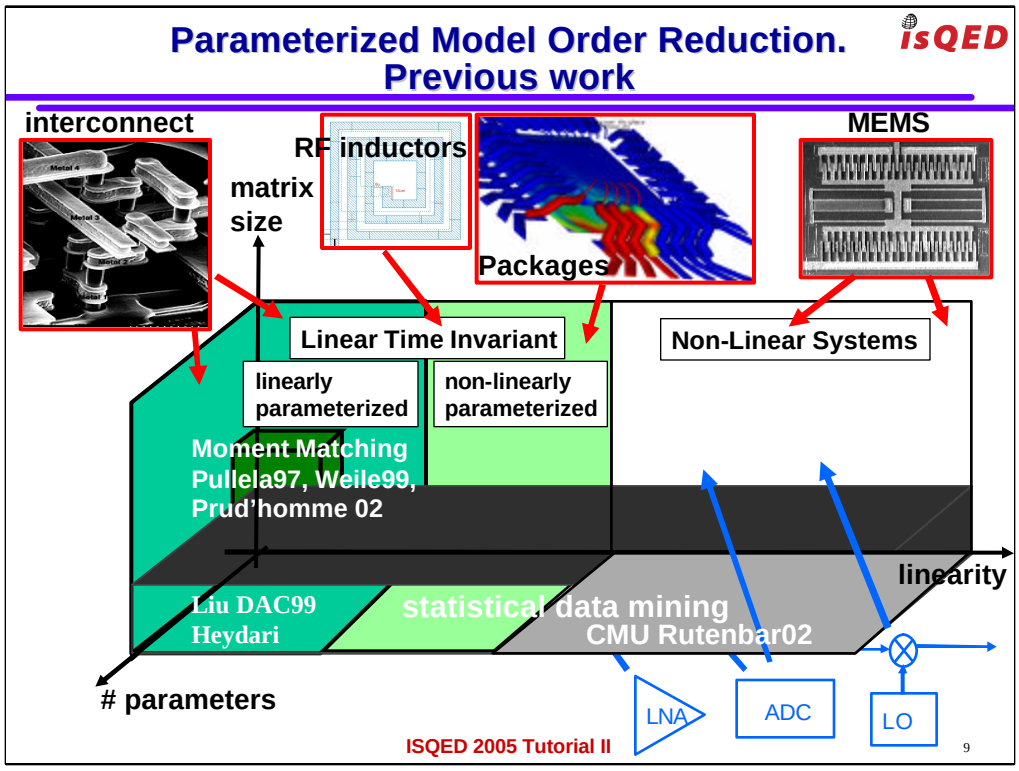
Here is where some of the typically electronic components can be situated according to such classification:

-the systems generated by field solvers applied on interconnects are typically linear, have a very large number of equations (or matrix size) and have a LINEAR dependency on the parameters RF inductors.

-RF inductors, and IC packages produce similar systems but the dependency on design parameters such as wire size and separation is NONLINEAR

-typical analog circuits such as Low Noise Amplifiers (LNA), Analog to Digital Converters (ADC) and Local Oscillators are characterized by smaller matrix size, large number of parameters and are NONLINEAR SYTEMS.

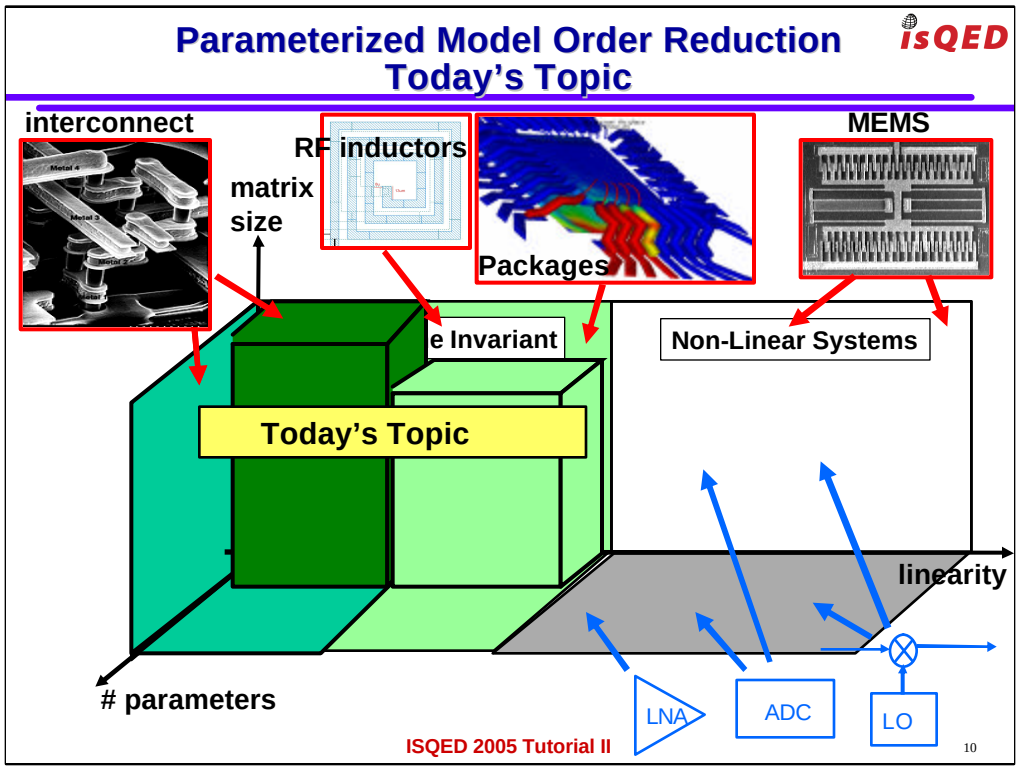
-Finally MicroElectroMechanical resonators are the most difficult of all: lots of parameters, large matrices and very nonlinear systems.



The available approaches to PMOR can be divided into 2 main classes:

-statistical data mining approaches that can handle more easily nonlinear systems but cannot handle very large matrices

-moment matching approaches that can potentially handle much larger size matrices



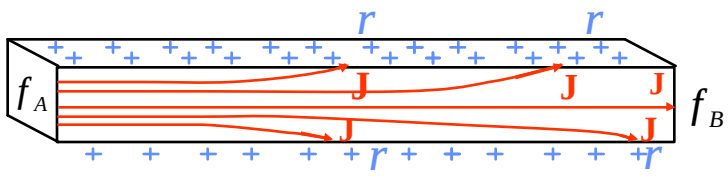
In this talk we will cover the moment matching approaches that are more relevant when reducing the size of the systems produced by field solvers in IC-package codesign problems

Outline

- Introduction
- Parameterized Model Order Reduction Classification
- **From Field Solvers to Parameterized Models**
- Case 1: Model Reduction with Geometrical Parameters
- Case 2: Model Reduction with Frequency Parameter
- Conclusions

Let's now see how one can assemble a dynamical linear system model from the output of one of the field solvers described in the previous presentation

Example: PEEC Mixed Potential Integral Equation [Ruehli MTT74]



$$\frac{\mathbf{J}(\mathbf{r})}{s} + jw\frac{m}{4p}\int_V \mathbf{J}(\mathbf{r}')\frac{e^{jk|\mathbf{r}-\mathbf{r}'|}}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}' = -\nabla f$$

resistive effect **magnetic coupling**

$$\frac{1}{4pe}\int_S r(\mathbf{r}_s')\frac{e^{jk|\mathbf{r}_s-\mathbf{r}_s'|}}{|\mathbf{r}_s-\mathbf{r}_s'|}d\mathbf{r}_s' = f(\mathbf{r}_s)$$

charge-voltage relation

$$\nabla\cdot\mathbf{J}(\mathbf{r}) = 0 \qquad \hat{\mathbf{n}}\cdot\mathbf{J}(\mathbf{r}) = jwr(\mathbf{r}_s)$$

current and charge conservation

Let for instance consider the Mixed potential Integral equation formulation used in the Partial Element Equivalent Circuit method by Ruehli.

One can use the equation in red to capture current distribution inside the conductor.

One can use the charge – voltage equation in blue to capture the charge distribution on the surface of the conductors

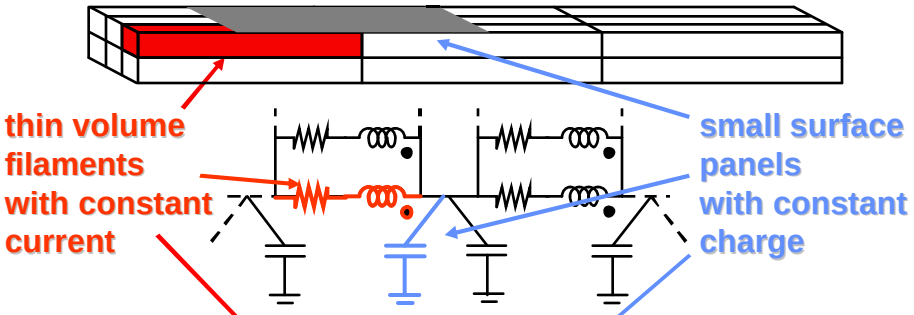
And one can link the two imposing current conservation in the interior of the conductors and charge conservation on the surface.

Observe that the main unknowns are the current distribution J in the interior and the charge density rho on the surface

PEEC Discretization Basis Functions [Ruehli MTT74, MIT course 6.336J and 16.920J]

isQED

- PEEC discretizes volumes in short thin filaments, small surface panels



- PEEC discretization gives branch equations:

$$\begin{bmatrix} R_c + jwL_c(w) & -\Theta \\ 0 & P_p(w) \end{bmatrix} \begin{bmatrix} I_c \\ q_p \end{bmatrix} = \begin{bmatrix} V_c \\ f_p \end{bmatrix}$$

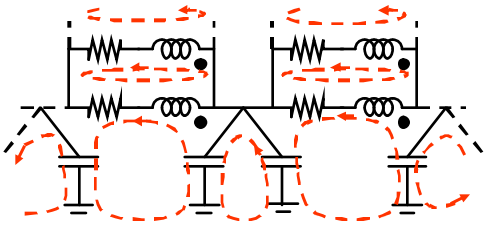
one can represent such unknowns using a collection of basis functions. For instance one can use a collection of small thin filaments for the current and a collection of small panels for the charges.

Using such basis functions and a standard Galerkin test procedure one can transform the previous equations into a set of linear algebraic equations representing the branch equations of an equivalent circuit where currents are modeled by equivalent partial inductors and charges are modeled by equivalent capacitors.

