**Main Trend:** *Interdisciplinary* Analysis, Design, Diagnosis and Optimization of *Complex* Dynamical Systems

- Power delivery network for Integrated Circuit (IC) or city/state
- Heat sink for 3D IC
- Blood/Oil delivery network
- Building/Mechanical frames
Trend: Typical Development Flow for Complex Systems is Hierarchical

Biomedical Implant Lab-on-chip
e.g. for monitoring glucose level in the blood

Body-Area Clothing Network
(Chandrakasan, MIT)

Entire Cardiovascular System
e.g. for non-invasive diagnoses,

Nano-fluidic Channels
(Voldman, Han, MIT)

Energy Harvesting MEMs
(Kim, Lang MIT)

Arterial Bifurcation
(Vassilevski, INM; Olshanskii U. Houston)

Step 1: Need efficient PDE solvers for emerging technologies

Will it work? Will it work? Will it work?

Component” Design

ρ(u_t + (u ⋅ ∇)u) = −∇P + μ∇^2 u + f

μA \frac{\partial^2 u}{\partial t^2} - h A \frac{\partial u}{\partial x} - Y A \frac{\partial^2 u}{\partial x^2} = \frac{\partial f}{\partial x}
**Trend: Typical Development Flow for Complex Systems is Hierarchical**

Step 1: Need efficient PDE solvers for emerging technologies

Will it work? Will it work? Will it work?

Component Design

- Nano-fluidic Channels (Voldman, Han, MIT)
- Energy Harvesting MEMs (Kim, Lang MIT)
- Arterial Bifurcation (Vassilevski, INM; Olshanskii U. Houston)

Step 2: Need techniques that generate dynamical models automatically from PDE solvers and from component measurements

Step 3: Need efficient simulators for networks of interconnected dynamical system components

**Step 2: Automatic Generation of Parameterized Reduced Order Models for Dynamical Systems**

- automatically
- with field solver accuracy
- small (only 10-15 Eqns.)
- geometrically parameterized

**Step 2. Parameter. Model Order Reduction or Model Synthesis**

\[
\frac{dx}{dt} = A(p) x(t) + B u(t)
\]

(1 Million Eqns)

**Step 1. from Field Solvers “guts” or Measurement Data**

- Interconnect
- RF inductors
- RF Power Combiners
- Microwave Wilkinson Divider
Background: the Standard Projection Framework
(graphically)

\[ V^T \begin{bmatrix} \dot{x} \\ dz \end{bmatrix} = A(p) \begin{bmatrix} \dot{x} \\ V^Tbu \end{bmatrix} \]

orthonormal projection

\[ \frac{dz}{dt} = A(p) \hat{x}(t) + \hat{bu}(t) \]

Key Question: how do you choose \( V \)?

How to choose \( V \)?

\[ \dot{x} = A(p)x + Bu \]

Curve in 3-D space

Rotate coordinate system

Curve lies in 2-D plane

Projection: Rotation + Truncation
Approaches for picking $V$

<table>
<thead>
<tr>
<th>Basic Technique</th>
<th>Control/Systems</th>
<th>Mechanical Aero/Astro</th>
<th>Statistics, E.D.A.</th>
<th>E.D.A. Num Linear Algebra</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBR, Hankel</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Moore 81</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Glover 84</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>POD, KL, PCA, SVD</td>
<td>Wilcox</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peraire91,</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMTBR04</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Moment,</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matching</td>
<td>AWE90,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PVL</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Felmann94</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rutishauser95</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Many contributions over the last couple of decades...

