SANDIA REPORT
SAND2016-9621
Unlimited Release
Printed July, 2016

Proceedings of the 2016 Parameterized Reduced Order Modeling Workshop

Matthew R. W. Brake, Bogdan I. Epureanu, and Harry R. Millwater

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-mission laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2016-9621
Unlimited Release
Printed July, 2016

Proceedings of the 2016 Parameterized Reduced Order Modeling Workshop

Matthew R. W. Brake[1], Bogdan I. Epureanu[2], and Harry R. Millwater[3]

[1]: Sandia National Laboratories
Component Science and Mechanics
P.O. Box 5800; M.S. 1070
Albuquerque, New Mexico 87185-1070

[2]: University of Michigan
Department of Mechanical Engineering
2350 Hayward
Ann Arbor, Michigan 48109

[3]: University of Texas, San Antonio
Department of Mechanical Engineering
One UTSA Circle
San Antonio, Texas 78249

Abstract

The 2016 Parameterized Reduced Order Modeling (PROM) Workshop was held in June, 2016, in Albuquerque, NM. This workshop included 30 researchers who took part in a two day discussion regarding the state of the art for PROMs, complimentary reduced order modeling (ROM) theories, and discussion of the future directions of PROM research. The goals of the workshop were three-fold: to assess the relative accuracy, efficiency, and merits of the different PROM methods; to discuss the state of the art for ROMs and how PROMs can benefit from these advances; and to define the pressing challenges for PROMs and a path for future research collaborations.

ACKNOWLEDGMENTS

The authors would like to thank Eliot Fang for his support of this research activity, and Michaela Negus for helping make it happen.

CONTENTS

NOMENCLATURE

FE	Finite Element
HD	Hyper Dual
POD	Proper Orthogonal Decomposition
PROM	Parameterized Reduced Order Model
ROM	Reduced Order Model

1. BACKGROUND

The focus of the 2016 Parameterized Reduced Order Modeling (PROM) Workshop is the development and accuracy of existing PROM tools. A number of theories for developing PROMs have recently been put forward [1, 2, 3, 4, 5, 6, 7], leading to the goals of this workshop:

- Assessing the relative accuracy, efficiency, and merits of different PROM approaches
- Discussing the state-of-the-art for reduced order models (ROMs) and how PROMs can benefit from these advances
- Defining the pressing challenges for PROMs and a path for future research collaborations.

The impetus for PROMs is found in modern engineering analysis, which must take into account the effects of aleatoric (parametric) uncertainty in the analysis of a system. As a real system is manufactured, part-to-part variations are introduced that can have significant ramifications on the functionality of the system. Thus, in order to account for these variations at the design stage, a methodology is needed to assess the performance of many (often thousands) of permutations of a design to qualify the performance of a manufactured system.

The most common method to simulate the performance of a system is via high fidelity modeling, such as using the finite element (FE) method. High fidelity computational simulations often can provide very accurate predictions; however, they have a very high computational cost. In order to develop simulations that are both efficient and sufficiently accurate, ROMs often are used as surrogates for a full order model in order to decrease the computational expense of analysis.

To model the perturbations that are found in manufactured systems without a systematic and efficient reduced order approach would be prohibitively expensive. For example, consider a scenario where it takes several weeks to develop a high quality mesh for one relatively simple component. To quantify the aleatoric uncertainty associated with manufacturing, thousands of perturbations of the ideal geometry are necessary, and each likely requires a new mesh. Even with factoring in time saved from some automation of the process, the number of man hours required to construct these meshes is on the order of decades. In addition, the computational time to analyze all of these models is on the order of years assuming that an entire super computer can be dedicated to the analysis. Clearly, decades of time are infeasible constraints to be incorporated into a design cycle. One method of accounting for these perturbations is to create a PROM of the system.

In what follows, the details of the 2016 PROM Workshop are presented. In Section 2, the programmatic details for the organization of the 2016 PROM Workshop are discussed. In Section 3, both presentations and discussion of presentations from the 2016 PROM Workshop are given. Finally, in Section 4, a summary of the discussion from the plenary section of the 20016 PROM Workshop is given, including the 12 main themes that were identified during the 2016 PROM Workshop's presentation sessions.

2. WORKSHOP ORGANIZATION

The workshop was held at the COSMIAC facility (located at 2350 Alamo Ave SE, #100, Albuquerque, NM), which is a facility jointly managed by the University of New Mexico and Air Force Research Laboratories. The workshop itself spanned two days: June 2nd and 3rd, 2016. To achieve the goals of the workshop, it was organized into five sessions: one session overviewing recent advances in PROM, two sessions highlighting recent advances in ROMs, one session consisting of PROM tutorials and solutions to a round robin problem distributed to several attendees in advance of the workshop, and one session focused on discussing future directions of PROM research.

2.1 Schedule

The agenda for the workshop followed:

June 2nd

Greetings	
7:30 – 8:15	Coffee and bagels
8:15 – 8:30	Welcome and introduction to the workshop

Session 1	
8:30 – 9:05	Bogdan Epureanu, NX-PROMs
9:05 – 9:40	Harry Millwater, Overview of the ZFEM Multicomplex Finite Element Method
9:40 – 10:15	Matthew Brake, Hyper Dual Numbers
10:15 – 10:50	Matthew Bonney, Meta-Modeling

11:00 – 1:00 Lunch

Session 2	
1:00 – 1:30	Laura Mainini, Multistep ROM Strategy to Support Real Time Data to Decisions
1:30 – 2:00	Judy Brown, Quantifying the Impact of Material-Model Error on Macroscale Quantities
2:00 – 2:30	Ben Pacini, Experimental ROMs

2:30 – 2:45 Break

Session 3	
2:45 – 3:15	Gustavo Castelluccio, Multiscale Modeling Applications
3:15 – 3:45	Manuel Garcia, 2-Dimensional Curvilinear Progressive Fracture Using Multicomplex FEM
3:45 – 4:15	Rob Kuether, Viscoelastic ROMs

June 3rd

7:30 – 8:00 Coffee and Bagels

Sessions 4 and 5	
8:00 – 8:05	Overview of Day 2
8:05 – 8:45	Jau-Ching, NX-PROM Round Robin and Tutorial
8:45 – 9:25	Jeff Fike, Hyper Dual Number Round Robin and Tutorial
9:25 – 10:05	Matthew Bonney, Meta-Modeling Round Robin and Tutorial
10:05 – 10:35	Andres Aguirre, A Library for Multi-Complex and Multi-Dual Numbers
10:35 – 10:45	Break if time allows
10:45 – 12:30	Plenary Discussion on the Future of PROM Research

2.2 Participants

Thirty researchers attended this invitation only workshop:

Attendee	Institute	Email
Andres Aguirre	EAFIT	aaguirr2@eafit.edu.co
Arturo Montoya	UT San Antonio	Arturo.Montoya@utsa.edu
Ben Pacini	Sandia	brpacin@sandia.gov
Bogdan Epureanu	Michigan	epureanu@umich.edu
Brenton Taft	AFRL	brenton.taft@us.af.mil
Brian Robbins	Sandia	barobbi@sandia.gov
David Day	Sandia	dmday@sandia.gov
Derek Hengeveld	AFRL	dhengeveld@loadpath.com
Garth Reese	Sandia	gmreese@sandia.gov
Gustavo Castelluccio	Sandia	gmcaste@sandia.gov
Harry Millwater	UT San Antonio	harry.millwater@utsa.edu
Jau-Ching Lu	Michigan	jauching@umich.edu
Jeffrey Fike	Sandia	jafike@sandia.gov
Joe Bishop	Sandia	jebisho@sandia.gov
Jordan Massad	Sandia	jemassa@sandia.gov
Judy Brown	Sandia	judbrow@sandia.gov
Kevin Irick	AFRL	kevin.irick.1.cttr@us.af.mil
Kevin Troyer	Sandia	kltroye@sandia.gov
Kirsten Peterson	Colorado State	kirstenpeterson999@gmail.com

Laura Mainini	MIT	lmainini@mit.edu
Lynn Munday	Sandia	lmunday@hotmail.com
Manuel Garcia	EAFIT	mgarcia@eafit.edu.co
Matthew Bonney	Wisconsin	msbonney@wisc.edu
Matthew Brake	Sandia	mrbrake@sandia.gov
Matthew Castanier	USARMY TARDEC	matthew.p.castanier.civ@mail.mil
Mikhail Mesh	Sandia	mmesh@sandia.gov
Pania Newell	Sandia	pnewell@sandia.gov
Rob Kuether	Sandia	rjkueth@sandia.gov
Scott Grutzik	Sandia	sjgrutz@sandia.gov
Vit Babuska	Sandia	vbabusk@sandia.gov

3. PROM PRESENTATIONS

In what follows, only the presentations from the PROM talks are reproduced. Many of the presentations from the second and third sessions are in the process of being published, and are thus withheld to protect the authors' interests.

3.1 Session 1 Presentations – PROM Methodology Overview

The first session of the 2016 PROM workshop focused on presenting the four main branches of PROM research. This set of four presentations is at a higher level to both introduce the methodologies and to demonstrate their strengths and weaknesses.

3.1.1 NX-PROMs, Bogdan Epureanu

The Next Generation PROMs (NX-PROMs) [1, 2] and their precursors developed by Bogdan Epureanu et al. at the University of Michigan and Matthew Castanier of the US Army TARDEC, represent some of the first work within the field of PROMs. The premise of this family of PROMs is that four perturbations of a model in a dimension of interest are calculated. These perturbations are then combined, using a special weighting function formulated based off of the element formulation from the high fidelity model, to create a finite difference-based PROM. This approach has proven very effective for single variable expansions, but more work is needed for multivariate expansions.

UNIVERSITY OF MICHIGAN

Next-Generation Parametric Reduced Order Models

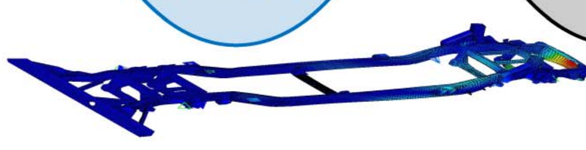
Bogdan I. Epureanu
Matt Castanier
Jau-Ching Lu
Sung Kwon Hong

Overview: Objectives

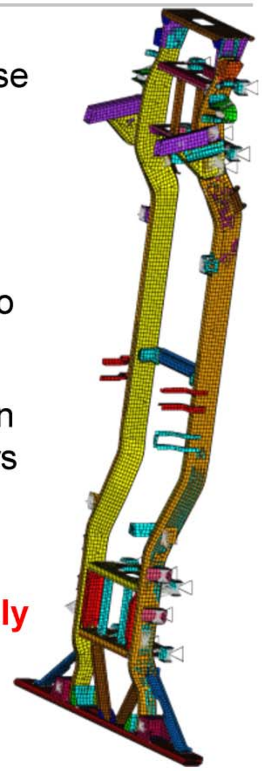
Develop new, advanced modeling and simulation capabilities for dynamic analysis of very complex (nonlinear) structures

Develop signal processing and damage identification technology for fast and accurate predictions, RBDO, monitoring, prognosis, and CBM

Enhance design capabilities for vehicle modifications, up-armoring, integrated monitoring, advanced hybrid-material structures



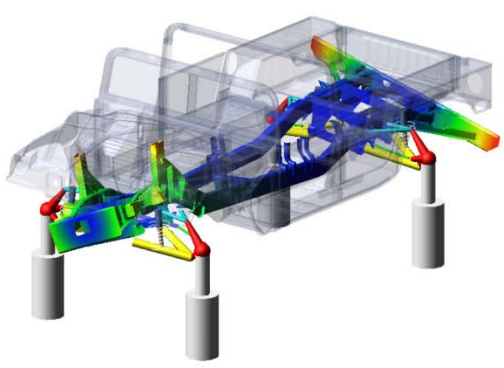
Overview: Challenges



1. Component-level **uncertainties**, **design changes** and **damages** affect system-level structural response
2. Innovative designs, RBDO, guidance for repairs, evaluation and measurement after repairs require fast re-analysis to **reduce computational cost**
3. Cracks create **nonlinear dynamics** (much harder to tackle) and crack lengths are difficult to **identify**
4. Monitoring, evaluation, measurement and inspection require **system information** which relies on sensors and signal processing, which is difficult for complex structures with both **uncertainties** & **damage**
5. Element-level structural characteristics have a **highly nonlinear dependence** on parameters for parametric reduced order models

Program Overview: Solutions

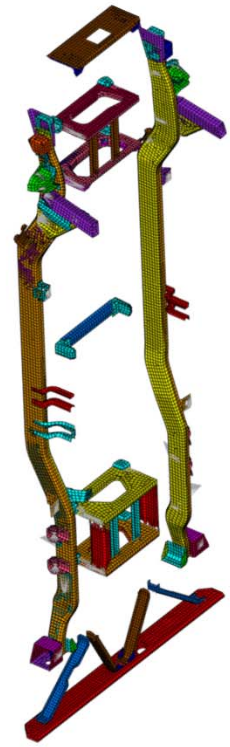
1. Develop novel **substructure-based methods** to construct ROMs
2. Create **new algorithms** to allow component-level models to be easily plugged back into ROM and enable fast re-analysis
3. Develop **novel parameterizations** for ROMs to treat element nonlinearity (**the next-generation PROMs**)
4. Develop new **signal processing** & **damage identification** methods



- ❑ Develop **new algorithms for mode approximations** (BMAs) to characterize the dynamics of complex nonlinear structures
- ❑ Develop **new signal processing technology** by generalizing EIDV for PROMs and BMAs (cracked complex structures with structural variability)
- ❑ Develop **new crack identification algorithms** which are enabled by the new PROMs and signal processing algorithms

Progress and Agenda

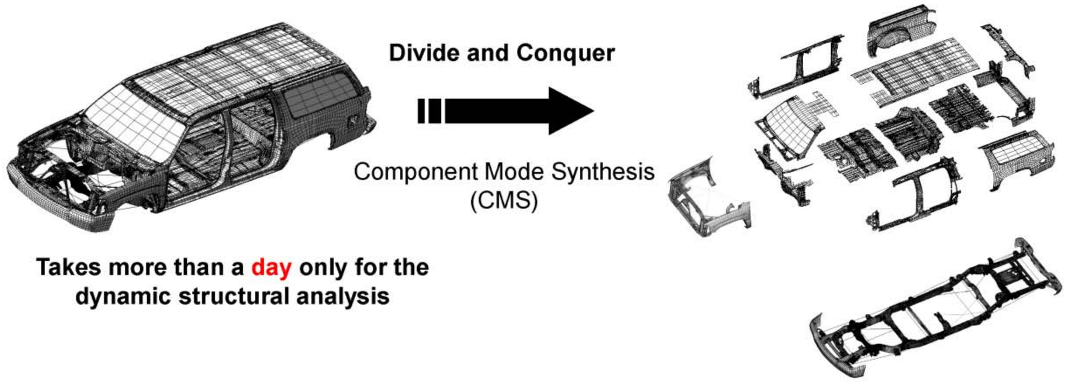
- New **signal processing algorithms**
  - Complex structures with variability
  - Oversampled locations in a frequency range
  - Damage detection/identification
  - **Results** Complex HMMWV frame and simple plate
    Optimal/minimum sensory data
- **Next-Generation PROMs**
  - Computational speed-up (modeling architecture)
  - Increased/Enhanced robustness
  - Key enhancements of element-level characteristics
  - **Results** Complex HMMWV frame and simple plate
    Integrated PROMs with optimal signal processing
- New **optimization techniques for up-armoring**
  - Implementation of vehicle performance optimization
  - Minimization of impacts on passenger comfort
  - Minimization of armor effects on vehicle/passengers
  - **Results** Optimization of attachment points
    Direct detection of "weakest point" for impacts



From ROMs to PROMs

Reduced Order Models: Overview

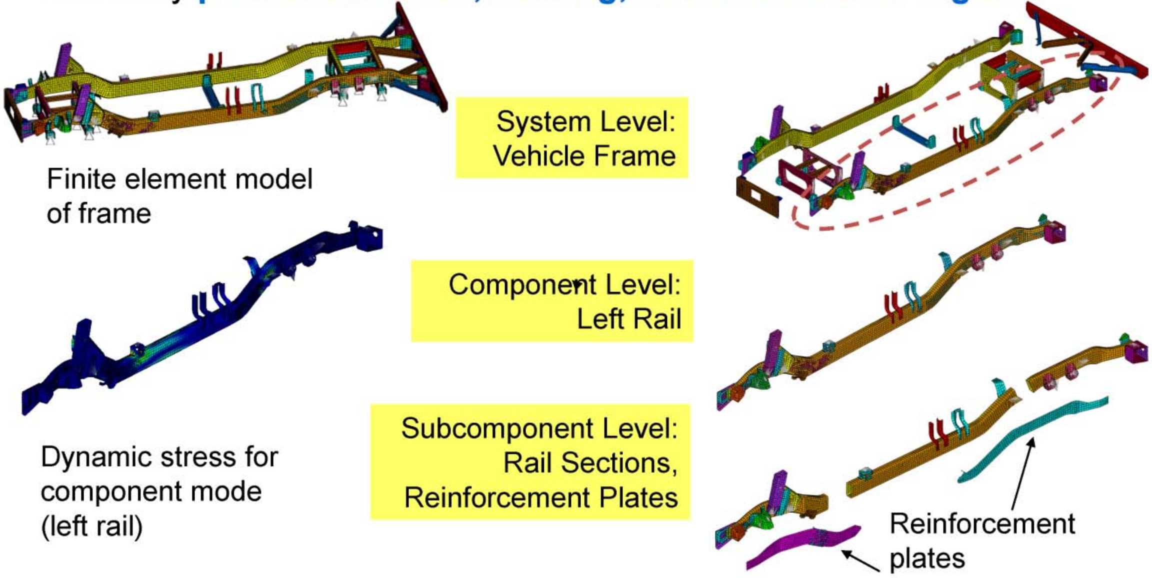
- Dynamic analysis of **invariant** complex structures
  - Projection by lower modes of the large-scale eigenvalue problem



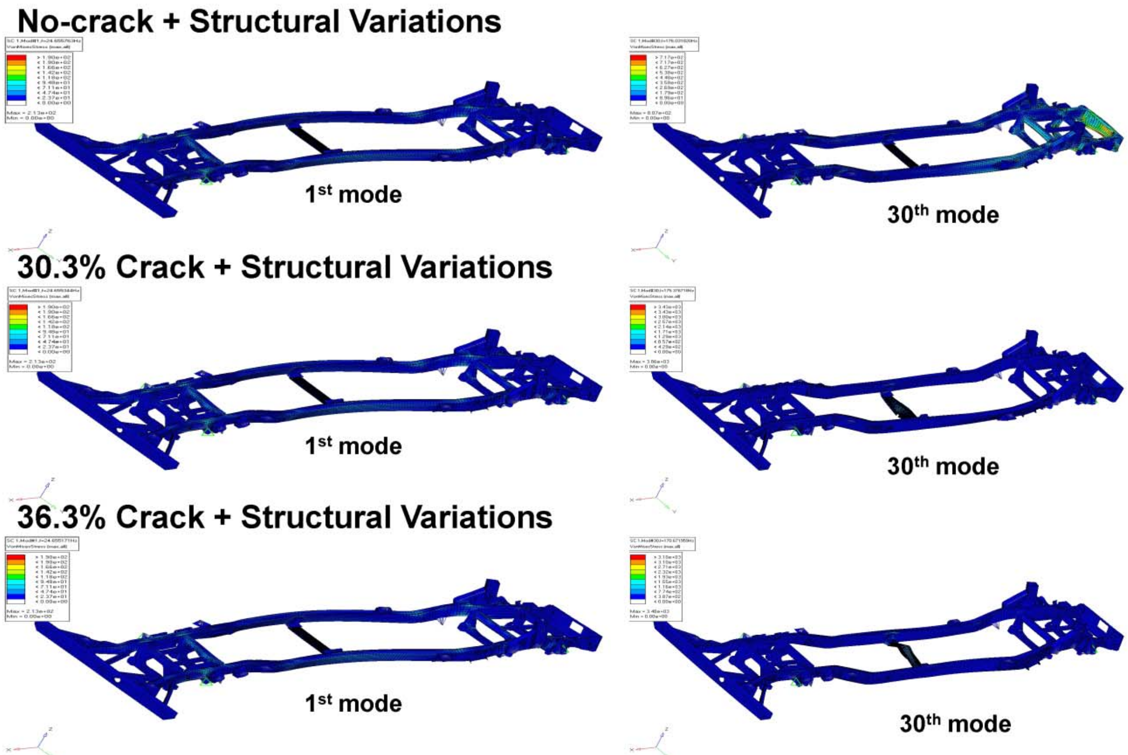
- Dynamic analysis of **damaged** complex structures
  - Projection by **proper basis** of the large-scale eigenvalue problem
  - Proper basis can be defined for **each damage type**: **cracks, dents and other structural variations** of complex structures

Reduced Order Models: Sub-Structuring

- Assemble ROMs of system (e.g., frame) from finite element **analyses of components and subcomponents**
- Efficiently **predict vibration, loading, stress in critical regions**



Why PROMs: Local Variations Lead to Global Changes



Reduced Order Models: Parametric Models (PROMs)

- Enable **fast re-analysis**
- Subcomponent dynamics evaluated at **sampled parameter values**
- System-level **response expressed as function of parameter changes**

- Global PROM (Parametric Reduced Order Models)

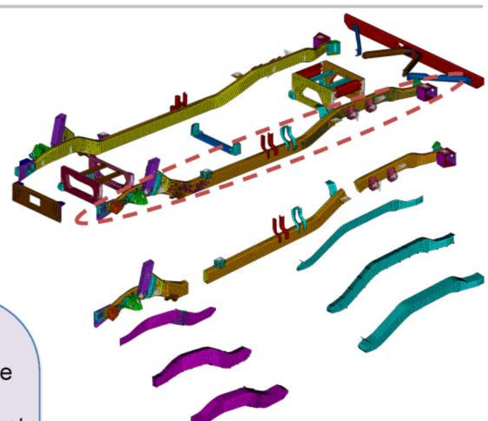
- Balmès: Collected eigenvectors at sampled points in the parameter space
  Problem: Overhead computational cost is very high to get the modal matrix to project the FE model

- CMB-PROM (Component Mode Basis PROM)

- Zhang (2005): Collect fixed interface normal modes and global interface mode and project the FE model
  Problem: Global analysis not substructural analysis

- Component PROM

- Park (2008): Developed PROM for substructural analysis
  Problem: A single design component is tackled



Reinforcement plate thickness changes

Multi-component PROM (MC-PROM)

Reduced Order Modeling: Framework

Analysis Framework

- Divide the global structure into substructures with or without damage
- Apply Craig-Bampton CMS (CB-CMS) for substructures which do not have any damage or variability
- Apply MC-PROM for the substructures with model variations (e.g. uncertainties)
- Apply BFA for cracked structure analysis

Assemble substructures to get the system-level response (entire structure)

Core technologies

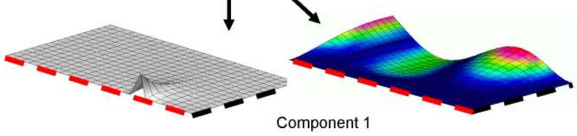
- CB-CMS
- Multi-Component PROM
- SMC-CMS
- Bilinear Frequency and Bilinear Mode Approximations

Efficient framework for damage detection and for structural predictions

Next-Generation PROMs

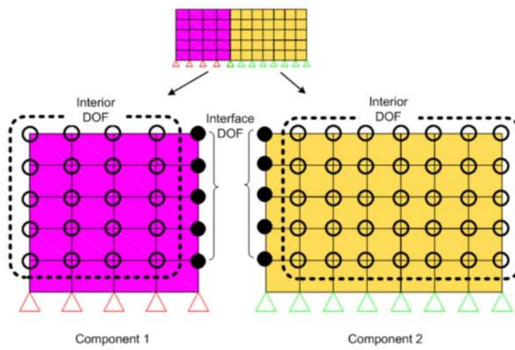
Next Generation PROMs: Basics of CMS

$$\begin{bmatrix} \mathbf{u}_A \\ \mathbf{u}_O \end{bmatrix}_{(i)} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{\Psi}_C & \mathbf{\Phi}_N \end{bmatrix}_{(i)} \begin{bmatrix} \mathbf{u}_A \\ \mathbf{p}_O \end{bmatrix}_{(i)}$$



Static modes: as many as interface DOF

Selected fixed-interface normal nodes



- i th component mass and stiffness matrix and force vectors

$$\mathbf{M}_i^{CBCMS} = \begin{bmatrix} \mathbf{m}_i^{C} & \mathbf{m}_i^{CN} \\ \mathbf{m}_i^{NC} & \mathbf{m}_i^{N} \end{bmatrix} \quad \mathbf{K}_i^{CBCMS} = \begin{bmatrix} \mathbf{k}_i^{C} & 0 \\ 0 & \mathbf{k}_i^{N} \end{bmatrix} \quad \mathbf{F}_i^{CBCMS} = \begin{Bmatrix} \mathbf{f}_i^{C} \\ \mathbf{f}_i^{N} \end{Bmatrix}$$

- Superscript C: Constraint part
- Superscript N: Internal part
- Subscript i: i th component

Key: How to create a good transformation matrix in the presence of parameter variability ?

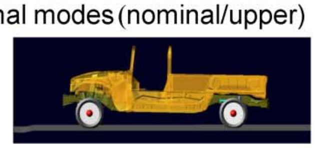
Next Generation PROMs: Transformation Matrix

❑ **Previous Approach**: static constraint modes and fixed-interface normal modes for the nominal case and the upper limit case

$$\mathbf{T}_{PROM} = \begin{bmatrix} \mathbf{T}_C & \mathbf{T}_N \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{\Psi}_C^0 & \mathbf{\Psi}_C^U & \mathbf{\Phi}_N^0 & \mathbf{\Phi}_N^U \end{bmatrix}$$

Static constraint modes ← → Fixed-interface normal modes

$\mathbf{T}_C$: Constraint modes (nominal/upper) $\mathbf{T}_N$: Fixed interface normal modes (nominal/upper)



❑ **Challenges**

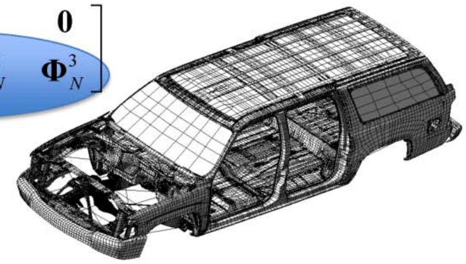
1. The transformation matrix (and the mass matrix): **only information from substructures with nominal and upper limit** (parameters) while stiffness matrices parameterized by 3rd order Taylor series
2. If the normal mode set $\mathbf{T}_N$ is not truncated, the **size of PROM** mass and stiffness matrices can be bigger than full order matrices
3. Taylor series needs **large number of matrix operations**, and the **accuracy of the parameterization** is limited

Next Generation PROMs: Enhancements

- **Improved accuracy/performance of the transformation matrix**
  - Enhance accuracy and robustness of **subspace of normal modes**
  - Enhance capability to capture **subspace of constraint modes**

- **Example: subspace of normal modes**

$$\mathbf{T}_{PROM} = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{\Psi}_C^0 & \mathbf{\Psi}_C^1 & \mathbf{\Psi}_C^2 & \mathbf{\Psi}_C^3 & \mathbf{\Phi}_N^0 & \mathbf{\Phi}_N^1 & \mathbf{\Phi}_N^2 & \mathbf{\Phi}_N^3 \end{bmatrix}$$



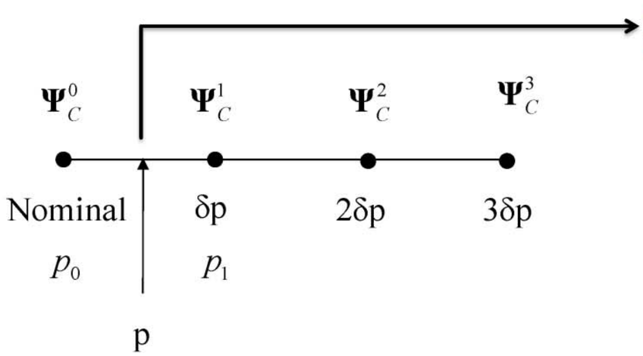
$$\left[\mathbf{\Phi}_N^0\ \mathbf{\Phi}_N^1\ \mathbf{\Phi}_N^2\ \mathbf{\Phi}_N^3\right] = \left[\mathbf{U}\ \ \tilde{}\ \right]\mathbf{\Sigma}\mathbf{V}^T \rightarrow \mathbf{T}_{PROM} = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{I} & \mathbf{I} & \mathbf{0} \\ \mathbf{\Psi}_C^0 & \mathbf{\Psi}_C^1 & \mathbf{\Psi}_C^2 & \mathbf{\Psi}_C^3 & \mathbf{U} \end{bmatrix}$$

$$\mathbf{M}_{PROM} = \mathbf{T}_{PROM}^T \mathbf{M}_{FEM} \mathbf{T}_{PROM} \qquad \mathbf{K}_{PROM} = \mathbf{T}_{PROM}^T \mathbf{K}_{FEM} \mathbf{T}_{PROM}$$

Next Generation PROMs: Enhancements

- **Example: subspace of constraint modes**

  Enhance robustness: reduce the number of static constraint modes



$$\bar{\mathbf{\Psi}}_C = \left(\frac{p_1 - p}{p_1 - p_0}\right)\mathbf{\Psi}_C^0 + \left(\frac{p - p_0}{p_1 - p_0}\right)\mathbf{\Psi}_C^1$$

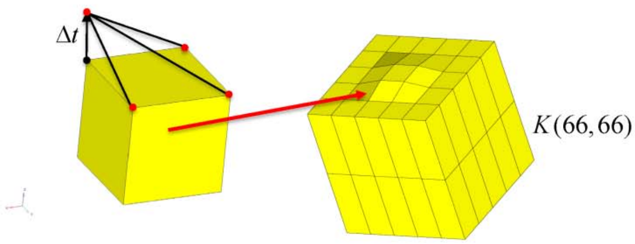
new static constraint modes

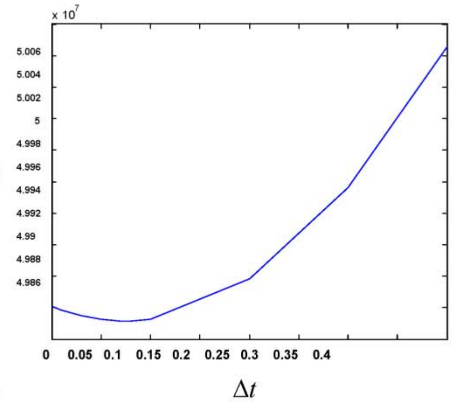
- **Key Feature:**

  New implementation **without reconstructing** PROM matrices
  Calculations: **just a simple linear combination** of partitions of the initially generated PROM matrices

Next Generation PROMs: Enhancements

Nonlinearity of Stiffness Matrix





Δt **:** thickness variation at a node

FEM formulation for stiffness matrix

$$\mathbf{K} = \int_V \mathbf{B}^T \mathbf{D}\mathbf{B}\, dV = \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} \mathbf{B}^T \mathbf{D}\mathbf{B}\, d\xi d\eta d\zeta = \sum_{i=1}^{8}\sum_{j=1}^{8}\sum_{k=1}^{8} W_i W_j W_k \mathbf{B}^T\left(\xi_i,\eta_j,\zeta_k\right)\mathbf{D}\mathbf{B}\left(\xi_i,\eta_j,\zeta_k\right)\det\left(\mathbf{J}\left(\xi_i,\eta_j,\zeta_k\right)\right)$$

Quadratic term of inverse of Jacobian included in denominator

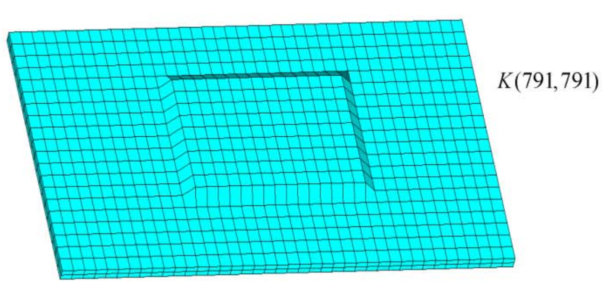
× Inverse of Jacobian included in denominator

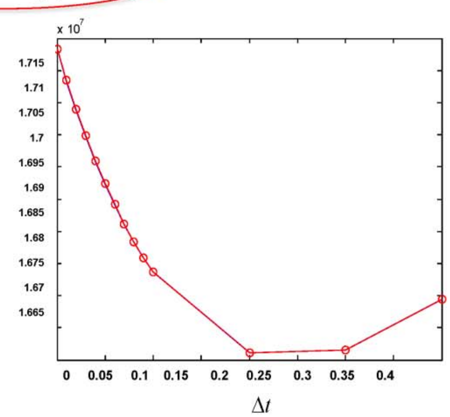
= **Cubic term of Jacobian inverse** (volume variations)

Next Generation PROMs: Enhancements

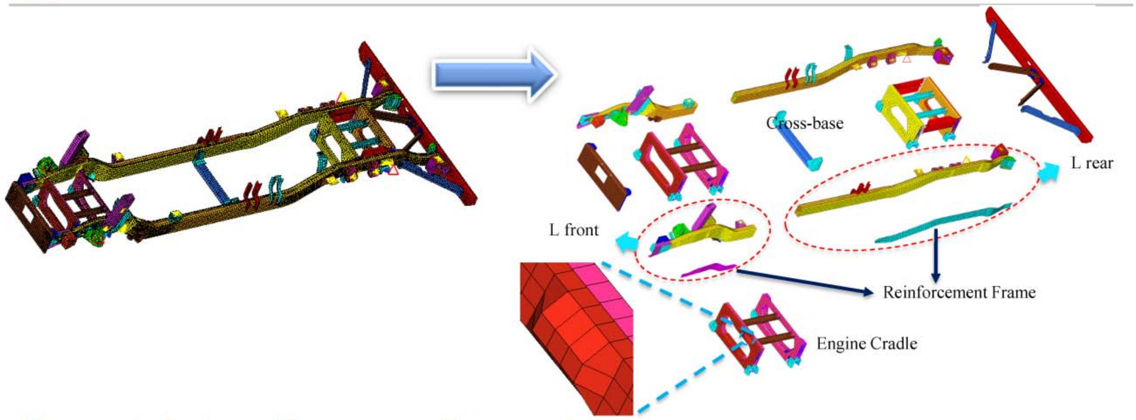
$$\mathbf{K}\left(p_0+\Delta p\right) \approx c_0\mathbf{K}_0 + c_1\mathbf{K}_1\Delta p + c_2\mathbf{K}_2\Delta p^2 + c_3\mathbf{K}_3\Delta p^3$$

$$\mathbf{K}\left(p_0+\Delta p\right) \approx \frac{c_0\mathbf{K}\left(p_0\right) + c_1\frac{\Delta p}{p_0}\mathbf{K}\left(p_0+\frac{\delta p}{p_0}\right) + c_2\left(\frac{\Delta p}{p_0}\right)^2\mathbf{K}\left(p_0+2\frac{\delta p}{p_0}\right) + c_3\left(\frac{\Delta p}{p_0}\right)^3\mathbf{K}\left(p_0+3\frac{\delta p}{p_0}\right) + c_4\left(\frac{\Delta p}{p_0}\right)^4\mathbf{K}\left(p_0+4\frac{\delta p}{p_0}\right)}{\left(1+\frac{\Delta p}{p_0}\right)\left(1+\frac{1}{2}\frac{\Delta p}{p_0}\right)\left(1+\frac{1}{3}\frac{\Delta p}{p_0}\right)}$$





Results: Vehicle Frame: Dents & Thickness Variations



Uncertainty + Damage Scenario

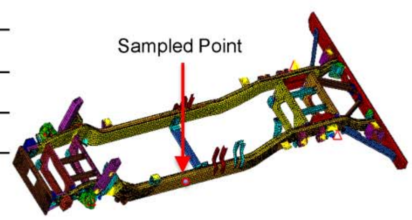
Each reinforcement frame has **thickness variation**

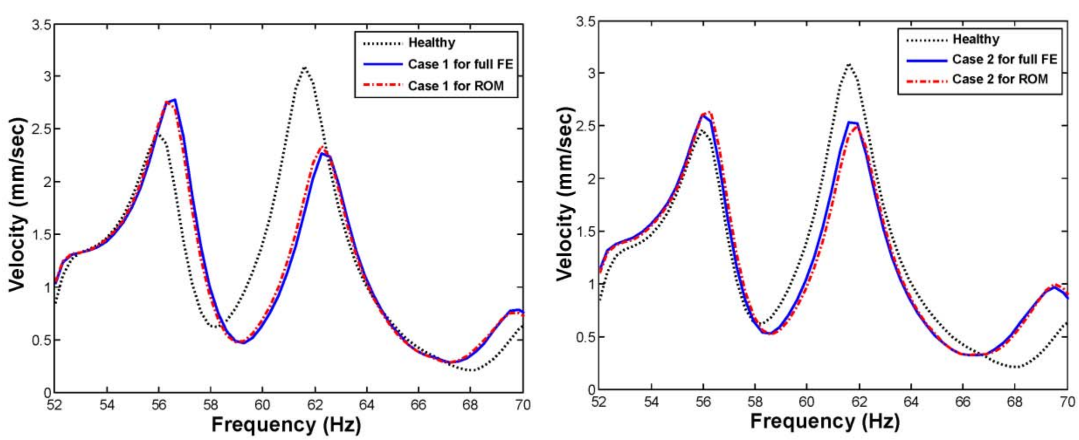
Engine cradle has a **dent**

Substructure	Thickness, Case1	Thickness, Case2
L rear	3.04 mm → 4.63 mm	3.04 mm → 5.58 mm
L front	3.04 mm → 5.38 mm	3.04 mm → 4.09 mm

Results: Vehicle Frame: Forced Response

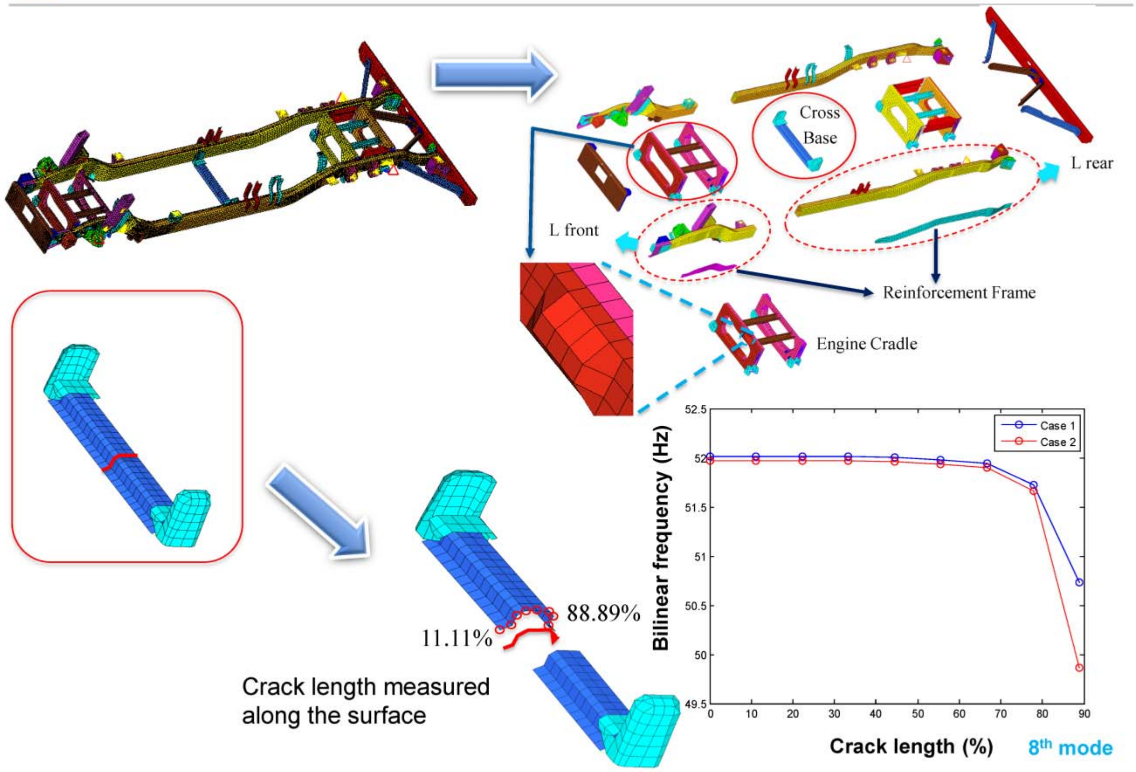
	Full order model		ROM
System DOF	119,808	$\times\frac{1}{3}$	2,420
Initial Analysis Time	60,125 (sec.)	→	21,956 (sec.)
Reanalysis Time	**60,125** (sec.)	→ $\times\frac{1}{100}$	**595** (sec.)