Mesh (Loop) Analysis [Kamon Trans Packaging98]

isQED

Imposing current conservation with mesh (loop) analysis (KVL)



$$\begin{bmatrix} R_c + jwL_c & 0 \\ M_b & \frac{P_p}{jw} \end{bmatrix} \begin{bmatrix} I_b \\ I_{mms} \end{bmatrix} = \begin{bmatrix} V_b \\ 0 \end{bmatrix} = \begin{bmatrix} V_s \\ 0 \end{bmatrix}$$

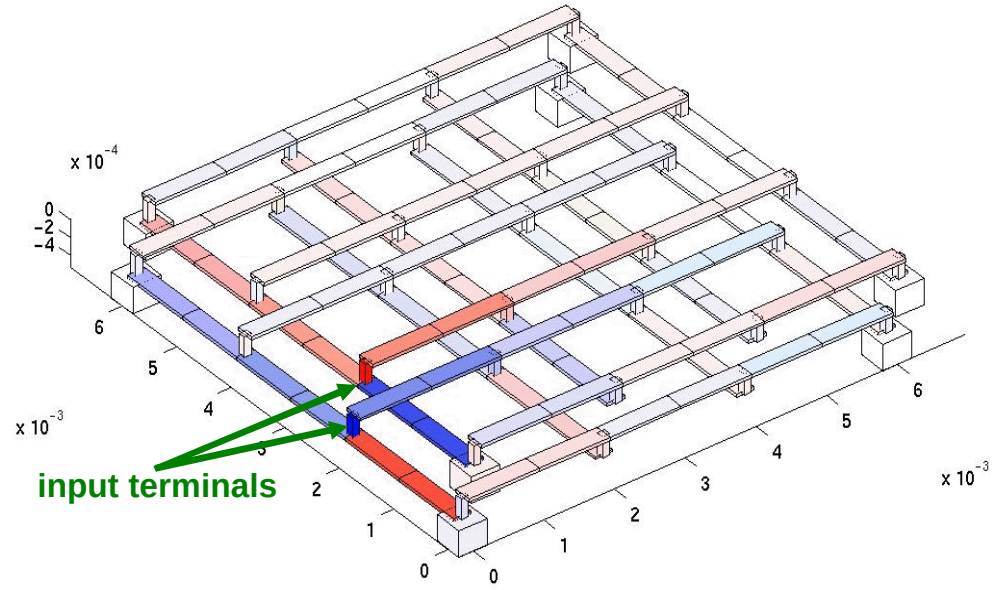
Finally one can impose the remaining two current and charge conservation equations using for instance a mesh analysis approach.

In other words one can write a Kirckof Voltage Law for each mesh in the equivalent circuit.

Using the PEEC branch equations and using simple network theory results one can easily assemble in this way a linear system that can be solved using for instance Krylov subspace iterative methods combined with a fast matrix vector product such as PFFT

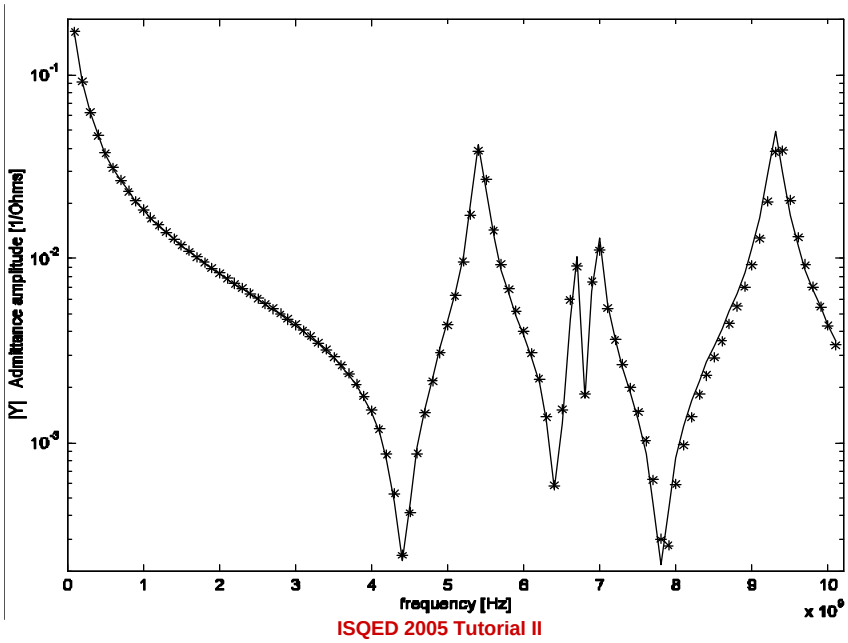
Example of Field Solver output: current distributions on a package power grid

isQED



In a field solver, solving the system can provide values for the currents (and charge) distribution everywhere in the system as shown for instance in this simulation of a large power distribution grid on an package.

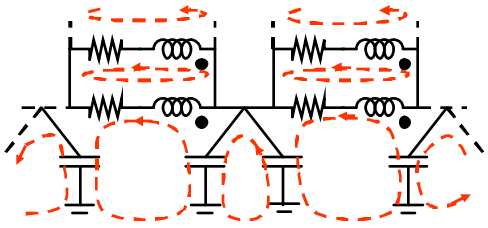
Example of a Field Solver output: package powergrid admittance amplitude



Another possible output of a field solver comes for instance from solving the system at several frequency points creating in this way a frequency response plot for the same package power grid example.

From Field Solvers to a Dynamical Linear System Model

Imposing current conservation with mesh (loop) analysis (KVL)



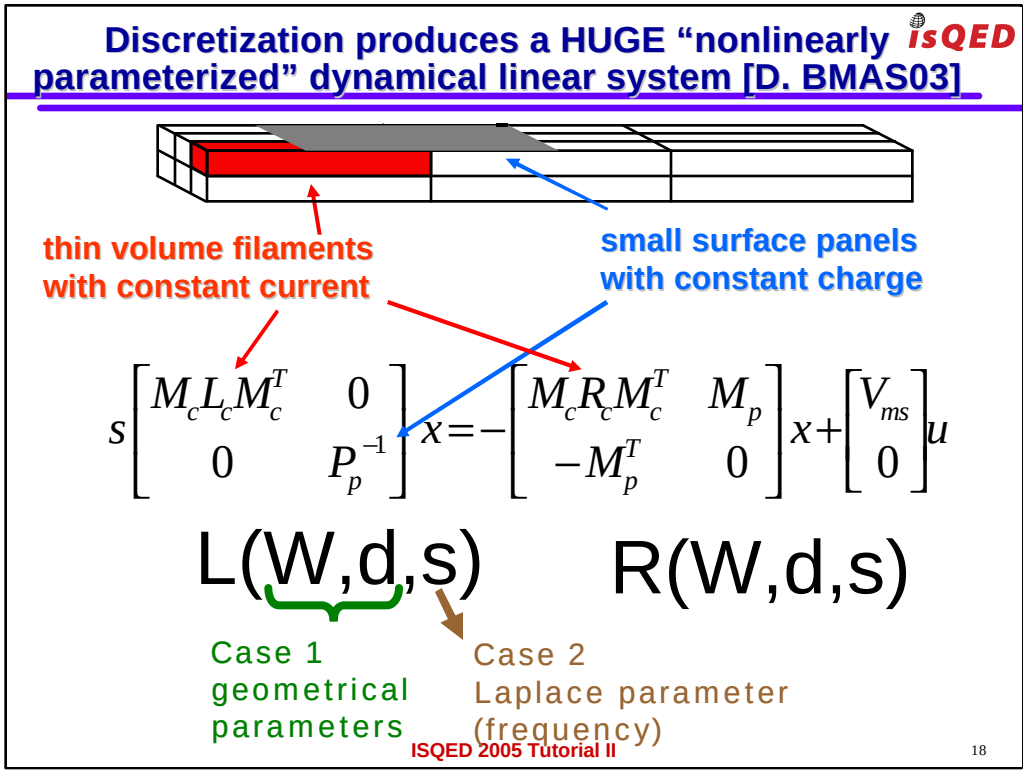
$$\begin{bmatrix} M_c & M_p \end{bmatrix} \begin{bmatrix} R_c + jwL_c & 0 \\ 0 & \frac{P_p}{jw} \end{bmatrix} \begin{bmatrix} M_c^T \\ M_p^T \end{bmatrix} I_m = V_{ms}$$

Multiply out and introduce state $x = \begin{bmatrix} I_m \\ f_p \end{bmatrix}$

The previous slides showed typical results of a field solver type of analysis. However here we want to assemble a dynamical model for the system.

Firs of all instead of working with the frequency omega w let's introduce the more general Laplace variable s=jw

Then we can identify a set of states, for instance the current in the mesh loops, and the voltages on the surface panels



Multiplying out the terms and rearranging them it is easy to rewrite the systems in terms of a dynamical linear system.

It is a "dynamical" system because please remember that sx on the left side of the equation has the meaning in the time domain of the derivative of x: dx/dt

You can see that the two large matrices L and R describing the model:

a) can be calculated using the partial inductance and coef of potential matrices produced by the field solver

b) have different values when the layout parameters or frequency are change

We will now discuss the reduction of the size of these matrices and will divide the discussion in two cases

- first we will discuss the case where the matrices do not depend only on geometrical parameters (for instance wire width W and separation d)
- then if time remains we will discuss a possible the dependency on frequency.

Outline

- Introduction
- Parameterized Model Order Reduction Classification
- From Field Solvers to Parameterized Models
- Case 1: Model Reduction with Geometrical Parameters
  - Polynomial Interpolation
  - Moment Matching (non-parameterized: PRIMA)
  - Moment Matching (parameterized)
  - Example RF inductor
- Case 2: Model Reduction with Frequency Parameter
- Conclusions

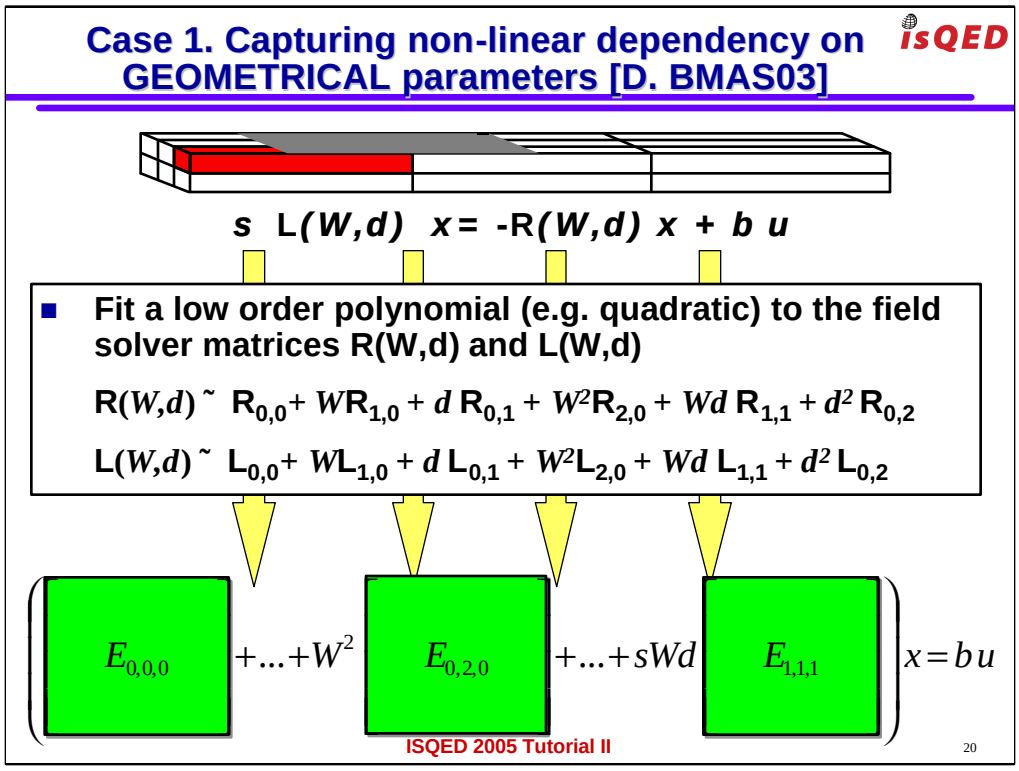
Let's now consider the case where the model matrices depend only on geometrical parameters.