<table>
<thead>
<tr>
<th>Linear Systems</th>
<th>Non-Linear Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Technique</td>
<td>Stability/Passivity</td>
</tr>
<tr>
<td>TBR, Hankel</td>
<td>Phillips Daniel02, Wong04</td>
</tr>
<tr>
<td>POD, KL, PCA, SVD</td>
<td>Wilcox Peraire91, PMTBR04</td>
</tr>
<tr>
<td>Moment, Matching</td>
<td>AWE90, Bond Daniel ICCAD08 indefinite $E, A^T$</td>
</tr>
<tr>
<td>Fitting, Optimiz Based, System ID</td>
<td>Gustav99</td>
</tr>
</tbody>
</table>
Generation of Parameterized Reduced Models for MRI coils optimization for SAR (heat) reduction

Choose values for coil geometrical parameters $p_n$

$n \leftarrow n + 1$

Eval. performance e.g. SAR($p_n$) using $x_n$

Call field solver $A(p_n)x_n = b_n$

Standard Optimization Loop e.g. for $\min_{p_n} \text{SAR}(p_n)$
MRI coils optimization for SAR (heat) reduction

Standard Optimization Loop e.g. for \[ \min_{p_n} \text{SAR}(p_n) \]

Choose values for coil geometrical parameters \( p_n \)

\[ n \leftarrow n + 1 \]

Solve reduced system

\[ V^T A(p_n) V \hat{x}_n = V^T b_n \]

Eval. performance e.g. SAR(\( p_n \)) using \( V \hat{x}_n \)

Parameterized Model Reduction of MRI coils can accelerate coil optimization for SAR (heat) reduction

<table>
<thead>
<tr>
<th>SAR computation</th>
<th>coil optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td>SemCad (time domain approach)</td>
<td>15h, 0.5GB</td>
</tr>
<tr>
<td>HFSS (FEM frequency domain)</td>
<td>2h, 250GB</td>
</tr>
<tr>
<td>Developing new Field Solvers [Hochman, Villena, Polimeridis, White, Daniel, ISMRM13]</td>
<td>3d 10cores offline +2.5-6.5s 30GB +800GB</td>
</tr>
<tr>
<td>Combined optimization loop with “on-demand construction” of Parameterized Reduced Model</td>
<td>~ 500-1000 conf/min</td>
</tr>
</tbody>
</table>


Parameterized Model Order Reduction can Accelerate the *Optimization* of Structural-Mechanical Systems

- Shape, property, dimension optimization for structural frames of bridges, buildings, automobiles, planes etc...

*Inverse Problems* in Complex Systems e.g. characterize water/oil/gas reservoirs from scattered seismic waves
**Inverse Problems in Complex Systems e.g. characterize water/oil/gas reservoirs from scattered seismic waves**

Standard Inverse Problem Loop e.g. identifying reservoir parameters

```
Choose values for coil geometrical parameters \( p_n \)

\[
\begin{align*}
n &\leftarrow n + 1 \\
\text{Call field solver} \\
A(p_n)x_n &= b_n \\
\text{Evaluate match of scattered waves with measurements}
\end{align*}
\]
```

---

**Parameterized Reduced Order Models for Acceleration of Inverse Problems e.g. characterize water/oil/gas reservoirs**

Standard Inverse Problem Loop e.g. identifying reservoir parameters

```
Choose values for coil geometrical parameters \( p_n \)

\[
\begin{align*}
n &\leftarrow n + 1 \\
\text{Solve reduced system} \\
V^T A(p_n) V x_n &= V^T b_n \\
\text{Evaluate match of scattered waves with measurements}
\end{align*}
\]
```
Inverse Problems in Complex Systems
e.g. non-invasive diagnosis of diseases of the cardiovascular system

Need Uncertainty Quantification Tools
(i.e. Stochastic Field Solvers) for Complex Systems

Need Efficient Uncertainty Quantification Tools for Microelectronic Manufacturing Process Variations
Need Efficient Uncertainty Quantification Tools for Microelectronic Manufacturing Process Variations

Problem Objectives for uncertainty quantification tools (i.e. Stochastic Field Solvers)

Given some statistical representation of the parameter variations $p$

Stochastic Field Solver

Generate some statistical representation for the output quantity of interest $y(p)$

\[ A(p)x(p) = b \]
\[ y(p) = c^T x(p) \]
**Sampling-Based Stochastic Field Solvers**

A(\(p\))x(\(p\)) = b

\(y(\textbf{p}) = c^T x(\textbf{p})\)