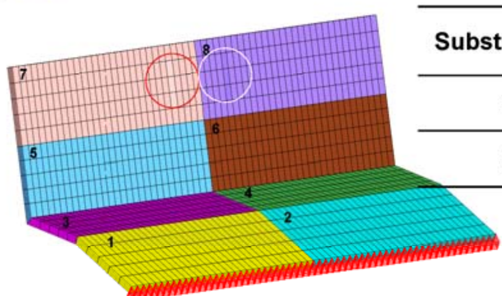


Forced response for cases 1 and 2

Results: Frame: Dents, Crack & Thickness Variations



Results: L-Shaped Structure: Solid Elements

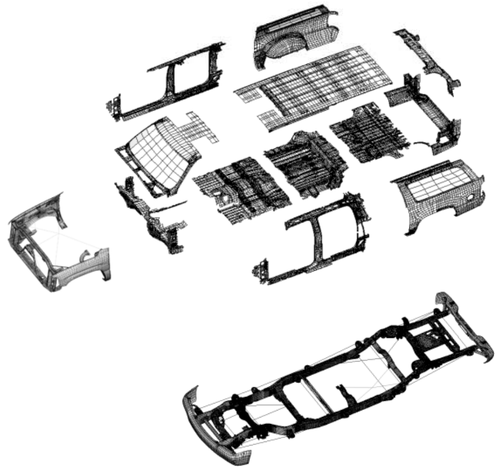


Substructure	Case 1 Thickness Variation	Case 2 Thickness Variation
7	10 mm → 10.12 mm	10 mm → 10.06 mm
8	10 mm → 10.12 mm	10 mm → 10.06 mm

Mode	Healthy (FEM)	Damaged (FEM)	Case 1 (previous PROM)	Case 1 (new PROM)
1	23.276778	23.173331	22.906386	23.173995
2	40.108421	40.255157	40.112458	40.256144
3	96.111376	95.732472	95.393447	95.733345
4	112.16800	112.99557	112.87563	112.99644
5	155.84310	**156.04702**	**155.77790**	**156.04771**
		Case 2 (FEM)	**Case 2 (previous PROM)**	**Case 2 (new PROM)**
1	23.224843	23.223324	23.062921	23.224843
2	40.182215	40.181979	40.407017	40.182215
3	95.923606	95.921692	95.307588	95.923606
4	112.58054	112.58008	113.88731	112.58054
5	155.95068	**155.94913**	156.18793	**155.95068**

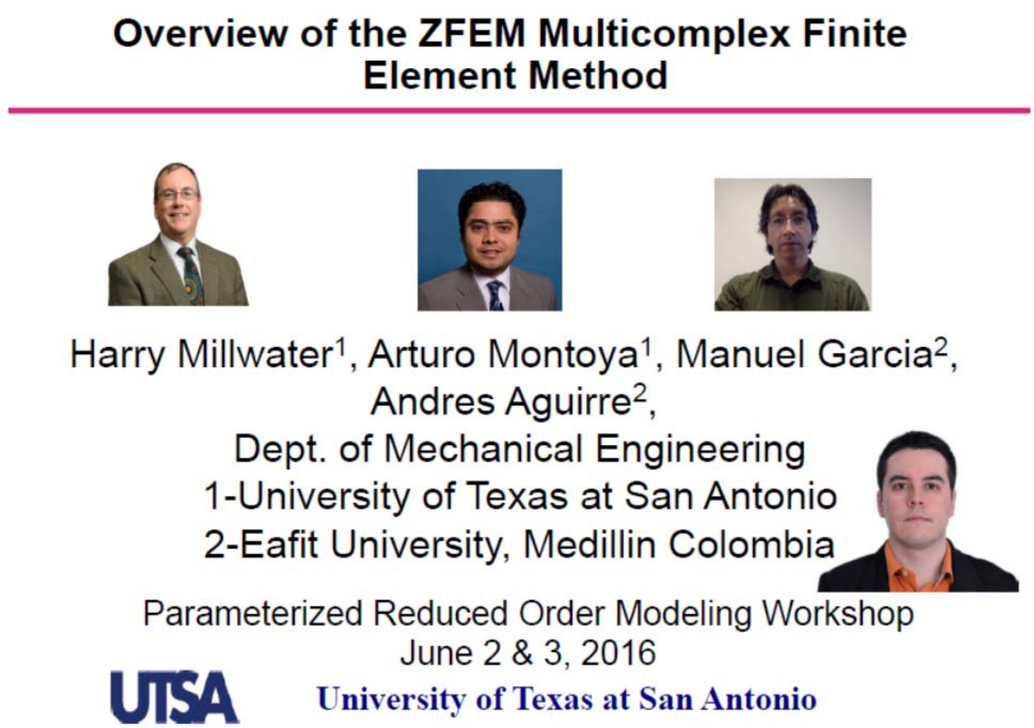
Summary

- Refined **parametric reduced order modeling**
  - Enhanced transformation matrix (significant computational savings)
  - New static constraint modes developed and implemented
  - Novel interpolation to capture element-level nonlinearity

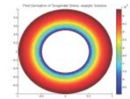


3.1.2 Multicomplex FEA, Harry Millwater and Manuel Garcia

The multicomplex method, based on higher order complex numbers in which multiple imaginary number systems are defined, is developed by Harry Millwater et al. at the University of Texas at San Antonio. The advantage of using these multicomplex numbers is that they allow for either higher order derivatives to be calculated (including cross derivatives) or for perturbations in multiple dimensions to be considered simultaneously. To date, this method has focused on modeling crack propagation [5, 6]. The advantage of this approach is two-fold: one, the multicomplex numbers allow for very accurate calculations of local derivatives, and two, the implementation in commercial FEA code is non-intrusive. Two presentations were given on this method, the first by Harry Millwater, and the second by Manuel Garcia.



Personnel



- Harry Millwater, Professor, ME, UTSA
- Arturo Montoya, Assistant Professor, CE, UTSA
- Manuel Garcia, Professor, ME, Eafit Univ., Medillin, Colombia
- Andres Aguirre, PhD student, ME, Eafit Univ., Medillin, Colombia
- David Wagner, PhD student, ME, UTSA
- Daniel Ramirez, PhD student, ME, UTSA
- Wes Fielder, MS students, UTSA
- Jose Garza, PhD ME UTSA, Dec 14
- Andrew Baines, MS ME UTSA, Dec 14

Daniel Ramirez

Jose Garza

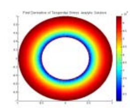
Wes Fielder

Andrew Baines

David Wagner

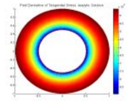
UTSA University of Texas at San Antonio

ZFEM Development Timeline



- 08 Millwater working on probabilistic sensitivities hears about CTSE method applied in aerodynamics
- 08 CTSE applied to fatigue code to compute lifing sensitivities wrt initial crack size, etc. (Voorhees, MSME UTSA)
  - A. Voorhees*, H.R. Millwater, R. Bagley, P. Golden, "Fatigue Sensitivity Analysis Using Complex Variable Methods," Int J Fatigue 40 (2012) 61-73, doi:10.1016/ j.ijfatigue.2012.01.016
- 09-10 2D complex FE code written in Matlab. Concept of imaginary nodal coordinates introduced. (Voorhees, MSME UTSA)
  - A. Voorhees*, H.R. Millwater, R.L. Bagley, "Complex Variable Methods for Shape Sensitivity of Finite Element Models," Finite Elem. Anal. Des., 47 (2011) 1146–1156, doi:10.1016/j.finel.2011.05.003
- 10-11 2D weight functions computed using CTSE with Matlab complex FE code (Wagner, MSME UTSA)
  - D. Wagner*, and H.R. Millwater, "2D Weight Function Development using a Complex Taylor Series Expansion Method," Engng Fract Mech 86 (2012), 23-37, 210- doi:10.1016/j.engfracmech.2012.02.006
- 11-12 Implementation into Abaqus using UEL. New method called ZFEM. (Wagner, MSME UTSA)
  - H.R. Millwater, D. Wagner*, A. Baines*, K. Lovelady*, "Improved WCTSE Method for the Generation of 2D Weight Functions through Implementation into a Commercial Finite Element Code," Engng Fract Mech, 109 (2013) 302-309, http://dx.doi.org/10.1016/j.engfracmech.2013.07.012
- 12 Extension of CTSE to multicomplex mathematics discovered by UTSA (Lantoine). Implemented into Abaqus. (Wagner, MSME UTSA)
- 13-16 Extension to 2D progressive fracture (Wagner, Garcia)
  - H.R. Millwater, D. Wagner, "A New Progressive Curvilinear Strain Energy-based Crack Growth Modeling Algorithm using Multicomplex Variable Finite Elements," Advanced Materials Research Vols. 891-892 (2014) pp 1015-1020 Online available at www.scientific.net, doi: 10.4028 / www.scientific.net / AMR.891-892.1015
- 13-14 ZFEM extended to plasticity in Abaqus (Montoya, Gomez-Farias, Fielder)
  - A. Montoya, R. Fielder*, A. Gomez-Farias*, H. Millwater, "Finite Element Sensitivity for Plasticity using Complex Variable Methods," J. Eng. Mech. 141 2 2015, DOI:10.1061/(ASCE)EM.1943-7889.0000837, 04014118.

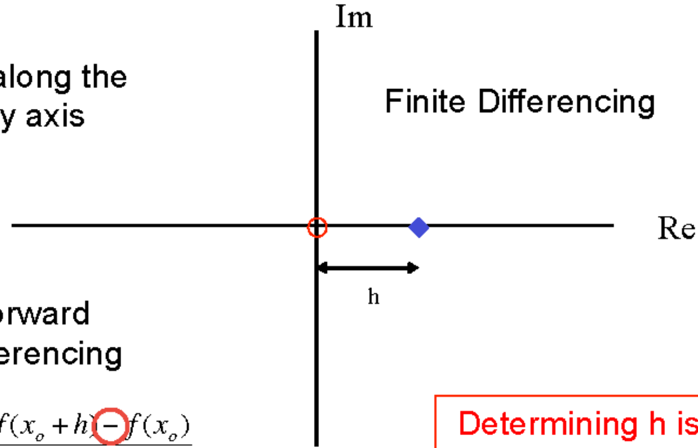
ZFEM Development Timeline



- 13-14 Application to Newmark-beta structural dynamics (Garza, PhD ME UTSA)
  - J. Garza* and H. Millwater, "Multicomplex Newmark-Beta Time Integration Method for Sensitivity Analysis in Structural Dynamics", AIAA Journal, Vol. 53, No. 5 (2015), pp. 1188- 1198, doi: http://arc.aiaa.org/doi/abs/10.2514/1.J053282l
- 14-15 High order probabilistic sensitivities demonstrated. Needed functions of matrices developed. (Garza, PhD ME UTSA)
  - J. Garza* and H.R. Millwater, "Higher-Order Probabilistic Sensitivity Calculations Using the Multicomplex Score Function Method," Probabilistic Engineering Mechanics, 45 (2016) 1-12, http://dx.doi.org/10.1016/j.probengmech.2015.12.001
- 14 ZFEM extended to creep in Abaqus (Gomez-Farias BSCE UTSA, Montoya)
  - A. Gomez-Farias*, A. Montoya, H.R. Millwater, "Complex Finite Element Sensitivity Method for Creep Analysis," International Journal of Pressure Vessels and Piping (2015), V 132-133, 27- 42, http://dx.doi.org/10.1016/j.ijpvp.2015.05.006
- 14 Application to 3D fracture demonstrated (Baines, MSME UTSA)
  - H.R. Millwater, D. Wagner*, A. Baines*, and A. Montoya, "A Virtual Crack Extension Method to Compute Energy Release Rates using a Complex-valued Finite Element Method," Engineering Fracture Mechanics 162 (2016) 95–111, http://dx.doi.org/10.1016/j.engfracmech.2016.04.002
- 15-16 Bioheat transfer (Garcia)
  - Sensitivity analysis in thermal modeling of radiofrequency ablation using the complex finite element method" by Monsalvo, J.; Garcia, M.; Millwater, H.; Feng, Y., Phys. Med. Biol.: PMB-103963 (Under review)
- 15-16 Thermoelastic analysis (Montoya)
  - Sensitivity Analysis in Thermoelastic Problems using the Complex Finite Element Method, Journal of Thermal Stresses (Under review)
- 15-16 Residual stresses (Fielder, MSME UTSA)
  - Residual Stress Sensitivitiy Analysis using a Complex Variable Finite Element Method (in progress)
- 15-16 Elasto-plastic fracture application demonstrated (Montoya)
  - In progress
- 16 Multicomplex Python and Fortran libraries (Garcia, Aguirre, PhD ME Eafit)

Finite Difference Method

Perturb along the imaginary axis



Finite Differencing

Forward Differencing

$$\frac{df(x_o)}{dx} \approx \frac{f(x_o+h) - f(x_o)}{h}$$

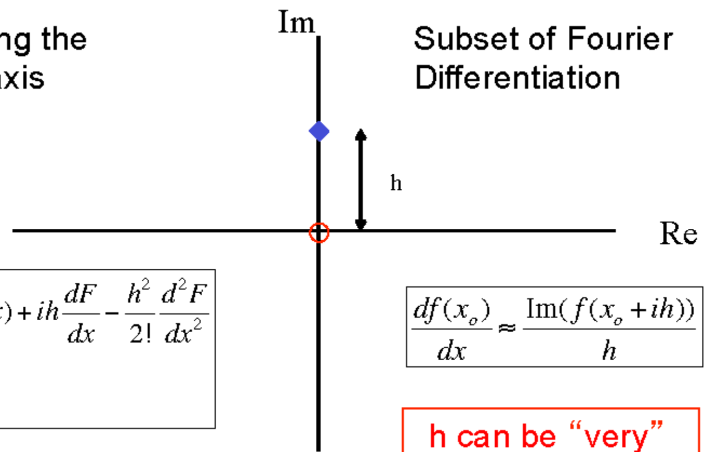
Determining h is problematic

UTSA University of Texas at San Antonio

Complex Taylor Series Expansion

Perturb along the imaginary axis

Subset of Fourier Differentiation



$$F(x+ih) = F(x) + ih\frac{dF}{dx} - \frac{h^2}{2!}\frac{d^2F}{dx^2} - \frac{ih^3}{3!}\frac{d^3F}{dx^3} + \cdots$$

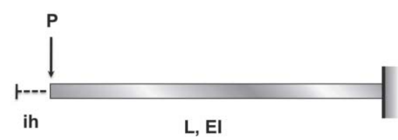
$$\frac{df(x_o)}{dx} \approx \frac{\mathrm{Im}(f(x_o+ih))}{h}$$

h can be "very" small ~ 10^{-30}

UTSA University of Texas at San Antonio

Finite Element Implementation

$$\begin{Bmatrix} P \\ 0 \end{Bmatrix} = \frac{EI}{L^3}\begin{bmatrix} 12 & 6L \\ 6L & 4L^2 \end{bmatrix}\begin{Bmatrix} \delta \\ \phi \end{Bmatrix}$$



$$\begin{Bmatrix} P \\ 0 \end{Bmatrix} = \frac{EI}{(L+ih)^3}\begin{bmatrix} 12 & 6(L+ih) \\ 6(L+ih) & 4(L+ih)^2 \end{bmatrix}\begin{Bmatrix} \delta \\ \phi \end{Bmatrix}$$

$$\delta = \left\{\frac{PL^3}{3EI} - \frac{PLh^2}{EI}\right\} + i\left\{\frac{PL^2h}{EI} - \frac{Ph^3}{3EI}\right\}$$

$$\frac{\partial\delta}{\partial L} = \mathrm{Im}[\delta]/h$$

UTSA University of Texas at San Antonio

Finite Element Implementation

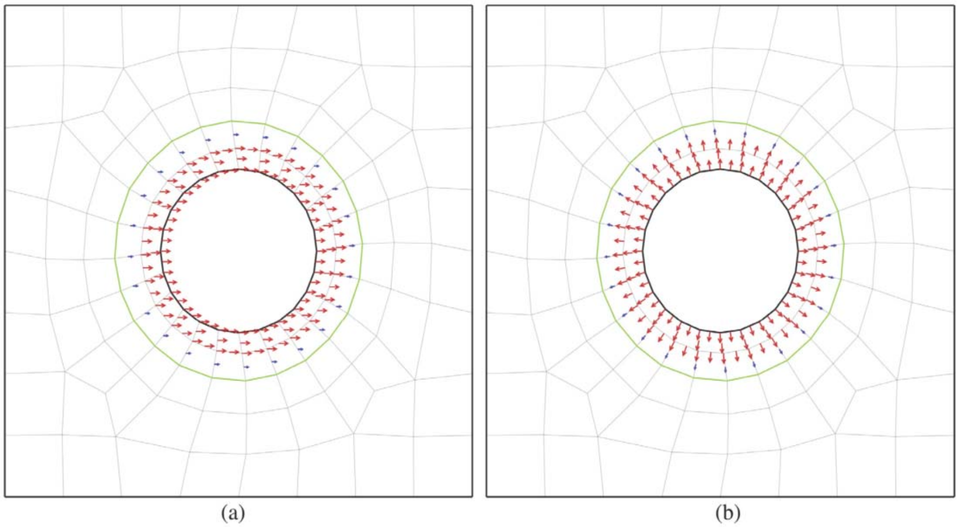
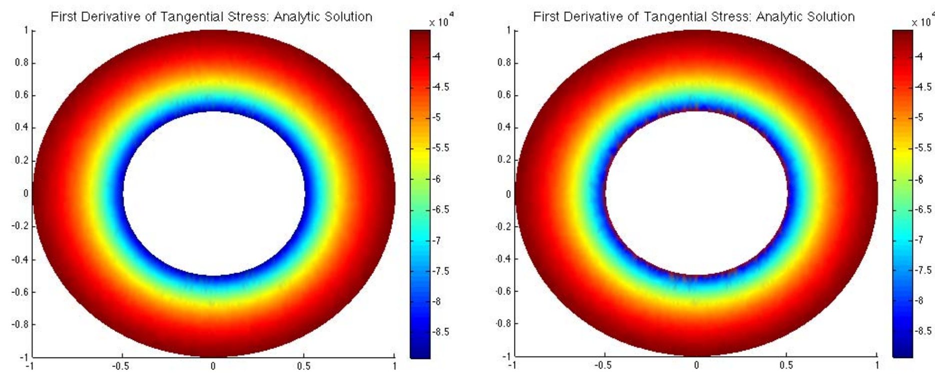


Fig. 1. (a) Horizontal perturbation of the hole location and (b) radial perturbation of the hole.

Example: 1^{st} Der. of σ_θ w.r.t. R_i



Analytic Solution

CTSE

UTSA Matlab code

A. Voorhees, H.R. Millwater, R.L. Bagley, "Complex Variable Methods for Shape Sensitivity of Finite Element Models," Finite Elem. Anal. Des., 47 (2011) 1146–1156, doi:10.1016/j.finel.2011.05.003

MCX Finite Element Implementation

$$\begin{Bmatrix} P \\ 0 \end{Bmatrix} = \frac{EI}{(L+i_1h+i_2h)^3}\begin{bmatrix} 12 & 6(L+i_1h+i_2h) \\ 6(L+i_1h+i_2h) & 4(L+i_1h+i_2h)^2 \end{bmatrix}\begin{Bmatrix} \delta \\ \varphi \end{Bmatrix}$$

2nd Order Example

$$\delta = \frac{P}{EI}\left[\frac{L^3}{3} - 2h^2L + i_1h\left(L^2 - \frac{4}{3}h^2\right) + i_2h\left(L^2 - \frac{4}{3}h^2\right) + 2i_{12}h^2L\right]$$

$$\mathrm{Re}[\delta] = \frac{PL^3}{3EI} + O(h^2)$$

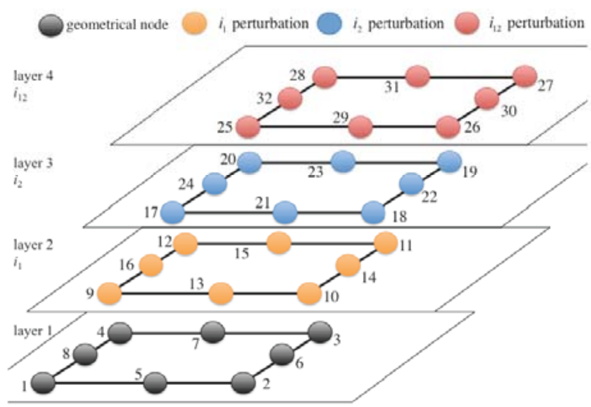
$$\frac{\partial\delta}{\partial L} = \frac{1}{h}\mathrm{Im}_1[\delta] = \frac{PL^2}{EI} + O(h^2)$$

1st Order Derivative

$$\frac{\partial^2\delta}{\partial L^2} \approx \frac{1}{h^2}\mathrm{Im}_{12}[\delta] = \frac{2PL}{EI} + O\left(h^2\right)$$

2nd Order Derivative

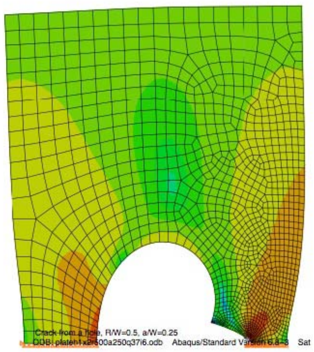
Imaginary Nodes (DOF)



University of Texas at San Antonio

Implementation into Abaqus

- Abaqus user element implementation (uel)
- 6 dof/node (3 real, 3 imag)
- Abaqus cannot solve complex stiffness matrix, represent as real



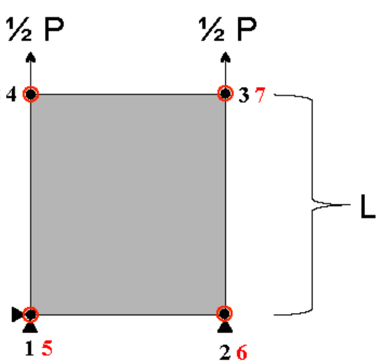
$$\begin{bmatrix} \mathrm{Re}\{\mathbf{P}\} \\ \mathrm{Im}\{\mathbf{P}\} \end{bmatrix} = \begin{bmatrix} \mathrm{Re}[\mathbf{K}] & -\mathrm{Im}[\mathbf{K}] \\ \mathrm{Im}[\mathbf{K}] & \mathrm{Re}[\mathbf{K}] \end{bmatrix} \begin{bmatrix} \mathrm{Re}\{\mathbf{U}\} \\ \mathrm{Im}\{\mathbf{U}\} \end{bmatrix}$$

n x n complex matrix solved as 2n x 2n real matrix

UTSA University of Texas at San Antonio

Obtaining Derivatives

- Shape sensitivity – input imaginary coordinates to represent shape change. (All Imag nodes not perturbed have a coordinate of zero)



<u>Nodal Coordinates</u>

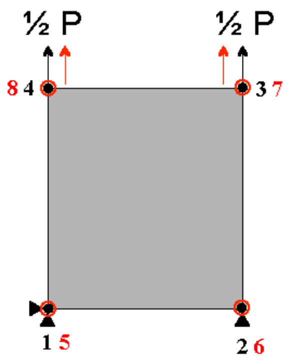
1 (0,0)
2 (1,0)
3 (0,1)
4 (1,1)
5 (0,0)
6 (0,0)
7 (0,1)h
8(0,1)h

$h = 10^{-10}$

UTSA University of Texas at San Antonio

Obtaining Derivatives

- Load sensitivity – Apply perturbation in loading to Imag nodes.



<u>Applied Loads</u>

3 (0,1/2 P)
4 (0,1/2 P)
7 (0,1/2)h
8(0,1/2)h

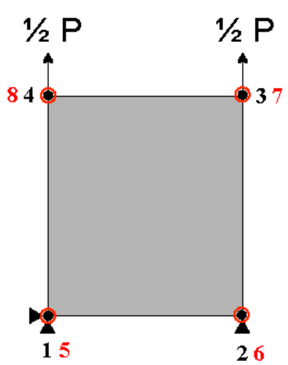
Imaginary coordinates all zero

$h = 10^{-10}$

University of Texas at San Antonio

Obtaining Derivatives

- Material sensitivity – Apply complex perturbation to constitutive matrix.



$$\frac{E}{(1+\nu)(1-2\nu)}\begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & 1/2(1-2\nu) \end{bmatrix}$$

$$E = E + ih$$
$$\nu = \nu + ih$$

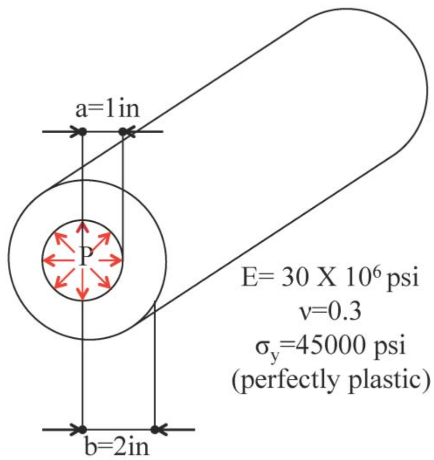
$h = 10^{-10}$

Imaginary coordinates all zero

University of Texas at San Antonio

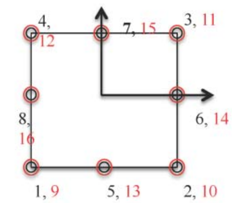
Plasticity & Creep

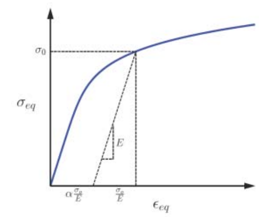
Arturo Montoya, CE, UTSA



Thick wall cylinder

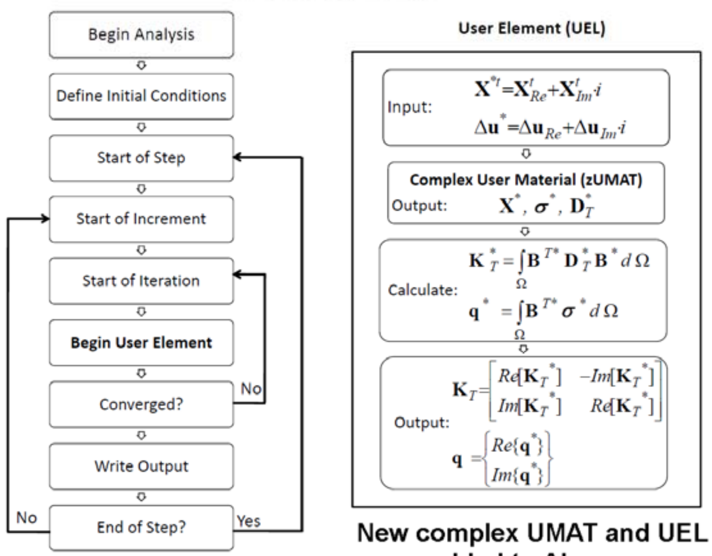
- ZFEM extended to nonlinear materials: plasticity and creep





ABAQUS Plasticity Implementation

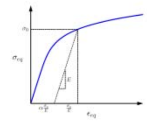
Nonlinear zFEM



17

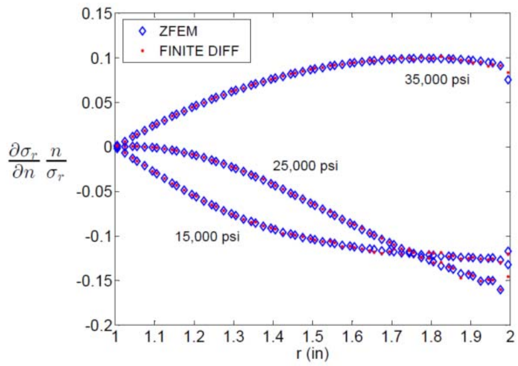
New complex UMAT and UEL added to Abaqus

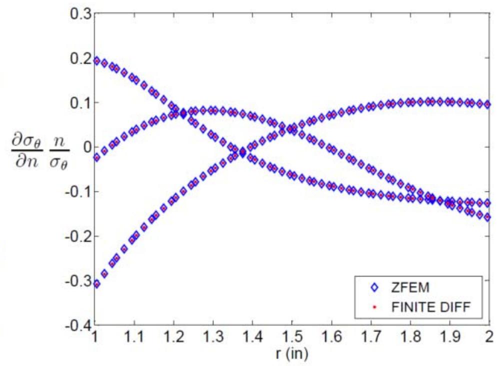
Stress Sensitivity Results



$$\varepsilon = \frac{\sigma}{E} + \alpha\left(\frac{\sigma}{\sigma_0}\right)^n \sigma$$

$$\frac{\partial \sigma_r}{\partial n}\frac{n}{\sigma_r} \qquad\qquad \frac{\partial \sigma_\theta}{\partial n}\frac{n}{\sigma_\theta}$$





UTSA **University of Texas at San Antonio**

Abaqus Creep Implementation

Arturo Montoya, CE, UTSA

```
c     Alpha Method Solution
         zstressprev = zstresst
       do j = 1,30
c     Stress at t + alpha*dtime
         if(j.eq.1) then
            zstress = zstresst
         else
            zstress = (1-alpha)*zstresst + alpha*zstress
         end if
c     Calculate deviatoric trial stress
         zmean = (one/three)*(zstress(1)+zstress(2)+zstress(3))
         do k = 1,3
            zdstr(k) = zstress(k) - zmean
         end do
         zdstr(4) = zstress(4)
c     Calculate effective trial stress
         zpj11 = zdstr(1)*zdstr(1)
         zpj22 = zdstr(2)*zdstr(2)
         zpj33 = zdstr(3)*zdstr(3)
         zpj12 = two*zdstr(4)*zdstr(4)
         zpj   = sqrt((three/two)*(zpj11+zpj22+zpj33+zpj12))
c     Creep strain
         dtime = passdtime
         do k= 1,4
            zdcrstran(k)=(three/two)*za*(zpj**(zxn-one))*zdstr(k)*dtime
         end do
         zdcrstran(4) = two*zdcrstran(4)
c     Elastic strain
         zdestran = zdstran - zdcrstran
c     New stress increment
         zdstress = matmul(zdds,zdestran)
         zstress = zstresst + zdstress
c     Convergence Check
         ztopstress = zstress - zstressprev
         zbotstress = zstress
         zy = zero
         zx = zero
         do k = 1,4
            zx = zx + zbotstress(k)*zbotstress(k)
            zy = zy + ztopstress(k)*ztopstress(k)
         end do
         ztopstressnorm = sqrt(zy)
         zbotstressnorm = sqrt(zx)
         zcriteria = ztopstressnorm/zbotstressnorm
         if(real(zcriteria).le.ctol) goto 10
         zstressprev = zstress
       end do
c
10       continue
```

Alpha Method

$${}^{t+\alpha\Delta t}\sigma = (1-\alpha)\,{}^{t}\sigma + \alpha\,{}^{t+\Delta t}\sigma$$

$$\Delta\varepsilon^c = \frac{3}{2} K \sigma_e^{n-1} \sigma' \Delta t$$

Creep Law

$$\Delta\varepsilon^e = \Delta\varepsilon - \Delta\varepsilon^c$$

$$\Delta\sigma = D\Delta\varepsilon^e$$

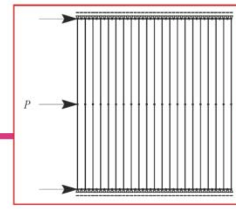
$${}^{t+\Delta t}\sigma = {}^{t}\sigma + \Delta\sigma$$

$$\frac{\left\| {}^{t+\Delta t}\sigma_{(k)} - {}^{t+\Delta t}\sigma_{(k-1)} \right\|_2}{\left\| {}^{t+\Delta t}\sigma_{(k)} \right\|_2} \le ctol$$

LOOP

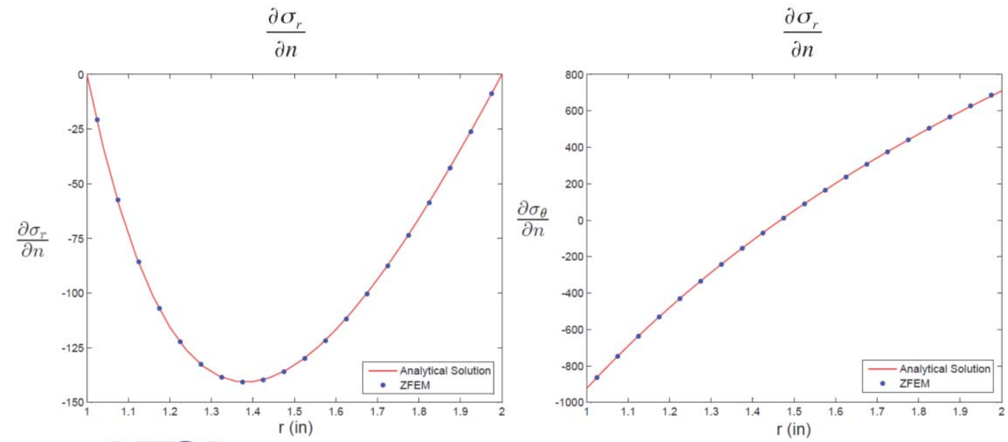
New complex UMAT and UEL added to Abaqus

Creep Model



- Material Sensitivity
  - Steady State

Power Law $\varepsilon = \alpha\sigma^n$	$n=3.5$ $\alpha=4e-24$



University of Texas at San Antonio

The University of Texas at San Antonio

Applications to Fracture Mechanics

University of Texas at San Antonio

Weight Function Development

Calculation of the partial derivative of the crack opening displacement with respect to crack length required

$$m(a,x) = \frac{E'}{2K_A}\frac{\partial u}{\partial a}$$

Standard research approaches:

Assume 3-4 term approximations to du/da or approximate the weight fn. directly.

$$m(x,a) = \frac{2}{\sqrt{2\pi(a-x)}}\left[1 + M_1(1-x/a)^{1/2} + M_2(1-x/a)^1 + \cdots + M_n(1-x/a)^{n/2}\right]$$

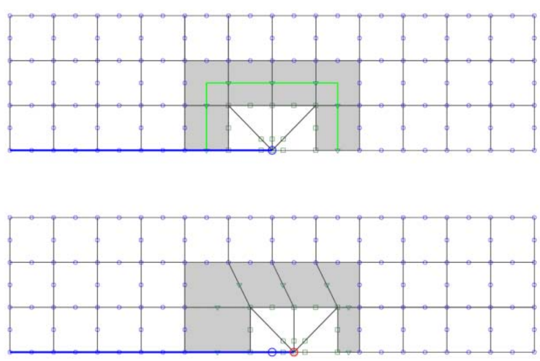
Use multiple reference solutions to solve for Mi

UTSA University of Texas at San Antonio

Perturbation of Crack Length

Crack tip element can be perturbed in the imaginary domain – no perturbation of real mesh

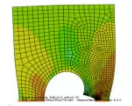
Perturb a no. of elements around the crack tip

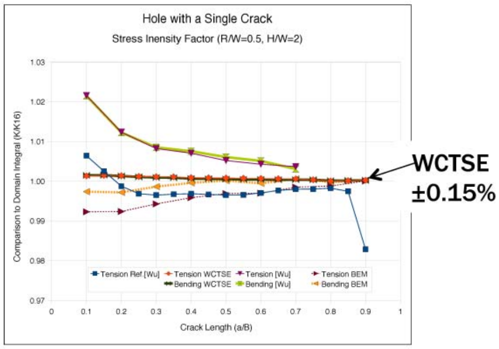


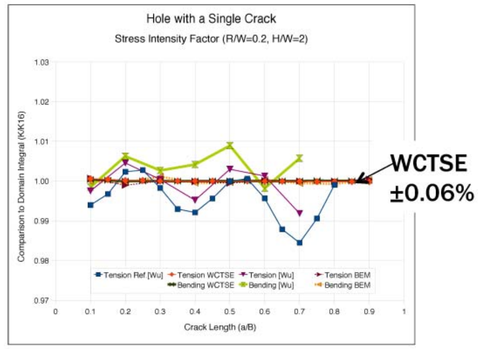
Perturbation of Crack Length in Imaginary domain

UTSA University of Texas at San Antonio

Accuracy: Crack from a Hole







WCTSE weight function consistently better than published weight fns.

UTSA

D. Wagner, and H. Millwater, "2D Weight Function Development using a Complex Taylor Series Expansion Method," Engng Fract Mech 86 (2012), 23-37

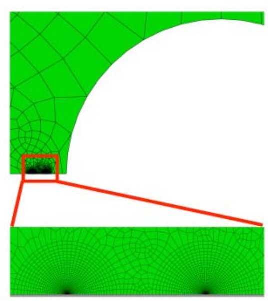
Calculation of Energy Release Rate

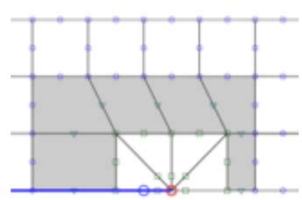
Energy release rate contained in strain energy

$$G = -\frac{\partial U}{\partial A} = -\frac{1}{h}\sum_{el=1}^{n} \mathrm{Im}[U_{el}]$$

Accuracy comparable to J integral

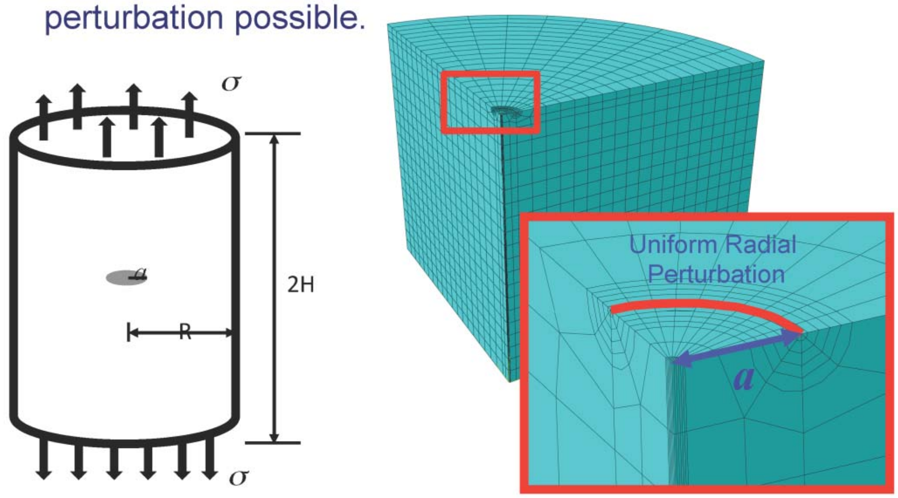
Perturbation Method	G (ZFEM)/J-Integral
Crack tip only	1.0032
Crack tip and Quarter Points of Contour 1	1.0005
4 Inner Contour Rings + Midpoints of Contour 5	1.0000





3D Fracture

- Perturb the crack front along the imaginary axis to determine energy release rate. Arbitrary crack front perturbation possible.



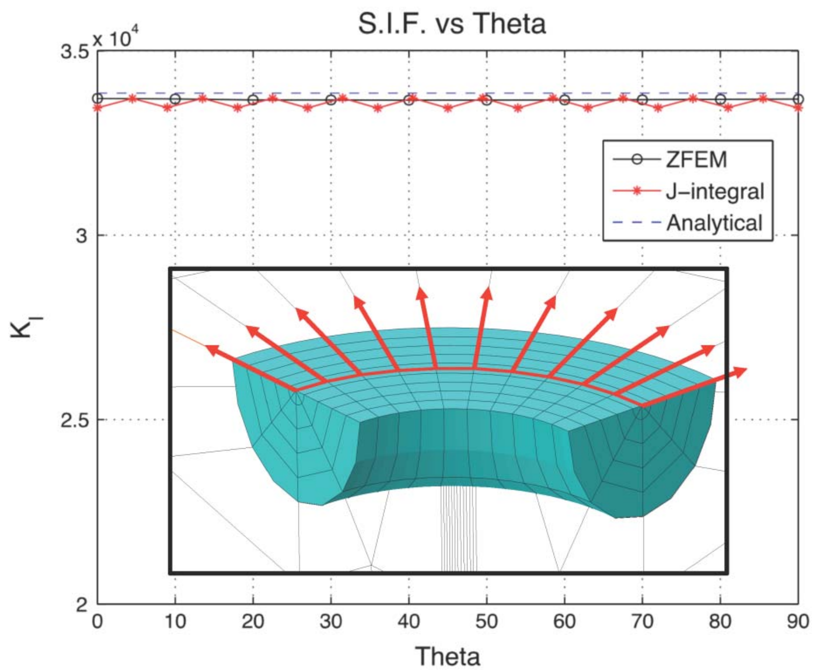
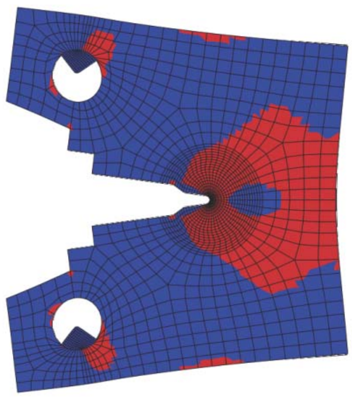
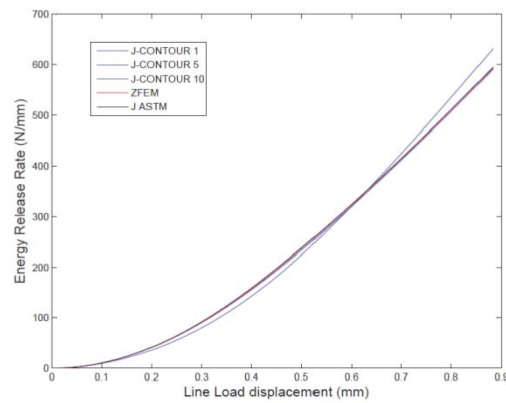


Fig. 19. Stress intensity factor along the crack front for embedded penny shaped crack.