The dependency on the parameters can be in general nonlinear. First we will see a simple method to cast such dependency in a easier to handle polynomial dependency

Then we will briefly review the standard non-parameterized moment matching reduction technique "PRIMA"

Then we extend the moment matching reduction technique to parameterized systems

Finally we show some implementation results on modeling for instance an RF inductor



The dependency of the system matrices L and R on the parameters (for instance wire width W and separation d) can be in general nonlinear L(W,d) and R(W,d)

As a first stem we can use for instance a simple fitting approach to cast such dependency in a easier to handle polynomial dependency.

Calculating Interpolation coefficients [D. BMAS03]

- **E.G. for a 2nd order polynomial fit: we need to calculate 6 coefficients**
- **Hence we need at least 6 equations imposing the fit in 6 test points**
- **However in general it is better to use more evaluation points than the minimum.**
- **For instance here we used a regular grid of 9 evaluation points for different combination of parameters. E.G.:**

(W,d) = (1um,1um), (1um,3um), (1um,5um), (3um,1um), (3um,3um), (3um,5um), (5um,1um), (5um,3um), (5um,5um)

For instance if we want to use a 2nd order polynomial fit we need to calculate 6 coefficients.

therefore we need to impose at least 6 equations

but it is generally more numerically robust to impose a much larger number of equations

and then use a least square solve to find the best 6 coefficients.

In this example we used 9 equations obtained evaluating the large matrices

L(W,d) and R(W,d) in 9 different points in the design space

Calculating Interpolation coefficients

- Use PEEC to generate system matrices $L_k = L(W_k, d_k)$ and $R_k = R(W_k, d_k)$ for each of the 9 combination of parameters in the test points

$$s\begin{bmatrix} M_c L_c M_c^T & 0 \\ 0 & P_p^{-1} \end{bmatrix} x = -\begin{bmatrix} M_c R_c M_c^T & M_p \\ -M_p^T & 0 \end{bmatrix} x + \begin{bmatrix} V_{ms} \\ 0 \end{bmatrix} u$$

$L_k = L(W_k, d_k)$ $\quad$ $R_k = R(W_k, d_k)$

Note that in order to calculate the large matrices L(W,d) and R(W,d) for any of the 9 combinations of (W,d)

we can simply use nine times our PEEC field solver

Calculating Interpolation coefficients

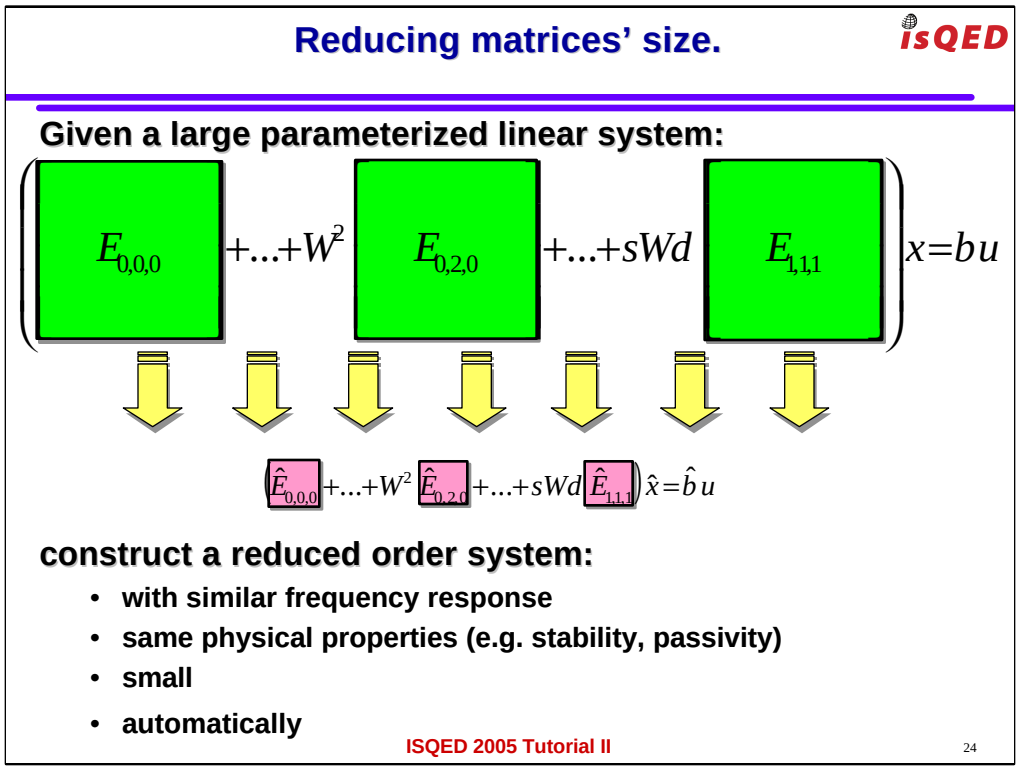
- **Use a least square method to find the best fit for the 6 coefficients of the 2nd order polynomial matching the 9 test grid points**

$$\begin{bmatrix} 1 & W_1 & d_1 & W_1^2 & W_1 d_1 & d_1^2 \\ 1 & W_2 & d_2 & W_2^2 & W_2 d_2 & d_2^2 \\ 1 & W_3 & d_3 & W_3^2 & W_3 d_3 & d_3^2 \\ 1 & W_4 & d_4 & W_4^2 & W_4 d_4 & d_4^2 \\ 1 & W_5 & d_5 & W_5^2 & W_5 d_5 & d_5^2 \\ 1 & W_6 & d_6 & W_6^2 & W_6 d_6 & d_6^2 \\ 1 & W_7 & d_7 & W_7^2 & W_7 d_7 & d_7^2 \\ 1 & W_8 & d_8 & W_8^2 & W_8 d_8 & d_8^2 \\ 1 & W_9 & d_9 & W_9^2 & W_9 d_9 & d_9^2 \end{bmatrix} \begin{bmatrix} R_{0,0}^{i,j} \\ R_{1,0}^{i,j} \\ R_{0,1}^{i,j} \\ R_{2,0}^{i,j} \\ R_{1,1}^{i,j} \\ R_{0,2}^{i,j} \end{bmatrix} = \begin{bmatrix} R_1^{i,j} \\ R_2^{i,j} \\ R_3^{i,j} \\ R_4^{i,j} \\ R_5^{i,j} \\ R_6^{i,j} \\ R_7^{i,j} \\ R_8^{i,j} \\ R_9^{i,j} \end{bmatrix}$$

Finally we can write the 9 equations one after the other and collect them into a system form.

Note that the indices (i,j) indicate the coefficient (i,j) of the matrix R(W,d)

Since the system has more equations than unknowns we can use a least square solve algorithm (e.g. QR)



Now that we have captured the dependency of the parameters in a simple polynomial form, the second step of the procedure consists in the reduction of the size of the matrices E.

We want to do that

- preserving the frequency response of the system for different values of the parameters

-preserving some physical properties such as stability and passivity

-we want the produced matrices to be very small

-and we want the entire procedure to be completely automatic

The final output that we give to the user (the designer) is a small model consisting of a few SMALL matrices.

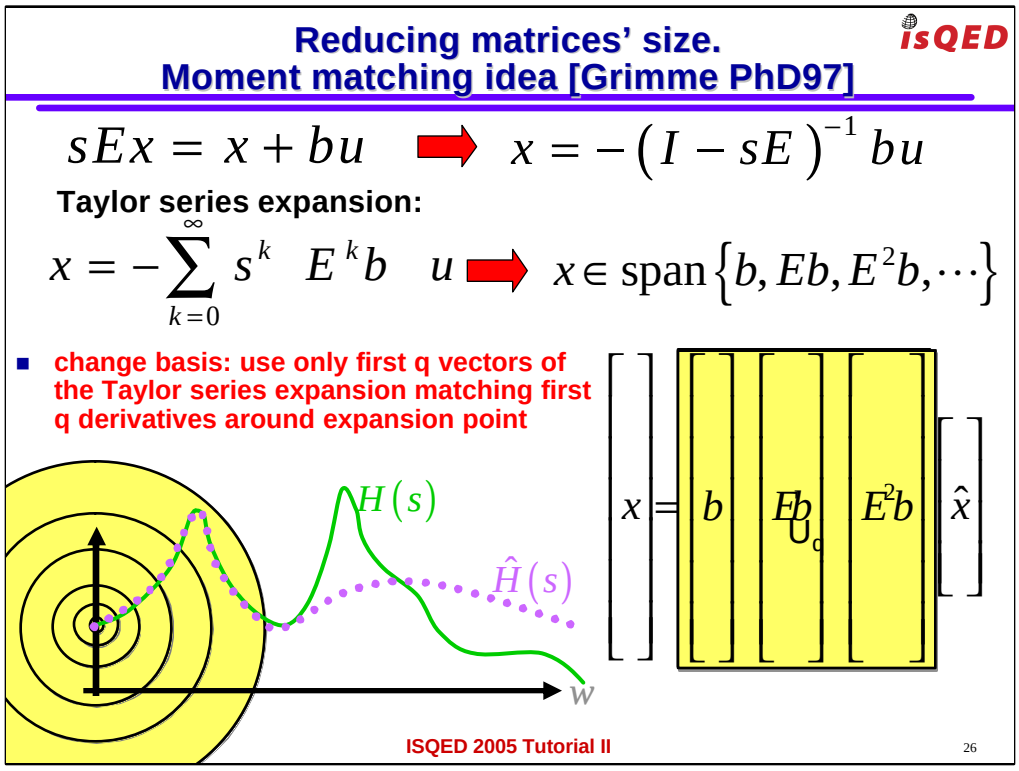
If the user wants to instantiate a new model for his/her own chosen value of parameters W and d he/she only

needs to multiply and sum up a few matrices of very small size

Outline

- Introduction
- Parameterized Model Order Reduction Classification
- From Field Solvers to Parameterized Models
- Case 1: Model Reduction with Geometrical Parameters
  - Polynomial Interpolation
  - Moment Matching (non-parameterized: PRIMA)
  - Moment Matching (parameterized)
  - Example RF inductor
- Case 2: Model Reduction with Frequency Parameter
- Conclusions

Let's now review briefly the standard non-parameterized moment matching reduction technique "PRIMA"



In PRIMA we have one single matrix E and parameter s.