### e.g. computing **statistical moments**:

\[
E[y(p)^k] = \int y(p)^k \text{PDF}(p) dp \approx \sum_{i=1}^{N} w_i (c^T A(p_i)^{-1} b)^k \text{PDF}(p_i)
\]

(i.e. solve deterministic system at many sample points)

### e.g. computing **coefficients for a polynomial chaos expansion**:

\[
y(p) = \sum_{i=1}^{K} y_i \Psi_i(p)
\]

\[
y_i = \int y(p) \Psi_i(p) \text{PDF}(p) dp \approx \sum_{i=1}^{N} w_i c^T A(p_i)^{-1} b \Psi(p_i) \text{PDF}(p_i)
\]

Standard iteration for computing statistical moments in sampling-based stochastic field solver

- **Pick a sample** \(p_n\) from your favorite quadrature integration rule
- **Call field solver**

\[
\text{A}(p_n)\textbf{x}_n = \textbf{b}_n
\]

- **Update** \(E[y(p)^k]\), or poly chaos coefficients
- \(n \leftarrow n + 1\)
Standard iteration for computing statistical moments in sampling-based stochastic field solver

Pick a sample \( p_n \) from your favorite quadrature integration rule

\[ n \leftarrow n + 1 \]

Call field solver

\[ V^T A(p_n) \tilde{X}_n = V^T b_n \]

Update \( E[y|p_n] \), or poly chaos coefficients

---

**Large 2D Example: 10x10 grid of conductors**

- 100 conductors in a 10x10 grid
- Each discretized using 50 spatial elements (total 5,000 unknowns)
- Widths are independent Gaussian random variables (20% std. dev.)

**Large 2D Example: 10x10 grid of conductors**

**Time & Memory Results**

- **All comparisons are for the same 5% accuracy**

<table>
<thead>
<tr>
<th>Method</th>
<th>Time</th>
<th>Memory</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stochastic Collocation</td>
<td>40 hours (estim.)</td>
<td>200 MB</td>
<td>20,301 solves for 2nd order quadr.</td>
</tr>
<tr>
<td>Monte-Carlo</td>
<td>22 hours</td>
<td>200 MB</td>
<td>10,000 solves</td>
</tr>
<tr>
<td>SMOR [Moselhy Daniel10]</td>
<td>15 min speedup 90x</td>
<td>200 MB</td>
<td>size of reduced model: 112</td>
</tr>
</tbody>
</table>


**Very Large 3D Example:**

**Surface Roughness**

Example description:

- large square parallel plate capacitor (N=21,000 discretization elements)
- with surface roughness (Gaussian, size=20x20 correlation lengths),

### Very Large 3D Example: Surface Roughness

#### Time & Memory Results

- MonteCarlo or Stochastic Collocation can only be estimated, since example is too large (323 uncorrelated parameters)
- All comparisons are for the same estimated 5% accuracy

<table>
<thead>
<tr>
<th>Method</th>
<th>Time</th>
<th>Memory</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stochastic Collocation</td>
<td>(2000 hours)</td>
<td>5 GB</td>
<td>209,628 solves for 2nd order quadr.</td>
</tr>
<tr>
<td>Monte-Carlo</td>
<td>(150 hours)</td>
<td>5 GB</td>
<td>15,000 solves</td>
</tr>
<tr>
<td>SMOR [Moselhy Daniel10]</td>
<td>10 hours speedup 15x</td>
<td>5 GB</td>
<td>size reduced model: 997</td>
</tr>
</tbody>
</table>


---

### Conclusions

- **Trend toward complex systems**
- **Handle complexity with hierarchy of system components**
- **Parameterized Model Reduction** generates models for system components that both:
  - Preserve accuracy of PDE solvers
  - Are fast to evaluate for many values of design parameters
- **Can use Parameterized Model Reduction in Complex Systems to accelerate**:
  - **Optimization** of component parameters (100x speedups)
  - **Inverse problems** or non-invasive diagnosis problems (speedups 50x)
  - **Uncertainty quantification** solvers (15x-90x speedups)