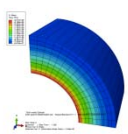
Elasto-Plastic Fracture



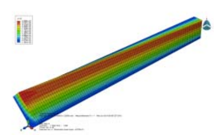


Yield Flag

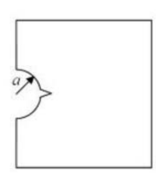
UTSA University of Texas at San Antonio



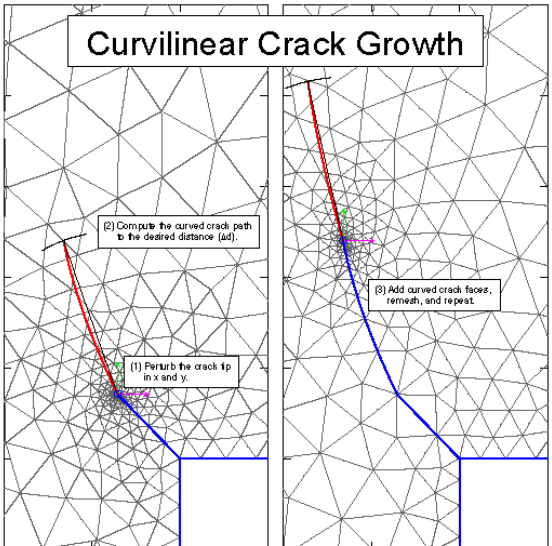
Discussion



- J and G_{ZFEM} accuracies equivalent
- Larger perturbations of crack region provide more accurate results
- ZFEM requires a special user element and longer run times
- ZFEM requires no additional coding to compute G - special case of the more general shape sensitivity capability
- G_{ZFEM} "possibly" more robust wrt mesh quality
- Derivatives of G using bicomplex analysis available, e.g., dG/dradius



Progressive Fracture



- Construct a 3rd order Taylor series of the strain energy using tricomplex elements.
- Predict the crack path along the max energy release
- Progress the crack, remesh, repeat

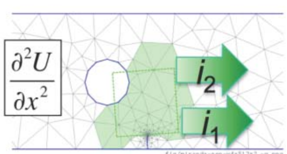
$$U(x_0,y_0) \approx c_{00} + c_{01}(y-y_0) + c_{02}(y-y_0)^2 + c_{03}(y-y_0)^3 + c_{10}(x-x_0) + c_{11}(x-x_0)(y-y_0) + c_{12}(x-x_0)(y-y_0)^2 + c_{20}(x-x_0)^2 + c_{21}(x-x_0)^2(y-y_0) + c_{30}(x-x_0)^3$$

UTSA University of Texas at San Antonio

Example: Quadratic Expansion

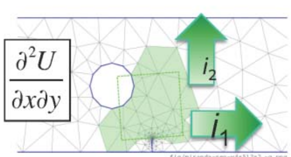
$$U(x_0,y_0) \approx c_{00} + c_{01}(y-y_0) + c_{02}(y-y_0)^2 + c_{10}(x-x_0) + c_{11}(x-x_0)(y-y_0) + c_{20}(x-x_0)^2$$

$\frac{\partial U}{\partial x}$ $\frac{\partial^2 U}{\partial x^2}$

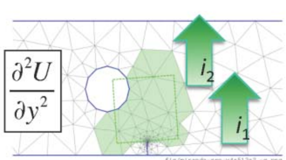


- 6 terms to compute
- 3 bicomplex analyses for single crack tip
  - i1,i2 – x (dUdx, dU²dx²)
  - i1,i2 – y (dUdy, dU²dy²)
  - i1- x , i2 – y (dUdx, dUdy, dUdxdy)

$\frac{\partial^2 U}{\partial x \partial y}$



$\frac{\partial U}{\partial y}$ $\frac{\partial^2 U}{\partial y^2}$



UTSA University of Texas at San Antonio

Validation Problem

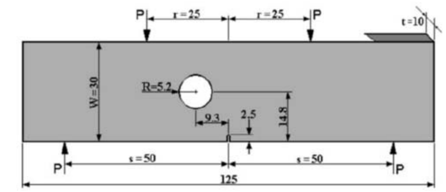
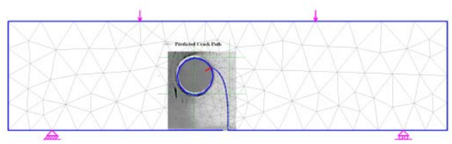
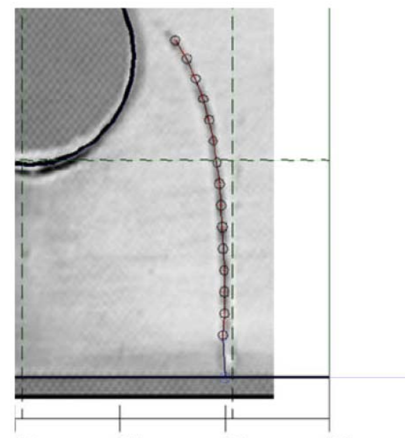


Fig. 7. Geometry of the modified SEN specimen (dimensions in mm).



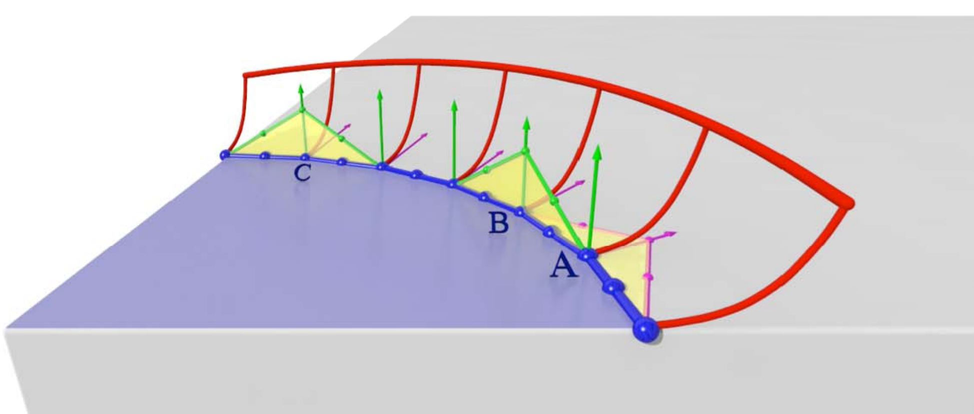


Fatigue life and crack path predictions in generic 2D structural components,
A.C.O. Miranda, M.A. Meggiolaro, J.T.P. Castro, L.F. Martha, T.N. Bittencourt, Engineering Fracture Mechanics 70 (2003) 1259–1279

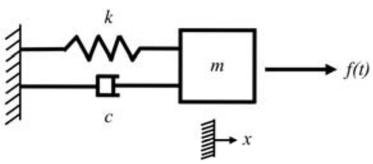
UTSA **University of Texas at San Antonio**

Future Work: 3D Fracture

Perturb the crack front in orthogonal directions
Construct a Taylor series of strain energy
Propagate the crack using gradient descent



Multicomplex Extension of Newmark-beta Algorithm



$$\mathbf{M}\ddot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{C}\dot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{K}\boldsymbol{\eta}^{t+\Delta t} = \mathbf{F}^{t+\Delta t}$$

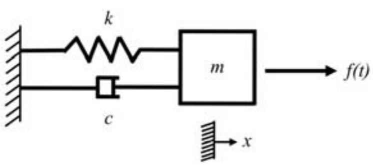
Multicomplex Newmark-beta algorithm

$$4\ddot{x}(t) + 4\dot{x}(t) + 17x(t) = 202\cos(3t), \qquad x(0) = 10, \; \dot{x}(0) = 0$$

$$[m] = \begin{bmatrix} 4 & -10^{-20} & -10^{-20} & 0 \\ 10^{-20} & 4 & 0 & -10^{-20} \\ 10^{-20} & 0 & 4 & -10^{-20} \\ 0 & 10^{-20} & 10^{-20} & 4 \end{bmatrix} \qquad \{x([m],t)\} = \begin{Bmatrix} x(t) \\ h\dfrac{\partial x(t)}{\partial m} \\ h\dfrac{\partial x(t)}{\partial m} \\ h^2\dfrac{\partial^2 x(t)}{\partial m^2} \end{Bmatrix}$$

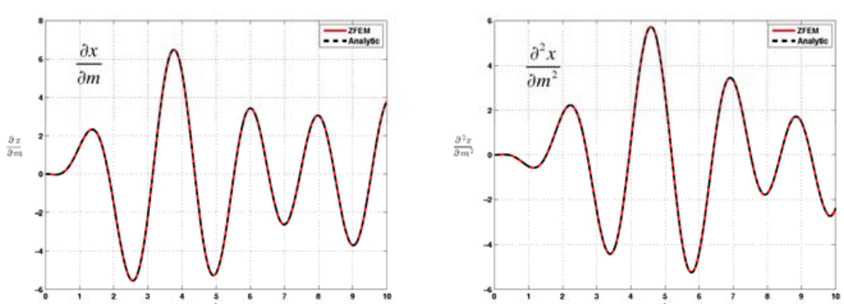
$$\begin{bmatrix} 4 & -10^{-20} & -10^{-20} & 0 \\ 10^{-20} & 4 & 0 & -10^{-20} \\ 10^{-20} & 0 & 4 & -10^{-20} \\ 0 & 10^{-20} & 10^{-20} & 4 \end{bmatrix} \begin{Bmatrix} \ddot{x}(t) \\ h\ddot{x}'(t) \\ h\ddot{x}'(t) \\ h^2\ddot{x}'(t) \end{Bmatrix} + \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} \begin{Bmatrix} \dot{x}(t) \\ h\dot{x}'(t) \\ h\dot{x}'(t) \\ h^2\dot{x}''(t) \end{Bmatrix} + \begin{bmatrix} 17 & 0 & 0 & 0 \\ 0 & 17 & 0 & 0 \\ 0 & 0 & 17 & 0 \\ 0 & 0 & 0 & 17 \end{bmatrix} \begin{Bmatrix} x(t) \\ hx'(t) \\ hx'(t) \\ h^2x''(t) \end{Bmatrix} = \begin{Bmatrix} 202\cos(3t) \\ 0 \\ 0 \\ 0 \end{Bmatrix}$$

Structural Dynamics



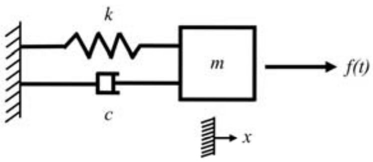
$$\mathbf{M}\ddot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{C}\dot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{K}\boldsymbol{\eta}^{t+\Delta t} = \mathbf{F}^{t+\Delta t}$$

Multicomplex Newmark-beta algorithm



University of Texas at San Antonio

Multicomplex Extension of Newmark-beta Algorithm



$$\mathbf{M}\ddot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{C}\dot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{K}\boldsymbol{\eta}^{t+\Delta t} = \mathbf{F}^{t+\Delta t}$$

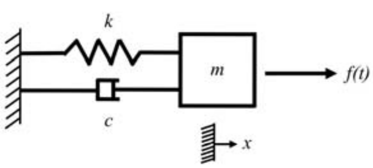
Multicomplex Newmark-beta algorithm

$$4\ddot{x}(t) + 4\dot{x}(t) + 17x(t) = 202\cos(3t), \qquad x(0) = 10,\ \dot{x}(0) = 0$$

$$[m] = \begin{bmatrix} 4 & -10^{-20} & -10^{-20} & 0 \\ 10^{-20} & 4 & 0 & -10^{-20} \\ 10^{-20} & 0 & 4 & -10^{-20} \\ 0 & 10^{-20} & 10^{-20} & 4 \end{bmatrix} \qquad \{x([m],t)\} = \begin{Bmatrix} x(t) \\ h\dfrac{\partial x(t)}{\partial m} \\ h\dfrac{\partial x(t)}{\partial m} \\ h^2\dfrac{\partial^2 x(t)}{\partial m^2} \end{Bmatrix}$$

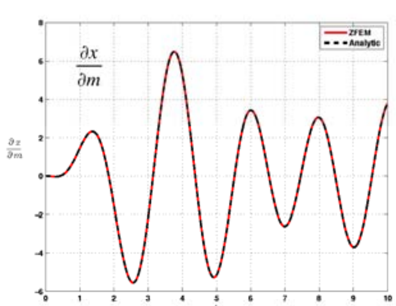
$$\begin{bmatrix} 4 & -10^{-20} & -10^{-20} & 0 \\ 10^{-20} & 4 & 0 & -10^{-20} \\ 10^{-20} & 0 & 4 & -10^{-20} \\ 0 & 10^{-20} & 10^{-20} & 4 \end{bmatrix} \begin{Bmatrix} \ddot{x}(t) \\ h\ddot{x}'(t) \\ h\ddot{x}'(t) \\ h^2\ddot{x}'(t) \end{Bmatrix} + \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} \begin{Bmatrix} \dot{x}(t) \\ h\dot{x}'(t) \\ h\dot{x}'(t) \\ h^2\dot{x}''(t) \end{Bmatrix} + \begin{bmatrix} 17 & 0 & 0 & 0 \\ 0 & 17 & 0 & 0 \\ 0 & 0 & 17 & 0 \\ 0 & 0 & 0 & 17 \end{bmatrix} \begin{Bmatrix} x(t) \\ hx'(t) \\ hx'(t) \\ h^2x''(t) \end{Bmatrix} = \begin{Bmatrix} 202\cos(3t) \\ 0 \\ 0 \\ 0 \end{Bmatrix}$$

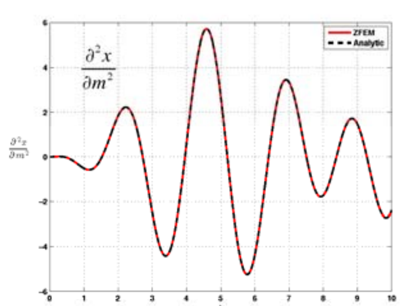
Structural Dynamics



$$\mathbf{M}\ddot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{C}\dot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{K}\boldsymbol{\eta}^{t+\Delta t} = \mathbf{F}^{t+\Delta t}$$

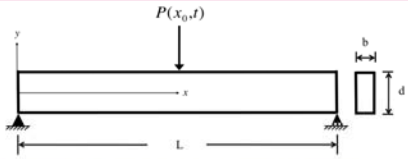
Multicomplex Newmark-beta algorithm





University of Texas at San Antonio

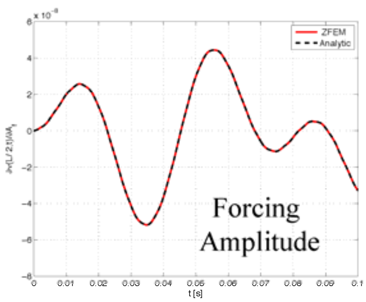
Structural Dynamics

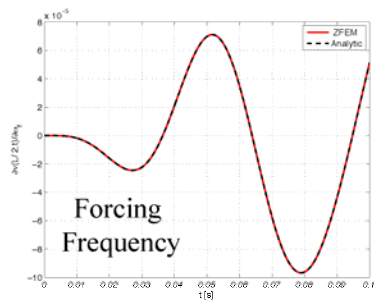


$$\mathbf{M}\ddot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{C}\dot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{K}\boldsymbol{\eta}^{t+\Delta t} = \mathbf{F}^{t+\Delta t}$$

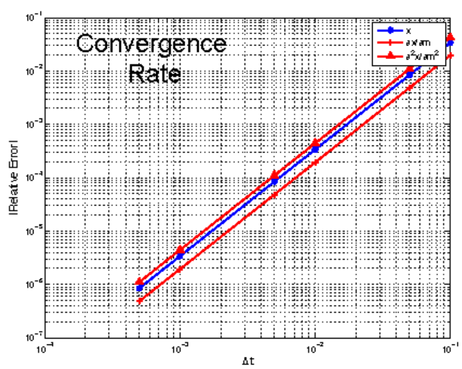
Newmark-beta algorithm complexified

- Derivatives wrt: nat. freq., mode shapes, initial conditions, cross sectional dimension, beam length, forcing amplitude, forcing frequency





Structural Dynamics

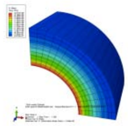


$$\mathbf{M}\ddot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{C}\dot{\boldsymbol{\eta}}^{t+\Delta t} + \mathbf{K}\boldsymbol{\eta}^{t+\Delta t} = \mathbf{F}^{t+\Delta t}$$

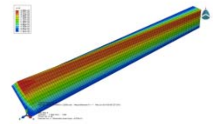
Newmark-beta algorithm complexified

$$\left[\begin{array}{c} \boldsymbol{\eta}^t = \mathrm{Re}(\boldsymbol{\eta}^t)_{102x1} \\ \dfrac{\partial \boldsymbol{\eta}^t}{\partial \theta} = \dfrac{\mathrm{Im}_1(\boldsymbol{\eta}^t)_{102x1}}{h} \\ \dfrac{\partial \boldsymbol{\eta}^t}{\partial \theta} = \dfrac{\mathrm{Im}_2(\boldsymbol{\eta}^t)_{102x1}}{h} \\ \dfrac{\partial^2 \boldsymbol{\eta}^t}{\partial \theta^2} = \dfrac{\mathrm{Im}_{12}(\boldsymbol{\eta}^t)_{102x1}}{h^2} \end{array}\right]_{4nx1}$$

$$K_{bicomplex} = \begin{bmatrix} \mathrm{Re}(K) & -\mathrm{Im}_1(K) & -\mathrm{Im}_2(K) & \mathrm{Im}_{12}(K) \\ \mathrm{Im}_1(K) & \mathrm{Re}(K) & -\mathrm{Im}_{12}(K) & -\mathrm{Im}_2(K) \\ \mathrm{Im}_2(K) & -\mathrm{Im}_{12}(K) & \mathrm{Re}(K) & -\mathrm{Im}_1(K) \\ \mathrm{Im}_{12}(K) & \mathrm{Im}_2(K) & \mathrm{Im}_1(K & \mathrm{Re}(K) \end{bmatrix}_{4nx4n}$$



Future Interests



- Large scale applications
  - 3D fracture
  - Residual stresses
  - Contact
  - Thermal fracture, thermal shock
- Extension to non-linear materials
  - Visco-plasticity
  - Composites, anisotropic materials
- Thermoelastic analysis
- Expanded element library
  - Plates and shells

Acknowledgements

- Efficient Sensitivity Methods for Probabilistic Lifing and Engine Prognostics, Pat Golden, AFRL/RXLMN, Aug. 2007-Sep. 2010
- Efficient Finite Element-based 3D Fracture Mechanics Crack Growth Analysis using Complex Variable Sensitivity Methods, DoD PETTT, Sep. 2010 - Aug. 2011
- Implementation of Complex Variable Finite Element Methods in Abaqus, DOD PETTT, Sep. 2011- Aug. 2012
- Enhanced Fracture Mechanics Crack Growth Analysis using Complex Variable Sensitivity Methods, AFOSR (David Stargel), May 2011-2014
- Probabilistic Residual Stress Modeling, AFRL through Clarkson Aerospace, Sept. 2012 – Nov. 2013
- A New Progressive Curvilinear Strain Energy-based Crack Growth Modeling Algorithm using Multicomplex Variable Finite Elements, ONR, Sept. 2013-Sept. 2016

UTSA University of Texas at San Antonio

Manuel Garcia’s presentation on progressive fracture:

Two Dimensional Curvilinear Progressive Fracture using a Multicomplex Finite Element Method

Manuel Garcia, David Wagner, Harry Millwater

Dept. of Mechanical Engineering,
University of Texas at San Antonio, TX
Depto of Mechanical Engineering,
Universidad EAFIT, Medellín, Colombia

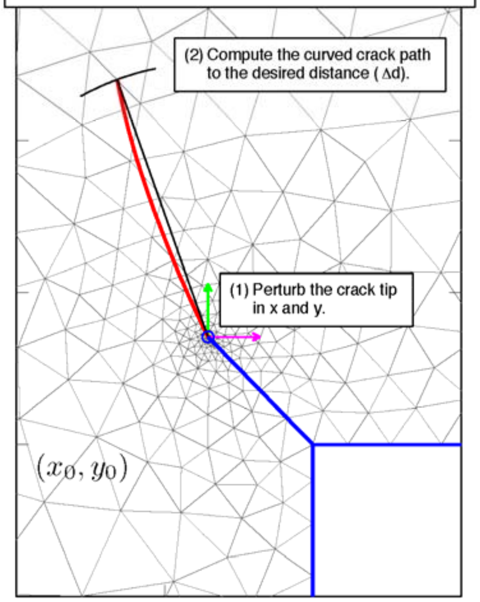
Parameterized Reduced Order Modeling Workshop
June 1-2, Sandia National Laboratories

UTSA

UNIVERSIDAD EAFIT

Progressive Fracture

Curvilinear Crack Growth



- Construct a *n-order* Taylor series approximation of the strain energy using multicomplex elements.

$$R(\boldsymbol{a}) = U(\boldsymbol{x}^0 + \boldsymbol{a}) = \sum_{k=0}^{n} \left\{ \sum_{j=0}^{k} \frac{1}{(k-j)!j!} \frac{\partial^k U(x_0, y_0)}{\partial r^{k+j} \partial s^j} (r)^{k-j} (s)^j \right\}$$

- Using *R(a)*, predict the crack path along the max energy release
- Progress the crack, remesh, repeat

Progressive Fracture

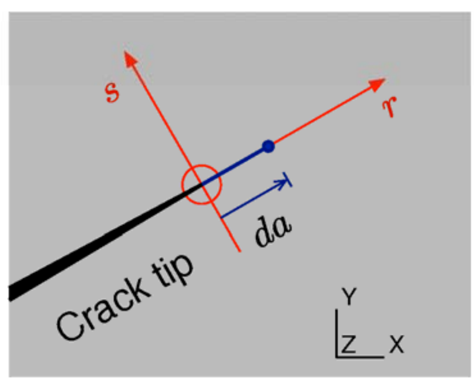
Strain energy function

$$U(\boldsymbol{u}) = \frac{1}{2} \int_{\Omega} \sigma(\boldsymbol{u}) : \varepsilon(\boldsymbol{u}) dV,$$

For a linear elastic body, the strain energy release rate if given by

$$G(a) = \frac{1}{b} \frac{dU}{da}$$

G = G(a) gives the strain energy release rate as the crack propagates in the direction of *a*.

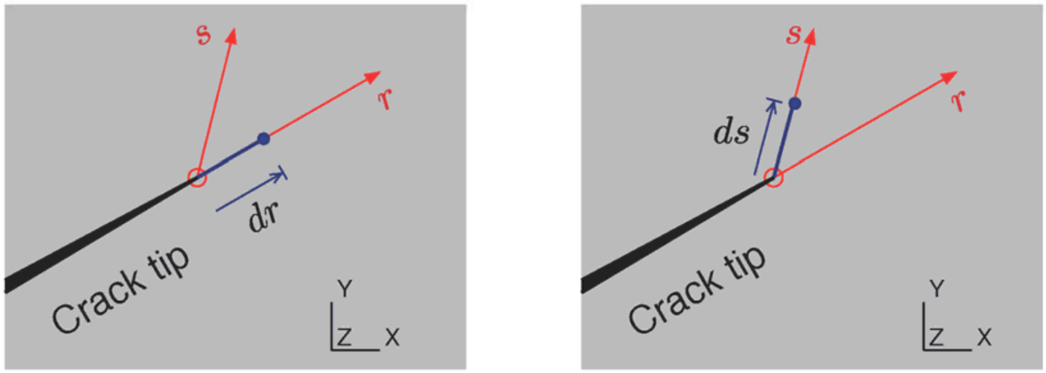


The crack propagates in the direction in which the energy release rate is maximum (maximum energy release rate criterion):

$$\max_a G(a) = \nabla_a U = \begin{pmatrix} \frac{\partial U}{\partial a_1} \\ \frac{\partial U}{\partial a_2} \end{pmatrix}$$

Progressive Fracture

To compute the gradient two perturbations are necessary $\nabla_a U = \begin{pmatrix} \frac{\partial U}{\partial r} \\ \frac{\partial U}{\partial s} \end{pmatrix}$

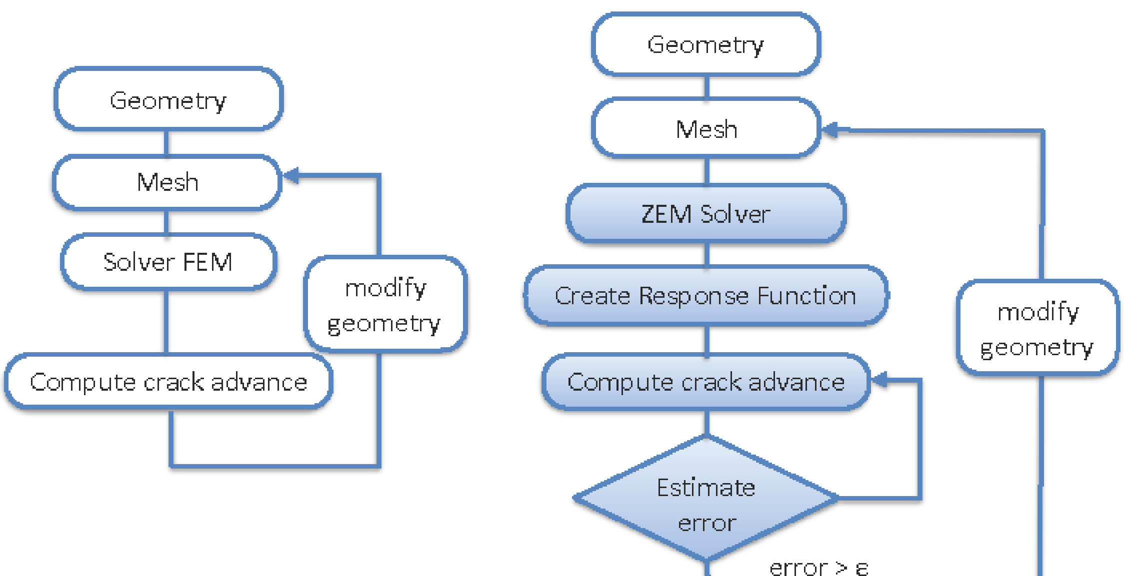


$$\frac{\partial U}{\partial r} \qquad \frac{\partial U}{\partial s}$$

Overview of the method

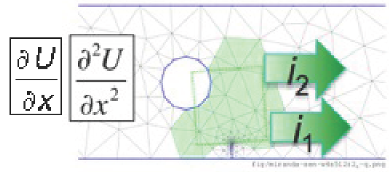
Typical first order method

Proposed high order method



ZFEM Solver

Example: Quadratic Expansion



Perturb in i2 and i2 complex directions

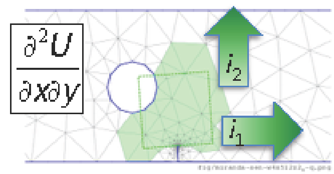
6 terms to compute

3 bicomplex analyses for single crack tip

$$U\left(x+h\left(i_1+i_2\right),\ y\right) \Rightarrow \frac{\partial U(x,y)}{\partial x}, \frac{\partial U^2(x,y)}{\partial x^2}$$

$$U\left(x,\ y+h\left(i_1+i_2\right)\right) \Rightarrow \frac{\partial U(x,y)}{\partial y}, \frac{\partial U^2(x,y)}{\partial y^2}$$

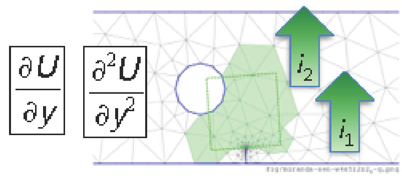
$$U\left(x+hi_1,\ y+hi_2\right) \Rightarrow \frac{\partial U^2(x,y)}{\partial x \partial y}$$



Form the response function

$$R(x,y) = U(x_0,y_0) + \frac{\partial U}{\partial x}x + \frac{\partial U}{\partial y}y + $$

$$+\frac{1}{2}\left(\frac{\partial^2 U}{\partial x^2}x^2 + \frac{\partial^2 U}{\partial y^2}y^2 + 2\frac{\partial^2 U}{\partial x \partial y}x\ y\right)$$



UTSA The University of Texas at San Antonio 6 UNIVERSIDAD EAFIT

Number of ZFEM Solutions

Multicomplex Order (m)	Polynomial Terms (p)	ZFEM Solutions (n_z)
1	3	2
2	6	3
3	10	4
4	15	5

$$p = \binom{2n+m}{2n} = \frac{(2n+m)!}{(2n)!\,m!}$$

$$n_z = \binom{2n+m-1}{m}$$

UTSA The University of Texas at San Antonio 7 UNIVERSIDAD EAFIT

Compute Crack Advance

Steepest descent

$$\partial_{kj}U = \frac{\partial^k U}{\partial {a_1}^{k-j} \partial {a_2}^{j}}$$

function SteepestAscent($\boldsymbol{x}_0$, $\partial_{kj}U$):

// create $R(\boldsymbol{a})$ as aa taylor series expansion

R=CreateTaylorPolynomial($\partial_{kj}U$) ;

$(x, y) = (x_0, y_0)$;

while *true* **do**

$\nabla R = \left(\frac{\partial R}{\partial x}, \frac{\partial R}{\partial y}\right)_{x,y}$ // gradient of R at (x, y)

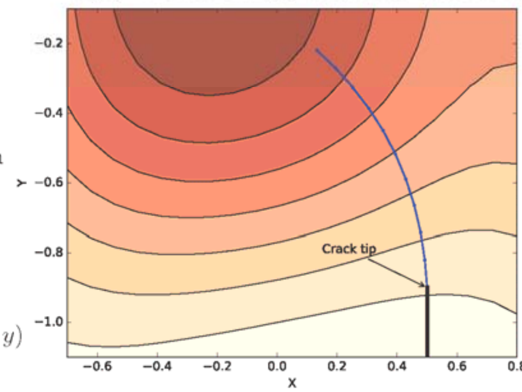
$\delta \boldsymbol{x} = \alpha \frac{\nabla R}{||\nabla R||}$ // move α in the direction of ∇R

$\boldsymbol{x} = \boldsymbol{x} + \delta \boldsymbol{x}$;

AddToPath($\boldsymbol{x}$) // store the point

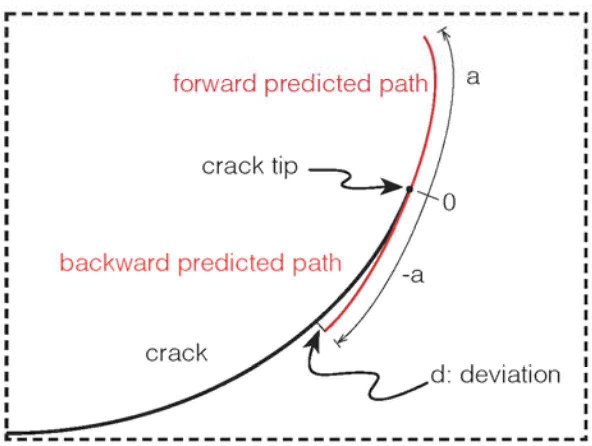
end

R(a)= Strain energy response function



Estimate Error

Adaptive/A Priori Step Size Estimation Based on Backward Deviation



- *a* is a curvilinear distance from the crack tip

- deviation target ε is defined as

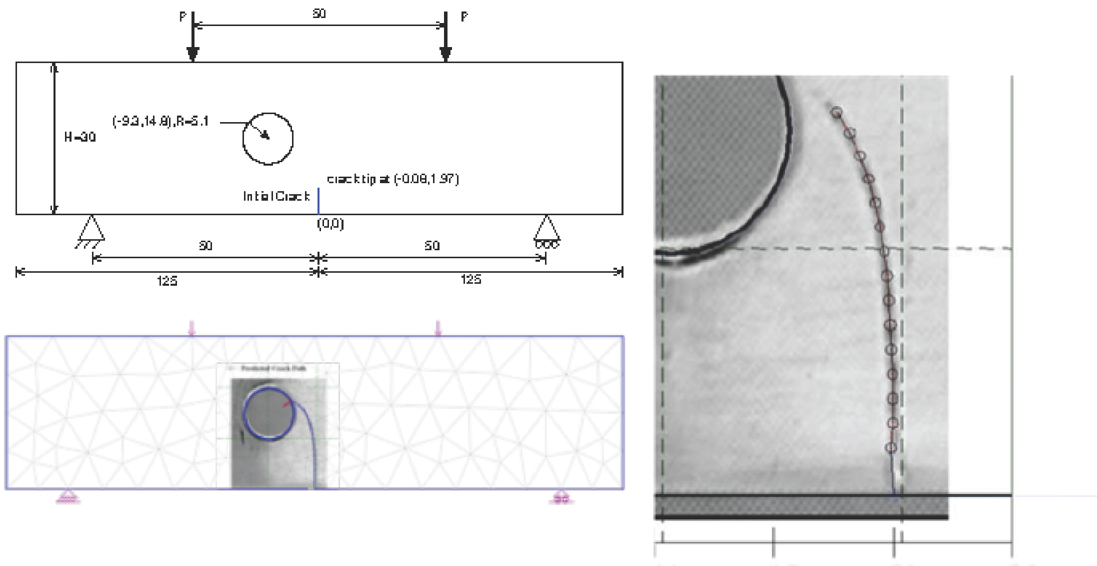
$$\varepsilon = \frac{\text{total deviation distance}}{\text{estimated crack lenght}}$$

- The crack is predicted by the R(a) function up to a distance *a* from the crack tip
- *a* is determine by the deviation target as

$$\varepsilon = \frac{d}{a}$$

- The steepest descent is run until the deviation target is reached

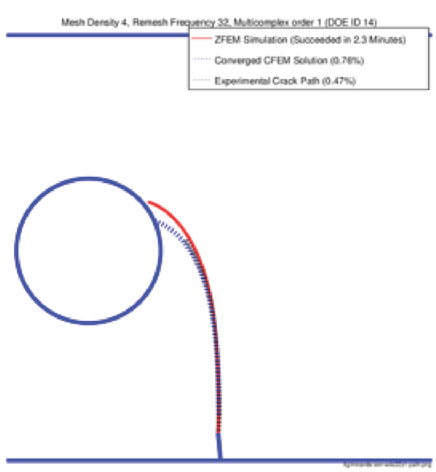
Validation Problem



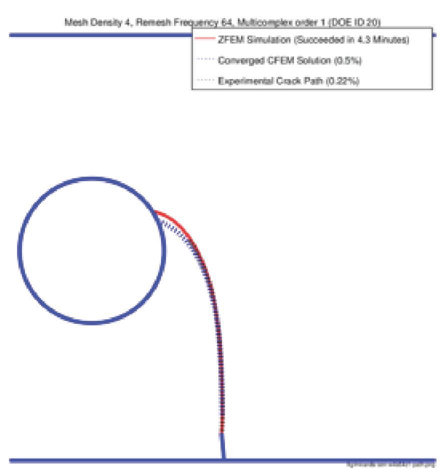
Fatigue life and crack path predictions in generic 2D structural components,
A.C.O. Miranda, M.A. Meggiolaro, J.T.P. Castro, L.F. Martha, T.N. Bittencourt, Engineering Fracture Mechanics 70 (2003) 1259–1279

Compare Results

First order C1 equal size step results



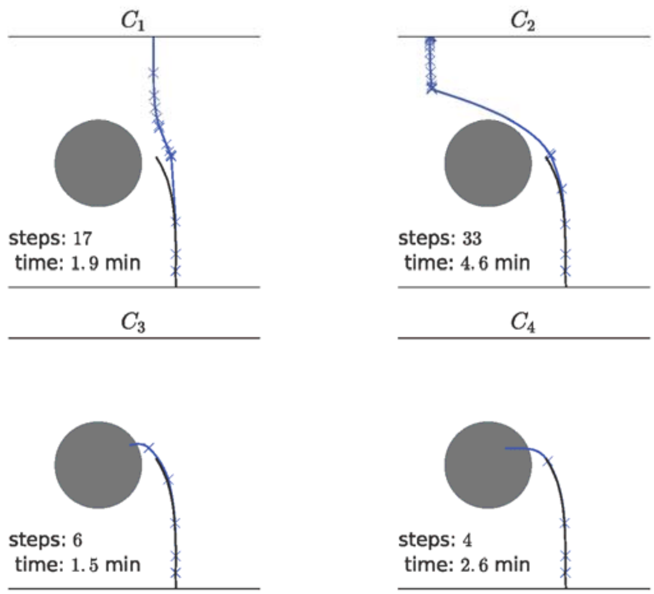
32 steps, 2.3 minutes



64 steps, 4.3 minutes

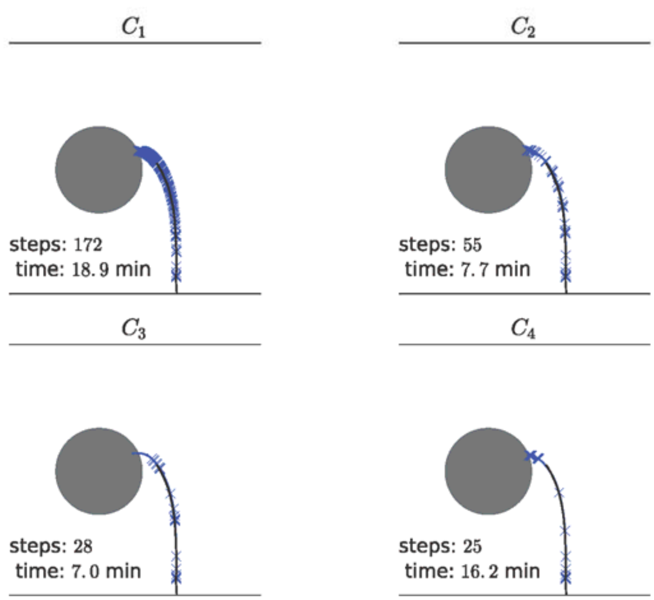
Compare Results

Deviation target ε = 0.1



Compare results

Deviation target c ε = 0.01



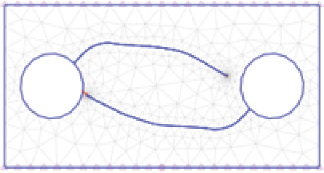
Conclusions

- Multicomplex variable finite element formulation, linear and nonlinear, allows calculation of arbitrary order derivatives.
  - Calculation of the energy release rate are a subset of the shape sensitivity capabilities
- Accuracy comparable to J integral formulations
  - Based on 2D simulations to date
- Progressive fracture algorithm – natural extension of strain energy release rate capabilities
- Adaptive methodology that adjusts the curvilinear step size as needed to ensure an accurate crack propagation path with optimal computational effort

UTSA The University of Texas at San Antonio 14 UNIVERSIDAD EAFIT

Future Work

- Ultimate goal: determine an adaptive robust methodology that adjusts the mesh frequency/density/Taylor series order as needed to ensure an accurate crack propagation path with optimal computational effort



- Extend to multiple crack tips
- 3D Single crack
- Integrate lifing methods with progressive fracture for lifing predictions
- Probabilistic progressive fracture using sampling methods

UTSA The University of Texas at San Antonio 15 UNIVERSIDAD EAFIT

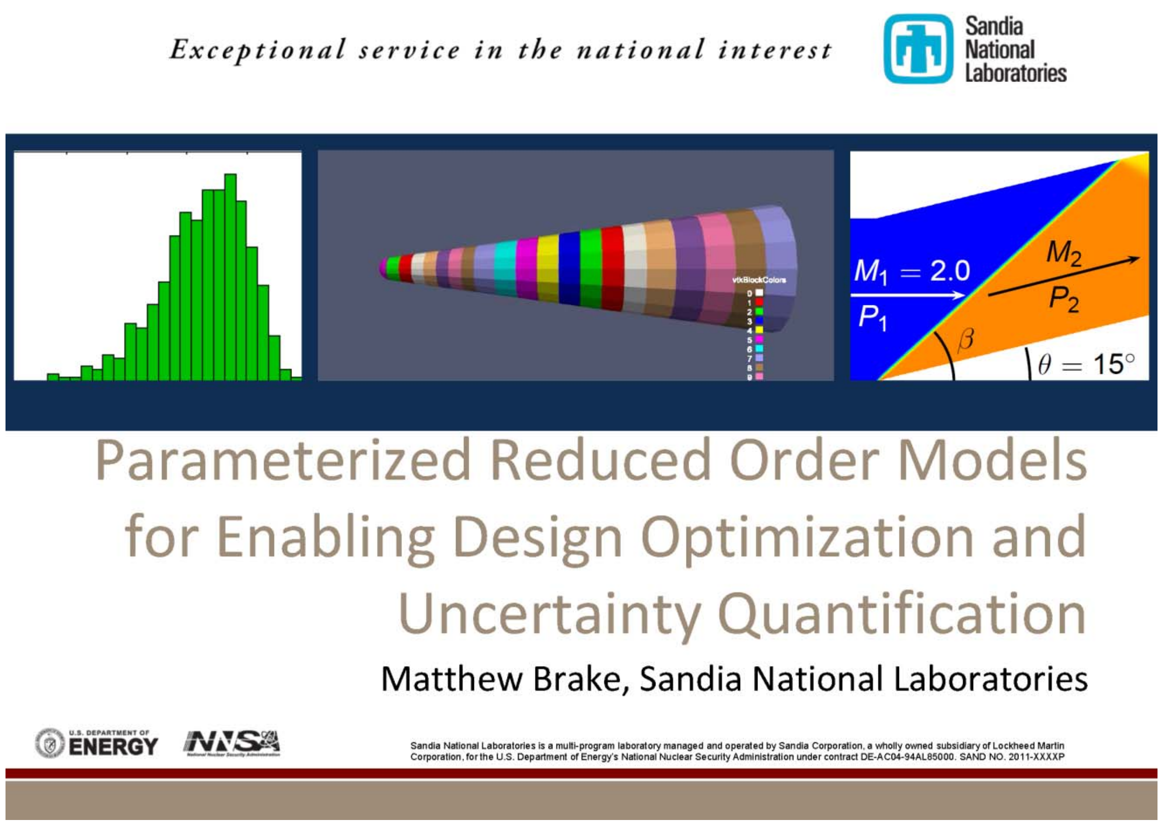
Acknowledgments

- A New Progressive Crack Growth Modeling Algorithm using Complex Variable Finite Elements, ONR contract N000141410004

- Sandia National Laboratories for travel Support

3.1.3 Hyper Dual Numbers, Matthew Brake

The Hyper Dual number approach to PROM development (developed by Matthew Brake and Jeff Fike at Sandia National Laboratories) combines the ideas developed within the NX-PROM research with the usage of higher order, generalized complex numbers (similar to the multicomplex number approach) to calculate derivatives. Dual numbers are defined as the non-zero square root of zero, and are best thought of as an orthogonal number system to the real number system. Because of their well-defined mathematical properties, dual numbers allow for the exact calculation of derivatives of functions. The PROMs based on dual numbers (and hyper-dual numbers for higher order representations – termed HD PROMs) allow for very accurate local perturbations based on a single finite element model [7]; however, the accuracy for large perturbations is not guaranteed as the derivative information is all developed locally.