First let's write the transfer function from the input u to the state x

Let's then write its Taylor series expansion in the variable s around some point in frequency (for instance here s=0)

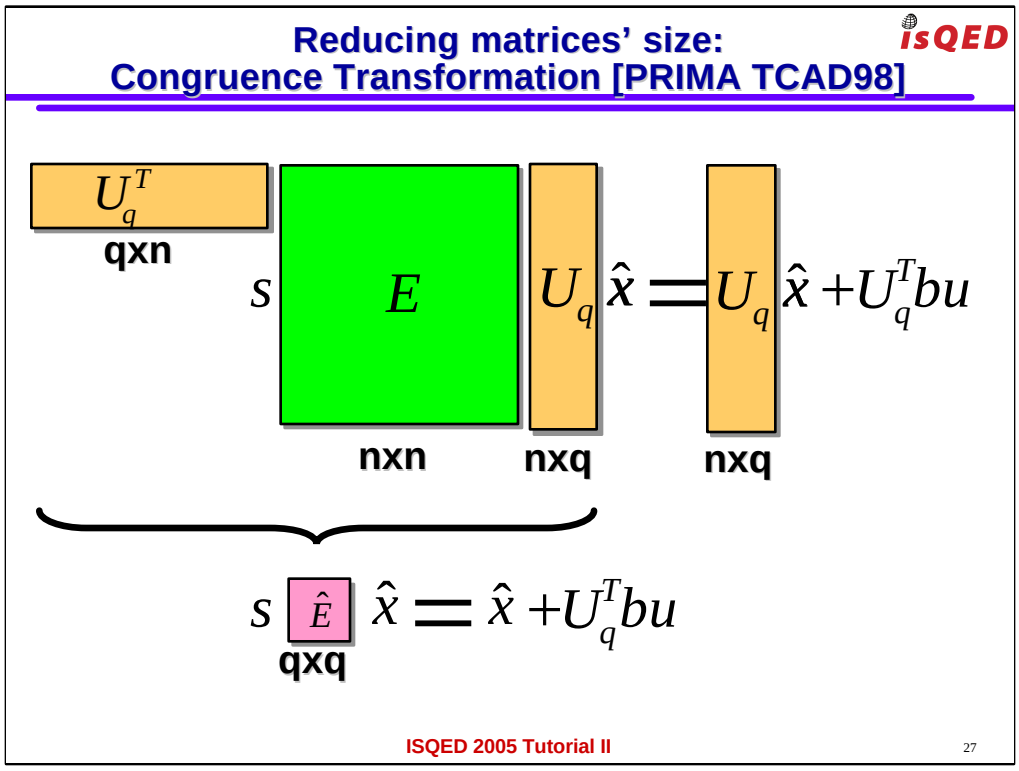
Let's look at what we wrote: we just wrote the state x as a linear combination of a whole bunch of vectors

for instance the vector b, and the vector Eb and the vector E^2b and so on and so forth…

Another way to express this concept is to say that the state x lives in the subspace generated by those vectors.

When adding each of those vectors I add one more term of the Taylor series expansion (also called moment) or in other words I match one more derivative with respect to s of the frequency response (which I am showing here with yellow circles).

If I want use only the first q=3 vectors to write x I will match only the first 2 moments (or derivatives).



substituting this change of variables in the original system I immediately recognize that I have not a system with a much smaller number of components in the reduced state.

However I still have a very large number of equations.

In order to reduce the number of equations I could for instance multiply on the left the whole system by some matrix.

PRIMA for instance uses the SAME matrix used for the change of basis.

This is NOT optimal in terms of accuracy

However it guarantees as we will see later that the final system is stable and passive.

For now in this slide let's just note graphically how the size of the system matrix has been reduced by the

multiplication on both sides by matrix U (congruence transformation)

Outline

- Introduction
- Parameterized Model Order Reduction Classification
- From Field Solvers to Parameterized Models
- Case 1: Model Reduction with Geometrical Parameters
  - Polynomial Interpolation
  - Moment Matching (non-parameterized: PRIMA)
  - Moment Matching (parameterized)
  - Example RF inductor
- Case 2: Model Reduction with Frequency Parameter
- Conclusions

let’s try now to extend this approach to the parameterized cased

Parameterized moment matching [D. TCAD04] [D. PhD04]

$$\left(E_{0,0,0} + \ldots + W^2 E_{0,2,0} + \ldots + sWd\, E_{1,1,1}\right) x = b\, u$$

$$x = \sum_m \left(s_1 E_1 + s_2 E_2 + \ldots + s_p E_p\right)^m b u$$

- **It is a p-variables Taylor series expansion**

$$x \in span\left\{b, E_1 b, E_2 b, \cdots, E_p b, E_1^2 b, \left(E_1 E_2 + E_2 E_1\right) b, \cdots\right\}$$

$$\begin{bmatrix} \\ x \\ \\ \end{bmatrix} = \boxed{U_q} \begin{bmatrix} \\ \hat{x} \\ \\ \end{bmatrix}$$

Once again change basis:

- **use first few vectors of the Taylor expansion,**
- **matching first few derivatives with respect to each parameter**

We now have many parameters and many matrices.

Let's redefine these monomial parameters with new names for simplicity.

We can recognize now that we have a function dependent on many variables (the new redefined parameters).

So we can use a MULTIVARIABLE Taylor series expansion.

If we do that in a similar way to what we did for a single parameter in PRIMA we can recognize

that the state x can be expressed as a linear combination of a whole bunch of vectors.

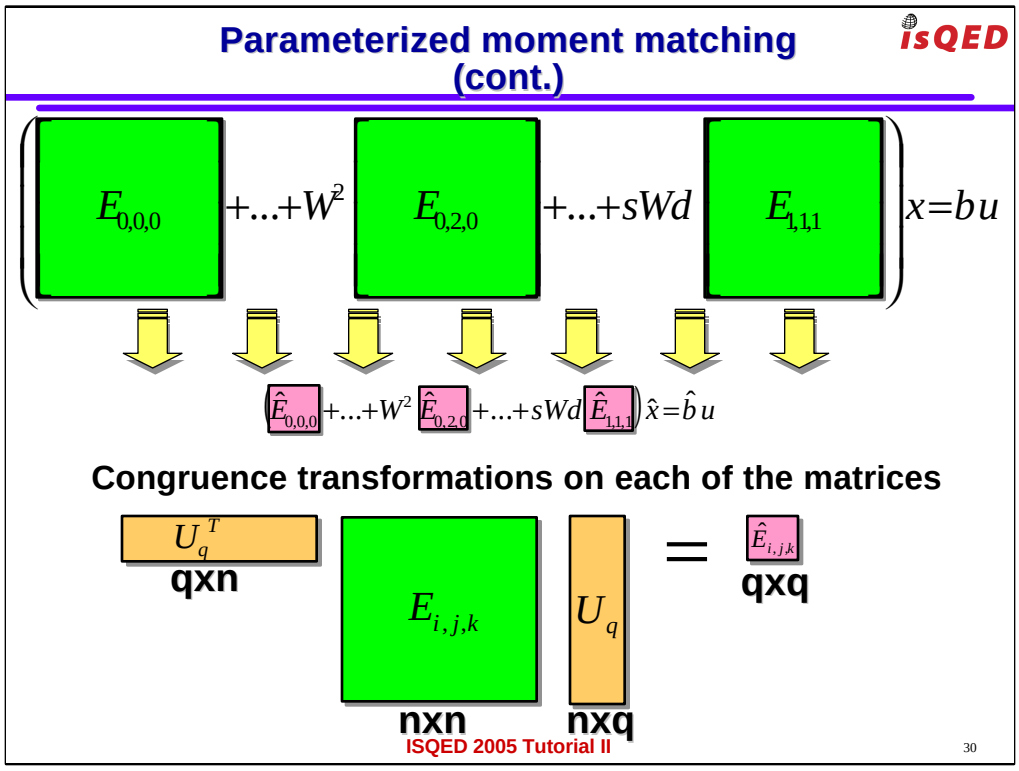
Each of these vectors will add one more term to the Taylor series expansion (that is one more moment

or derivative with respect to some parameter).

If we want to approximate the system we can just truncate that Taylor expansion as before to the

first few q moments. Mechanically we do that with a change of variable where the change of basis

matrix as before has in the columns the first few q vectors of the subspace



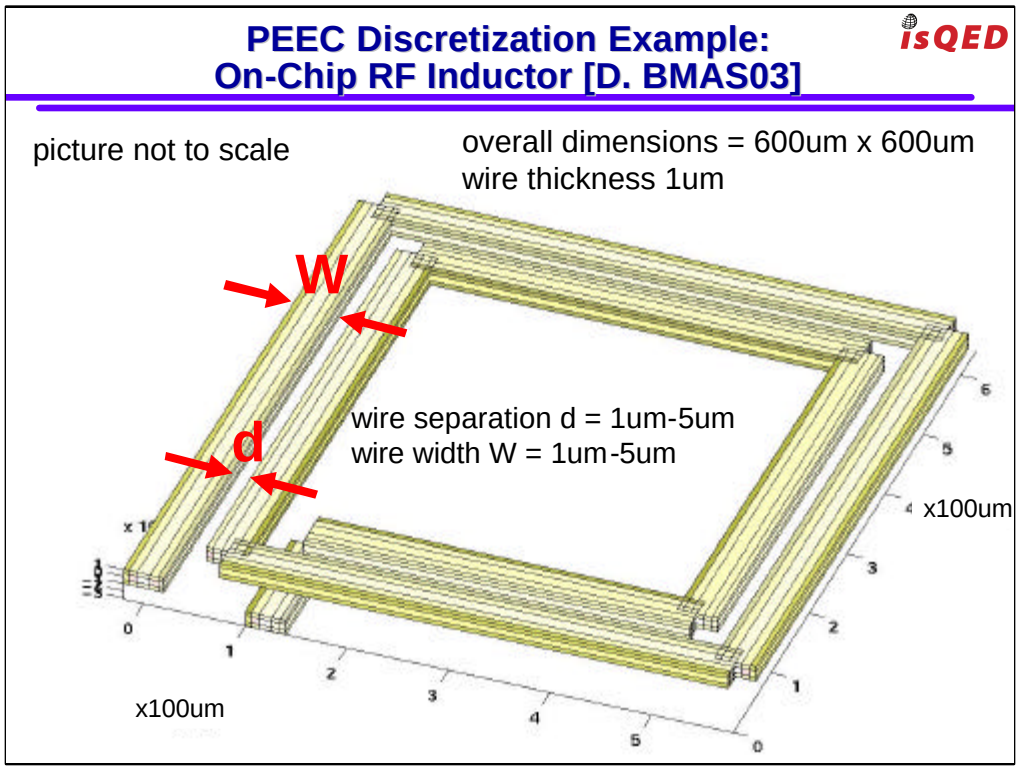
We can now do the same steps we did before:

1) substitute the change of variables
2) and premultiply the system by the same matrix U

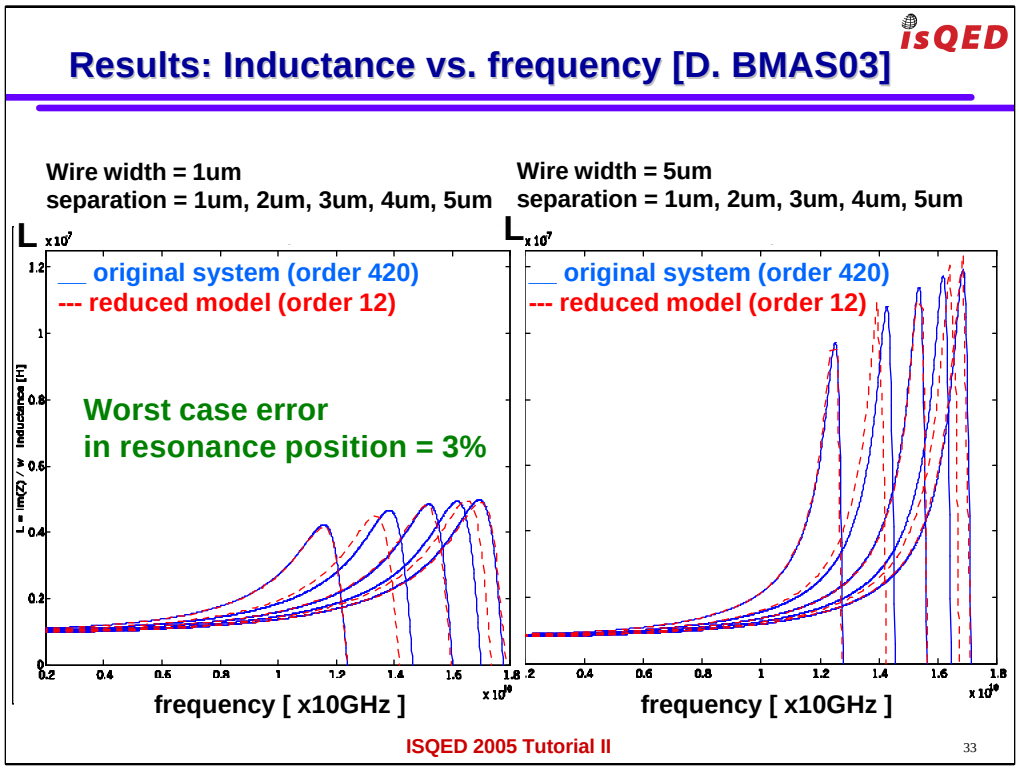
We notice that the size of each of the final resulting matrix is reduced to simply qxq VERY SMALL

Outline

- Introduction
- Parameterized Model Order Reduction Classification
- From Field Solvers to Parameterized Models
- Case 1: Model Reduction with Geometrical Parameters
  - Polynomial Interpolation
  - Moment Matching (non-parameterized: PRIMA)
  - Moment Matching (parameterized)
  - Example RF inductor
- Case 2: Model Reduction with Frequency Parameter
- Conclusions



In example we constructed a model for an RF inductor that can be instantiated instantaneously for different values of wire with W and wire separation d.



After the parameterized model is produced we verified its accuracy by

1) instantiating it for different values of wire with and separation
2) and comparing it to field solver results run on layout constructed with those same wire width and separation

On the left we show models instantiate with wire width 1um. on the left 5um

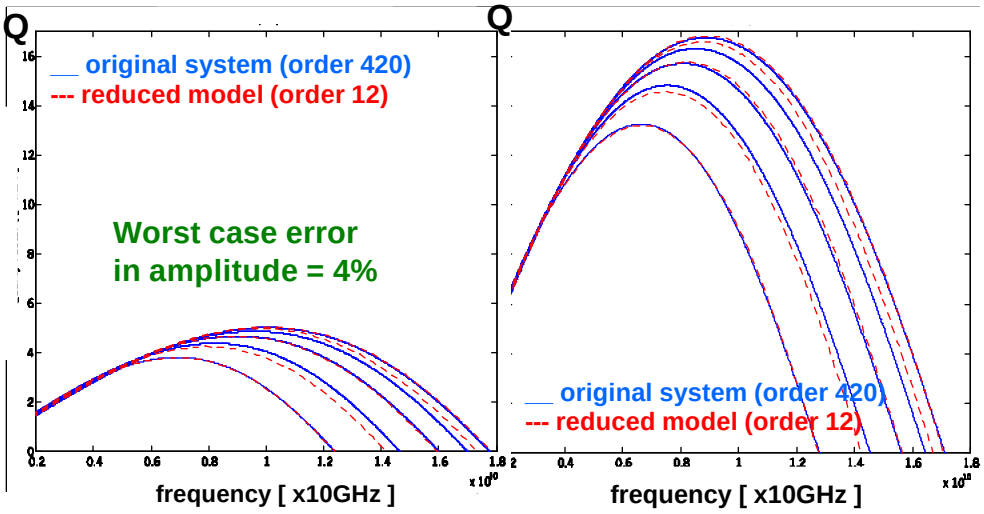
the red dashed lines are the result of imaginary part of the frequency response divided by jw vs. frequency

for the reduced model size 12 instantiated for different values of wire separation.

those lines compare quite nicely with the reference blue continuous values produced by the field solver with matrices size 420

Worst case error on frequency position is 3%

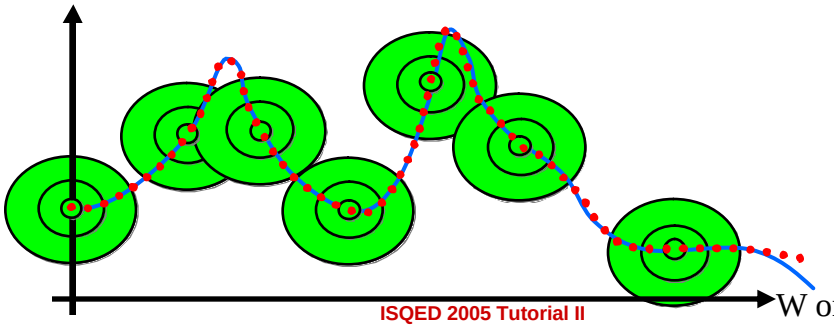
Results: Quality factor (Q=wL/R) vs. frequency



and worst case error on the magnitude of the quality factor is 4%

Open issues in the PMOR Matrix Reduction step

- **Model order grows as $O(p^m)$ where p = # parameters and m = # derivatives matched for each parameter**
  - **however model order is linear in # of parameters when matching only one derivative per parameter (m = 1) and still produces good accuracy in our experiments.**
  - **furthermore, for higher accuracy instead of increasing # of matched derivatives, can instead match multiple points (or combine the two approaches)**



There are still several open issues regarding this approach.

For instance if p is the number of parameters and m is the number of derivatives matched for each parameter

then the order of the produced reduced model grows with a nasty exponential complexity $O(p^m)$

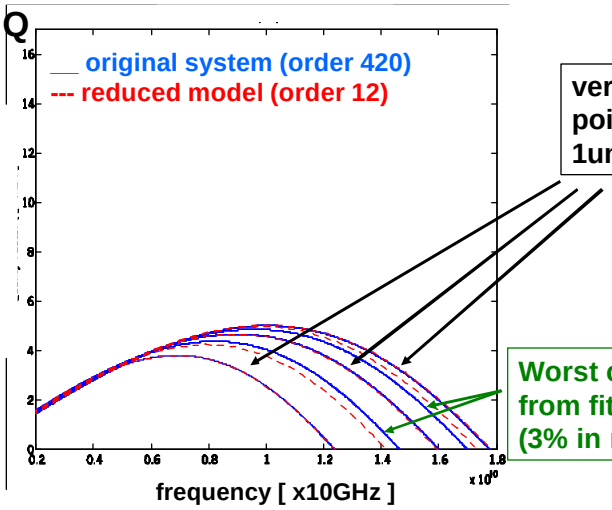
Fortunately If one matches only one derivative (m=1) per parameter the order grows only linearly with the number of parameters.

But is that enough accuracy? For some applications probably yes: you will be the judge of that since the RF inductor example corresponds exactly to that case.

For higher accuracy, instead of matching more derivatives, one could try matching several points

Open issues in the PMOR non-linear parameter interpolation step

Wire width = 1um
separation = 1um, 2um, 3um, 4um, 5um



very little error in the points used for fitting 1um, 3um, 5um

so the critical step is the fitting!!

Worst case errors far from fitting points 2um, 4um (3% in resonance position)

Let's look at another open issue.

There are two kinds of error introduced by our procedure.

The error of the first polynomial fitting step.

And the error of the actual matrix reduction step.

If we look very closely at the quality factor matching results we see that the parameterized reduced

model match VERY well for separation values d=1um, 3um and 5um

A little less well instead for 2um and 4um.

You may remember that the values 1um, 3um and 5um are the values we chose

to use when we did the polynomial fitting the matrices R(W,d) and L(W,d)

The error in those points is very small and must be due to only to the matrix reduction step which is then

working very well.

Instead the error in d=2um and 4um is larger. Hence we can see that in this case

the error of the first polynomial fitting step is larger that the matrix reduction step.

In general one should try to balance the two steps and obtain similar errors in both steps for an optimal job.

Outline

- Introduction
- Parameterized Model Order Reduction Classification
- From Field Solvers to Parameterized Models
- Case 1: Model Reduction with Geometrical Parameters
- **Case 2: Model Reduction with Frequency Parameter**
  - Preserving Stability and Passivity
  - Globally Convergent Interpolation
  - Example 2 lines over substrate, full-wave
- Conclusions

Let's consider now the case where the parameter is Frequency with a bit more attention

Distributed Linear Systems

$$E(s, s_1, \ldots, s_p)x = bu$$

$$y = c^T x$$

where the dependency on the Laplace variable s is not linear

$$E(s, s_1, \ldots, s_p) \neq s\,E(s_1, \ldots, s_p) - A(s_1, \ldots, s_p)$$

Examples:

- **full-wave PEEC**
- **PEEC using layered-media Green functions (e.g. for handling substrate or dielectrics)**
- **frequency-dependent basis functions**
- **frequency dependent discretizations**

Distributed systems are systems whose dependency on the frequency parameter is not linear (or more precisely affine)

This may happens for instance

-when the field solver uses a fullwave formulation

-when layered media is treated using green functions

-when one uses higher order frequency dependent basis functions

-or when one uses a frequency dependent discretization

Polinomial interpolation for frequency [Phillips96]

$$E(s)x = bu \qquad y = c^T x$$

- **Polynomial approximation e.g. Taylor expansion, or a polynomial interpolation for E(s)**

$$\left(E_0 + sE_1 + s^2 E_2 + \cdots + s^M E_M\right)x = bu$$

- **Convert to non-distributed model reduction problem**

$$\tilde{x} = [x \quad sx \quad s^2 x \quad \cdots \quad s^M x]$$
$$s\tilde{E}\tilde{x} = \tilde{x} + bu$$

- **Performance: Fast and accurate in the frequency band of interest**
- **Problem: Can not be used in a time domain circuit simulator because does not guarantee stability and passivity**