Acknowledgements

- Jeffrey Fike, postdoc
- Sean Topping, undergraduate intern
- Matt Bonney, graduate intern

Uncertainty in Large Models

- Aleatoric (parametric) uncertainty:
  - Manufacturing tolerances –
    - Geometric variations
    - Material property variations
  - Can result in thousands of design variables
- Models of complex structures often include hundreds of thousands or millions of elements…

3

The Challenge Inherent in Modern Design for High Consequence Applications[1]

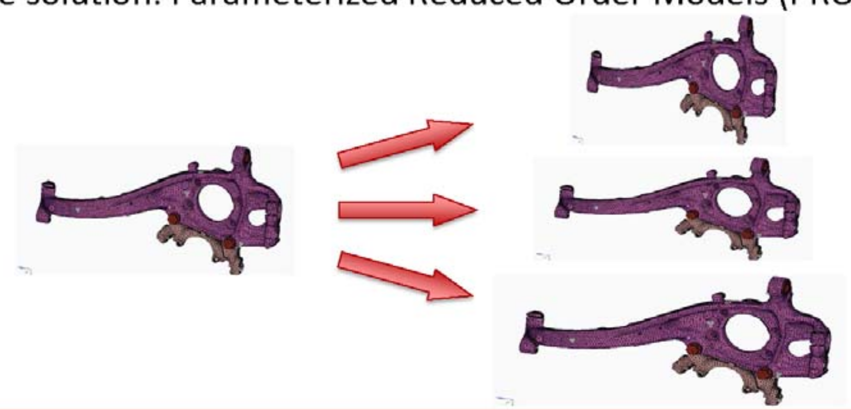
- High fidelity FEA leads to desire for FEA modeling/verification
  - Contrast with approach taken in the 50s…
- Uncertainties omnipresent
  - Environmental specifications, manufacturing tolerances, defects, epistemic sources, etc.
- Result: robust design requirements
  - Can require thousands of perturbed models
- Rough estimate of time to robustly design a single component at SNL:
  - 10 years of human effort, plus 3 years of a dedicated super computer using high fidelity FEA…
- Need for an efficient, automated process…

4

[1]: High consequence – systems where consequences of failure are non-trivial, such as airplanes or automobiles

Enabling Technologies for UQ

- Given needs for UQ (and optimization), what theoretical basis will enable it?
  - Fast simulations
  - Ability to incorporate variations without remeshing
  - Confidence in accuracy
- One solution: Parameterized Reduced Order Models (PROMs)



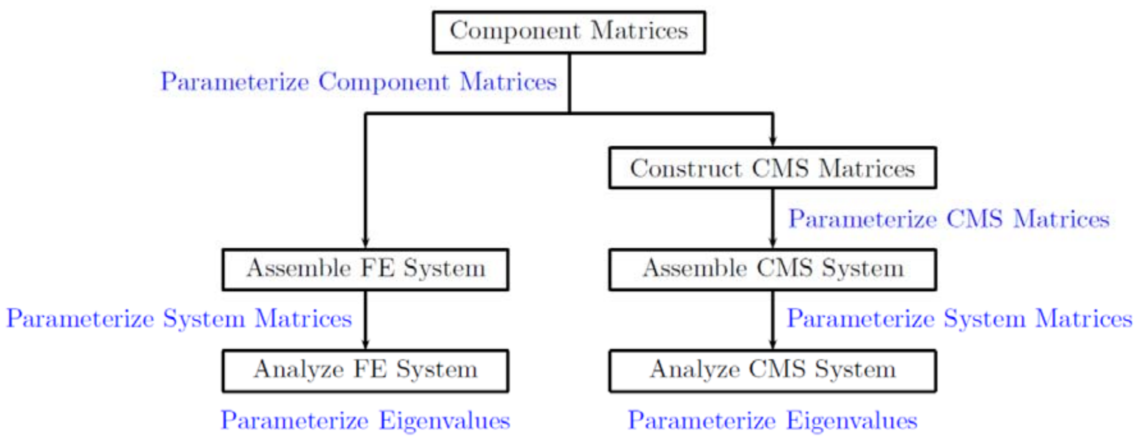
5

Outline

- Context
- Finite difference implementation of methodology
- Hyper dual number basis
- Extension to large, FE systems

6

Possible Parameterizations

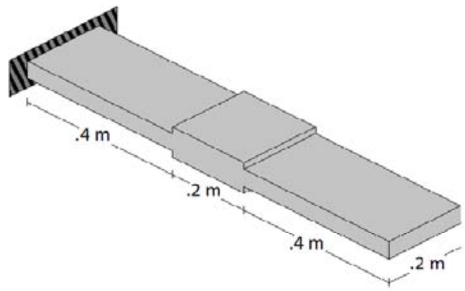


- What is the optimal path for parameterization?
- How many terms should be taken in the expansions?
- Existing research focused on parameterizing the CMS matrices

7

Candidate System

- Simple system considered since there is an analytical answer



  - Length: 1 meter
  - Thickness: 50 millimeters
  - Width: 200 millimeters
  - Material: 6061 Aluminum (E = 68.9 GPa, rho = 2700 kg/m^3)
  - Feature: Middle of beam, 0.2 meters long
  - Boundary conditions: Clamped-Free and Pinned-Pinned

8

Overview of Method (Details to follow…)

- Given a linear subsystem expressed as

$$[M]\{\ddot{u}\} + [K]\{u\} = \{0\}$$

- The CB model and eigenvalues of the nominal system are readily available
- These quantities are then parameterized in terms of the variables of interest using

$$f(a+x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \ldots\frac{f^{(n)}(a)}{n!}(x-a)^n$$

- Challenge in specifying derivatives

9

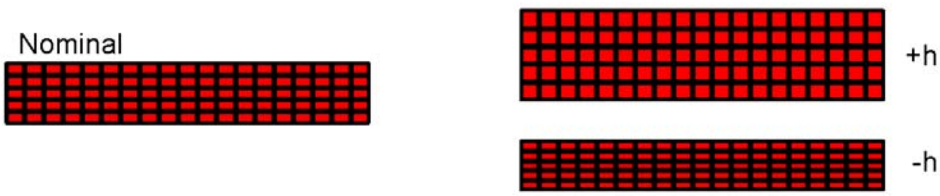
Calculation of Derivatives

- First approach: finite difference approximations.

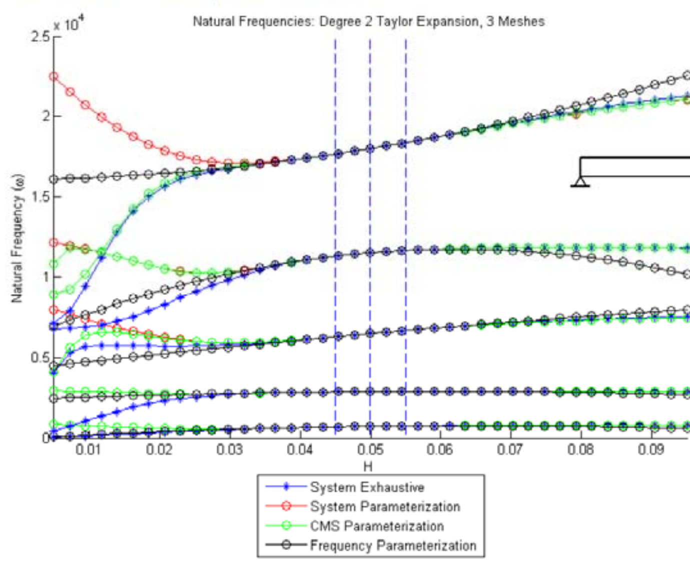
$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

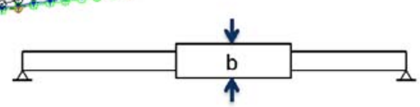
$$f^{(4)}(x) \approx \frac{f(x-2h) - 4f(x-h) + 6f(x) - 4f(x+h) + f(x+2h)}{h^4}$$

- Thus, derivative information can be calculated from perturbations of the system's model



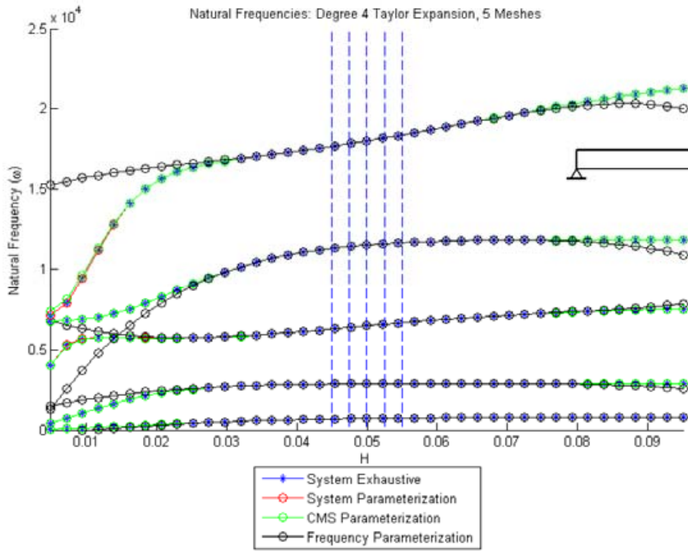
Varying Defect Thickness
2nd Order Expansion

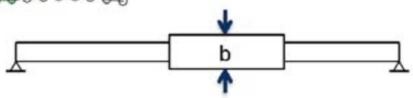




- Exhaustive and analytical solution lie atop one another
- Dashed blue lines indicates regime for calculating PROM

Varying Defect Thickness 4th Order Expansion

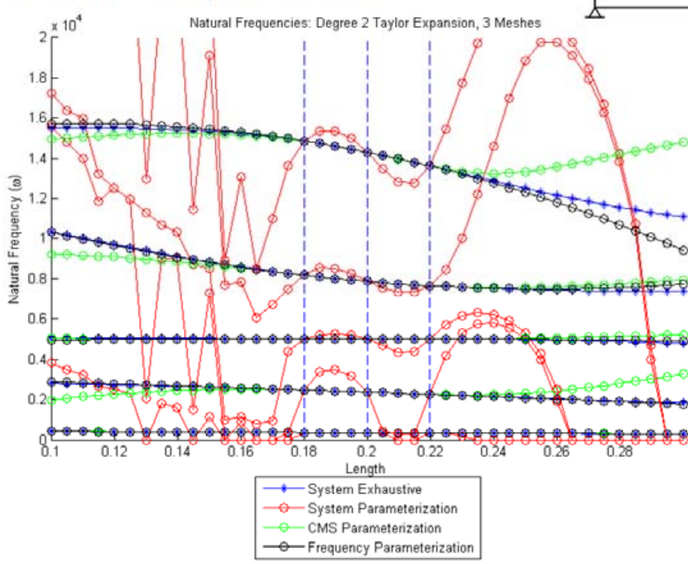




- In general, can achieve agreement well outside of the region used to calculate PROMs, but requires multiple derivatives…

12

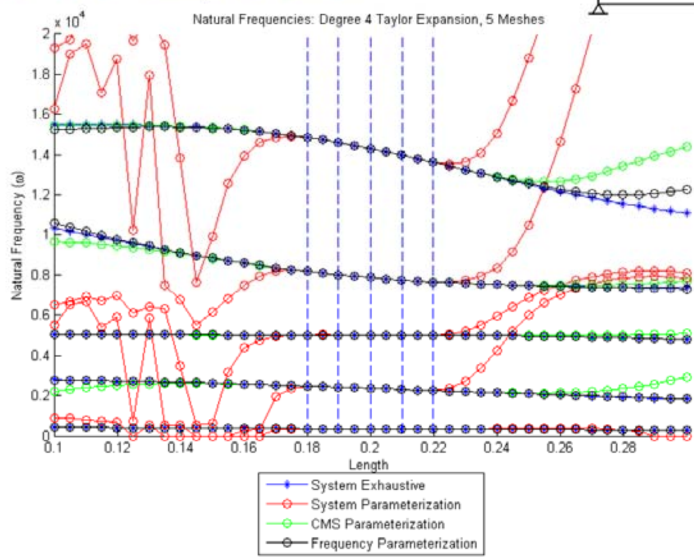
Varying Defect Length 2nd Order Expansion



- In general, system parameterization least accurate for geometrical variations

13

Varying Defect Length
4th Order Expansion



- Though, with sufficient derivatives, even the system level PROMs are predictive over the region used to calculate them…

14

Observations on the Finite Difference Based PROMs

- Fourth order system expansion is fairly accurate
  - Highest order term in system matrices is of third order
- System expansion is inaccurate for node shifting model variations
  - Terms with close proximity to zero; small deviations -> large error

- Reduced order model accuracy on par with eigenspace parameterization
  - Some applications only are interested in frequency characteristics, which would help guide choice in parameterization level
- PROM accuracy is good for large model variations

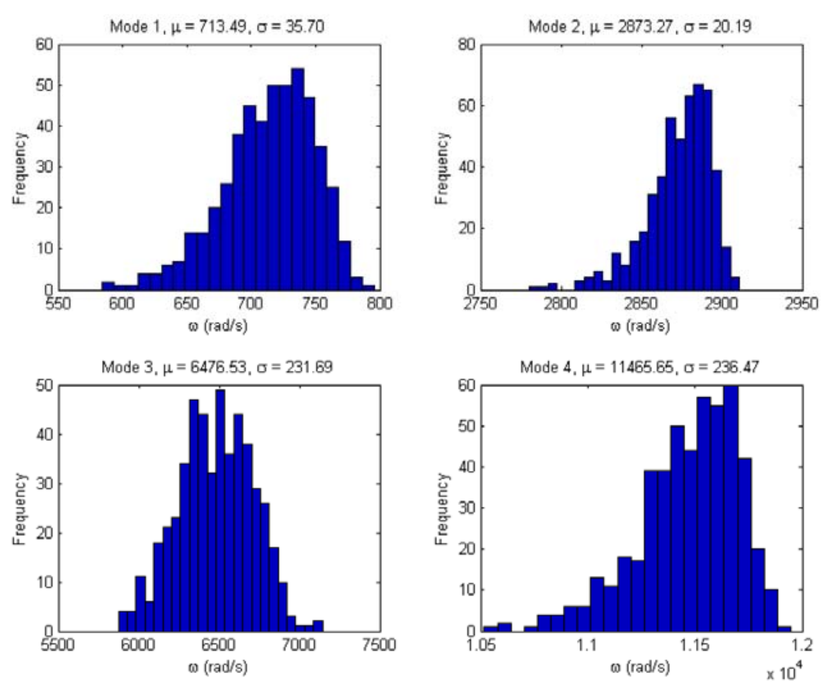
15

Multivariate Parameterization

- Generate multivariate expansion using N-dimensional Taylor series approximation for 5 variables simultaneously
- Specify highest order derivative (including mixed derivative terms)
- Generate Latin Hypercube Sample (LHS) based on probability distribution of parameters
- Plug samples into parameterized models and compare with true system response

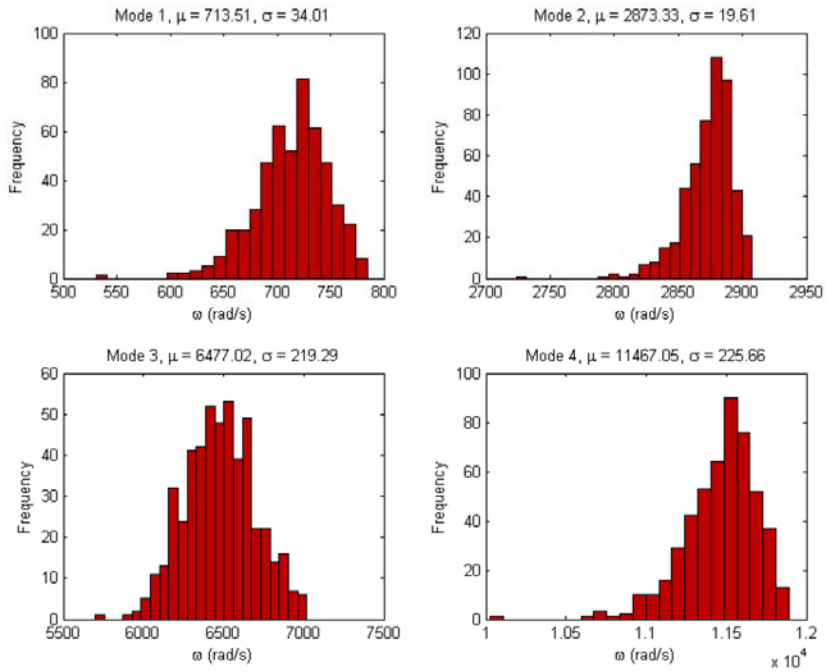
16

Exhaustive Sweep



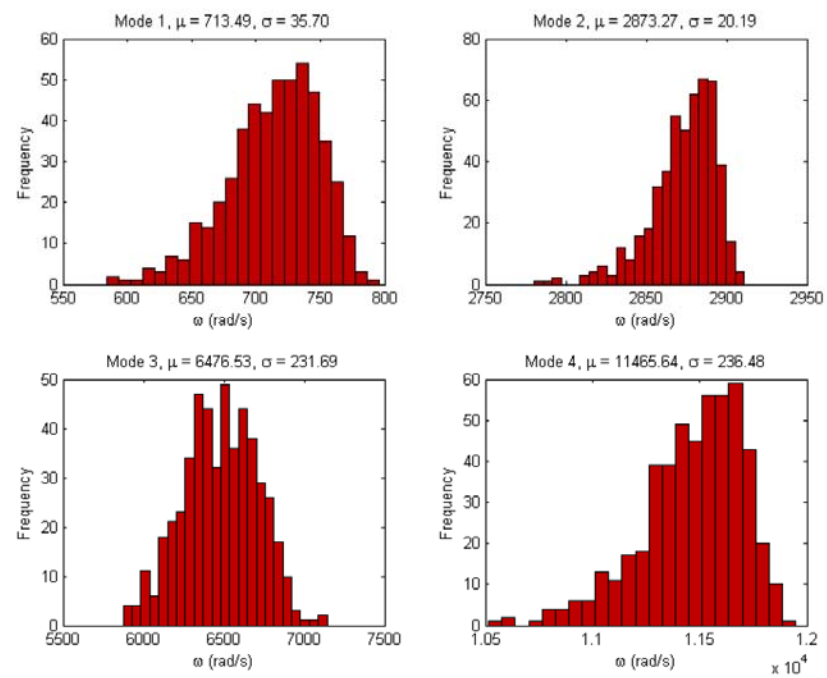
17

System Parameterization, 2nd Order



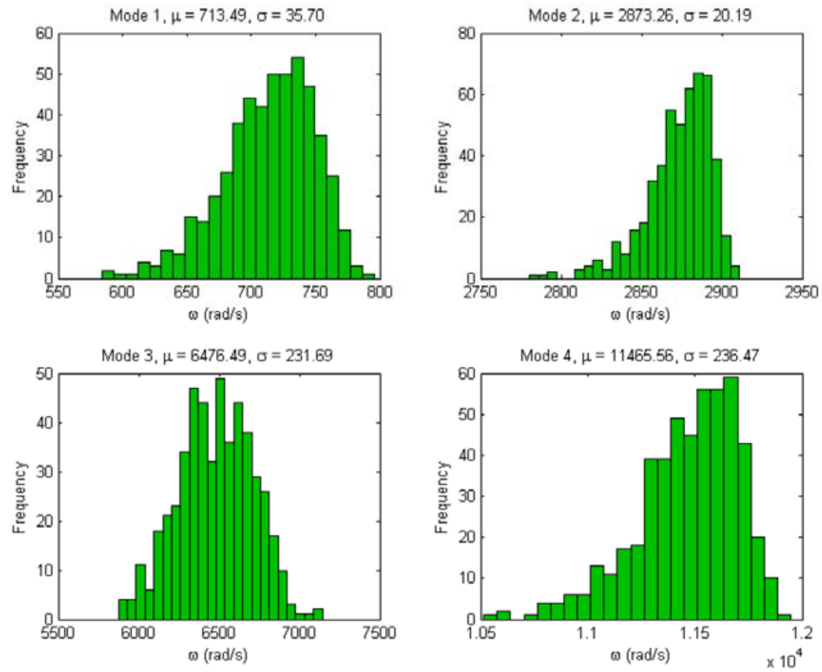
18

System Parameterization, 4th Order



19

ROM Parameterization, 4^{th} Order



Observations on the Multivariate Expansions of Finite Difference Based PROMs

- Even a second order expansion is good for high component variations
  - Means and standard deviations within ~10% of exhaustive approach
  - Histograms relatively similar
  - **Uses 51 meshes**
- Fourth order expansion is almost exact
  - Means and standard deviations are well within .01% of exhaustive
  - Histograms almost indistinguishable
  - **Uses 301 meshes**

Revisiting the Derivatives

Sandia National Laboratories

- Recall the finite difference expansions

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

$$f^{(4)}(x) \approx \frac{f(x-2h) - 4f(x-h) + 6f(x) - 4f(x+h) + f(x+2h)}{h^4}$$

- For two dimensions, even larger expressions results
- The number of meshes needed grows geometrically with the number of free variables.
- Quickly becomes intractable for a real problem with multiple dimensions of interest…

22

Alternatives Exist!

Sandia National Laboratories

- Parameterize a single element and propagate through models…
  - Book-keeping challenge…
- Replace finite difference based expansions with complex step approximations…
  - See the recent work by Millwater's group [1]…
- Alternatively, use hyper dual numbers…

23

[1]: A. Voorhees, H.R. Millwater, and R. Bagley, *Finite Elements in Analysis and Design*, **47**, pp. 1146-1156, 2011.

What Are Dual Numbers?

- Branch of generalized complex numbers
  - Ordinary complex numbers, $E^2 = i^2 = -1$
  - Double numbers, $E^2 = e^2 = 1$ (Clifford, 1873)
  - Dual numbers, $E^2 = \varepsilon^2 = 0$ (Study, 1903)
- The complex step approximation for a Taylor series

$$f(x + hE) = f(x) + hEf'(x) + \frac{1}{2!}h^2E^2f''(x) + \frac{1}{3!}h^3E^3f'''(x) + \cdots$$

simplifies based off of the choice for *E*...

24

The Complex Step Expansion

- Ordinary complex numbers ($E^2 = i^2 = -1$)

$$f(x + hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2f''(x) + \cdots\right)}_{\text{Real}} + h\underbrace{\left(f'(x) - \frac{1}{3!}h^3f'''(x) + \cdots\right)i}_{\text{Imaginary}}$$

- Double numbers ($E^2 = e^2 = 1$)

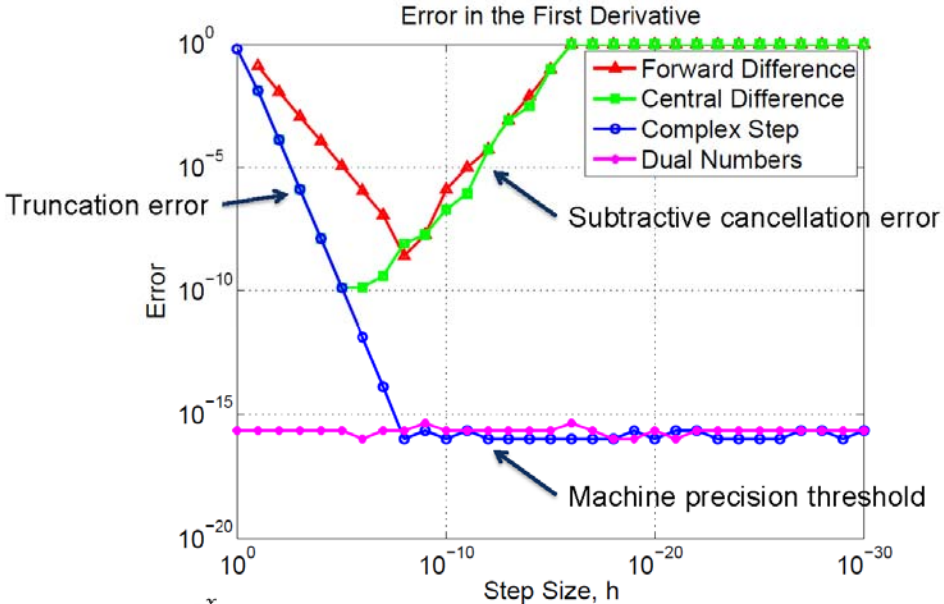
$$f(x + he) = \underbrace{\left(f(x) + \frac{1}{2!}h^2f''(x) + \cdots\right)}_{\text{Real}} + h\underbrace{\left(f'(x) + \frac{1}{3!}h^3f'''(x) + \cdots\right)e}_{\text{Non-Real}}$$

- Dual numbers ($E^2 = \varepsilon^2 = 0$)

$$f(x + h\varepsilon) = \underbrace{f(x)}_{\text{Real}} + \underbrace{hf'(x)\varepsilon}_{\text{Non-Real}}$$

25

Accuracy of First Derivative Calculations



$$f(x) = \frac{e^x}{\sqrt{(\sin x)^3 + (\cos x)^3}}$$

26

What About Higher Derivatives?

- Complex step method requires a differencing operation, which leads to subtractive cancellation error…
- Hyper dual numbers are dual numbers defined in multiple dimensions (Fike, 2011 & 2012)

$$x = x_0 + x_1\varepsilon_1 + x_2\varepsilon_2 + x_3\varepsilon_1\varepsilon_2$$
$$\varepsilon_1^2 = \varepsilon_2^2 = 0$$
$$\varepsilon_1 \neq \varepsilon_2 \neq 0$$
$$\varepsilon_1\varepsilon_2 = \varepsilon_2\varepsilon_1 \neq 0$$

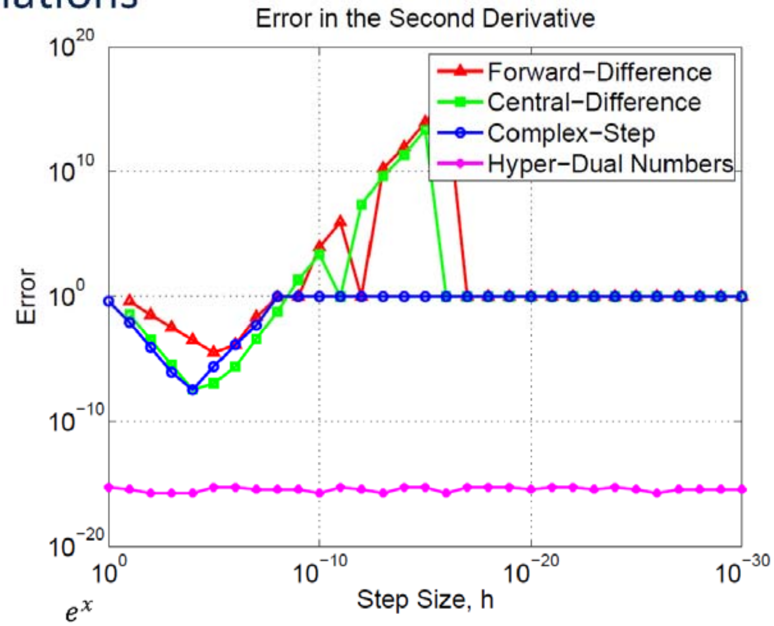
- This leads to the expansion

$$f(x + h_1\varepsilon_1 + h_2\varepsilon_2 + 0\varepsilon_1\varepsilon_2) = f(x) + h_1 f'(x)\varepsilon_1 + h_2 f'(x)\varepsilon_2 + h_1 h_2 f''(x)\varepsilon_1\varepsilon_2$$

which has exact first and second derivatives

27

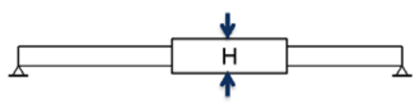
Accuracy of Second Derivative Calculations

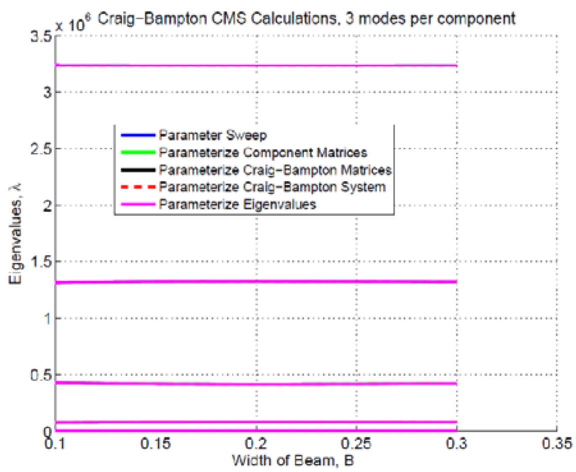


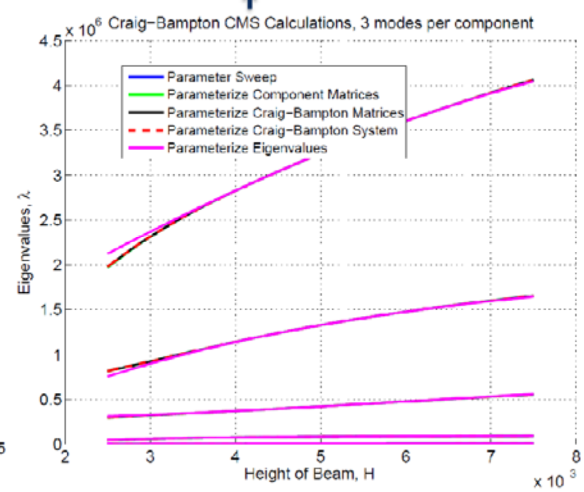
$$f(x) = \frac{e^x}{\sqrt{(\sin x)^3 + (\cos x)^3}}$$

28

Results of PROMs Constructed With Hyper Dual Numbers



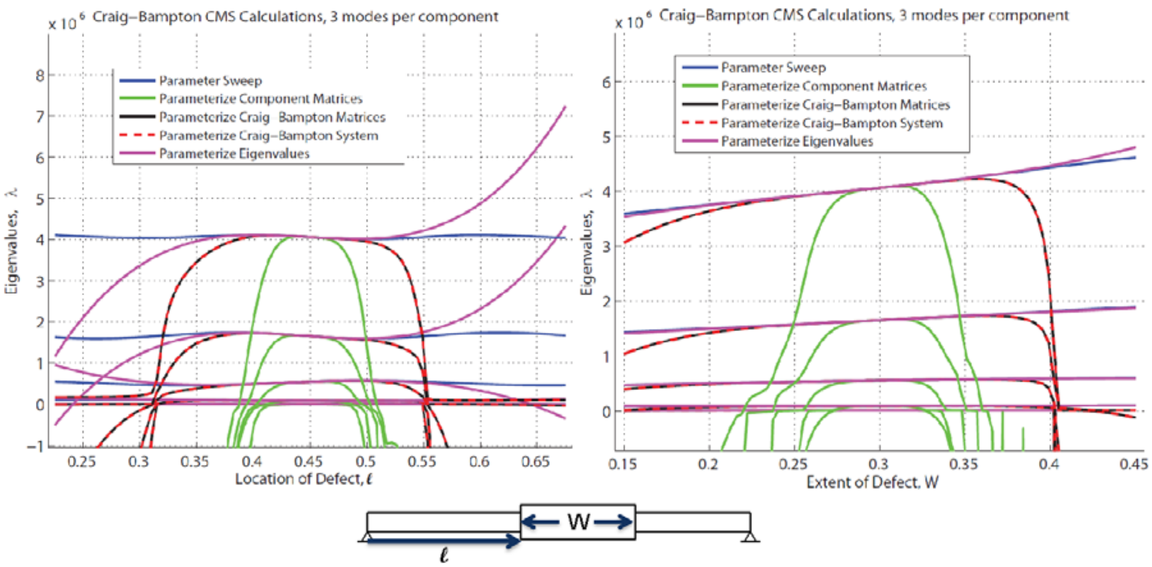




- Cubic expansion, based off of a single mesh of the beam

29

Results of PROMs Constructed With Hyper Dual Numbers

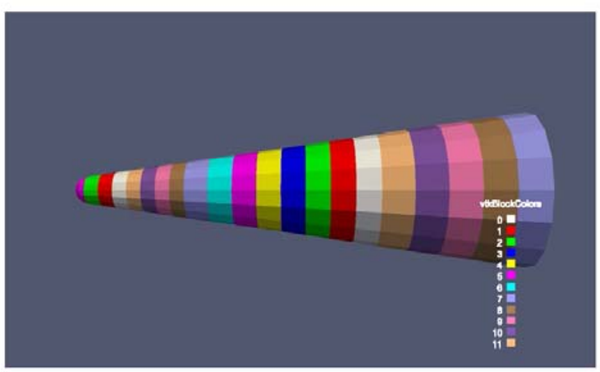


- Accuracy can be further improved with a meta-modeling approach, but that necessitates more meshes…

30

Application to Large Systems

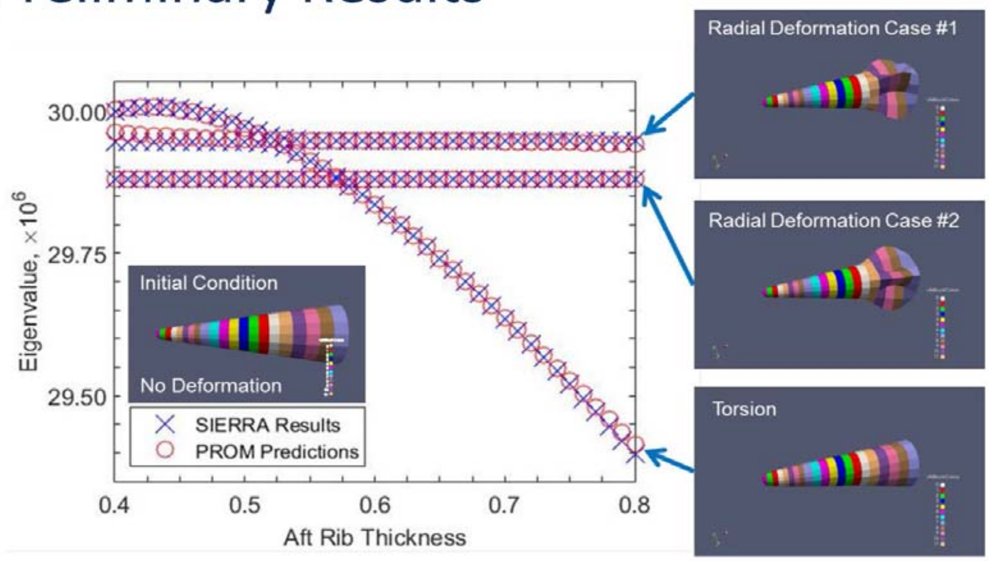
- Mathematics the exact same as for previous system
- The challenge is in prescribing geometric dimensions for variation…



- Several internal components; design challenge: ribs connecting exterior to interior

31

Preliminary Results



- Computational time for the PROM approximately 1/40th that of the high fidelity model, and no additional costs to consider geometry changes.

32

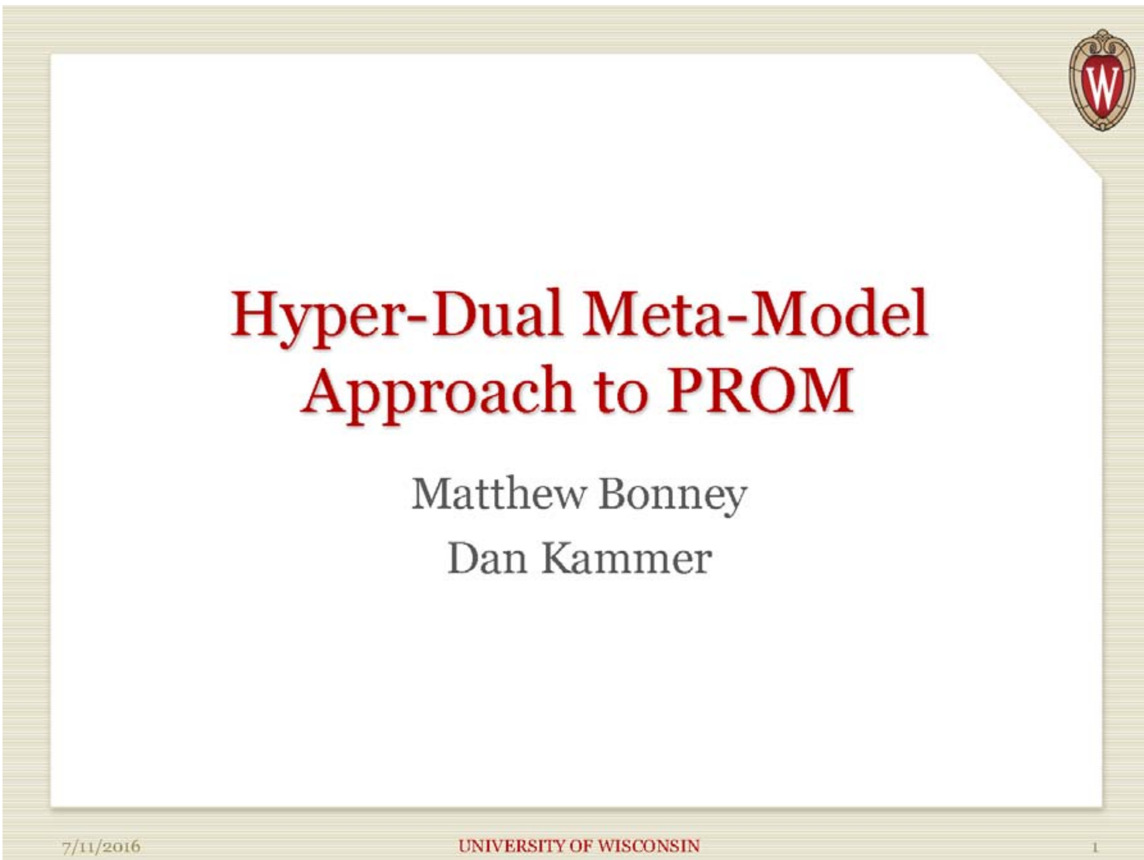
Summary & Conclusions

- Hyper dual numbers are a branch of generalized complex numbers with the property $\varepsilon^2 = 0$, $\varepsilon \neq 0$
- Building hyper dual numbers into our FEA code allows us to develop parameterized reduced order models (PROMs) with a single mesh
- Multiple levels of parameterization are investigated, and the results indicate that this parameterization technique is an effective and efficient approach to modeling
- Generally, the closer a parameterization is to the high fidelity FEA model, the worse that the PROM constructed from it will perform
- Results match analytical solutions very well for PROMs constructed from Craig-Bampton models or Eigen representations

33

3.1.4 Meta-Modeling, Matthew Bonney

The meta-modeling approach, developed by Daniel Kammer and Matthew Bonney of the University of Wisconsin, Madison, use sets of HD PROMs to develop globally accurate PROMs based off of a small number of numerical models. The advantage of this approach is that it does not depend on a single type of PROM formulation (it can be applied to HD PROMs, NX PROMs, or other types of PROMs), and that it can result in a globally accurate formulation for multivariate expansions. The trade-off, of course, is the high computational times necessitated by multiple PROM formulations.