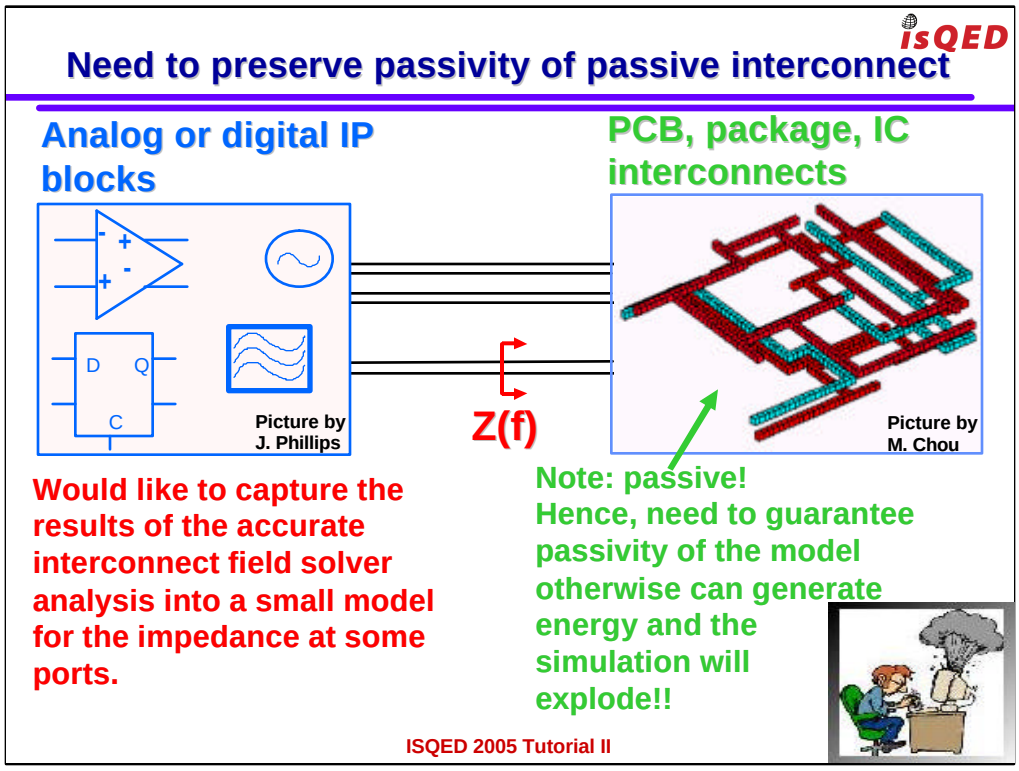
One could try to use the same “polynomial” fitting approach that we used for geometrical parameters.

The first attempt along these lines is due to Joel Phillips.

The approach is fast an accurate in the frequency band of interest

unfortunately one the produced model is used within a TIME DOMAIN simulator, some numerical instability problems can often occur.

This is due to the fact that often the models produced by polynomial fitting are not stable nor passive

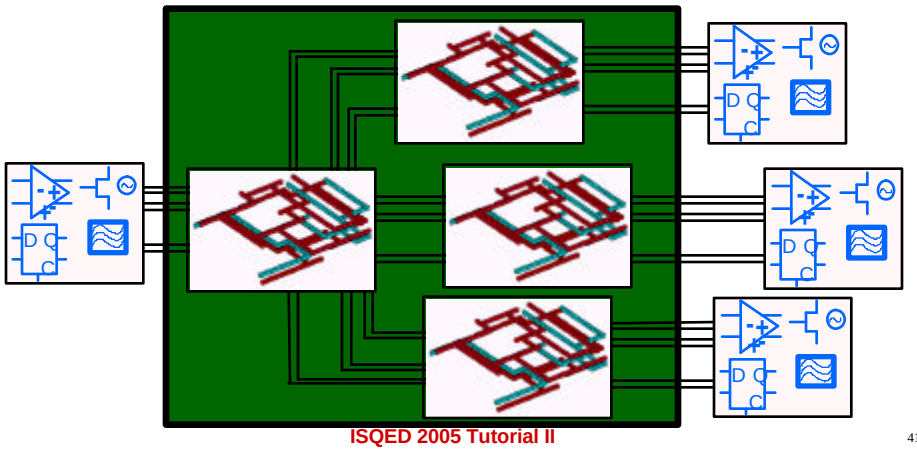


Usually we are instead in producing small models of the PCB, package, and IC interconnect wires that connect circuit components.

Such systems are intrinsically passive, hence the model we produce for them needs to be passive as well otherwise the time domain simulation may explode as it is clearly illustrated in the picture in the corner ☺

The Composition of Passive Models is Passive

- The composition of "stable" models is not necessarily stable
- But the composition of "passive and stable models" is passive and stable.



ISQED 2005 Tutorial II

41

Furthermore, we would like the designers to be able to freely connect our models in the same way they connect their actual components to create larger systems.

Unfortunately the interconnection of stable models may not be guaranteed to be stable

But fortunately the interconnection of any passive models is always a passive model (and hence also stable)

Therefore it is important to produce models that are not only guaranteed stable but also guaranteed passive.

Passivity condition on transfer function [Willems72]

- **For systems with immittance matrix representation, passivity is equivalent to positive-realness of the transfer function**

$$y(s) = H(s)u(s)$$

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

$\overline{\mathbf{H}(s)} = \mathbf{H}(\bar{s})$ for $\mathrm{Re}(s) > 0$ ⟸ (impulse response is real)

$\mathbf{H}(s) + \mathbf{H}(s)^H \geq 0, \text{for } s = jw$ ⟸ (no negative resistors)

It means positive resistance (conductance) for any frequency.
Note: it is a global property!!!

mathematically, for immitance systems (that is systems whose input and outputs are currents and voltage, or the opposite)

passivity is equivalent to “positive realness of the transfer function”

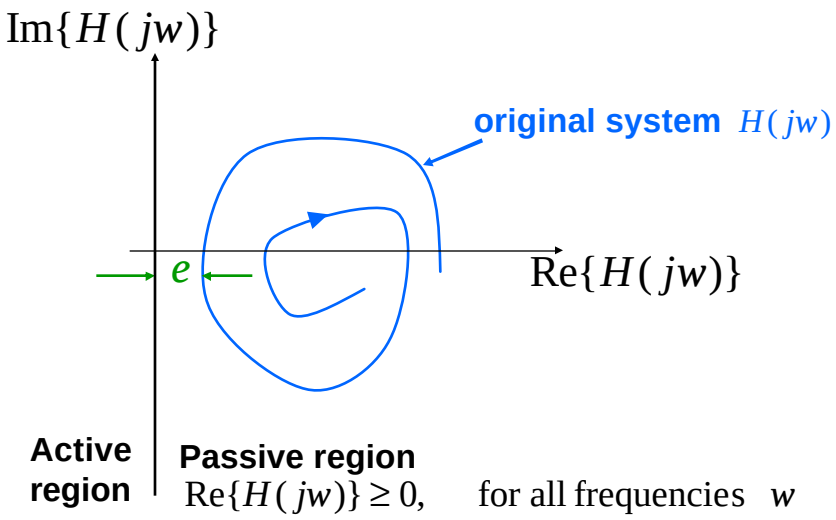
or in other words

1) H(s) has no unstable poles in the right half plane
2) the impulse response is “real”, so the system is a physical system with real coefficients
3) the real part of the transfer function is always positive for any frequency (or in other words the system dissipates energy at all frequencies)

Note that the most important property is the 3rd and the most important part of it is that the real part is positive FOR ALL frequencies

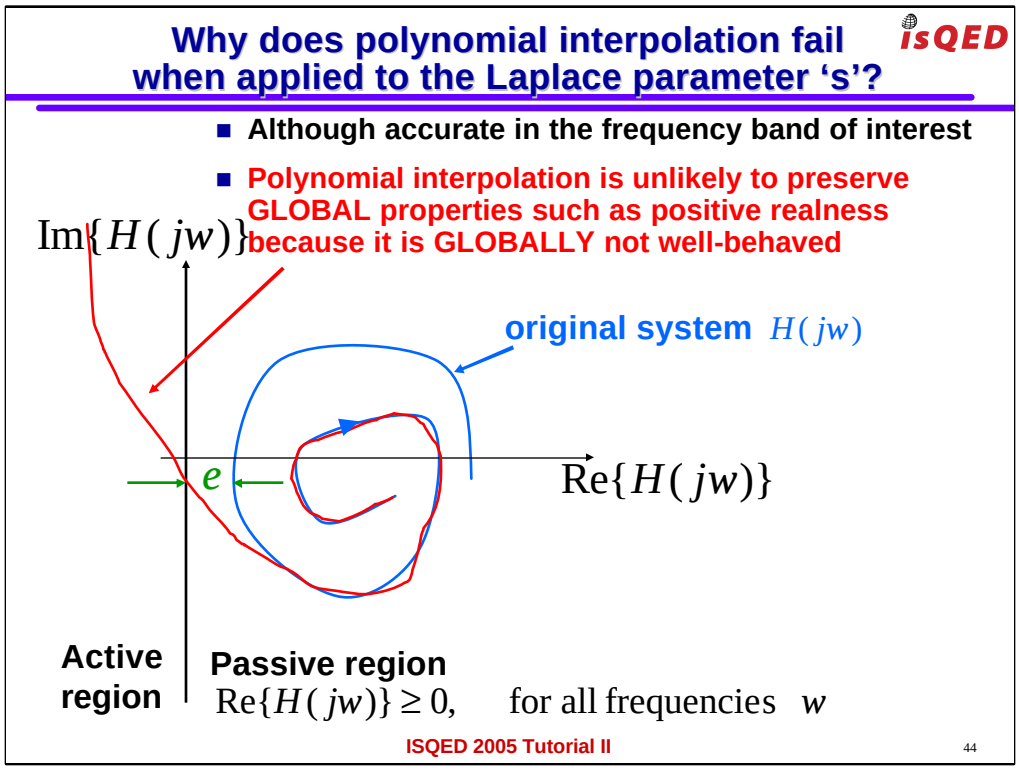
so passivity is a GLOBAL property of the system.

Positive real transfer function in the complex plane for different frequencies



One graphical way to visualize the passivity property is to draw the path of the transfer function in the complex plain for all frequencies.

If the system is passive H(jw) will always be at least at a distance epsilon from the imaginary axis, completely contained in the passive right region.



A polynomial fitting approach can achieve a very accurate matching in a large band of frequency of interest to the user.

However it can have a very very inaccurate matching for much higher frequencies where the users THINKS he/she does not care.

But in reality he/she MUST care, because if the matching is very inaccurate it could potentially go for some frequency into the

active region.

In other words if the system is excited by some small noise at those frequencies it will generate lots of energy and the numerical simulation

can quickly become unstable.

The problem with the polynomial interpolation approach is that it is NOT well behaved GLOBALLY

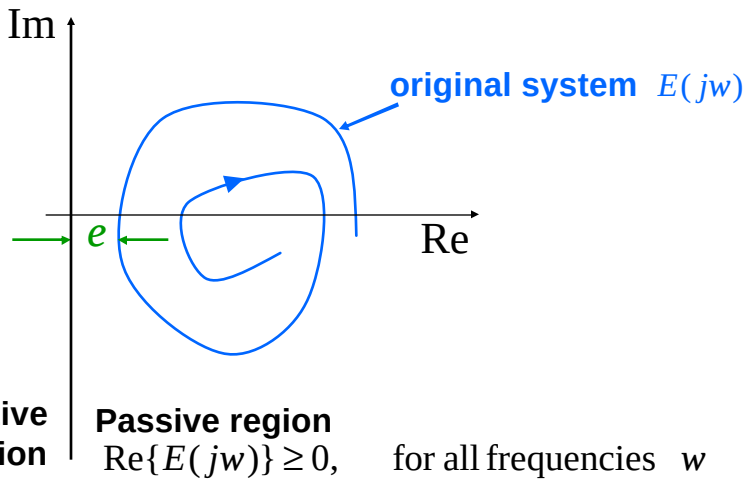
Outline

- Introduction
- Parameterized Model Order Reduction Classification
- From Field Solvers to Parameterized Models
- Case 1: Model Reduction with Geometrical Parameters
- Case 2: Model Reduction with Frequency Parameter
  - Preserving Stability and Passivity
  - Globally Convergent Interpolation
  - Example 2 lines over substrate, full-wave
- Conclusions

one solution is therefore to loot for some other interpolation that is globally convergent.