Hyper-Dual (HD) Numbers

- Multi-dimensional expansion of Dual numbers
  - $\epsilon^2 = 0, \epsilon \neq 0$
- Truncates Taylor series
  - Truncation order depends on how many dual numbers
  - Can generate derivatives with no subtractive or truncation errors
- $f(x + h_1\epsilon_1 + h_2\epsilon_2 + 0\,\epsilon_{12}) = f(x) + h_1 f'(x)\epsilon_1 + h_2 f'(x)\epsilon_2 + h_1 h_2 f''(x)\epsilon_{12}$
- Theoretically, easy to add as many desired dual numbers
  - Computationally, not so easy

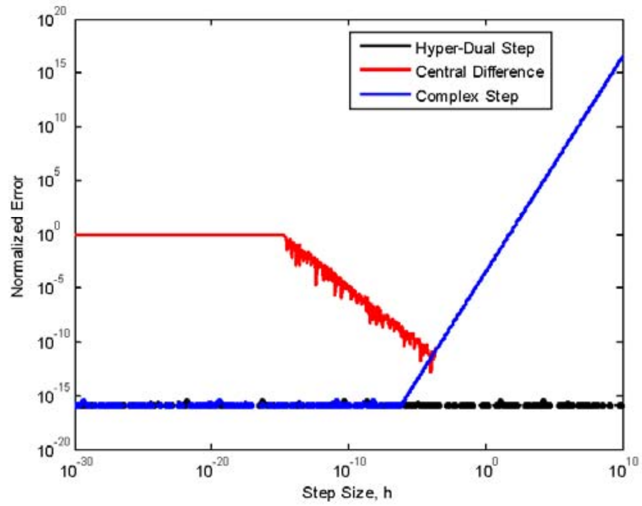
7/11/2016 UNIVERSITY OF WISCONSIN 2

HD Numbers Implementation

- Implementation
  - Matlab
  - Finite Elements
    - SIERRA
    - 3D Beams and Axisymmetric Solid in Matlab
- 2 types of function evaluations
  - Algorithmic
  - Analytical derivation

7/11/2016 UNIVERSITY OF WISCONSIN 3

Accuracy of HD Step



7/11/2016 UNIVERSITY OF WISCONSIN 4

Using a Hyper-Dual Step

Pros

- Single code evaluation
- No truncation error
- Exact derivatives
- Independent of step size

Cons

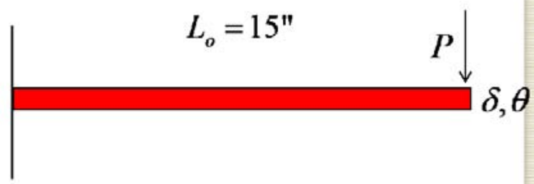
- Each code evaluation can become expensive
- Requires HD solver
- Only uses information from a single point
  - Limits range of effectiveness

7/11/2016 UNIVERSITY OF WISCONSIN 5

Hyper-Dual Meta-Model (HDM)

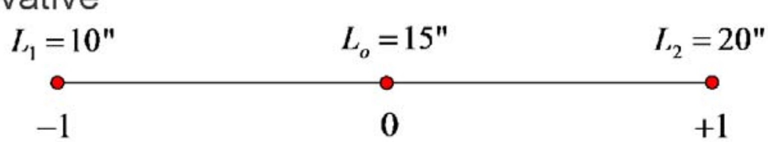
- Improve accuracy range of HD step
  - Combines the accuracy of HD step and range of finite difference
- Take information from multiple HD evaluations
- Uses basis functions to characterize data
  - Enforce function value and derivatives at each function evaluation
  - Polynomials, Splines, Sines
- Can be parameterized at any level
  - Can be more efficient at system level

Simple Example



- Deflection of cantilever tip due to applied force
- Parameterize stiffness matrix in terms of beam length *L*, very nonlinear
- Evaluate stiffness at 3 meshes with first derivative

$$EI\begin{bmatrix} 12/L^3 & 6/L^2 \\ 6/L^2 & 4/L \end{bmatrix}\begin{Bmatrix} \delta \\ \theta \end{Bmatrix} = \begin{Bmatrix} P \\ 0 \end{Bmatrix}$$



Simple Example Cont.

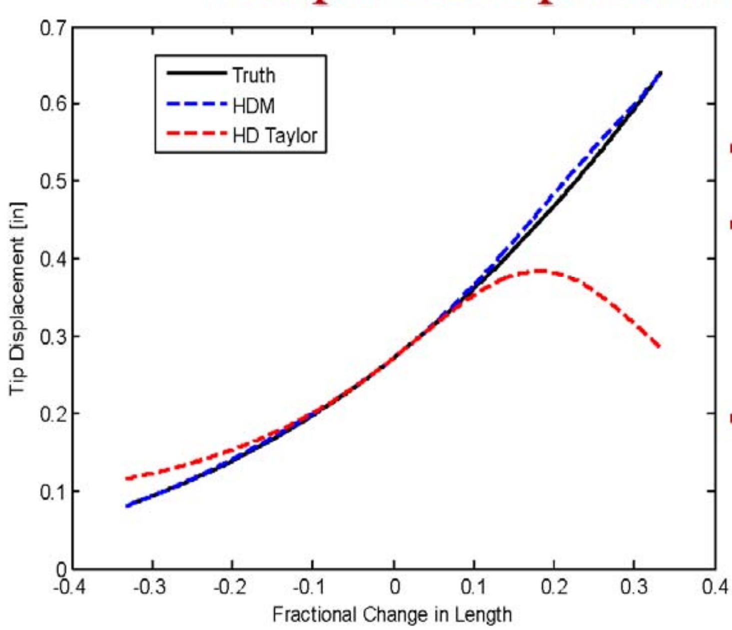
- Fit polynomial to stiffness matrix in terms of dimensionless length, γ
- Match values and first derivative at each evaluation
- Solve for unknown matrix coefficients

$$K = K_0 + K_1\gamma + K_2\gamma^2 + K_3\gamma^3 + K_4\gamma^4 + K_5\gamma^5$$

$$\frac{dK}{d\gamma} = K_1 + 2K_2\gamma + 3K_3\gamma^2 + 4K_4\gamma^3 + 5K_5\gamma^4$$

$$\begin{bmatrix} I & -I & I & -I & I & -I \\ I & 0 & 0 & 0 & 0 & 0 \\ I & I & I & I & I & I \\ 0 & I & -2I & 3I & -4I & 5I \\ 0 & I & 0 & 0 & 0 & 0 \\ 0 & I & 2I & 3I & 4I & 5I \end{bmatrix} \begin{bmatrix} K_0 \\ K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \end{bmatrix} = \begin{bmatrix} K(L_1) \\ K(L_o) \\ K(L_2) \\ K'(L_1) \\ K'(L_o) \\ K'(L_2) \end{bmatrix}$$

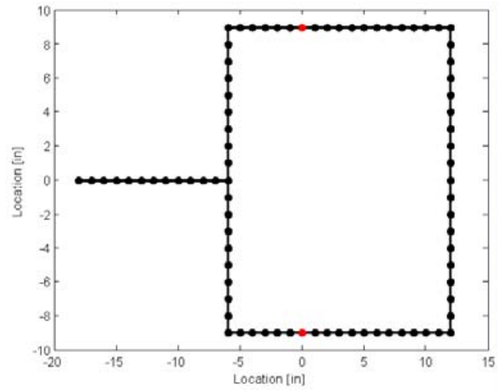
Simple Example Results



- First order HDM
- 2[nd] order HD w/ Taylor Series
- Second order HDM lies on top of Truth curve

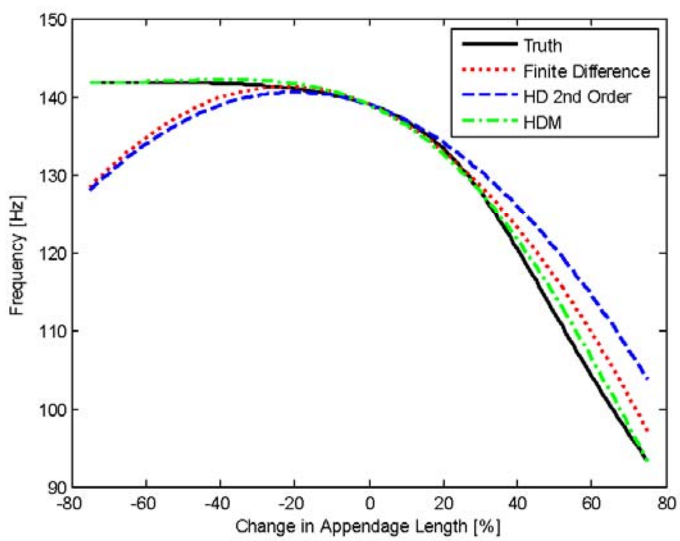
Application To Finite Elements

- Planar frame using substructuring
- Change length of appendage
- Nominal 12”
  - Vary from 3” to 21”
- Compare HD 2nd order, Finite Difference, and HDM
- Parameterized at frequency level



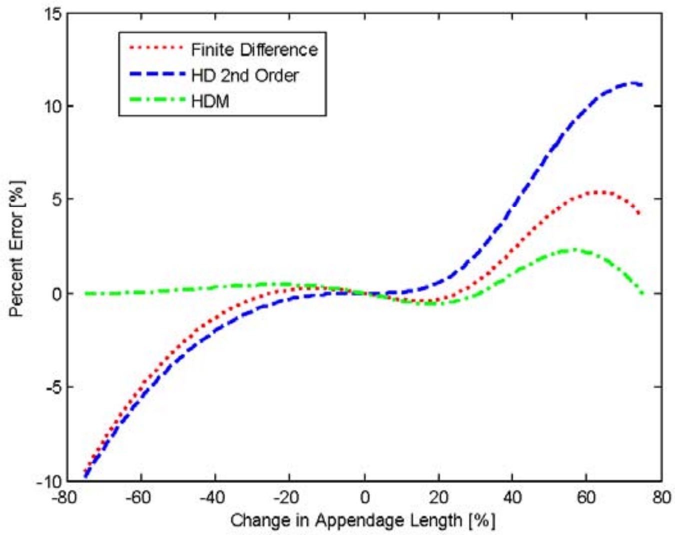
7/11/2016 UNIVERSITY OF WISCONSIN 10

FE Results



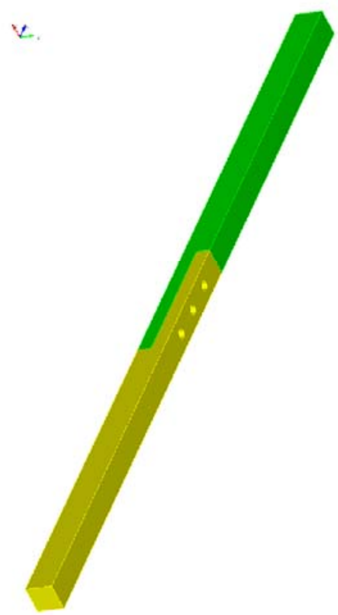
7/11/2016 UNIVERSITY OF WISCONSIN 11

FE Results Cont.

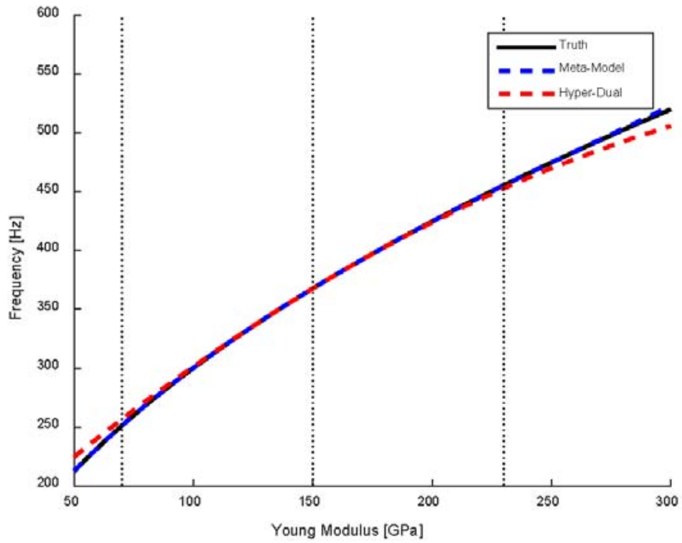


Now for the Hard Stuff

- Apply to complicated FE system
  - Brake-Reuss Beam
- Change in Young's Modulus
- 50-300 Gpa
- Parameterized at frequency level
- Uses Sierra to perform HD calculations

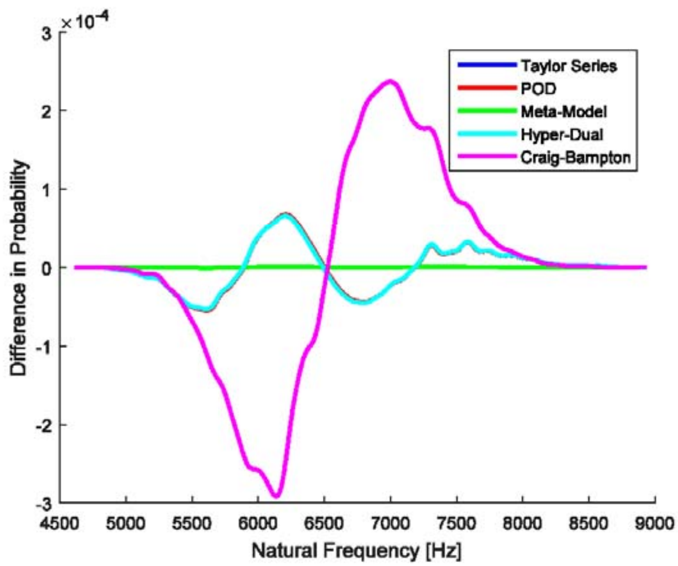


Parameter Sweep



7/11/2016 UNIVERSITY OF WISCONSIN 14

Distribution Propagation



7/11/2016 UNIVERSITY OF WISCONSIN 15

Quantitative Results

- Computational Time
  - HD : 21.40 sec
  - HDM : 0.24 sec
- RMS Error
  - HD : 1.488 %
  - HDM : 0.127 %
- Distribution Propagation
  - HD : 0.031 %
  - HDM : 0.009 %

Summary

- Hyper-Dual Meta-Model combines the accuracy of a Hyper-Dual step and the accuracy range of Finite Difference.
  - Perform multiple HD code evaluations
  - Apply basis function to match output and derivate at each code evaluation
- Applied to 3 different systems
  - 1 analytical, 1 material property, 1 geometric changes
  - Parameter Sweep and Distribution Propagation

Questions

Portions of the work presented in this proposal are conducted with support from Sandia National Laboratories. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

3.2 Summary of Sessions 2 & 3 – Complimentary Theories

The second and third sessions of the 2016 PROM Workshop focused on complimentary ROM techniques. The goals of this session were to inform the community about recent advances in other areas of ROM research, and to determine if there was any potential for adoption of those techniques into PROM methodologies. The five talks during these two sessions highlighted several topics:

- Multiscale modeling for material microstructure models ("Quantifying the Impact of Material-Model Error on Macroscale Quantities," by Judy Brown, and "Multiscale Modeling Applications," by Gustavo Castelluccio, both of Sandia National Laboratories);
- Proper Orthogonal Decompositions (POD) combined with Self Organizing Maps for real time data to decision ROMs (by Laura Mainini, MIT);
- Nonlinear ROM development ("Experimentally derived ROMs" by Ben Pacini and "Viscoelastic ROMs" by Rob Kuether, both of Sandia National Laboratories).

In particular, these talks focused on nonlinear models (both due to the material model and due to the structural model), alternative ROM strategies (such as the POD), and multiscale modeling frameworks (see [8], for instance). Themes that emerged from these presentations, in addition to opportunities to combine these theories with the PROM methodologies, are further discussed in Section 4.

3.3 Session 4 Presentations – Implementation Details and Round Robin Results

The last session of presentations at the 2016 PROM Workshop focused on two topics: one, to discuss the details of implementation for each methodology, and two, present the results of a round robin challenge organized specifically for this workshop.

3.3.1 NX-PROM Round Robin and Tutorial, Jau-Ching Lu

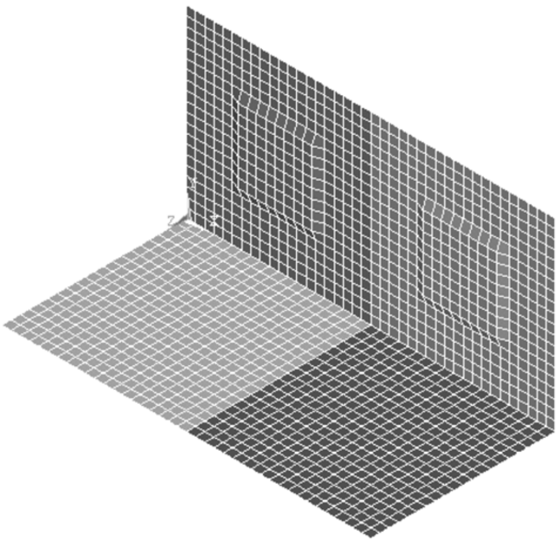
UNIVERSITY OF MICHIGAN

NX-PROM: Interpolation of matrix

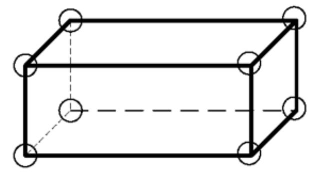
Jau-Ching Lu
Bogdan I. Epureanu
Matt Castanier
Sung Kwon Hong

1

System level FEM



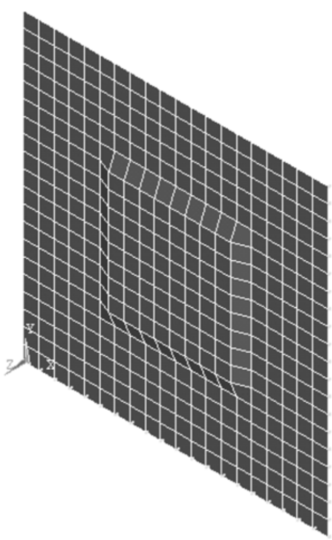
Element type: brick

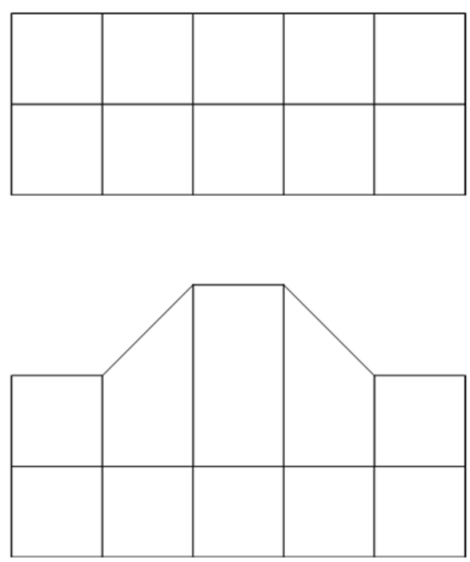


8 nodes @ each corner

2

Thickness variation



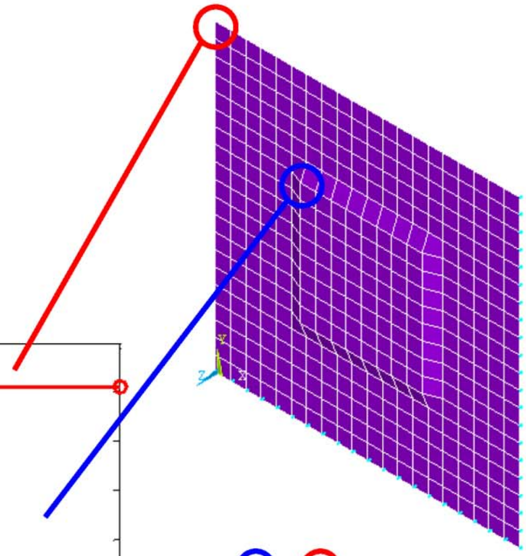


Thickness variation doesn't change the DOFs

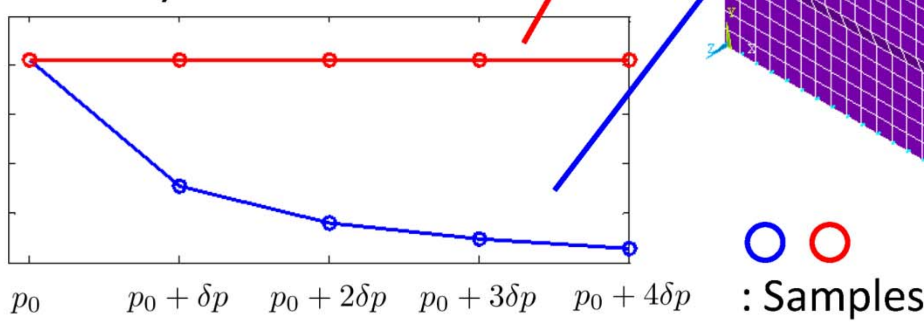
3

Effect of thickness variation

The stiffness **nonlinearly** varies with thickness variation



An entry of the K

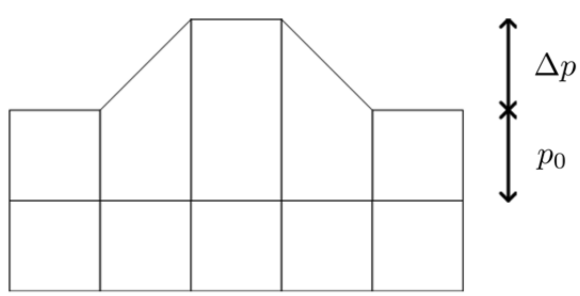


4

Stiffness interpolation

$$K(p_0+\Delta p) = \frac{K_0^{eq} + K_1^{eq}\Delta p + K_2^{eq}\Delta p^2 + K_3^{eq}\Delta p^3 + K_4^{eq}\Delta p^4}{D(\Delta p)}$$

$$D(\Delta p) = (1+\frac{\Delta p}{p_0})(1+\frac{1}{2}\frac{\Delta p}{p_0})(1+\frac{1}{3}\frac{\Delta p}{p_0})$$

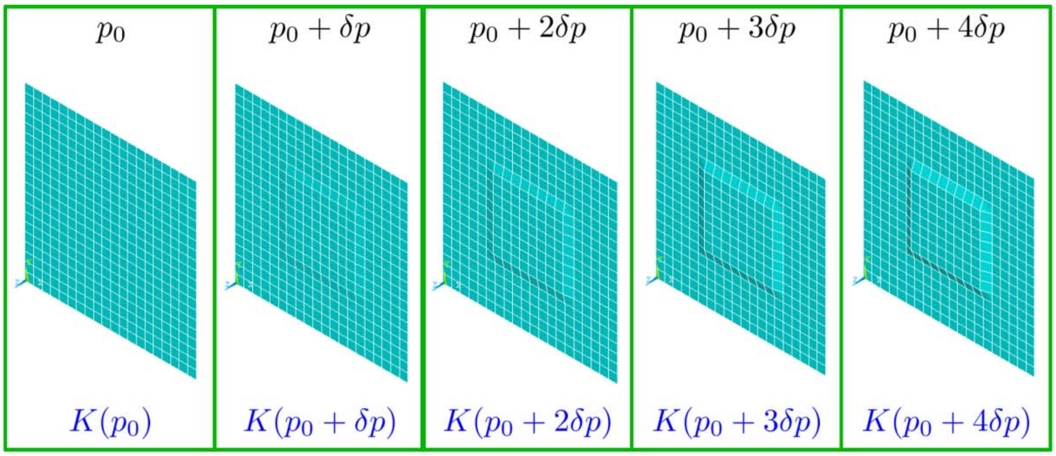


5

Sampling of K

$$K(p_0 + \Delta p) = \frac{K_0^{eq} + K_1^{eq}\Delta p + K_2^{eq}\Delta p^2 + K_3^{eq}\Delta p^3 + K_4^{eq}\Delta p^4}{D(\Delta p)}$$

5 unknowns: $K_0^{eq} \dots K_4^{eq}$ →5 equations



6

Sampling of K

$$K(p_0 + \Delta p) = \frac{K_0^{eq} + K_1^{eq}\Delta p + K_2^{eq}\Delta p^2 + K_3^{eq}\Delta p^3 + K_4^{eq}\Delta p^4}{D(\Delta p)}$$

5 samples in component level

$$\Delta p = 0 \rightarrow K(p_0) = K_0^{eq}$$

$$\Delta p = \delta p \rightarrow K(p_0 + \delta p) = \frac{1}{D(\delta p)}(K_0^{eq} + K_1^{eq}\delta p + \cdots + K_4^{eq}\delta p^4)$$

$$\Delta p = 2\delta p \rightarrow K(p_0 + 2\delta p) = \frac{1}{D(2\delta p)}(K_0^{eq} + K_1^{eq}2\delta p + \cdots + K_4^{eq}(2\delta p)^4)$$

$$\Delta p = 3\delta p \rightarrow K(p_0 + 3\delta p) = \frac{1}{D(3\delta p)}(K_0^{eq} + K_1^{eq}3\delta p + \cdots + K_4^{eq}(3\delta p)^4)$$

$$\Delta p = 4\delta p \rightarrow K(p_0 + 4\delta p) = \frac{1}{D(4\delta p)}(K_0^{eq} + K_1^{eq}4\delta p + \cdots + K_4^{eq}(4\delta p)^4)$$

7

Matrix-vector form

$$\begin{bmatrix} K(p_0) \\ K(p_0+\delta p) \\ K(p_0+2\delta p) \\ K(p_0+3\delta p) \\ K(p_0+4\delta p) \end{bmatrix} = \begin{bmatrix} K_0^{eq} \\ \frac{1}{D(\delta p)}(K_0^{eq} + K_1^{eq}\delta p + \cdots + K_4^{eq}\delta p^4) \\ \frac{1}{D(\delta p)}(K_0^{eq} + K_1^{eq}2\delta p + \cdots + K_4^{eq}(2\delta p)^4) \\ \frac{1}{D(\delta p)}(K_0^{eq} + K_1^{eq}3\delta p + \cdots + K_4^{eq}(3\delta p)^4) \\ \frac{1}{D(\delta p)}(K_0^{eq} + K_1^{eq}4\delta p + \cdots + K_4^{eq}(4\delta p)^4) \end{bmatrix}$$

$$\begin{bmatrix} K(p_0) \\ K(p_0+\delta p) \\ K(p_0+2\delta p) \\ K(p_0+3\delta p) \\ K(p_0+4\delta p) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{1}{D(\delta p)} & \frac{\delta p}{D(\delta p)} & \frac{\delta p^2}{D(\delta p)} & \frac{\delta p^3}{D(\delta p)} & \frac{\delta p^4}{D(\delta p)} \\ \frac{1}{D(2\delta p)} & \frac{2\delta p}{D(2\delta p)} & \frac{(2\delta p)^2}{D(2\delta p)} & \frac{(2\delta p)^3}{D(2\delta p)} & \frac{(2\delta p)^4}{D(2\delta p)} \\ \frac{1}{D(3\delta p)} & \frac{3\delta p}{D(3\delta p)} & \frac{(3\delta p)^2}{D(3\delta p)} & \frac{(3\delta p)^3}{D(3\delta p)} & \frac{(3\delta p)^4}{D(3\delta p)} \\ \frac{1}{D(4\delta p)} & \frac{4\delta p}{D(4\delta p)} & \frac{(4\delta p)^2}{D(4\delta p)} & \frac{(4\delta p)^3}{D(4\delta p)} & \frac{(4\delta p)^4}{D(4\delta p)} \end{bmatrix} \begin{bmatrix} K_0^{eq} \\ K_1^{eq} \\ K_2^{eq} \\ K_3^{eq} \\ K_4^{eq} \end{bmatrix}$$

8

Obtain the unknowns

$$\begin{bmatrix} K_0^{eq} \\ K_1^{eq} \\ K_2^{eq} \\ K_3^{eq} \\ K_4^{eq} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{1}{D(\delta p)} & \frac{\delta p}{D(\delta p)} & \frac{\delta p^2}{D(\delta p)} & \frac{\delta p^3}{D(\delta p)} & \frac{\delta p^4}{D(\delta p)} \\ \frac{1}{D(2\delta p)} & \frac{2\delta p}{D(2\delta p)} & \frac{(2\delta p)^2}{D(2\delta p)} & \frac{(2\delta p)^3}{D(2\delta p)} & \frac{(2\delta p)^4}{D(2\delta p)} \\ \frac{1}{D(3\delta p)} & \frac{3\delta p}{D(3\delta p)} & \frac{(3\delta p)^2}{D(3\delta p)} & \frac{(3\delta p)^3}{D(3\delta p)} & \frac{(3\delta p)^4}{D(3\delta p)} \\ \frac{1}{D(4\delta p)} & \frac{4\delta p}{D(4\delta p)} & \frac{(4\delta p)^2}{D(4\delta p)} & \frac{(4\delta p)^3}{D(4\delta p)} & \frac{(4\delta p)^4}{D(4\delta p)} \end{bmatrix}^{-1} \begin{bmatrix} K(p_0) \\ K(p_0+\delta p) \\ K(p_0+2\delta p) \\ K(p_0+3\delta p) \\ K(p_0+4\delta p) \end{bmatrix}$$

$$\begin{bmatrix} K_0^{eq} \\ K_1^{eq} \\ K_2^{eq} \\ K_3^{eq} \\ K_4^{eq} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} \\ A_{31} & A_{32} & A_{33} & A_{34} & A_{35} \\ A_{41} & A_{42} & A_{43} & A_{44} & A_{45} \\ A_{51} & A_{52} & A_{53} & A_{54} & A_{55} \end{bmatrix} \begin{bmatrix} K(p_0) \\ K(p_0+\delta p) \\ K(p_0+2\delta p) \\ K(p_0+3\delta p) \\ K(p_0+4\delta p) \end{bmatrix}$$

9

Re-write the equations

$$
\begin{aligned}
K_0^{eq} &= A_{11}K(p_0) + A_{12}K(p_0+\delta p) + \cdots + A_{15}K(p_0+4\delta p)\\
K_1^{eq} &= A_{21}K(p_0) + A_{22}K(p_0+\delta p) + \cdots + A_{25}K(p_0+4\delta p)\\
K_2^{eq} &= A_{31}K(p_0) + A_{32}K(p_0+\delta p) + \cdots + A_{35}K(p_0+4\delta p)\\
K_3^{eq} &= A_{41}K(p_0) + A_{42}K(p_0+\delta p) + \cdots + A_{45}K(p_0+4\delta p)\\
K_4^{eq} &= A_{51}K(p_0) + A_{52}K(p_0+\delta p) + \cdots + A_{55}K(p_0+4\delta p)
\end{aligned}
$$

$$
\begin{aligned}
K(p_0+\Delta p) = &\frac{K_0^{eq} + K_1^{eq}\Delta p + K_2^{eq}\Delta p^2 + K_3^{eq}\Delta p^3 + K_4^{eq}\Delta p^4}{D(\Delta p)}\\
&= b_0K(p_0) + b_1K(p_0+\delta p) + b_2K(p_0+2\delta p) + b_3K(p_0+3\delta p) + b_4K(p_0+4\delta p)
\end{aligned}
$$

10

Interpolation equation

$$
\begin{aligned}
K(p_0+\Delta p)&\\
&= b_0K(p_0) + b_1K(p_0+\delta p) + b_2K(p_0+2\delta p) + b_3K(p_0+3\delta p) + b_4K(p_0+4\delta p)
\end{aligned}
$$

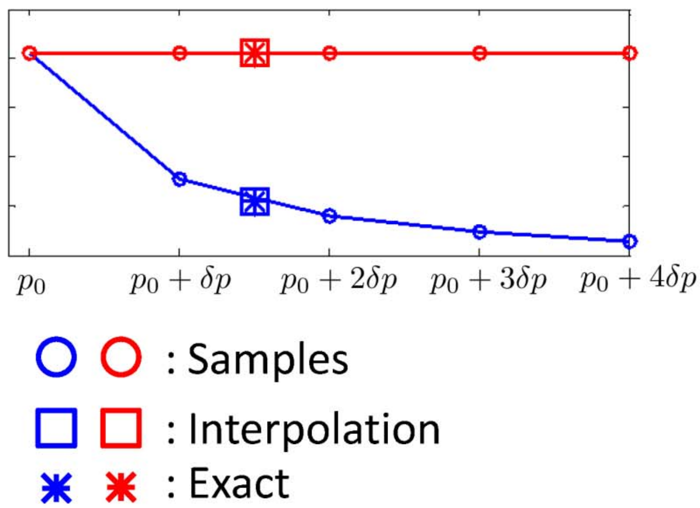
$$
\begin{aligned}
b_0 &= (A_{11} + A_{21}\Delta p + \cdots + A_{51}\Delta p^4)/D(\Delta p)\\
b_1 &= (A_{12} + A_{22}\Delta p + \cdots + A_{52}\Delta p^4)/D(\Delta p)\\
b_2 &= (A_{13} + A_{23}\Delta p + \cdots + A_{53}\Delta p^4)/D(\Delta p)\\
b_3 &= (A_{14} + A_{24}\Delta p + \cdots + A_{54}\Delta p^4)/D(\Delta p)\\
b_4 &= (A_{15} + A_{25}\Delta p + \cdots + A_{55}\Delta p^4)/D(\Delta p)
\end{aligned}
$$

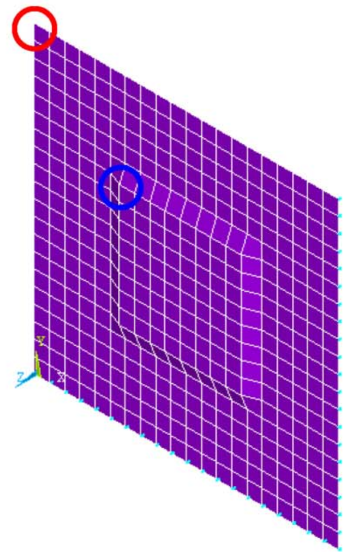
For each Δp, only $b_0 \dots b_4$ need to be re-calculated.

11

Results: entries

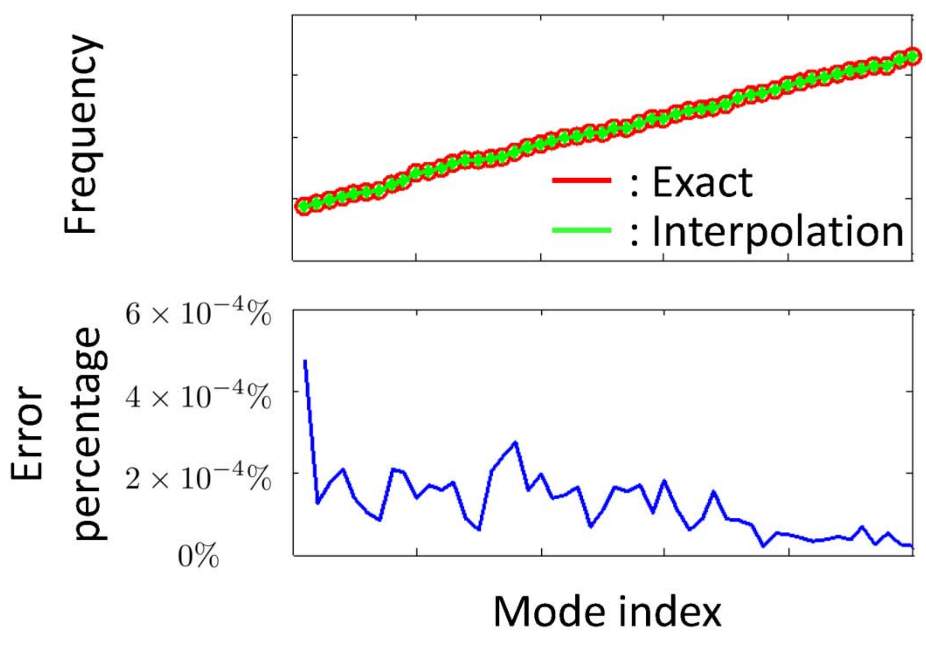
An entry of the K





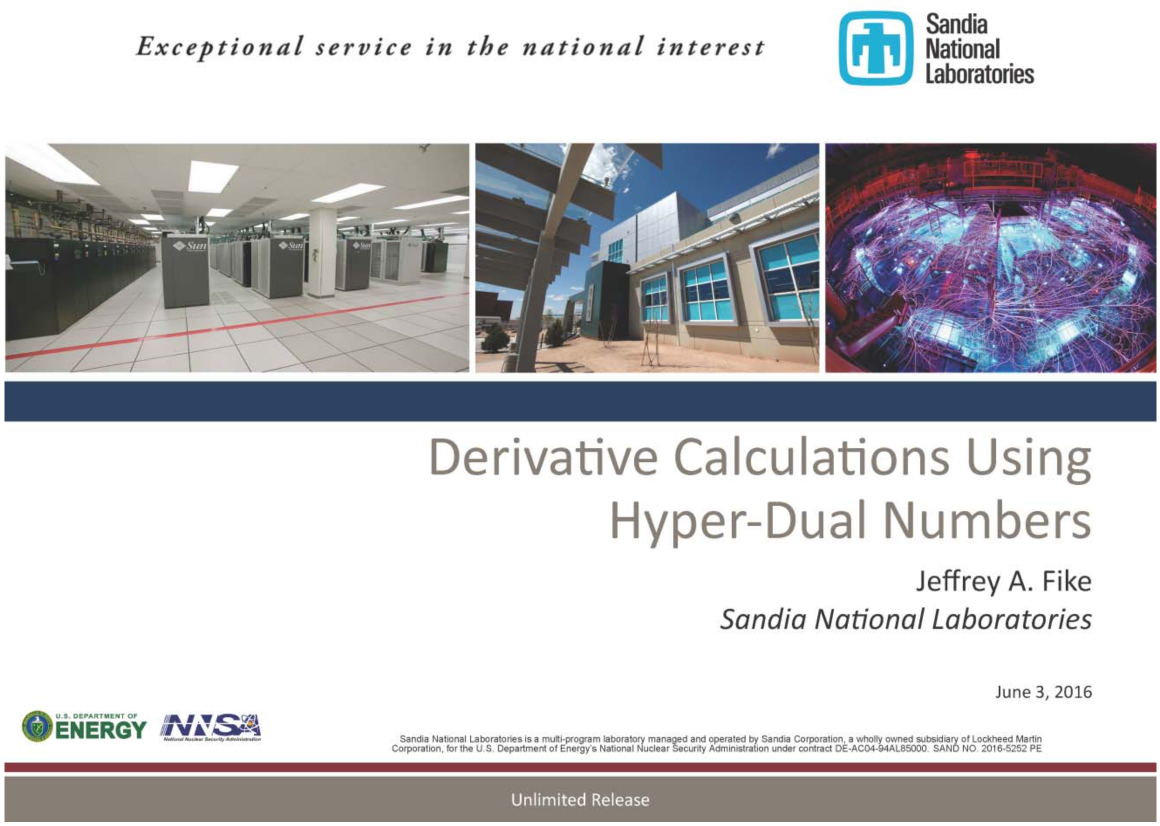
12

Results: natural frequency



13

3.3.2 Hyper Dual Number Round Robin and Tutorial, Jeff Fike



Derivative Calculations Using Hyper-Dual Numbers

Hyper-Dual Numbers [Fike and Alonso 2011] are an extension of Dual Numbers [Study 1903], one type of Generalized Complex Number.

Ordinary Complex Numbers can be used to compute accurate first derivatives. [Martins, Kroo, and Alonso 2000 and Martins, Sturdza, and Alonso 2003]

- Dual Numbers can be used in a similar manner to produce *exact* first derivatives. [Piponi 2004, Leuck and Nagel 1999]

Hyper-Dual Numbers enable exact calculations of second (or higher) derivatives.

Outline

Derivative Calculations

Mathematical Properties of Hyper-Dual Numbers

Implementation and Use of Hyper-Dual Numbers

Other Details

First-Derivative Finite-Difference Formulas

Forward-difference (FD) Approximation:

$$\frac{\partial f(\mathbf{x})}{\partial x_j} = \frac{f(\mathbf{x} + h\mathbf{e}_j) - f(\mathbf{x})}{h} + \mathcal{O}(h)$$

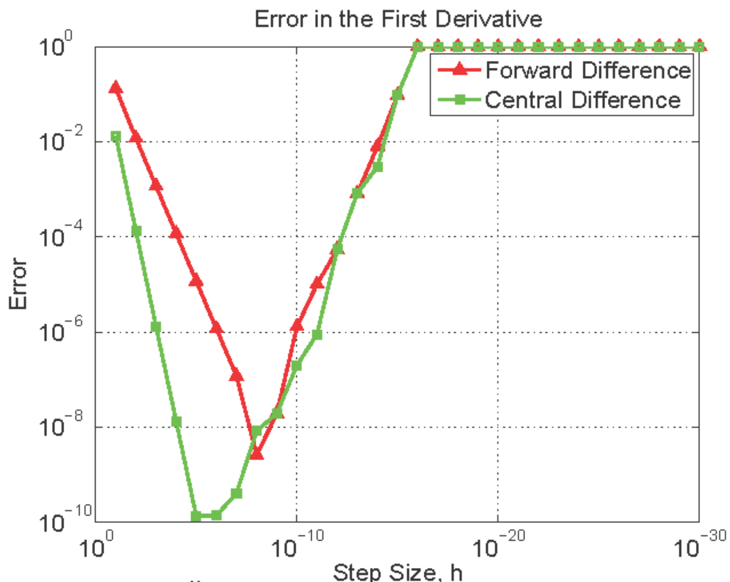
Central-Difference (CD) approximation:

$$\frac{\partial f(\mathbf{x})}{\partial x_j} = \frac{f(\mathbf{x} + h\mathbf{e}_j) - f(\mathbf{x} - h\mathbf{e}_j)}{2h} + \mathcal{O}(h^2)$$

Subject to truncation error and subtractive cancellation error

- Truncation error is associated with the higher order terms that are ignored when forming the approximation.
- Subtractive cancellation error is a result of performing these calculations on a computer with finite precision.