Observation: practical systems have some loss at all frequency

- Most systems are non-ideal i.e. contain some small loss e at any frequency w i.e. they can be described by strictly positive real matrices $E(jw)$

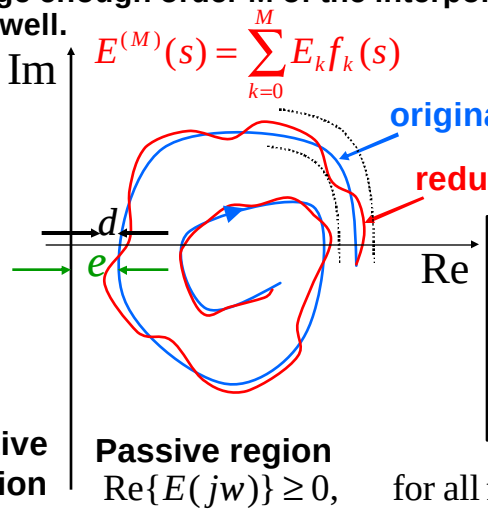


Let's first note that most practical systems are STRICLTLY passive, meaning that the path in the active region never touches that active region

and they are actually at least some epsilon away from it. (on in other words there is always some loss mechanism at any frequency).

Using global uniformly convergent interpolants [D. DAC02] [D. PhD04]

- **If E(s) is strictly positive real, a GLOBALLY and UNIFORMLY convergent interpolant will eventually get close enough (for a large enough order M of the interpolant) and be positive-real as well.**



- **Proof: just choose accuracy of interpolation d smaller than minimum distance from imaginary axis e**

If we use a “globally and uniformly convergent set of basis functions phi(s) then we can guarantee that

for ANY frequency the path of the reduced system can be restricted to a distance delta that we can make as small as we

want as long as we pick enough interpolation basis functions.

well the solution to make sure we have a PASSIVE reduced system is to make sure we pick enough interpolation

functions such that delta < epsilon. In that case the reduced system path is guaranteed to be never cross into the active region.

A good example of uniformly convergent interpolants: the Laguerre basis functions [D. DAC02] [D. PhD04]

isQED

- **Consider the family of basis functions:**

$$f_k(s) = \left(\frac{l - s}{l + s} \right)^k; \quad s = jw; \quad k = 0,1,\ldots,\infty$$

- **They form a complete, rational, orthonormal basis over the imaginary axis which gives a uniformly convergent interpolant**

- **No poles in RHP (stable)**

- $\overline{f}_k(s) = f_k(\overline{s})$ **(real time-domain representation)**

Now we need to find at least one set of globally and uniformly convergent basis functions.

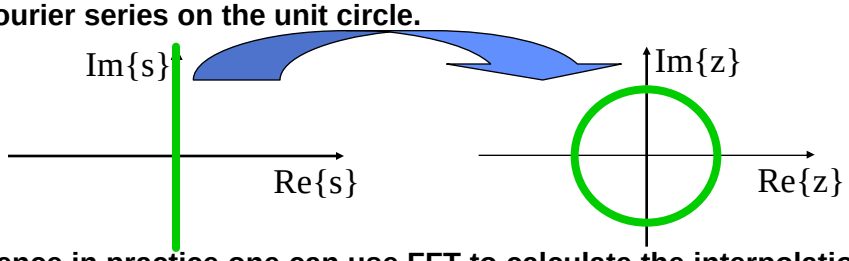
For instance the following Laguerre basis could be a good choice:

1) they are a complete set, so they can represent any function we want
2) they are rational, so they can be manipulated easily and will produce an easily synthesizable model
3) they are orthonormal so it will be easy to calculate the coefficients for the interpolation using ONE single inner product
4) they have no poles in the right half place (RHP) so they are intrinsically stable
5) they have the conjugate symmetry property so they are intrinsically associated with a “real” time domain representation

Calculation of interpolation coefficients

$$E^{(M)}(s) = \sum_{k=0}^{M} E_k f_k(s) \qquad f_k(s) = \left(\frac{1-s}{1+s}\right)^k = z^k$$

- **Note: it is a bilinear transform that maps the Laguerre basis to Fourier series on the unit circle.**



- **Hence in practice one can use FFT to calculate the interpolation coefficients: very efficient!**
- **Note: FFT coefficients typically drop quickly and the series can be truncated to the first few M coefficients because field solver matrices E(s) are often smooth.**

The most useful property of all is that it is computationally very easy to calculate the coefficients E_k for the interpolation.

One can simply realized that the Laguerre functions are some sort of bilinear transformation that frequencies s on an imaginary axis

into frequencies z on a unit circle.

in other words they can be interpreted as “powers of z” and hence the coefficients of the interpolation simply becomes

coefficients of a Discrete Fourier Transform that can be calculated in NlogN time using an FFT.

Finally one may wonder: “what if I need a very very large number of interpolation functions in order to get an accuracy delta

good enough to guarantee passivity???”

well that could in general be possible, however if the initial function is smooth enough than we know from the properties

of the Fourier transforms that the Fourier coefficients will drop very quickly and only a few of them will be enough to get an accurate

interpolation of the smooth function.

Reduction procedure [D. DAC02] [D. PhD04]

	Matrix sizes	System order
■ **Start from original system described by causal, strictly positive-real matrices** $$E(s)x = bu$$	**~ 3,000**	**infinite**
1) Evaluate and squash them at uniformly spaced points on the unit circle using congruence transformation which preserves positive realness $U^T E(s_k) U, \quad k=1,2,...,64$ $b_r = U^T b$	**~ 6**	**~ 6 x 64**

Let's now summarize the main steps of the procedure

we start from the original system that is produced by a fullwave or green function based field solver.

The matrices are large e.g. 3000x3000 and the order of the system is infinite because those matrices are frequency dependent

We can construct a change of basis matrix using for instance a multipoint scheme where each column of U is simply the state of the system for some frequency point. (Zhenhai: you don't need to say this.

I wrote it here only if people ask how you can get U)

As a first step of the procedure we can evaluate the large system matrix at some points (e.g 64) along

the imaginary axis corresponding to equally spaced points on the unit circle.

using U and the congruence transformation we can reduce the size of those matrices to for instance 6x6.

The order of the system is now 6x64 since we have powers of z up to order 64.

Reduction procedure

	Matrix sizes	System order
3) Calculate first few (e.g 5) FFT of the reduced system matrix coefficient $\left[\sum_{k=0}^{4}\tilde{E}_k z^k\right]\tilde{x}=b_r u$	**6**	**6 x 5**
4) Introduce extended state and realize a single matrix discrete time system $\tilde{x}=[x \quad zx \quad z^2x \quad \cdots \quad z^M x]$ $z\tilde{E}\tilde{x}=\tilde{A}\tilde{x}+\tilde{b}u$	**6 x 5**	**6 x 5**
5) Transform to continuous time $s\hat{E}\hat{x}=\hat{A}\hat{x}+\hat{b}u$	**6 x 5**	**6 x 5**

We can now calculate the FFT coefficients of those 64 small matrices.

The FFT coefficients will be some other 6x6 matrices.

Since they drop very quickly we can use maybe only the first 5 of them.

We have now obtained a reduced system with matrices of size 6 and the total order is 6x5

since there are powers of z up to 5.

The final steps 4) and 5) are simple algebraic steps where we substitute back s for z and we obtain a final system in the variable s of order 30

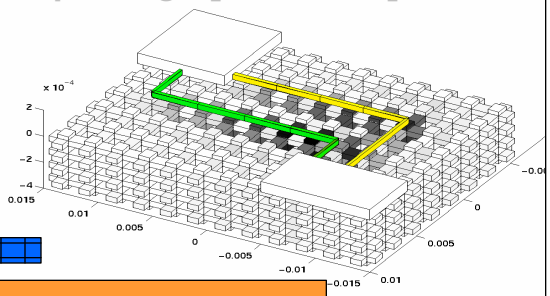
Details of all this can be found in [Daniel DAC02 and in the PhD thesis Daniel PhD04]

Outline

- Introduction
- Parameterized Model Order Reduction Classification
- From Field Solvers to Parameterized Models
- Case 1: Model Reduction with Geometrical Parameters
- Case 2: Model Reduction with Frequency Parameter
  - Preserving Stability and Passivity
  - Globally Convergent Interpolation
  - Example 2 wires on a MCM package, full-wave
- Conclusions

let's look how well it works on 2 wires over a multichip module package using fullwave and green function field solver

An implementation example: Two wires on a MCM package [D. DAC02]



package

- Discretize Maxwell equations in integral form using PEEC

$$[\, s\,\mathbf{L}(W,d,s) - \mathbf{R}(W,d,s)\,]\; x = b\; u$$

- NOTE: system matrices are frequency dependent because the substrate is handled by layered Green functions

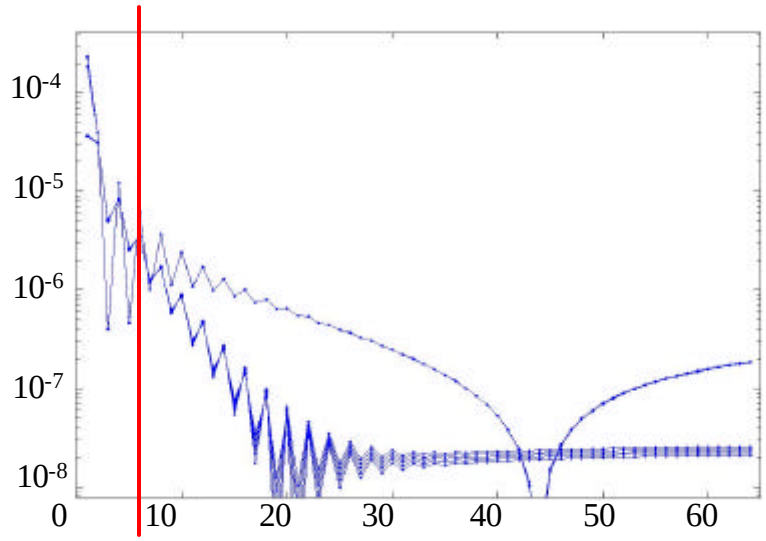
let's look how well it works on 2 wires over a multichip module package using fullwave and green function field solver

in this example the matrices L(W,d,s) and R(W,d,s) are frequency dependent because we used green functions to capture the package substrate

An implementation example: Two wires on a MCM package

isQED

FFT coefficients of L (s)



Here is a plot of some of the FFT coefficients of L(s)

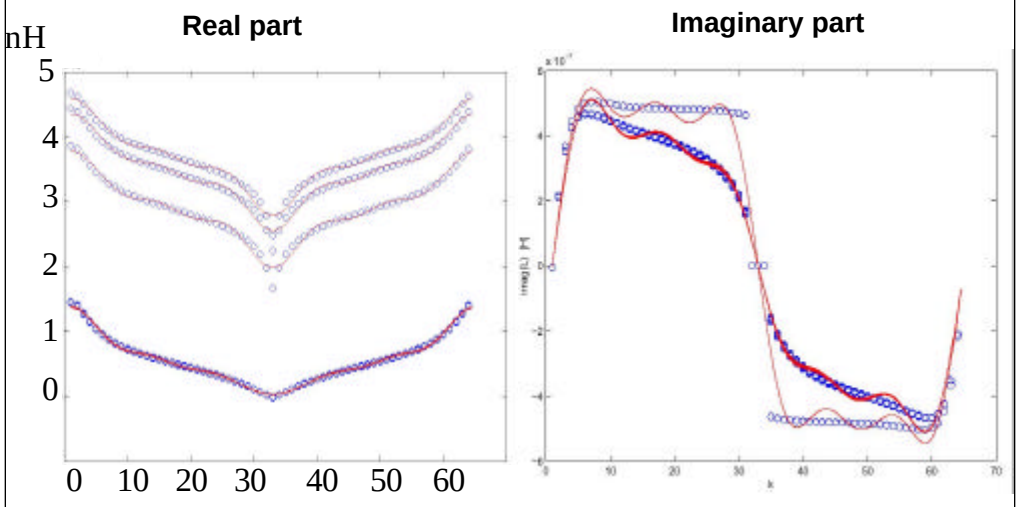
you can see that, as promised the coefficients drop by 3 orders of magnitude after they first 5

this is because typically the partial inductance and coefficient of potentials in L(s) have a smooth dependency on frequency

An implementation example: Two wires on a MCM package

isQED

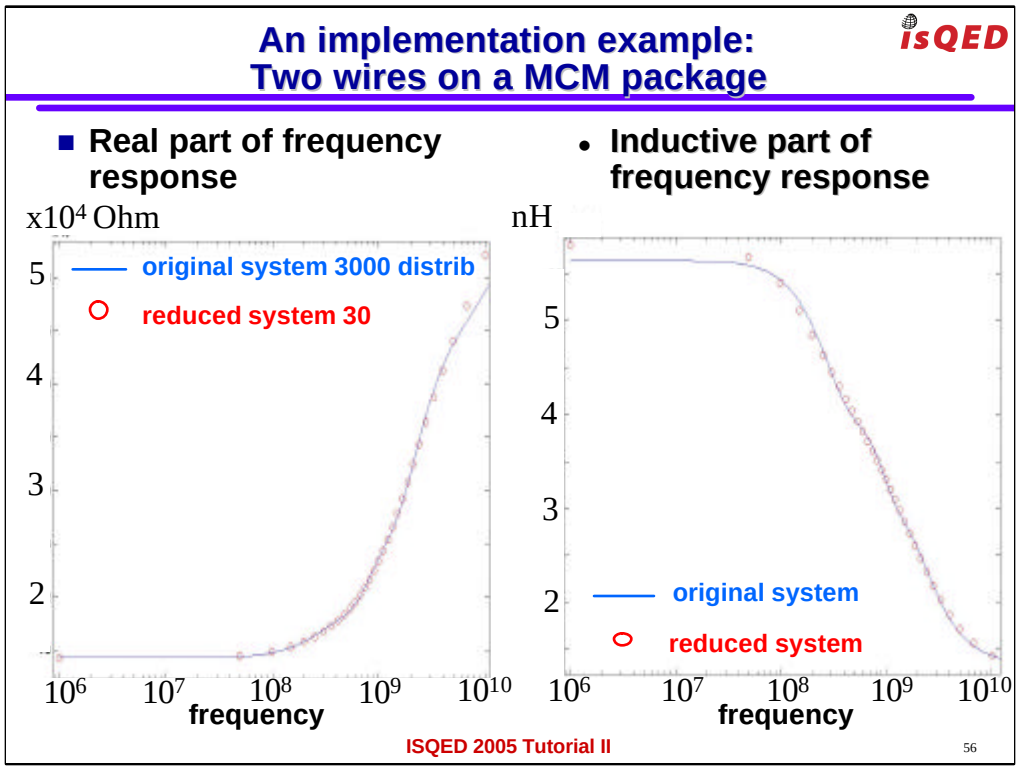
L *(s)* reconstructed from first 5 out of 64 FFT coefficients and compared to original L *(s)*



As a proof of the previous observation,

if we use only those first 5 fft coefficients to reconstruct the coefficients of L(s) we see quite a good matching

(compare blue circles with straight red line)



finally the overall transfer function of the reduced system of size 30 in red and the original system size 3000 match quite nicely

both in the real and in the imaginary part

Open issues for distributed systems

- **Guaranteeing positive realness relies on accuracy of the uniform interpolant. Hence if the matrices are NOT smooth, we might need a large order of the interpolant.**
  - **working on internal matrices might give smoother matrices**
- **Laguerre basis functions are an efficient choice since once can use FFT to calculate interpolation coeff.**
  - **However equally spaced points on the unit circle correspond to non-equally spaced points on the imaginary axis accumulating around a reference center frequency.**

There are several open issues in this approach.

For instance, let me repeat that we guarantee passivity by relying on accuracy of a globally convergent set of basis functions (e.g. Laguerre)

That can be achieved for smooth functions by few Interpolants.

But other formulations may not have smooth frequency dependency in their matrices and in that case we would need many more fft coefficients.

Another issue comes from the observation that the equally spaced points on the unit circle do not correspond to equally spaced points

the imaginary frequency axis. This might be desirable for some applications but not for others.

Conclusions

- From Field Solvers to large Non-Linearly Parameterized dynamical models of interconnect or RF inductors
- Model Reduction for Non-Linear dependency on geometrical parameters:
  - can use simple polynomial interpolation
  - moment matching and congruence transformation
  - e.g. RF inductor (no substrate)
- Model Reduction for Non-linear dependency in 's' (distributed systems):
  - e.g. full-wave, substrate layered green functions, high order basis functions
  - can use globally convergent interpolant implemented with FFT

In conclusion in this 3rd part of this tutorial we have shown an example of how one can construct large non-linearly parameterized dynamical models from the output of field solver based parasitic extractors such as PEEC.

The non-linear dependency when the parameters are geometrical can be handled using a simple polynomial interpolation approach

combined with a moment matching congruence transformation. We have shown an RF inductor example where an

EMQS PEEC field solver and the substrate was neglected so that the system matrices are not frequency dependent

Finally we have seen that if one uses fullwave solvers or green functions, the system matrices may be frequency dependent

and in that case one cannot use a simple polynomial interpolation because it is NOT globally convergent and it does not guarantee passivity.

For passivity one could use instead Laguerre basis that are globally convergent and their coefficients can be efficiently calculated using

a simple FFT.

References
from Field Solvers to Reduced Order Models

[PEEC Ruehli 74] A. E. Ruehli, "Equivalent circuit models for three dimensional multiconductor systems", Trans Microwave Theory and Techniques, 1974, v 22, p 216-221.

[Kamon Trans Packaging 98] M. Kamon, N. Marques, L. M. Silveira, J. K. White, Automatic Generation of Accurate Circuit Models", Trans on Packaging, Aug 1998.

[MIT course 6.336J] http://stellar.mit.edu/S/course/6/fa04/6.336J/

[MIT course 16.920J] http://ocw.mit.edu/OcwWeb/Aeronautics-and-Astronautics/16-920JNumerical-Methods-for-Partial-Differential-EquationsSpring2003/CourseHome/

[Grimme PhD97] Eric Grimme, “Krylov Projection Methods for Model Reduction", PhD Thesis, University of Illinois at Urbana-Champaign", 1997.

[PRIMA TCAD98] A. Odabasioglu, M. Celik, L. Pileggi , "PRIMA: passive reduced-order interconnect macromodeling algorithm", TCAD, v 17, n 8, Aug 1998.

[Phillips96] J. R. Phillips, E. Chiprout, D. D. Ling, "Efficient full-wave electromagnetic analysis via model-order reduction of fast integral transforms", DAC, June 1996.

[Willems72] J. C. Willems, “Dissipative Dynamical Systems”, Arch. Rational Mechanics and Analysis, 1972, v 45, p 321-393.

References
Parameterized Model Order Reduction

[Pullela97] S. Pullela N. Menezes L.T. Pileggi, "Moment-Sensitivity-Based Wire Sizing for Skew Reduction in On-Chip Clock Nets", Trans on CAD, v 16, n 2, p 210-215, Feb, 1997.

[Weile99] D. S. Weile E. Michielssen Eric Grimme K. Gallivan, "A Method for Generating Rational Interpolant Reduced Order Models of Two-Parameter Linear Systems", Applied Mathematics Letters, v 12, p 93-102, 1999.

[Prud'homme 02] C. Prud'homme, D. Rovas, K. Veroy, Y. Maday, A.T. Patera, G. Turinici, "Reliable real-time solution of parameterized partial differential equations: Reduced-basis output bounds methods", Journal of Fluids Engineering, 2002.

[Liu DAC99] Y Liu, L T. Pileggi, A J. Strojwas", Model Order-Reduction of RCL Interconnect Including Variational Analysis", DAC, p 201-206, 1999.

[Heydari ICCAD01] P. Heydari, M. Pedram, "Model Reduction of Variable-Geometry Interconnects Using Variational Spectrally-Weighted Balanced Truncation", ICCAD, 586—591, 2001.

[Rutenbar DAC02] H Liu, Singhee, A, Rutenbar, R.A., Carley, L.R, "Remembrance of circuits past: macromodeling by data mining in large analog design spaces", DAC 2002, p 437-42.

References
Parameterized Model Order Reduction (cont.)

[D. TCAD04] L. Daniel, C. S. Ong, S. C. Low, K. H. Lee, J. White, A Multiparameter Moment Matching Model Reduction Approach for Generating Geometrically Parameterized Interconnect Performance Models", IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, v 23, n 5, p 678-93, May 2004.

[D. BMAS03] L. Daniel, J. White, "Automatic generation of geometrically parameterized reduced order models for integrated spiral RF-inductors", Proceedings of the 2003 IEEE International Workshop on Behavioral Modeling and Simulation, p 18-23, San Jose, CA, 2003.

[D. APS04] L. Daniel, J. White, “Numerical Techniques for Extracting Geometrically Parameterized Reduced Order Interconnect Models from Full-wave Electromagnetic Analysis”, IEEE AP-S Symposium, June 2004.

[D. DAC02] L. Daniel, J. Phillips, "Model Order Reduction for Strictly Passive and Causal Distributed Systems", IEEE/ACM 39th Design Automation Conference, New Orleans, Jun 2002.

[D. PhD04] Luca Daniel, "Simulation and Modeling Techniques for Signal Integrity and Electromagnetic Interference on High Frequency Electronic Systems," Ph.D. Thesis, University of California at Berkeley, May 2003.