Accuracy of Finite-Difference Calculations



$$f(x) = \frac{e^x}{\sqrt{\sin^3 x + \cos^3 x}}$$

First-Derivative Complex-Step Approximation

Taylor series with an imaginary step:

$$f(x+hi) = f(x) + hf'(x)i - \frac{1}{2!}h^2 f''(x) - \frac{h^3 f'''(x)}{3!}i + \ldots$$

Unlimited Release

First-Derivative Complex-Step Approximation

Taylor series with an imaginary step:

$$f(x+hi) = f(x) + hf'(x)i - \frac{1}{2!}h^2 f''(x) - \frac{h^3 f'''(x)}{3!}i + \ldots$$

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h\underbrace{\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right)i}_{\text{imaginary}}$$

June 3, 2016 6

Unlimited Release

First-Derivative Complex-Step Approximation

Taylor series with an imaginary step:

$$f(x+hi) = f(x) + hf'(x)i - \frac{1}{2!}h^2 f''(x) - \frac{h^3 f'''(x)}{3!}i + \ldots$$

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h\underbrace{\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right)i}_{\text{imaginary}}$$

First-Derivative Complex-Step Approximation: [Martins, Kroo, and Alonso 2000 and Martins, Sturdza, and Alonso 2003]

$$f'(x) = \frac{\text{Im}\left[f(x+hi)\right]}{h} + \mathcal{O}(h^2)$$

- First derivatives are subject to truncation error but are not subject to subtractive cancellation error.

Generalized Complex Numbers

Generalized Complex Numbers [Kantor 1989] consist of one real part and one non-real part, $a + bE$

Generalized Complex Numbers

Generalized Complex Numbers [Kantor 1989] consist of one real part and one non-real part, $a + bE$

Addition:

$$(a + bE) + (c + dE) = (a + c) + (b + d)E$$

Unlimited Release

Generalized Complex Numbers

Generalized Complex Numbers [Kantor 1989] consist of one real part and one non-real part, $a + bE$

Addition:

$$(a + bE) + (c + dE) = (a + c) + (b + d)E$$

Multiplication:

$$(a + bE)(c + dE) = ac + (ad + bc)E + bd\,E^2$$

Generalized Complex Numbers

Generalized Complex Numbers [Kantor 1989] consist of one real part and one non-real part, $a + bE$

Addition:

$$(a + bE) + (c + dE) = (a + c) + (b + d)\,E$$

Multiplication:

$$(a + bE)\,(c + dE) = ac + (ad + bc)\,E + bd\,E^2$$

Three types based on choice for the non-real part, E:

- Ordinary Complex Numbers $E^2 = i^2 = -1$
- Double Numbers $E^2 = e^2 = 1$ [Clifford 1873]
- Dual Numbers $E^2 = \epsilon^2 = 0$ [Study 1903]

Generalized Complex Numbers

Ordinary Complex Numbers ($E^2 = i^2 = -1$):

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + \underbrace{h\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right)i}_{\text{imaginary}}$$

Generalized Complex Numbers

Ordinary Complex Numbers ($E^2 = i^2 = -1$):

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + \underbrace{h\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right)i}_{\text{imaginary}}$$

Double Numbers ($E^2 = e^2 = 1$):

$$f(x+he) = \underbrace{\left(f(x) + \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + \underbrace{h\left(f'(x) + \frac{1}{3!}h^2 f'''(x) + \ldots\right)e}_{\text{non-real}}$$

Unlimited Release

Generalized Complex Numbers

Ordinary Complex Numbers ($E^2 = i^2 = -1$):

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + \underbrace{h\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right)i}_{\text{imaginary}}$$

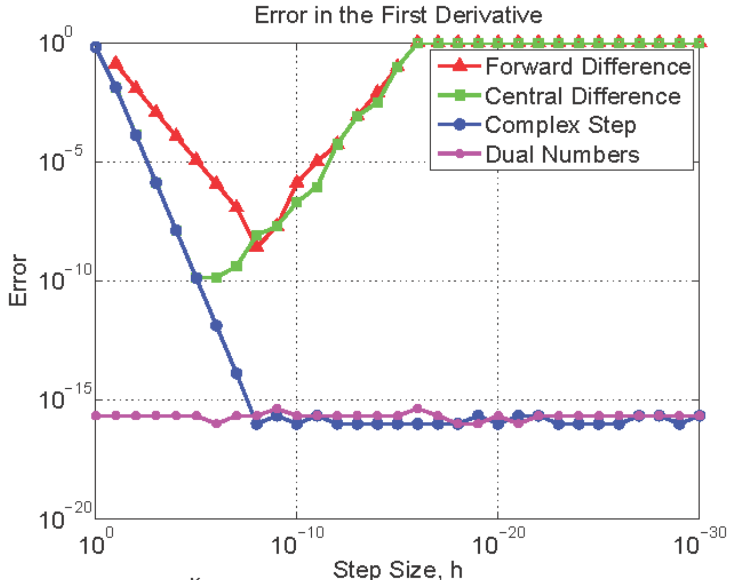
Double Numbers ($E^2 = e^2 = 1$):

$$f(x+he) = \underbrace{\left(f(x) + \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + \underbrace{h\left(f'(x) + \frac{1}{3!}h^2 f'''(x) + \ldots\right)e}_{\text{non-real}}$$

Dual Numbers ($E^2 = \epsilon^2 = 0$):

$$f(x + h\epsilon) = \underbrace{f(x)}_{\text{real}} + \underbrace{hf'(x)\epsilon}_{\text{non-real}}$$

Accuracy of First-Derivative Calculations



$$f(x) = \frac{e^x}{\sqrt{\sin^3 x + \cos^3 x}}$$

Second-Derivative Calculations?

Ordinary Complex Numbers ($E^2 = i^2 = -1$):

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h\underbrace{\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right)i}_{\text{imaginary}}$$

Double Numbers ($E^2 = e^2 = 1$):

$$f(x+he) = \underbrace{\left(f(x) + \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h\underbrace{\left(f'(x) + \frac{1}{3!}h^2 f'''(x) + \ldots\right)e}_{\text{non-real}}$$

Dual Numbers ($E^2 = \epsilon^2 = 0$):

$$f(x + h\epsilon) = \underbrace{f(x)}_{\text{real}} + \underbrace{hf'(x)\epsilon}_{\text{non-real}}$$

Second-Derivative Complex-Step

Sandia National Laboratories

One Second-Derivative Complex-Step Approximation:

$$f''(x) = \frac{2\left(f(x) - \mathrm{Re}[f(x+ih)]\right)}{h^2} + \mathcal{O}(h^2)$$

- Second derivatives are subject to subtractive cancellation error

Unlimited Release

Second-Derivative Complex-Step

Sandia National Laboratories

One Second-Derivative Complex-Step Approximation:

$$f''(x) = \frac{2\left(f(x) - \mathrm{Re}[f(x+ih)]\right)}{h^2} + \mathcal{O}(h^2)$$

- Second derivatives are subject to subtractive cancellation error

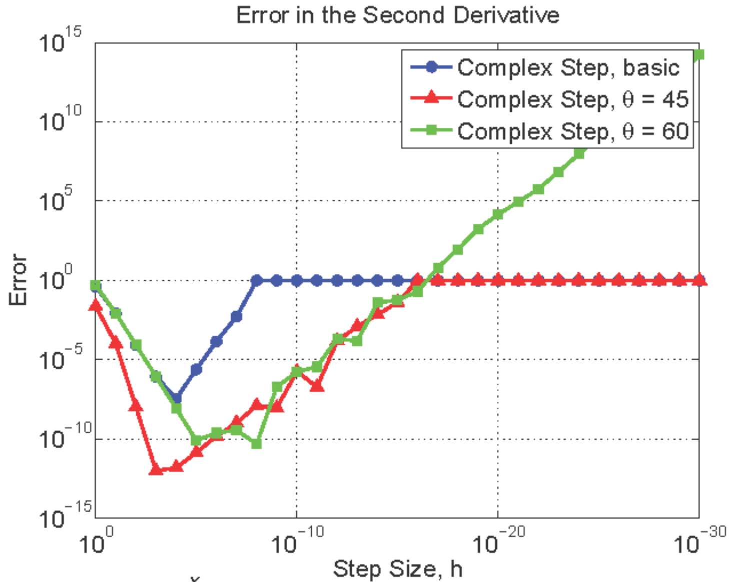
Alternative approximations: [Lai 2008]

$$f''(x) = \frac{\mathrm{Im}\left[f(x+i^{1/2}h) + f(x+i^{5/2}h)\right]}{h^2} + \mathcal{O}(h^4) \;:\; \theta = 45^\circ$$

$$f''(x) = \frac{2\,\mathrm{Im}\left[f(x+i^{2/3}h) + f(x+i^{8/3}h)\right]}{\sqrt{3}h^2} + \mathcal{O}(h^2) \;:\; \theta = 60^\circ$$

- These alternatives may offer improvements, but they are still subject to subtractive cancellation error

Alternative Complex-Step Approximations



$$f(x) = \frac{e^x}{\sqrt{\sin^3 x + \cos^3 x}}$$

Multiple Non-Real Parts

To avoid subtractive cancellation error:

- Second-derivative term should be the **leading term** of a non-real part
- First-derivative is already the leading term of a non-real part

Suggests that we need a number with **multiple non-real parts**

- Use higher-dimensional extensions of generalized complex numbers

Quaternions

Quaternions: one real part and three non-real parts

$$\begin{aligned} i^2 = j^2 = k^2 &= -1 \\ ijk &= -1 \end{aligned}$$

Unlimited Release

Quaternions

Quaternions: one real part and three non-real parts

$$\begin{aligned} i^2 = j^2 = k^2 &= -1 \\ ijk &= -1 \end{aligned}$$

Taylor series for a generic step, d:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \ldots$$

Quaternions

Quaternions: one real part and three non-real parts

$$i^2 = j^2 = k^2 = -1$$
$$ijk = -1$$

Taylor series for a generic step, d:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \ldots$$

For a quaternion step:

$$d = h_1 i + h_2 j + 0k$$
$$d^2 = -\left(h_1^2 + h_2^2\right)$$

- d^2 is real, second derivative only appears in the real part

Quaternions

Second-Derivative Quaternion-Step Approximation:

$$f''(x) = \frac{2\left(f(x) - \text{Re}[f(x + h_1 i + h_2 j + 0k)]\right)}{h_1^2 + h_2^2} + \mathcal{O}(h_1^2 + h_2^2)$$

- Subject to subtractive-cancellation error

Quaternion multiplication is not commutative, $ij = k$ but $ji = -k$

Quaternions

Second-Derivative Quaternion-Step Approximation:

$$f''(x) = \frac{2\left(f(x) - \text{Re}[f(x + h_1 i + h_2 j + 0k)]\right)}{h_1^2 + h_2^2} + \mathcal{O}(h_1^2 + h_2^2)$$

- Subject to subtractive-cancellation error

Quaternion multiplication is not commutative, $ij = k$ but $ji = -k$

Instead, consider a number with three non-real components E_1, E_2, and $(E_1 E_2)$ where multiplication is commutative, i.e. $E_1 E_2 = E_2 E_1$

Enforce Multiplication to be Commutative

Taylor series:

$$f(x + d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \ldots$$

Enforce Multiplication to be Commutative

Taylor series:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \dots$$

$$\begin{aligned}
d &= h_1 E_1 + h_2 E_2 + 0 E_1 E_2 \\
d^2 &= h_1^2 E_1^2 + h_2^2 E_2^2 + 2 h_1 h_2 E_1 E_2 \\
d^3 &= h_1^3 E_1^3 + 3 h_1 h_2^2 E_1 E_2^2 + 3 h_1^2 h_2 E_1^2 E_2 + h_2^3 E_2^3 \\
d^4 &= h_1^4 E_1^4 + 6 h_1^2 h_2^2 E_1^2 E_2^2 + 4 h_1^3 h_2 E_1^3 E_2 + 4 h_1 h_2^3 E_1 E_2^3 + h_2^4 E_2^4
\end{aligned}$$

Unlimited Release

Enforce Multiplication to be Commutative

Taylor series:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \dots$$

$$\begin{aligned}
d &= h_1 E_1 + h_2 E_2 + 0 E_1 E_2 \\
d^2 &= h_1^2 E_1^2 + h_2^2 E_2^2 + 2 h_1 h_2 E_1 E_2 \\
d^3 &= h_1^3 E_1^3 + 3 h_1 h_2^2 E_1 E_2^2 + 3 h_1^2 h_2 E_1^2 E_2 + h_2^3 E_2^3 \\
d^4 &= h_1^4 E_1^4 + 6 h_1^2 h_2^2 E_1^2 E_2^2 + 4 h_1^3 h_2 E_1^3 E_2 + 4 h_1 h_2^3 E_1 E_2^3 + h_2^4 E_2^4
\end{aligned}$$

- d^2 is first term with a non-zero $(E_1 E_2)$ component
- Second derivative is the leading term of the $(E_1 E_2)$ part
- As long as multiplication is commutative, and $E_1 E_2 \neq 0$, second-derivative approximations can be formed that are not subject to subtractive-cancellation error

Several Possible Number Systems

The requirement that $E_1E_2 = E_2E_1$ produces the constraint:

$$(E_1E_2)^2 = E_1E_2E_1E_2 = E_1E_1E_2E_2 = E_1^2E_2^2$$

Unlimited Release

Several Possible Number Systems

The requirement that $E_1E_2 = E_2E_1$ produces the constraint:

$$(E_1E_2)^2 = E_1E_2E_1E_2 = E_1E_1E_2E_2 = E_1^2E_2^2$$

This leaves many possibilities for the definitions of E_1 and E_2:

Several Possible Number Systems

The requirement that $E_1E_2 = E_2E_1$ produces the constraint:

$$(E_1E_2)^2 = E_1E_2E_1E_2 = E_1E_1E_2E_2 = E_1^2E_2^2$$

This leaves many possibilities for the definitions of E_1 and E_2:

- $E_1^2 = E_2^2 = -1$ which results in $(E_1E_2)^2 = 1$
  - Circular-Fourcomplex Numbers [Olariu 2002]
  - Multicomplex Numbers [Price 1991]

Unlimited Release

Several Possible Number Systems

The requirement that $E_1E_2 = E_2E_1$ produces the constraint:

$$(E_1E_2)^2 = E_1E_2E_1E_2 = E_1E_1E_2E_2 = E_1^2E_2^2$$

This leaves many possibilities for the definitions of E_1 and E_2:

- $E_1^2 = E_2^2 = -1$ which results in $(E_1E_2)^2 = 1$
  - Circular-Fourcomplex Numbers [Olariu 2002]
  - Multicomplex Numbers [Price 1991]

- Constrain $E_1^2 = E_2^2 = (E_1E_2)^2$
  - $E_1^2 = E_2^2 = (E_1E_2)^2 = 1$ Hyper-Double Numbers [Fike 2012]
  - $E_1^2 = E_2^2 = (E_1E_2)^2 = 0$ Hyper-Dual Numbers [Fike 2011]

Several Possible Number Systems

The requirement that $E_1E_2 = E_2E_1$ produces the constraint:

$$(E_1E_2)^2 = E_1E_2E_1E_2 = E_1E_1E_2E_2 = E_1^2E_2^2$$

This leaves many possibilities for the definitions of E_1 and E_2:

- $E_1^2 = E_2^2 = -1$ which results in $(E_1E_2)^2 = 1$
  - Circular-Fourcomplex Numbers [Olariu 2002]
  - Multicomplex Numbers [Price 1991]
- Constrain $E_1^2 = E_2^2 = (E_1E_2)^2$
  - $E_1^2 = E_2^2 = (E_1E_2)^2 = 1$ Hyper-Double Numbers [Fike 2012]
  - $E_1^2 = E_2^2 = (E_1E_2)^2 = 0$ Hyper-Dual Numbers [Fike 2011]

All are free from subtractive-cancellation error

- Truncation error can be reduced below machine precision
- Effectively exact

Hyper-Dual Numbers

Hyper-dual numbers have one real part and three non-real parts:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2$$

$$\epsilon_1^2 = \epsilon_2^2 = 0$$

$$\epsilon_1 \neq \epsilon_2 \neq 0$$

$$\epsilon_1\epsilon_2 = \epsilon_2\epsilon_1 \neq 0$$

Hyper-Dual Numbers

Hyper-dual numbers have one real part and three non-real parts:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2$$

$$\epsilon_1^2 = \epsilon_2^2 = 0$$

$$\epsilon_1 \neq \epsilon_2 \neq 0$$

$$\epsilon_1\epsilon_2 = \epsilon_2\epsilon_1 \neq 0$$

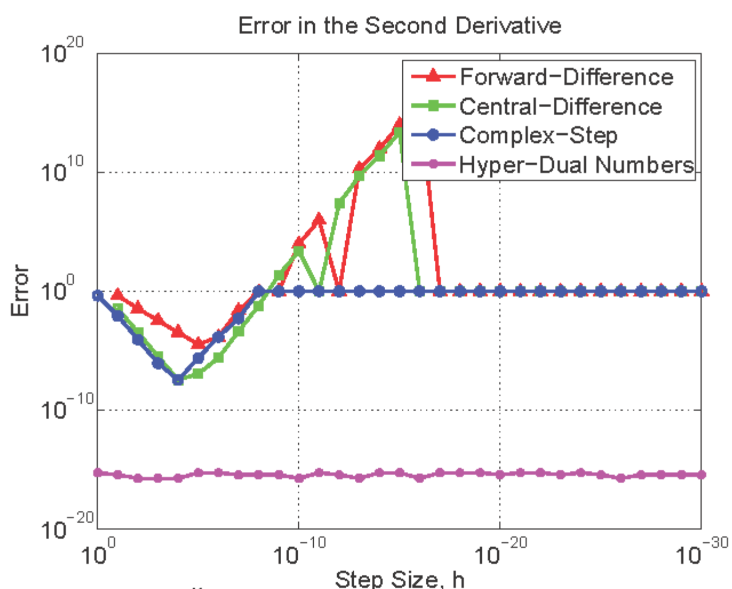
Taylor series truncates exactly at second-derivative term:

$$f(x+h_1\epsilon_1+h_2\epsilon_2+0\epsilon_1\epsilon_2) = f(x)+h_1 f'(x)\epsilon_1+h_2 f'(x)\epsilon_2+h_1h_2 f''(x)\epsilon_1\epsilon_2$$

- No truncation error and no subtractive-cancellation error
- Lack of higher order terms makes implementation easier

[Fike 2011 and Fike 2012]

Accuracy of Second-Derivative Calculations



$$f(x) = \frac{e^x}{\sqrt{\sin^3 x + \cos^3 x}}$$

Using Hyper-Dual Numbers

Evaluate a function with a hyper-dual step:

$$f(\mathbf{x} + h_1\epsilon_1\mathbf{e}_i + h_2\epsilon_2\mathbf{e}_j + \mathbf{0}\epsilon_1\epsilon_2)$$

Derivative information can be found by examining the non-real parts:

$$\frac{\partial f(\mathbf{x})}{\partial x_i} = \frac{\epsilon_1 part\left[f(\mathbf{x} + h_1\epsilon_1\mathbf{e}_i + h_2\epsilon_2\mathbf{e}_j + \mathbf{0}\epsilon_1\epsilon_2)\right]}{h_1}$$

$$\frac{\partial f(\mathbf{x})}{\partial x_j} = \frac{\epsilon_2 part\left[f(\mathbf{x} + h_1\epsilon_1\mathbf{e}_i + h_2\epsilon_2\mathbf{e}_j + \mathbf{0}\epsilon_1\epsilon_2)\right]}{h_2}$$

$$\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} = \frac{\epsilon_1\epsilon_2 part\left[f(\mathbf{x} + h_1\epsilon_1\mathbf{e}_i + h_2\epsilon_2\mathbf{e}_j + \mathbf{0}\epsilon_1\epsilon_2)\right]}{h_1 h_2}$$

Outline

Derivative Calculations

Mathematical Properties of Hyper-Dual Numbers

Implementation and Use of Hyper-Dual Numbers

Other Details

Arithmetic Operations

Consider two Hyper-Dual Numbers:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \qquad b = b_0 + b_1\epsilon_1 + b_2\epsilon_2 + b_3\epsilon_1\epsilon_2$$

Unlimited Release

Arithmetic Operations

Consider two Hyper-Dual Numbers:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \qquad b = b_0 + b_1\epsilon_1 + b_2\epsilon_2 + b_3\epsilon_1\epsilon_2$$

Addition:

$$a + b = (a_0 + b_0) + (a_1 + b_1)\,\epsilon_1 + (a_2 + b_2)\,\epsilon_2 + (a_3 + b_3)\,\epsilon_1\epsilon_2$$

June 3, 2016 22

Unlimited Release

Arithmetic Operations

Consider two Hyper-Dual Numbers:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \qquad b = b_0 + b_1\epsilon_1 + b_2\epsilon_2 + b_3\epsilon_1\epsilon_2$$

Addition:

$$a + b = (a_0 + b_0) + (a_1 + b_1)\,\epsilon_1 + (a_2 + b_2)\,\epsilon_2 + (a_3 + b_3)\,\epsilon_1\epsilon_2$$

Multiplication:

$$\begin{aligned} a * b \;=\; & (a_0 * b_0) + (a_0 * b_1 + a_1 * b_0)\,\epsilon_1 + (a_0 * b_2 + a_2 * b_0)\,\epsilon_2 \\ & + (a_0 * b_3 + a_1 * b_2 + a_2 * b_1 + a_3 * b_0)\,\epsilon_1\epsilon_2 \end{aligned}$$

Arithmetic Operations

Consider two Hyper-Dual Numbers:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \qquad b = b_0 + b_1\epsilon_1 + b_2\epsilon_2 + b_3\epsilon_1\epsilon_2$$

Addition:

$$a + b = (a_0 + b_0) + (a_1 + b_1)\,\epsilon_1 + (a_2 + b_2)\,\epsilon_2 + (a_3 + b_3)\,\epsilon_1\epsilon_2$$

Multiplication:

$$\begin{aligned} a * b \;=\; & (a_0 * b_0) + (a_0 * b_1 + a_1 * b_0)\,\epsilon_1 + (a_0 * b_2 + a_2 * b_0)\,\epsilon_2 \\ & + (a_0 * b_3 + a_1 * b_2 + a_2 * b_1 + a_3 * b_0)\,\epsilon_1\epsilon_2 \end{aligned}$$

- Hyper-Dual addition: 4 real additions
- Hyper-Dual multiplication: 9 real multiplications and 5 additions

Other Operations

The inverse:

$$\frac{1}{a} = \frac{1}{a_0} - \frac{a_1}{a_0^2}\epsilon_1 - \frac{a_2}{a_0^2}\epsilon_2 - \left(\frac{2a_1a_2}{a_0^3} - \frac{a_3}{a_0^2}\right)\epsilon_1\epsilon_2$$

- Only exists for $a_0 \neq 0$

This suggests a definition for the norm:

$$norm\,(a) = \sqrt{a_0^2}$$

This in turn implies that comparisons should only be made based on the real part.

- i.e. $a > b$ is equivalent to $a_0 > b_0$
- This allows the code to follow the same execution path as the real-valued code.

Mathematical Properties of Hyper-Dual Numbers

- Additive associativity, i.e. $(a + b) + c = a + (b + c)$,
- Additive commutativity, i.e. $a + b = b + a$,
- Additive identity, there exists a zero element, $z = 0 + 0\epsilon_1 + 0\epsilon_2 + 0\epsilon_1\epsilon_2$, such that $a + z = z + a = a$,
- Additive inverse, i.e. $a + (-a) = (-a) + a = 0$,
- Multiplicative associativity, i.e. $(a * b) * c = a * (b * c)$,
- Multiplicative commutativity, i.e. $a * b = b * a$,
- Multiplicative identity, there exists a unitary element, $1 + 0\epsilon_1 + 0\epsilon_2 + 0\epsilon_1\epsilon_2$, such that $a * 1 = 1 * a = a$,
- Left and right distributivity, i.e. $a * (b + c) = (a * b) + (a * c)$ and $(b + c) * a = (b * a) + (c * a)$.

These properties make hyper-dual numbers a commutative unital associative algebra.

Mathematical Properties of Hyper-Dual Numbers

Hyper-Dual Numbers are a commutative unital associative algebra.

Hyper-Dual Numbers are not a field (a commutative division algebra)

A division algebra requires the properties on the previous slide, plus a multiplicative inverse

- i.e. there exists an inverse, a^{-1}, such that $a * a^{-1} = a^{-1} * a = 1$ for every $a \neq 0 + 0\epsilon_1 + 0\epsilon_2 + 0\epsilon_1\epsilon_2$

Hyper-Dual Numbers have an inverse for every a with $norm(a) \neq 0$ (i.e. $a_0 \neq 0$)

Hyper-Dual Functions

Differentiable functions can be defined using the Taylor series for a generic hyper-dual number:

$$f(a) = f(a_0) + a_1 f'(a_0)\epsilon_1 + a_2 f'(a_0)\epsilon_2 + \left(a_3 f'(a_0) + a_1 a_2 f''(a_0)\right)\epsilon_1\epsilon_2$$

For instance:

$$a^3 = a_0^3 + 3a_1 a_0^2 \epsilon_1 + 3a_2 a_0^2 \epsilon_2 + \left(3a_3 a_0^2 + 6a_1 a_2 a_0\right)\epsilon_1\epsilon_2$$

$$\begin{aligned} \sin a &= \sin a_0 + a_1 \cos a_0 \epsilon_1 + a_2 \cos a_0 \epsilon_2 \\ &\quad + \left(a_3 \cos a_0 - a_1 a_2 \sin a_0\right)\epsilon_1\epsilon_2 \end{aligned}$$

Example Evaluation

A simple example hyper-dual function evaluation:

$$f(x) = \sin^3 x$$

This function can be evaluated as:

$$\begin{aligned} t_0 &= x \\ t_1 &= \sin t_0 \\ t_2 &= t_1^3 \end{aligned}$$

Unlimited Release

Example Evaluation

A simple example hyper-dual function evaluation:

$$f(x) = \sin^3 x$$

This function can be evaluated as:

$$\begin{aligned} t_0 &= x + h_1\epsilon_1 + h_2\epsilon_2 + 0\epsilon_1\epsilon_2 \\ t_1 &= \sin t_0 \\ &= \sin x + h_1 \cos x \epsilon_1 + h_2 \cos x \epsilon_2 - h_1 h_2 \sin x \epsilon_1 \epsilon_2 \\ t_2 &= t_1^3 \\ &= \sin^3 x + 3h_1 \cos x \sin^2 x \epsilon_1 + 3h_2 \cos x \sin^2 x \epsilon_2 \\ &\quad -\frac{3}{4} h_1 h_2 \left(\sin x - 3 \sin 3x\right) \epsilon_1 \epsilon_2 \end{aligned}$$

June 3, 2016 27

Unlimited Release

Outline

Derivative Calculations

Mathematical Properties of Hyper-Dual Numbers

Implementation and Use of Hyper-Dual Numbers

Other Details

Hyper-Dual Number Implementation

To use hyper-dual numbers, every operation in an analysis code must be modified to operate on hyper-dual numbers instead of real numbers

- Basic Arithmetic Operations: Addition, Multiplication, etc.
- Logical Comparison Operators: $\geq$, $\neq$, etc.
- Mathematical Functions: exponential, logarithm, sine, absolute value, etc.
- Input/Output Functions to write and display hyper-dual numbers

Hyper-dual numbers are implemented as a class using operator overloading in C++, CUDA, MATLAB and Fortran

- Change variable types, but body and structure of code is unaltered
- MPI datatype and reduction operations also implemented
- Implementations publicly available: `http://adl.stanford.edu/hyperdual`
- Implementations by others for Python and Julia

Variations of Hyper-Dual Numbers

Dual numbers produce exact first derivatives

Hyper-dual numbers, as described so far, produce exact second-derivatives

Third (or higher) derivatives can be computed by including additional non-real parts

- Third derivatives require an ϵ_3 term and its combinations

  $d = h_1\epsilon_1 + h_2\epsilon_2 + h_3\epsilon_3 + 0\epsilon_1\epsilon_2 + 0\epsilon_1\epsilon_3 + 0\epsilon_2\epsilon_3 + 0\epsilon_1\epsilon_2\epsilon_3$

Derivatives of complex-valued functions can be computed by defining hyper-dual numbers with complex-valued components

Vector-mode version propagates entire gradient and Hessian

- Eliminates redundant calculations, but increased memory requirements [Fike 2012]

Unlimited Release

Analysis Codes Using Hyper-Dual Numbers

Hyper-Dual Numbers can be applied to codes of arbitrary complexity in order to compute exact derivatives of output quantities of interest with respect to input parameters.

- Computational Fluid Dynamics
  - JOE, a parallel unstructured, 3-D, unsteady Reynolds-averaged Navier-Stokes code developed at Stanford University as part of PSAAP (the Department of Energy's Predictive Science Academic Alliance Program)
- Structural Dynamics
  - Sierra/SD (aka Salinas), a massively parallel, high-fidelity, structural dynamics finite element analysis code developed by Sandia National Laboratories

Converting Codes to Use Hyper-Dual Numbers

At a high level, converting a code to use Hyper-Dual Numbers requires little more than changing the variables types from real numbers to hyper-dual numbers.

- In some cases, there can be more effort required
- Requires modifying the source code
- Some codes make use of external libraries for which the source code is unavailable
  - Linear Solvers
  - Eigenvalue Solvers

Converting Codes to Use Hyper-Dual Numbers

At a high level, converting a code to use Hyper-Dual Numbers requires little more than changing the variables types from real numbers to hyper-dual numbers.

- In some cases, there can be more effort required
- Requires modifying the source code
- Some codes make use of external libraries for which the source code is unavailable
  - Linear Solvers
  - Eigenvalue Solvers

Hyper-Dual numbers can still be used to compute derivatives even if not all parts of a code can be modified

- Requires replicating the effect of a hyper-dual calculation, i.e. returning hyper-dual valued output containing the required derivative information

Differentiating the Solution of a Linear System

Solving the system:

$$\mathbf{A}(\mathbf{x})\mathbf{y}(\mathbf{x}) = \mathbf{b}(\mathbf{x})$$

Differentiating both sides with respect to the i^{th} component of **x** gives

$$\frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i}\mathbf{y}(\mathbf{x}) + \mathbf{A}(\mathbf{x})\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i} = \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_i}$$

Differentiating this result with respect to the j^{th} component of **x** gives

$$\frac{\partial^2 \mathbf{A}(\mathbf{x})}{\partial x_j \partial x_i}\mathbf{y}(\mathbf{x}) + \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i}\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_j} + \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j}\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i} + \mathbf{A}(\mathbf{x})\frac{\partial^2 \mathbf{y}(\mathbf{x})}{\partial x_j \partial x_i} = \frac{\partial^2 \mathbf{b}(\mathbf{x})}{\partial x_j \partial x_i}$$

Differentiating the Solution of a Linear System

This can be solved as:

$$\begin{bmatrix} \mathbf{A}(\mathbf{x}) & 0 & 0 & 0 \\ \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i} & \mathbf{A}(\mathbf{x}) & 0 & 0 \\ \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j} & 0 & \mathbf{A}(\mathbf{x}) & 0 \\ \frac{\partial^2 \mathbf{A}(\mathbf{x})}{\partial x_j \partial x_i} & \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j} & \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i} & \mathbf{A}(\mathbf{x}) \end{bmatrix} \begin{Bmatrix} \mathbf{y}(\mathbf{x}) \\ \frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i} \\ \frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_j} \\ \frac{\partial^2 \mathbf{y}(\mathbf{x})}{\partial x_j \partial x_i} \end{Bmatrix} = \begin{Bmatrix} \mathbf{b}(\mathbf{x}) \\ \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_i} \\ \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_j} \\ \frac{\partial^2 \mathbf{b}(\mathbf{x})}{\partial x_j \partial x_i} \end{Bmatrix}$$

Or

$$\mathbf{A}(\mathbf{x})\mathbf{y}(\mathbf{x}) = \mathbf{b}(\mathbf{x})$$

$$\mathbf{A}(\mathbf{x})\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i} = \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_i} - \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i}\mathbf{y}(\mathbf{x})$$

$$\mathbf{A}(\mathbf{x})\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_j} = \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_j} - \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j}\mathbf{y}(\mathbf{x})$$

$$\mathbf{A}(\mathbf{x})\frac{\partial^2 \mathbf{y}(\mathbf{x})}{\partial x_j \partial x_i} = \frac{\partial^2 \mathbf{b}(\mathbf{x})}{\partial x_j \partial x_i} - \frac{\partial^2 \mathbf{A}(\mathbf{x})}{\partial x_j \partial x_i}\mathbf{y}(\mathbf{x}) - \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i}\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_j} - \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j}\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i}$$

Derivatives of Eigenvalues and Eigenvectors

Eigenvalues and eigenvectors are solutions of the equation

$$(K - \lambda_\ell M)\,\phi_\ell = F_\ell \phi_\ell = 0$$

The first derivative of an eigenvalue is

$$\frac{\partial \lambda_\ell}{\partial x_i} = \phi_\ell^T \left(\frac{\partial K}{\partial x_i} - \lambda_\ell \frac{\partial M}{\partial x_i} \right) \phi_\ell$$

The first derivative of the eigenvector is

$$\frac{\partial \phi_\ell}{\partial x_i} = z_i + c_i \phi_\ell$$

where

$$F_\ell z_i = -\frac{\partial F_\ell}{\partial x_i} \phi_\ell$$

and

$$c_i = -\frac{1}{2} \phi_\ell^T \frac{\partial M}{\partial x_i} \phi_\ell - \phi_\ell^T M z_i$$

Outline

Derivative Calculations

Mathematical Properties of Hyper-Dual Numbers

Implementation and Use of Hyper-Dual Numbers

Other Details

Matrix Representation of Generalized Complex Numbers

Ordinary Complex Numbers:

$$a + bi = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} = a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + b \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Double Numbers:

$$a + be = \begin{bmatrix} a & b \\ b & a \end{bmatrix} = a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + b \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Dual Numbers:

$$a + b\epsilon = \begin{bmatrix} a & 0 \\ b & a \end{bmatrix} = a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + b \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$$

Matrix Representation of Hyper-Dual Numbers

Ordinary Complex Numbers:

$$a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \quad = \quad \begin{bmatrix} a_0 & 0 & 0 & 0 \\ a_1 & a_0 & 0 & 0 \\ a_2 & 0 & a_0 & 0 \\ a_3 & a_2 & a_1 & a_0 \end{bmatrix}$$

Questions?

Unlimited Release

Exceptional service in the national interest

Sandia National Laboratories

Round Robin: PROMs Using Hyper-Dual Numbers

Jeffrey A. Fike and Matthew R. W. Brake
Sandia National Laboratories

June 3, 2016

U.S. DEPARTMENT OF ENERGY NNSA

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2016-5251 PE

Outline

Sandia National Laboratories

Parameterized Reduced-Order Models Using Hyper-Dual Numbers

PROM Round Robin Test Case

PROM Comparison

Parameterized Reduced-Order Models (PROMs)

Create parameterized models using a Taylor series expansion about the nominal design:

$$\tilde{f}(x+\Delta x) = f(x) + (\Delta x)f'(x) + \frac{(\Delta x)^2}{2}f''(x) + \frac{(\Delta x)^3}{6}f'''(x) + \ldots.$$

Quantity of interest $f(x)$:

- Mass and stiffness matrices from Finite-Element Analysis (FEA)
- Outputs of FEA, such as displacements or natural frequencies

Perturbations Δx: variations in geometry or material properties

Terminology:

- Parameterized Full-Order Model if applied to FEA quantities
- Parameterized Reduced-Order Model (PROM) if applied to a Reduced-Order Model (ROM)
  - Craig-Bampton (C-B) Component Mode Synthesis (CMS) approach [Craig and Bampton 1968]

Finite Element Implementation

Salinas (a.k.a. Sierra/SD) was modified to operate on hyper-dual numbers and can produce up to exact third derivatives of outputs with respect to input parameters

- Inputs:
  - Derivatives with respect to Material Properties
  - Derivatives with respect to Geometric Perturbations
    - Geometric perturbations are currently computed internally from the nominal mesh by using a small set of geometric transformations to modify the nodal coordinates
    - Ongoing work to get geometric sensitivies from mesh generator
- Outputs:
  - Derivatives of Eigenvalues and Eigenvectors
  - Derivatives of Mass and Stiffness Matrices
  - Derivatives of Craig-Bampton Reduced Matrices

Outline

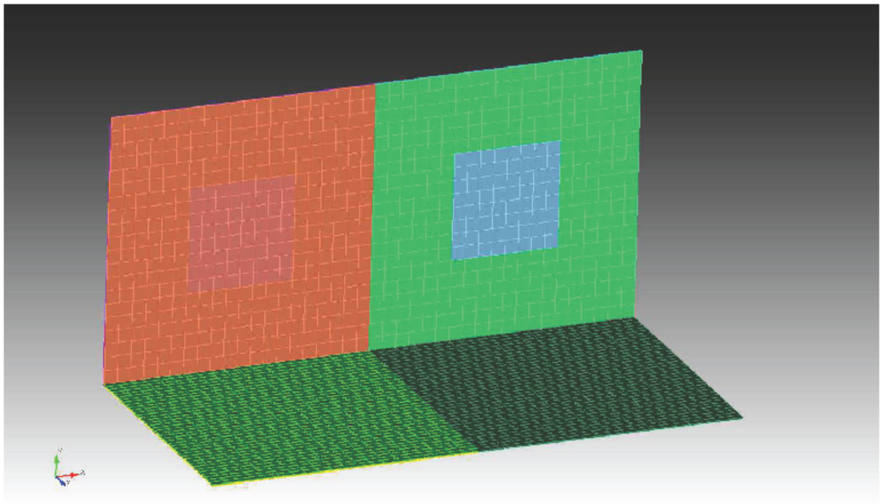
Parameterized Reduced-Order Models Using Hyper-Dual Numbers

PROM Round Robin Test Case

PROM Comparison

Unlimited Release

Test Case Geometry



- Plates are nominally 0.4 mm thick
- Square patch in center of vertical plates increased by up to 6 mm

Comparison Between Codes

First step: Try to get Salinas to agree with provided ANSYS results.

Unlimited Release

Comparison Between Codes

First step: Try to get Salinas to agree with provided ANSYS results.

Natural frequencies for first 5 vibration modes for unperturbed case:

	ANSYS	Salinas (original)	Salinas (fixed)
Mode 1	117.8 Hz	117.9 Hz	117.8 Hz
Mode 2	408.9 Hz	298.5 Hz	409.0 Hz
Mode 3	666.1 Hz	666.6 Hz	666.1 Hz
Mode 4	779.0 Hz	727.5 Hz	779.0 Hz
Mode 5	1792.6 Hz	1226.4 Hz	1792.4 Hz

Comparison Between Codes

First step: Try to get Salinas to agree with provided ANSYS results.

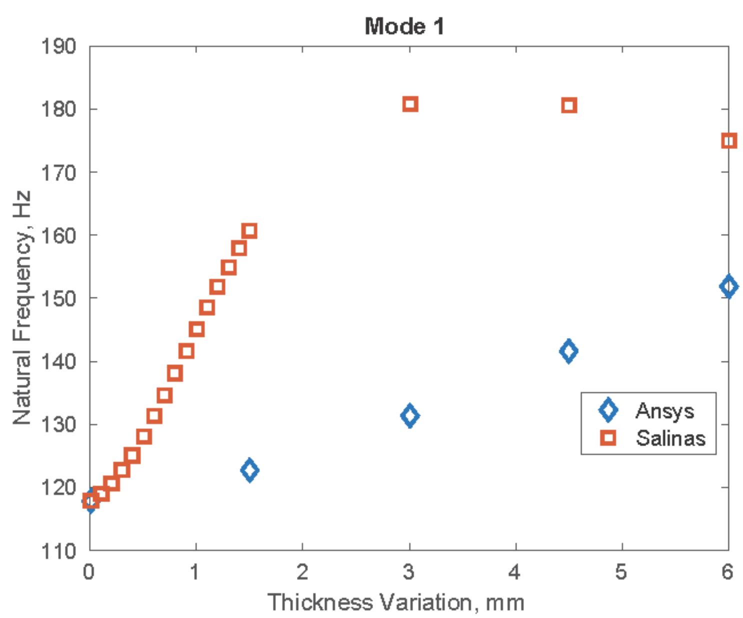
Natural frequencies for first 5 vibration modes for unperturbed case:

	ANSYS	Salinas (original)	Salinas (fixed)
Mode 1	117.8 Hz	117.9 Hz	117.8 Hz
Mode 2	408.9 Hz	298.5 Hz	409.0 Hz
Mode 3	666.1 Hz	666.6 Hz	666.1 Hz
Mode 4	779.0 Hz	727.5 Hz	779.0 Hz
Mode 5	1792.6 Hz	1226.4 Hz	1792.4 Hz

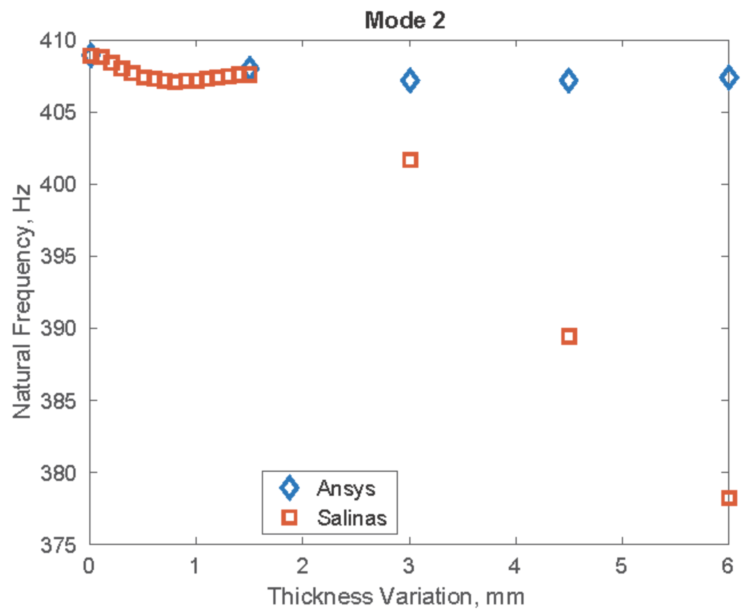
~~First and third modes match well, others do not.~~ (Fixed)

The original Salinas runs used meshes with a couple issues, fixing these issues results in a much better comparison.

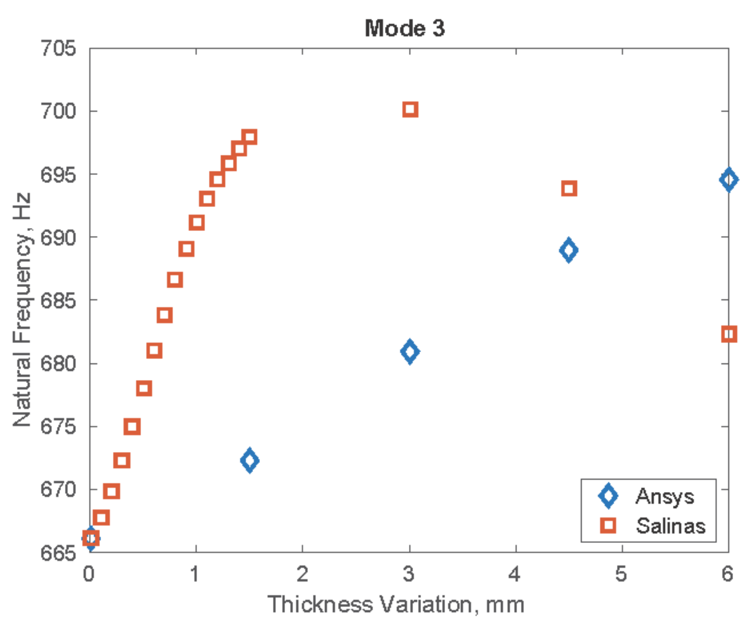
Natural Frequency as Thickness is Varied, Mode 1



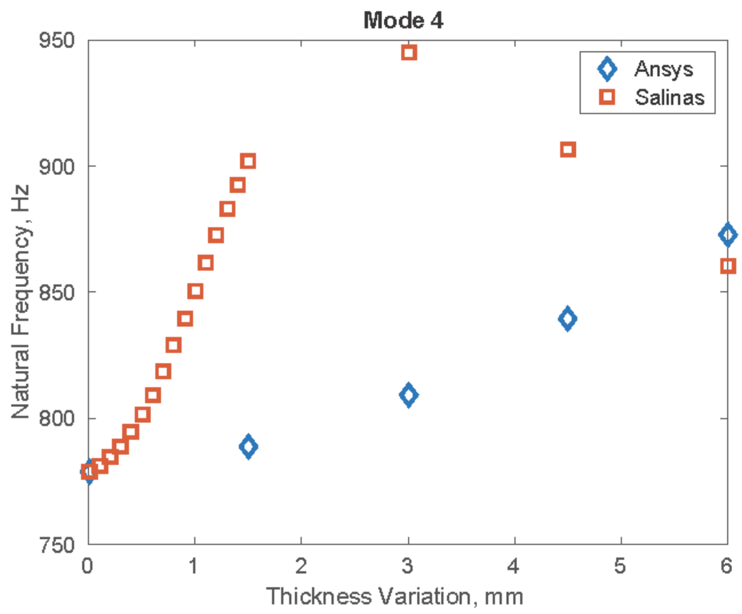
Natural Frequency as Thickness is Varied, Mode 2



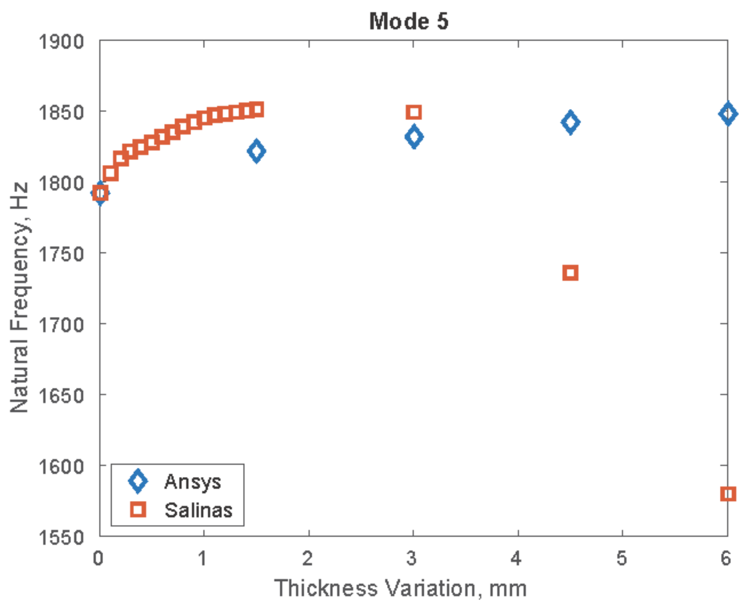
Natural Frequency as Thickness is Varied, Mode 3



Natural Frequency as Thickness is Varied, Mode 4



Natural Frequency as Thickness is Varied, Mode 5

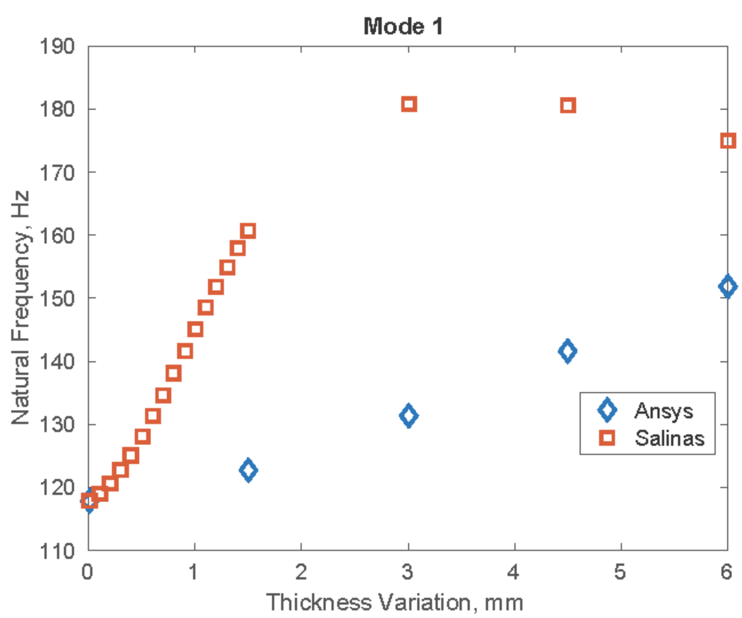


Discussion

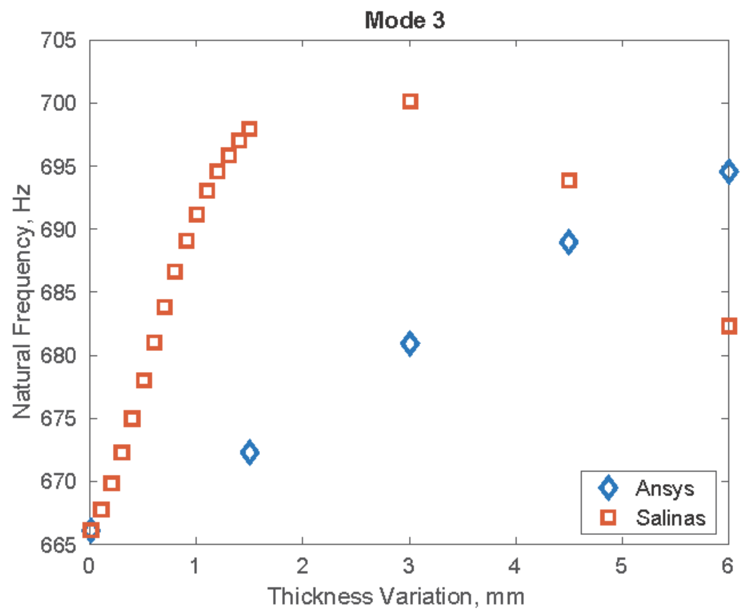
There are significant differences in the results

- Mode 1 and Mode 3 match well at 0 mm perturbation
  - ~~Other modes do not match as well even for 0mm~~ (Fixed)
- For these results, the plates are 2 elements thick
  - ~~Refining the mesh shifts the behavior, but does not drive Salinas results towards ANSYS results~~
- The behavior as the thickness is increased is different
  - ANSYS seems to be almost linear in most cases
  - Salinas exhibits non-linear behavior
  - Are thicknesses for ANSYS results correct?
- The element type choice in Salinas has a large impact
  - The chosen type is closer to a commercial code than the default (according to the Salinas documentation)
- Non-linear behavior suggests that HD PROMs constructed from 0mm case will not be very accurate for large perturbations

Natural Frequency as Thickness is Varied, Mode 1



Natural Frequency as Thickness is Varied, Mode 3

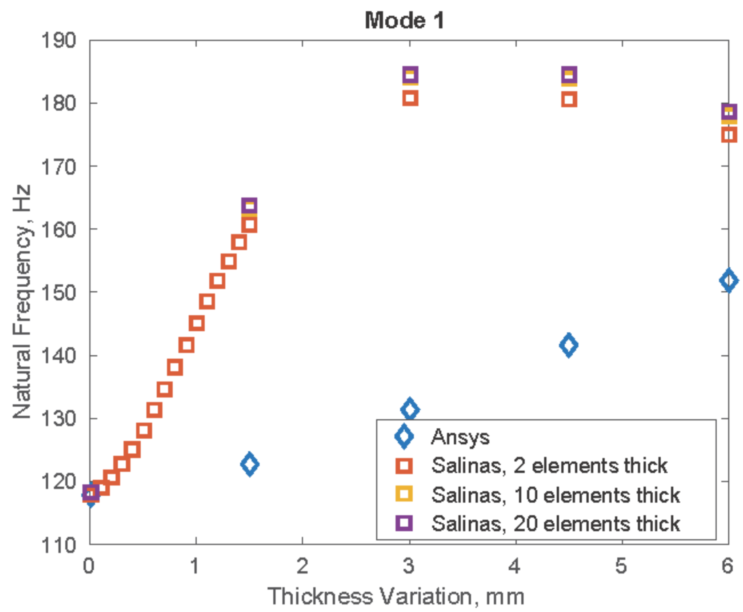


Discussion

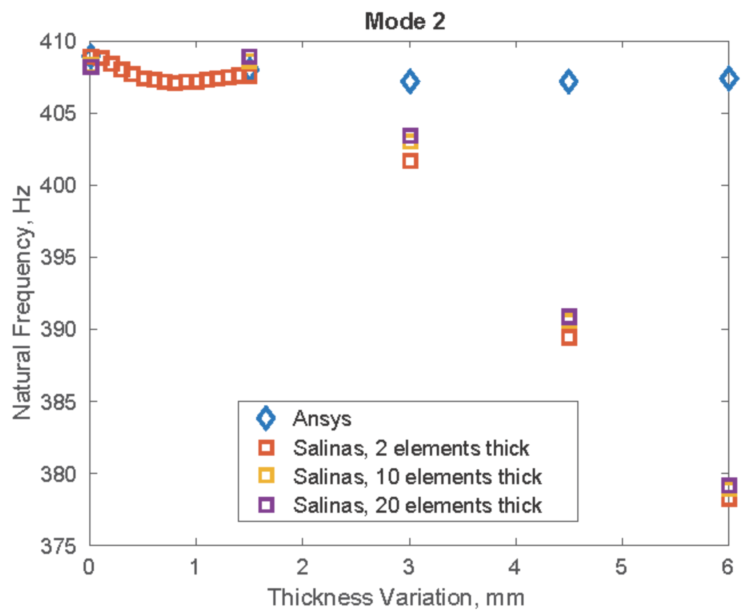
There are significant differences in the results

- Mode 1 and Mode 3 match well at 0 mm perturbation
  - ~~Other modes do not match as well even for 0mm~~ (Fixed)
- For these results, the plates are 2 elements thick
  - ~~Refining the mesh shifts the behavior, but does not drive Salinas results towards ANSYS results~~
- The behavior as the thickness is increased is different
  - ANSYS seems to be almost linear in most cases
  - Salinas exhibits non-linear behavior
  - Are thicknesses for ANSYS results correct?
- The element type choice in Salinas has a large impact
  - The chosen type is closer to a commercial code than the default (according to the Salinas documentation)
- Non-linear behavior suggests that HD PROMs constructed from 0mm case will not be very accurate for large perturbations

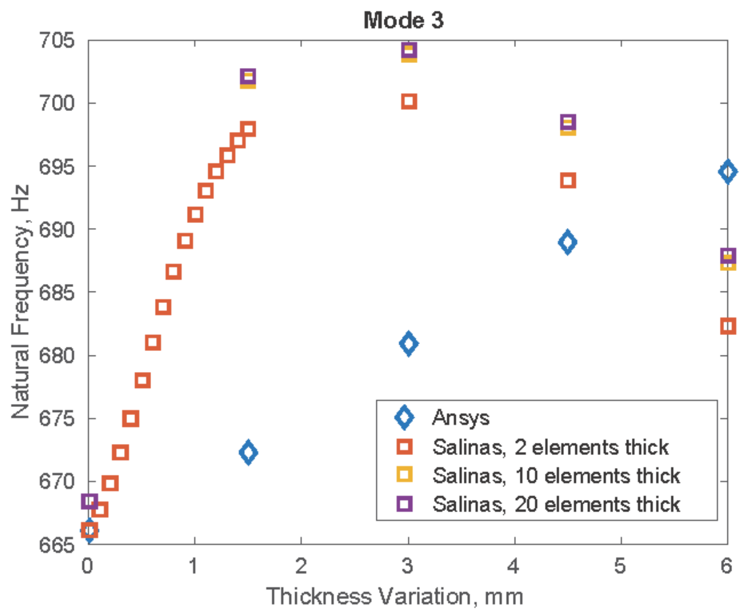
Mesh Convergence Study, Mode 1



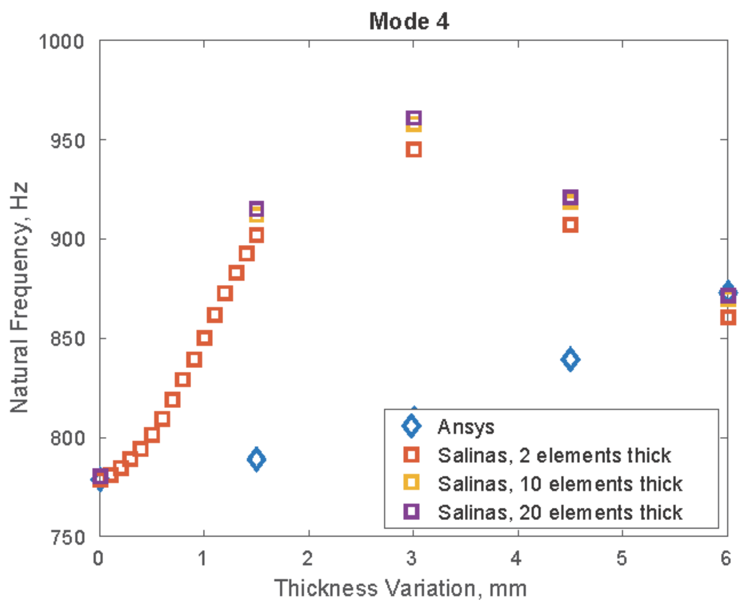
Mesh Convergence Study, Mode 2



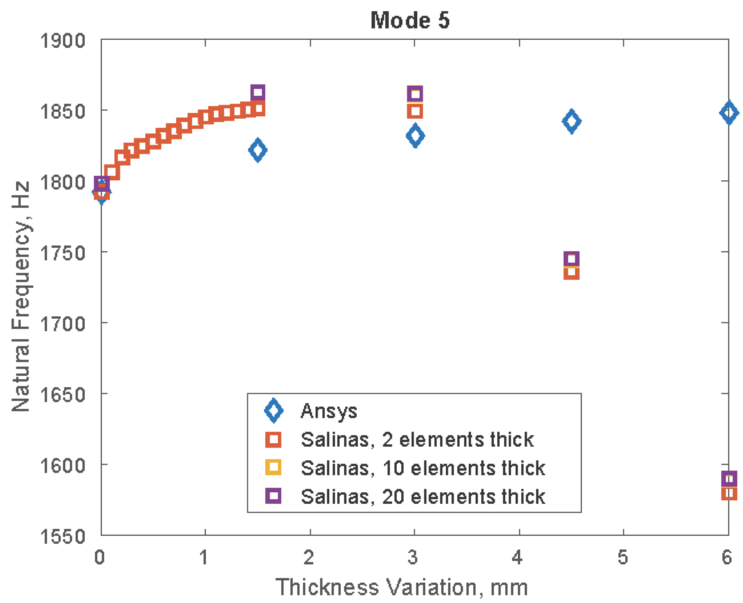
Mesh Convergence Study, Mode 3



Mesh Convergence Study, Mode 4



Mesh Convergence Study, Mode 5

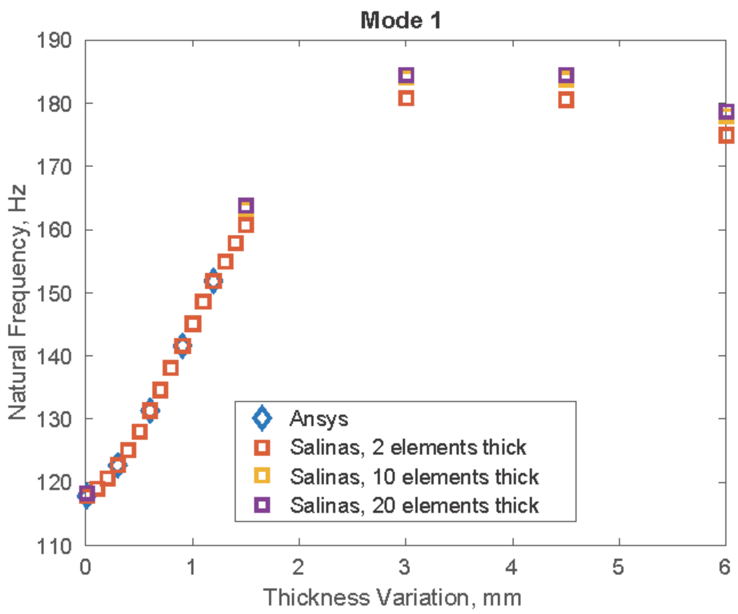


Discussion

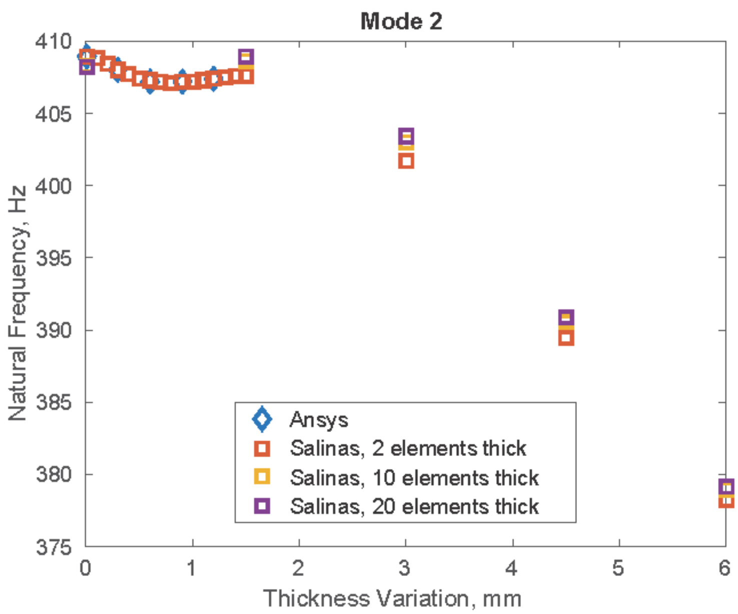
There are significant differences in the results

- Mode 1 and Mode 3 match well at 0 mm perturbation
  - ~~Other modes do not match as well even for 0mm~~ (Fixed)
- For these results, the plates are 2 elements thick
  - ~~Refining the mesh shifts the behavior, but does not drive Salinas results towards ANSYS results~~
- The behavior as the thickness is increased is different
  - ANSYS seems to be almost linear in most cases
  - Salinas exhibits non-linear behavior
  - Are thicknesses for ANSYS results correct?
- The element type choice in Salinas has a large impact
  - The chosen type is closer to a commercial code than the default (according to the Salinas documentation)
- Non-linear behavior suggests that HD PROMs constructed from 0mm case will not be very accurate for large perturbations

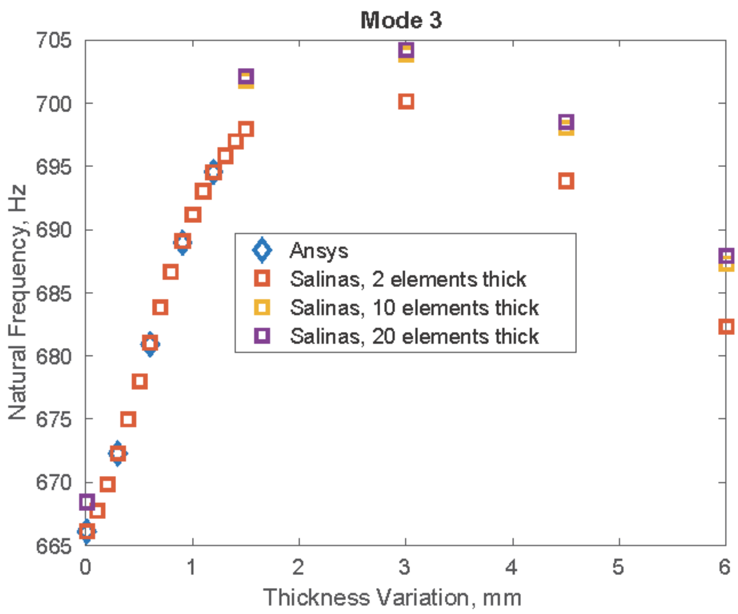
Change Thickness for ANSYS Results, Mode 1



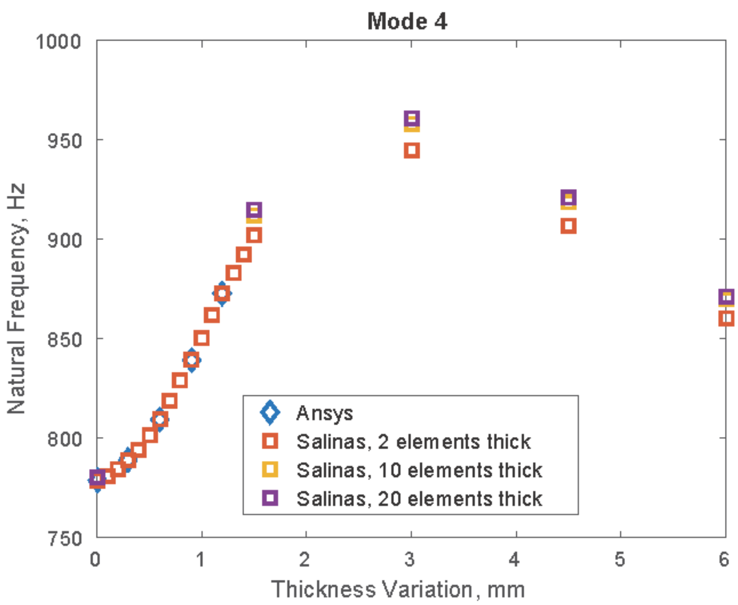
Change Thickness for ANSYS Results, Mode 2



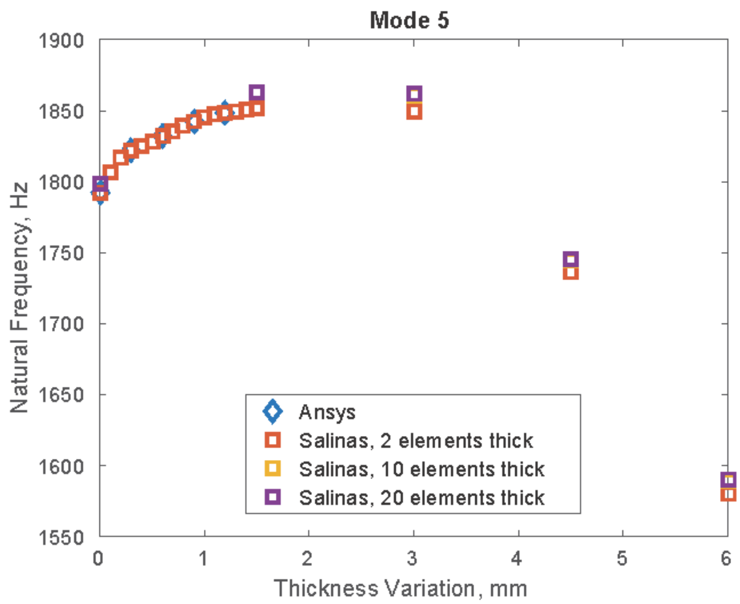
Change Thickness for ANSYS Results, Mode 3



Change Thickness for ANSYS Results, Mode 4



Change Thickness for ANSYS Results, Mode 5



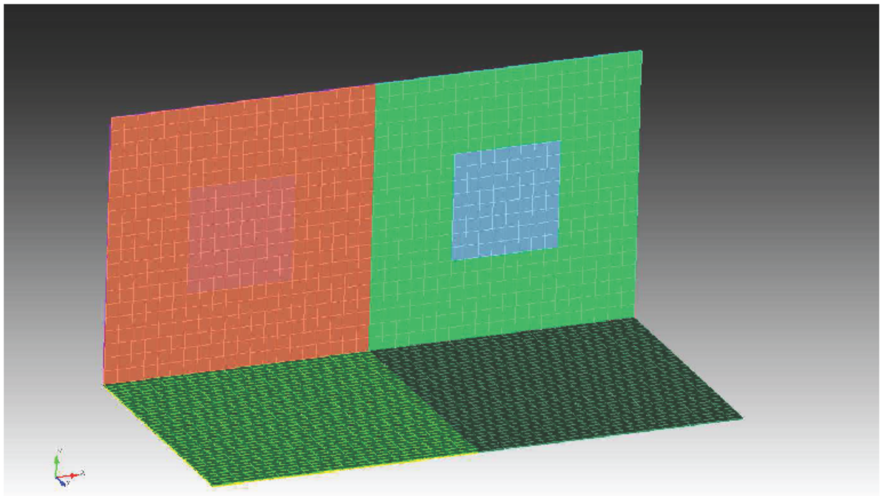
Unlimited Release

Discussion

There are significant differences in the results

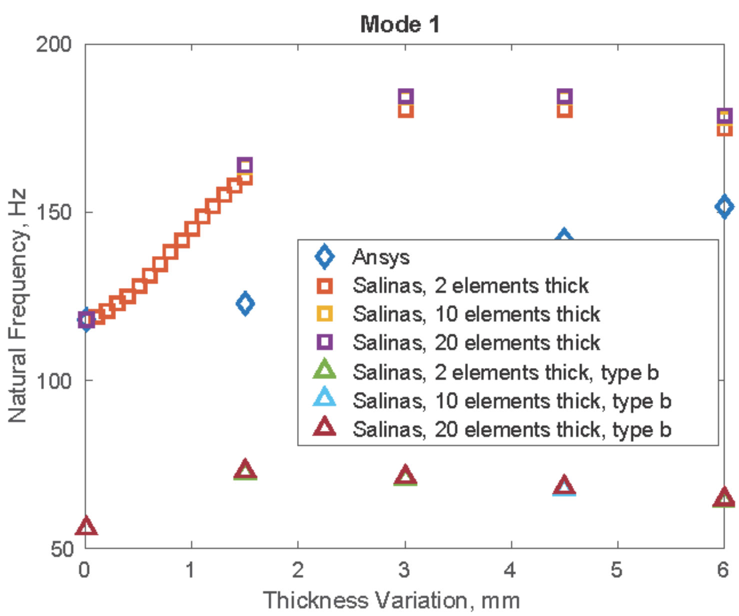
- Mode 1 and Mode 3 match well at 0 mm perturbation
  - ~~Other modes do not match as well even for 0mm~~ (Fixed)
- For these results, the plates are 2 elements thick
  - ~~Refining the mesh shifts the behavior, but does not drive Salinas results towards ANSYS results~~
- The behavior as the thickness is increased is different
  - ANSYS seems to be almost linear in most cases
  - Salinas exhibits non-linear behavior
  - Are thicknesses for ANSYS results correct?
- The element type choice in Salinas has a large impact
  - The chosen type is closer to a commercial code than the default (according to the Salinas documentation)
- Non-linear behavior suggests that HD PROMs constructed from 0mm case will not be very accurate for large perturbations

Additional Salinas Runs

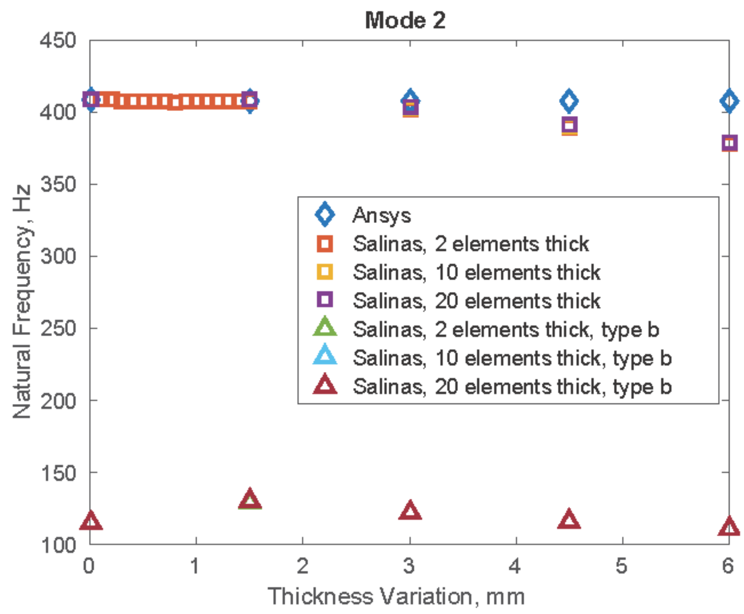


- Original elements were, 3mm x 3mm. ~~Refine to 1mm x 1mm~~
- Try using default hex element type

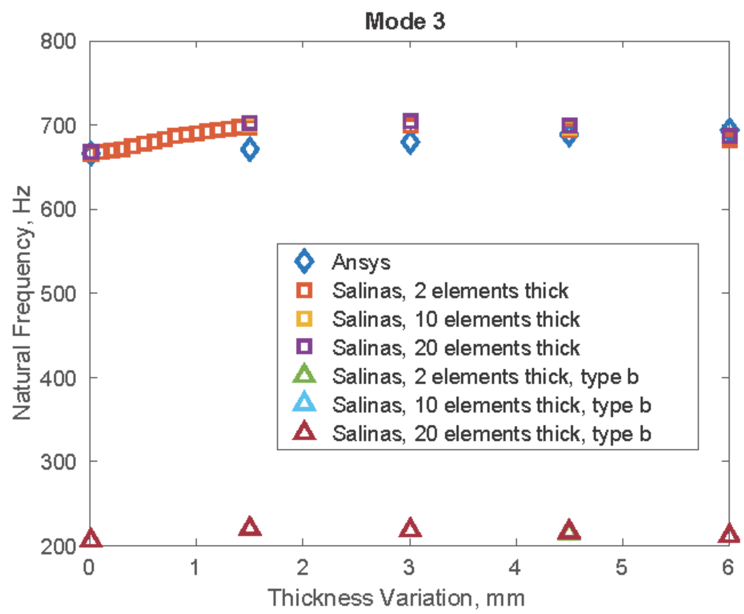
Additional Salinas Runs, Mode 1



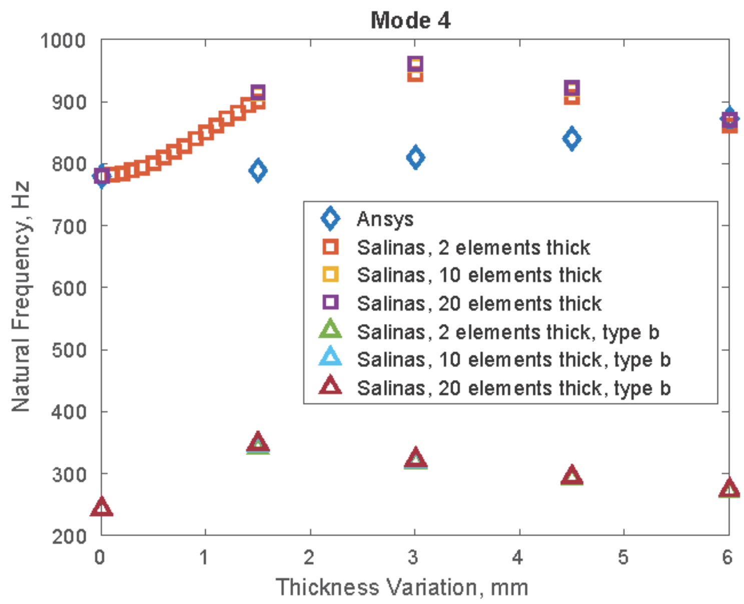
Additional Salinas Runs, Mode 2



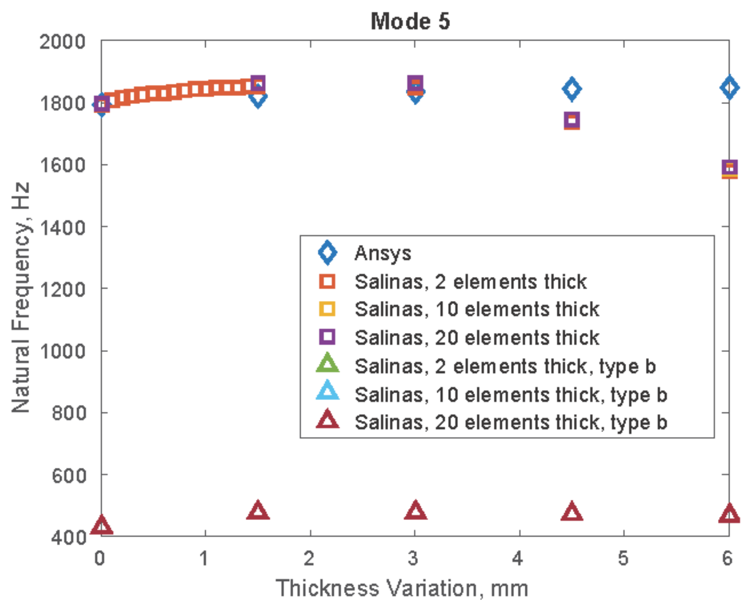
Additional Salinas Runs, Mode 3



Additional Salinas Runs, Mode 4



Additional Salinas Runs, Mode 5



Discussion

There are significant differences in the results

- Mode 1 and Mode 3 match well at 0 mm perturbation
  - ~~Other modes do not match as well even for 0mm~~ (Fixed)
- For these results, the plates are 2 elements thick
  - ~~Refining the mesh shifts the behavior, but does not drive Salinas results towards ANSYS results~~
- The behavior as the thickness is increased is different
  - ANSYS seems to be almost linear in most cases
  - Salinas exhibits non-linear behavior
  - Are thicknesses for ANSYS results correct?
- The element type choice in Salinas has a large impact
  - The chosen type is closer to a commercial code than the default (according to the Salinas documentation)
- Non-linear behavior suggests that HD PROMs constructed from 0mm case will not be very accurate for large perturbations

Outline

Parameterized Reduced-Order Models Using Hyper-Dual Numbers

PROM Round Robin Test Case

PROM Comparison

PROM Comparison

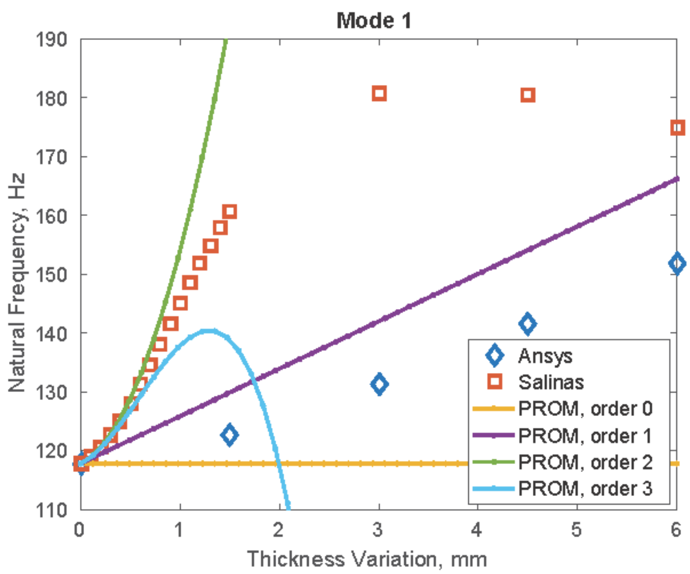
Two PROM Comparisons:

- PROM constructed using information from 0mm case
  - Not expected to be accurate for entire range of variation due to non-linear behavior
  - Accurate only for small variations
- PROM constructed using information from 3mm case
  - Captures behavior better for larger variations

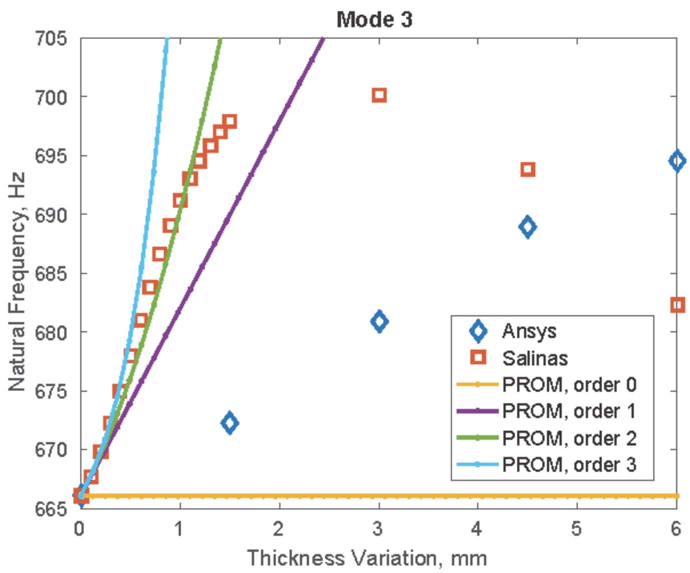
Vary order of parameterization from 0 (constant) to 3 (cubic)

Focus on Mode 1 and Mode 3, which matched better with ANSYS results

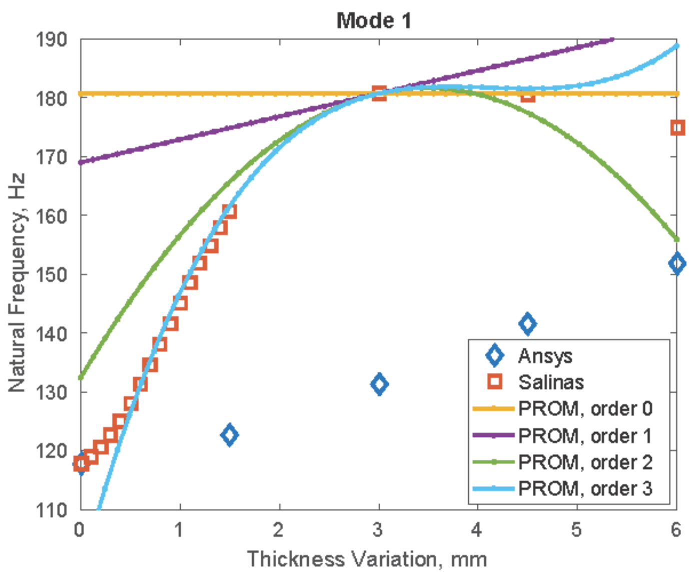
PROM Constructed from 0mm Case



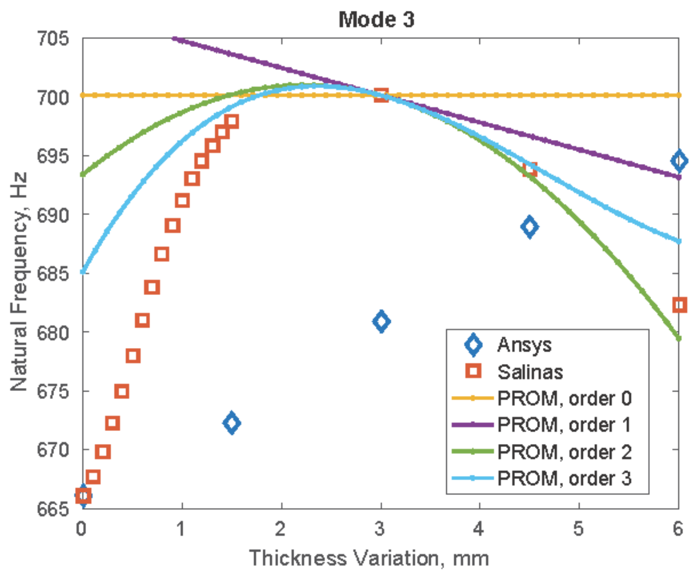
PROM Constructed from 0mm Case



PROM Constructed from 3mm Case



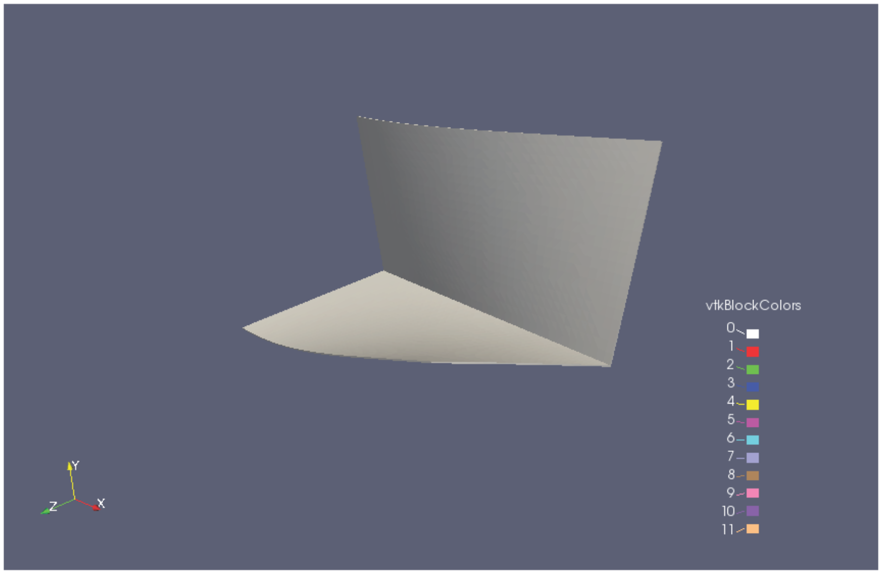
PROM Constructed from 3mm Case



Questions?

Unlimited Release

Mode Shapes: Mode 1



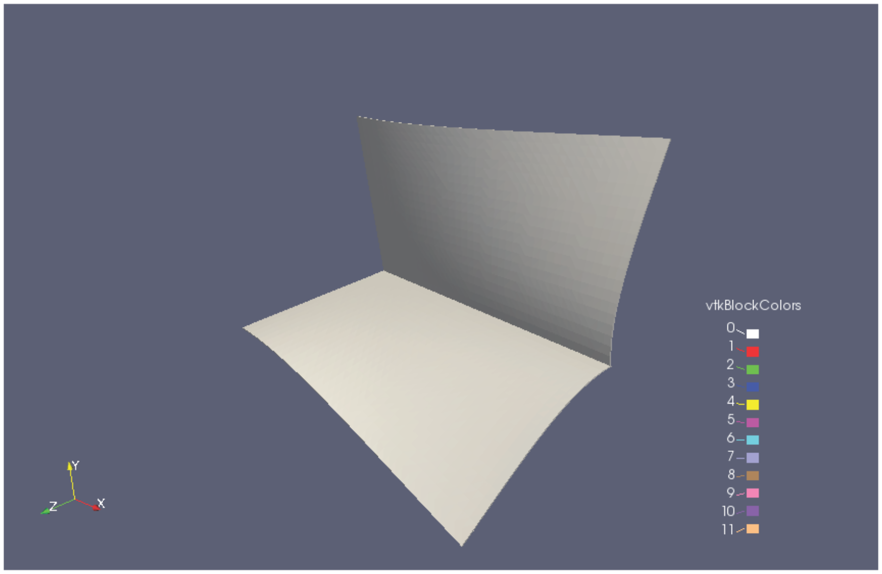
June 3, 2016

43

Unlimited Release

Unlimited Release

Mode Shapes: Mode 2

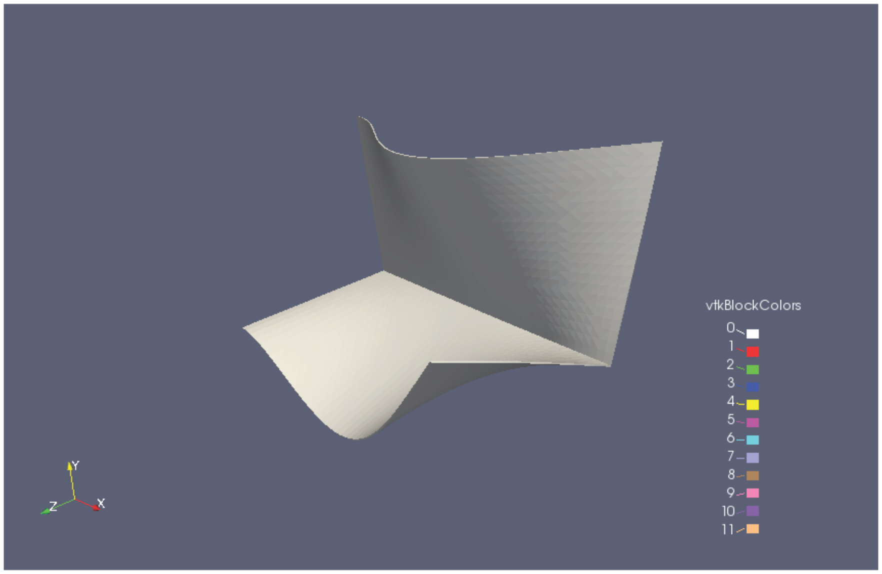


June 3, 2016

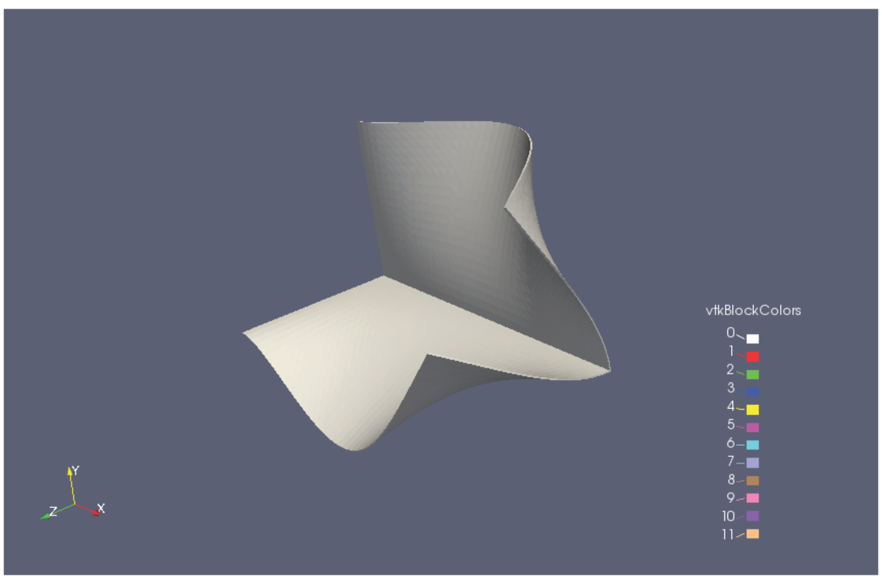
44

Unlimited Release

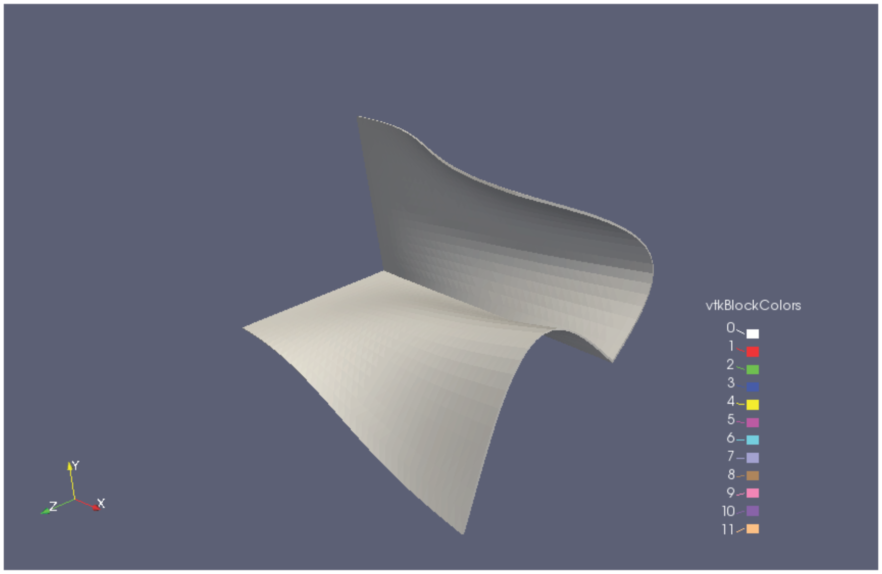
Mode Shapes: Mode 3



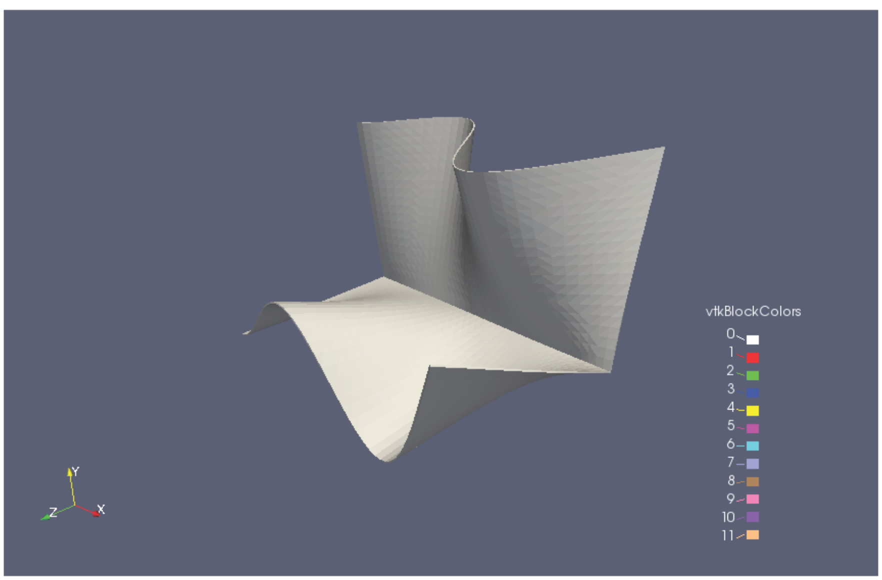
Mode Shapes: Mode 4



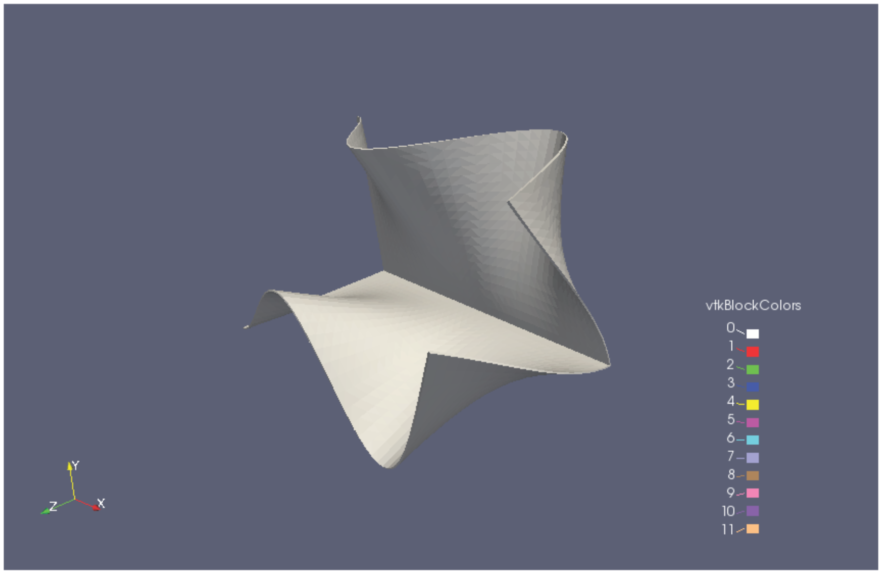
Mode Shapes: Mode 5



Mode Shapes: Mode 6

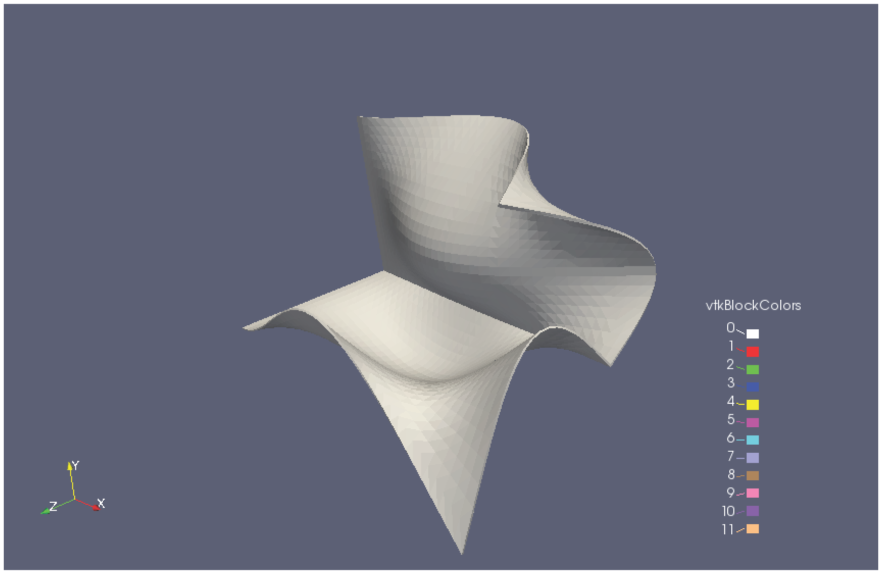


Mode Shapes: Mode 7

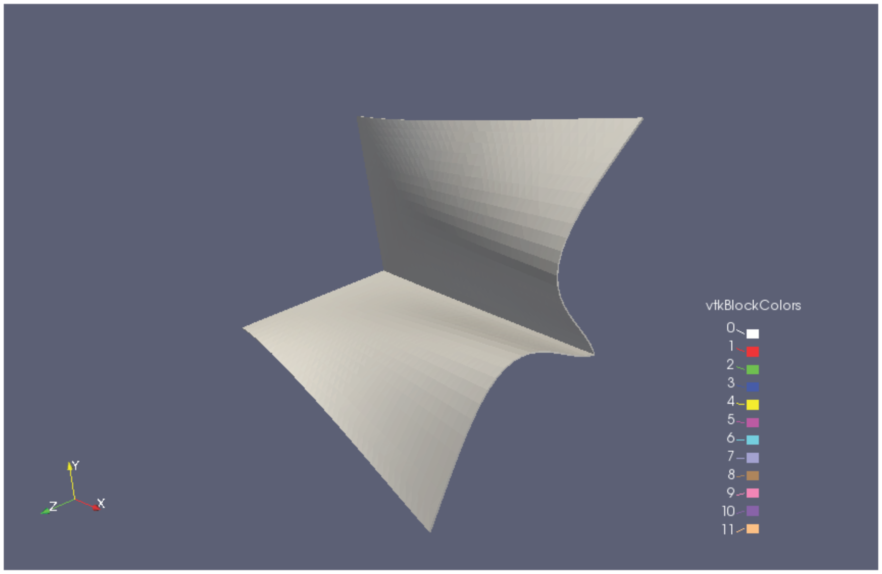


Unlimited Release

Mode Shapes: Mode 8

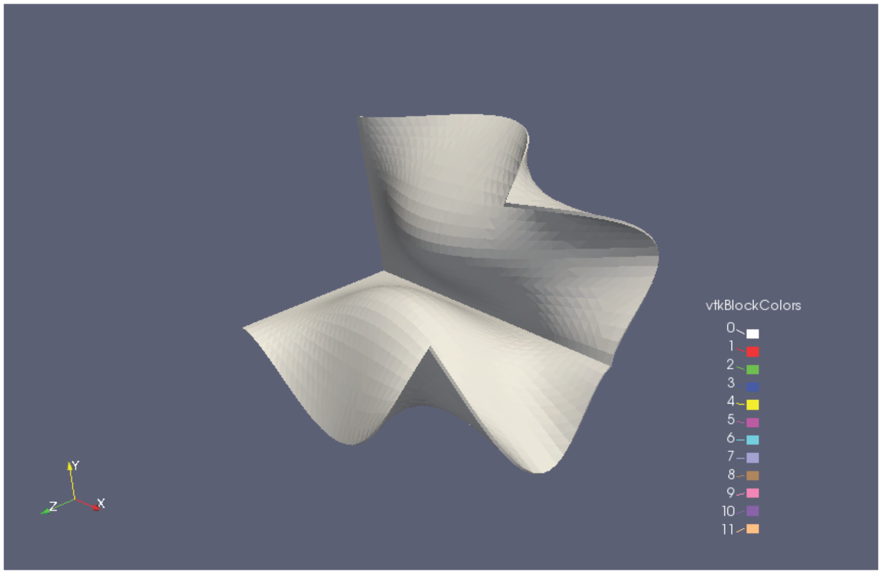


Mode Shapes: Mode 9

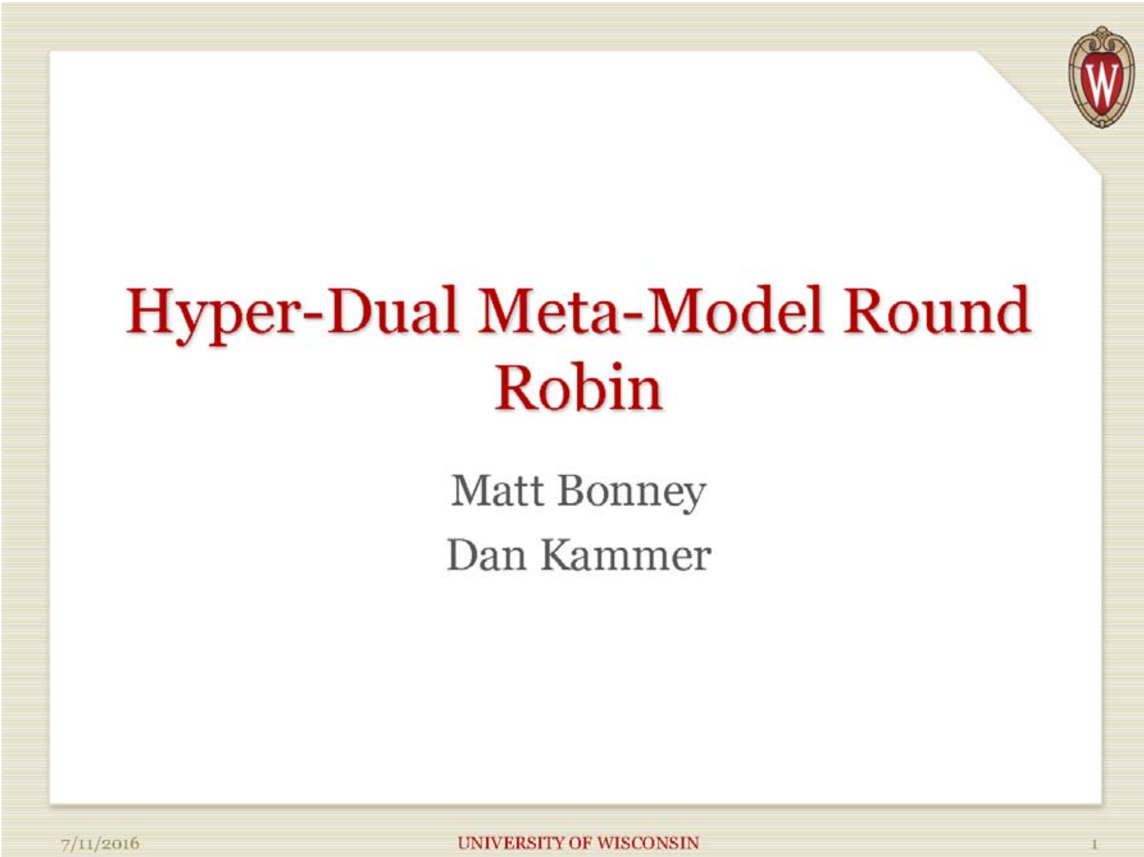


Unlimited Release

Mode Shapes: Mode 10



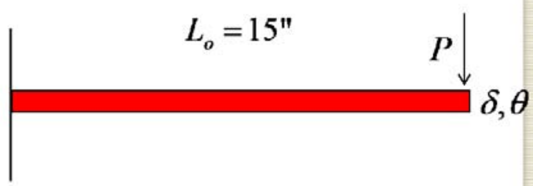
3.3.3 Meta-Modeling Round Robin and Tutorial, Matthew Bonney



Hyper-Dual Meta-Model (HDM) Review

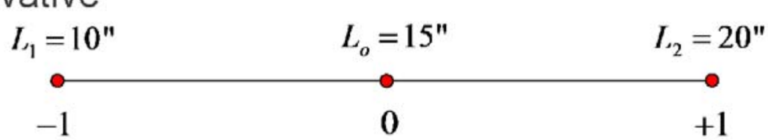
- Combines the accuracy of the Hyper-Dual step and the range of finite difference
- Perform multiple Hyper-Dual model evaluations
  - Use basis function to characterize output and sensitivities at each evaluation point
  - Most effective if evaluation points at or near the extremes of the parameter values
  - Requires least amount of code modification if performed at system level

Simple Example



- Deflection of cantilever tip due to applied force
- Parameterize stiffness matrix in terms of beam length *L*, very nonlinear
- Evaluate stiffness at 3 meshes with first derivative

$$EI\begin{bmatrix} 12/L^3 & 6/L^2 \\ 6/L^2 & 4/L \end{bmatrix}\begin{Bmatrix} \delta \\ \theta \end{Bmatrix} = \begin{Bmatrix} P \\ 0 \end{Bmatrix}$$



Simple Example Cont.

- Fit polynomial to stiffness matrix in terms of dimensionless length, γ
- Match values and first derivative at each evaluation
- Solve for unknown matrix coefficients

$$K = K_0 + K_1\gamma + K_2\gamma^2 + K_3\gamma^3 + K_4\gamma^4 + K_5\gamma^5$$

$$\frac{dK}{d\gamma} = K_1 + 2K_2\gamma + 3K_3\gamma^2 + 4K_4\gamma^3 + 5K_5\gamma^4$$

$$\begin{bmatrix} I & -I & I & -I & I & -I \\ I & 0 & 0 & 0 & 0 & 0 \\ I & I & I & I & I & I \\ 0 & I & -2I & 3I & -4I & 5I \\ 0 & I & 0 & 0 & 0 & 0 \\ 0 & I & 2I & 3I & 4I & 5I \end{bmatrix} \begin{bmatrix} K_0 \\ K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \end{bmatrix} = \begin{bmatrix} K(L_1) \\ K(L_o) \\ K(L_2) \\ K'(L_1) \\ K'(L_o) \\ K'(L_2) \end{bmatrix}$$

7/11/2016 UNIVERSITY OF WISCONSIN 4

Hyper-Dual Implementation

- Generated HD step in SIERA
- Collect data from multiple points
  - Extremes of interval and possibly mid-points
  - Up to engineering intuition
- Use first and second derivatives
- Can parameterize at system or output level
  - Changes accuracy and computational requirements for HDM generation
- Perform post-processing in Matlab

7/11/2016 UNIVERSITY OF WISCONSIN 5

Hyper-Dual in Matlab

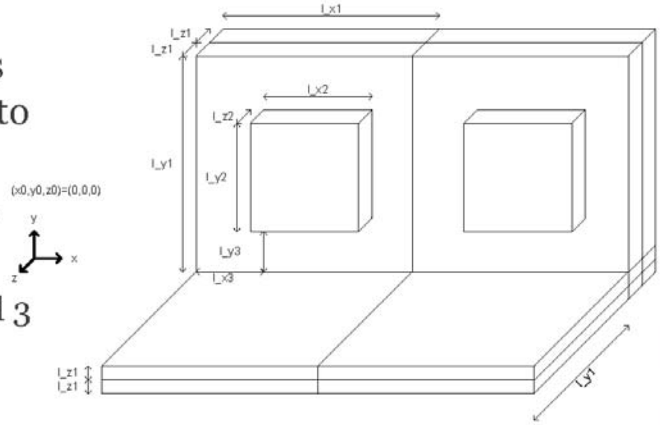
- Currently object programming on term-by-term basis
  - Requires less memory for processing compared to matrix implementation
- Requires specific order to be programmed individually
- Currently 2^{nd} and 3^{rd} derivatives available
- Must redefine basic operations (+-*/)
- More advance functions can be implemented in 2 ways, via algorithm or analytically

Hyper-Dual Functions

- Create folder listed as @hyperdual2 and add to Matlab path
  - Within folder contains all functions
  - Basic functions (+,-,*,/,^)
  - Logical function (==,~=,>,<)
  - Ease of use functions (display, sort)
  - More advanced functions
    - Eig, sqrt, inv, norm, eye, zeros, diag

Round Robin System

- Are you tired of this system yet?
- Look at thickness variation from 0 to 6 mm
- Polynomial basis function
  - 2 data points and 3 data points
- Issues with overfitting the data

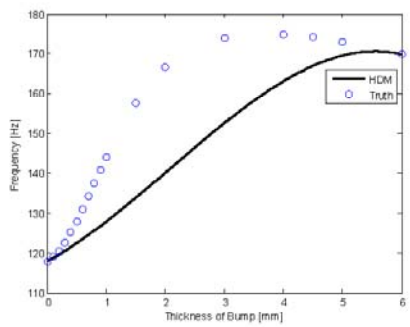


Simple Option

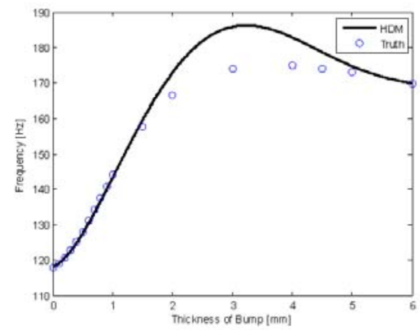
- 2 Data points
  - Match output and first derivative
    - 3rd Order polynomial
  - Match output and first two derivatives
    - 5th Order polynomial
- Uses zero thickness and 6mm thickness from Sierra
  - Compare to multiple Sierra “real” simulations
- Parameterized at Eigen frequency level

Results – 1st Natural Frequency

Match First Derivatives

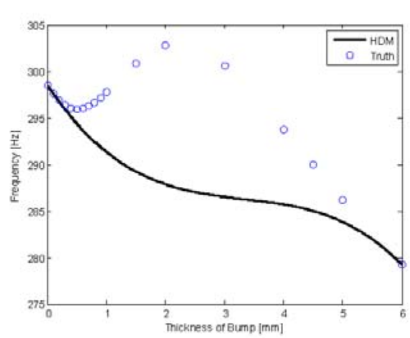


Match First & Second Derivative

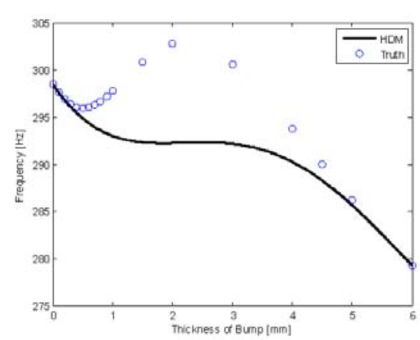


Results – 2nd Natural Frequency

Match First Derivatives

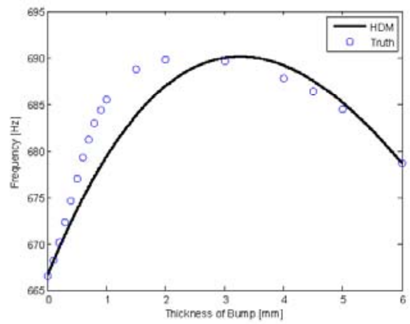


Match First & Second Derivative

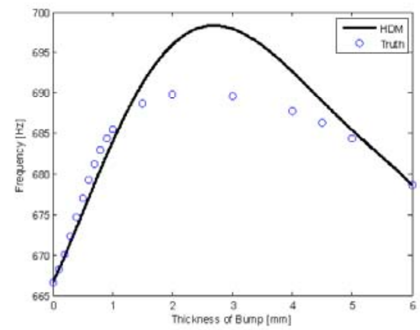


Results – 3rd Natural Frequency

Match First Derivatives



Match First & Second Derivative

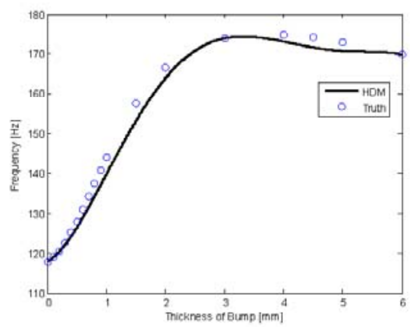


Collect More Data

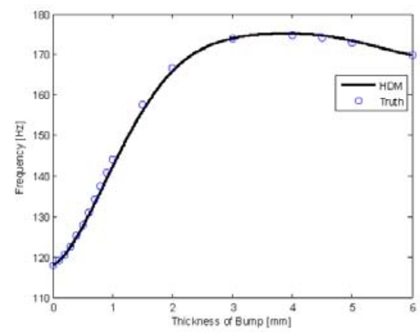
- Use 3 data points
  - Match 1st derivatives
    - 5th Order Polynomial
  - Match first two derivatives
    - 8th Order Polynomial
    - Over fits the data if extrapolating
- Uses zero, 3mm, and 6mm thickness from Sierra
  - Compare to multiple Sierra "real" simulations
- Parameterized at Eigen frequency level

Results – 1st Natural Frequency

Match First Derivatives

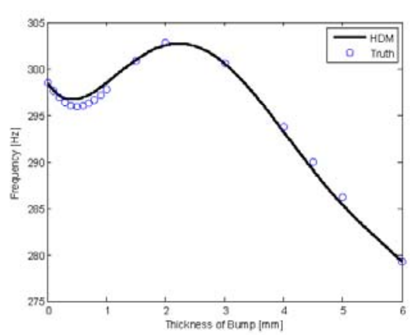


Match First & Second Derivative

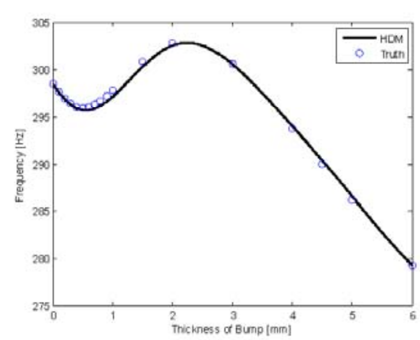


Results – 2nd Natural Frequency

Match First Derivatives

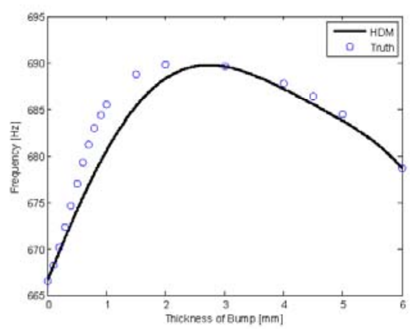


Match First & Second Derivative

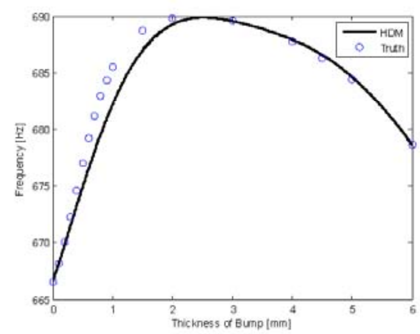


Results – 3rd Natural Frequency

Match First Derivatives



Match First & Second Derivative



Implementation Problems

- Unit matching
  - Derivatives can be given as non-dimensional or dimensional
  - HDM uses new non-dimensional parameter γ
- 2 steps
  - Model formulation
  - Model usage
- Without other truth data, easy to over fit the response
- Model formulation can be very computationally expensive for larger problems

3.3.4 Library for Multi-Complex and Multi-Dual Numbers, Andres Aguirre

A Python library for operating multicomplex and multidual numbers

Andrés M. Aguirre
Manuel J. Garcia
Harry R. Millwater

Parametrized Reduced Order Modeling Workshop

June 2 & 3, 2016

UTSA

University of Texas at San Antonio
Universidad EAFIT

UNIVERSIDAD EAFIT

Multicomplex and Hyperduals

We are going to use the notation "multiduals", because multicomplex and hyperduals share a lot of interesting properties:

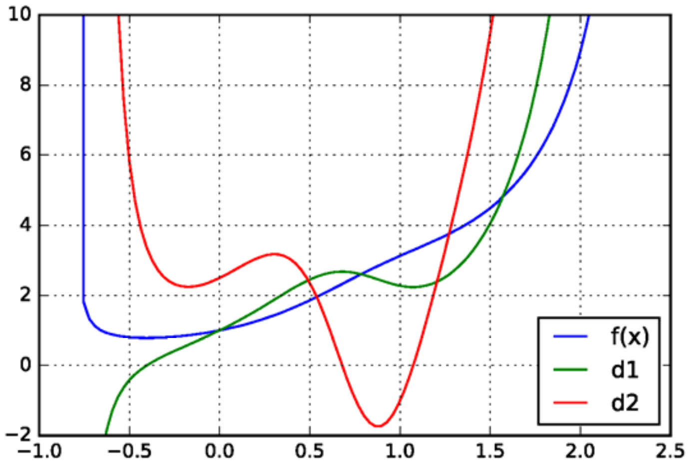
- Numbers with multiple imaginary parts.
- Belong to the superset of hypercomplex numbers (which also includes complex, dual, quaternions, etc).
- Useful for computation of high order derivatives (machine precision).

UTSA   UNIVERSIDAD EAFIT   2/26

Derivatives test

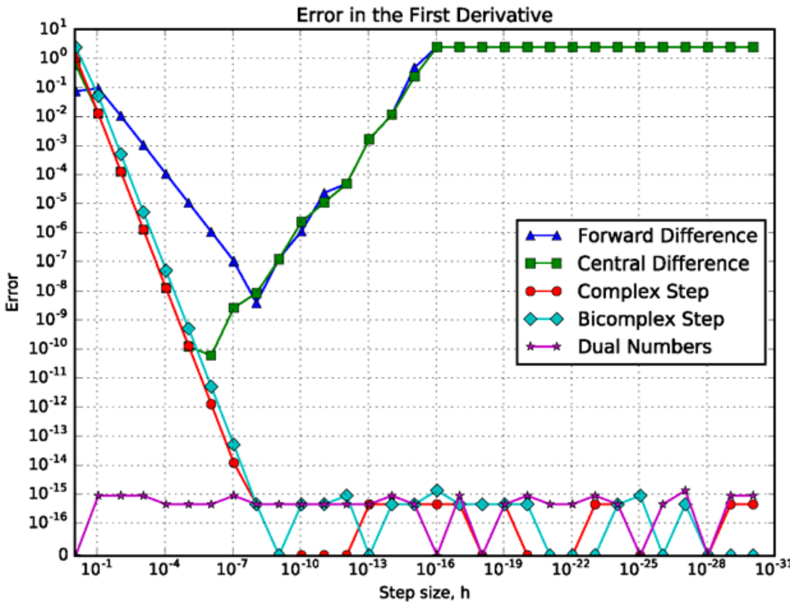
Using the example function from Fike and Alonso, 2011:

$$f(x) = \frac{e^x}{\sqrt{\sin^3(x) + \cos^3(x)}}$$



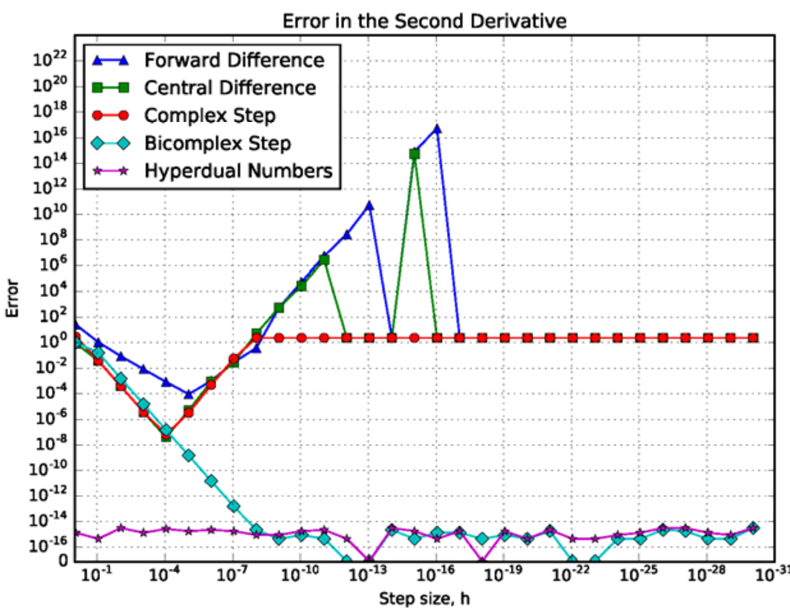
UTSA   UNIVERSIDAD EAFIT   3/26

First order derivative test



UTSA. UNIVERSIDAD EAFIT 4/26

Second order derivative test



UTSA. UNIVERSIDAD EAFIT 5/26

Examples of Multicomplex and Multidual

Multicomplex:

$c = c_0 + c_1\, i_1\ , \quad c \in \mathbb{C}_1\ , \quad c_0, c_1 \in \mathbb{R}.$ Complex

$b = b_0 + b_1\, i_1 + b_2\, i_2 + b_3\, i_1 i_2\ , \quad b \in \mathbb{C}_2\ ,$ Bicomplex

$b_0, \ldots b_3 \in \mathbb{R}.$

Multidual:

$d = d_0 + d_1\, \epsilon_1\ , \quad d \in \mathbb{D}_1\ , \quad d_0, d_1 \in \mathbb{R}.$ Dual

$h = h_0 + h_1\, \epsilon_1 + h_2\, \epsilon_2 + h_3\, \epsilon_1 \epsilon_2\ , \quad h \in \mathbb{D}_2\ ,$ Bidual

$h_0, \ldots h_3 \in \mathbb{R}.$

UTSA UNIVERSIDAD EAFIT 6/26

Rules for imaginary units

Multicomplex use a rule based on complex numbers.

$$i_p^2 = -1 \quad \forall p,$$

$$i_p i_q = i_q i_p \quad \forall p \neq q,$$

$$p, q \in \mathbb{N}.$$

Multiduals use one based on dual numbers.

$$\epsilon_p^2 = 0 \quad \forall p,$$

$$\epsilon_p \epsilon_q = \epsilon_q \epsilon_p \quad \forall p \neq q,$$

$$p, q \in \mathbb{N}.$$

UTSA UNIVERSIDAD EAFIT 7/26

Elementary functions for dual numbers

They are based on the Taylor series expansion.

$$f(z) = \sum_{k=0}^{\infty} \frac{f^{(n)}(a)}{n!}(z-a)^n$$

If $z \in \mathbb{D}_1 , \quad z = x_0 + x_1\,\epsilon , \quad a = x_0 ,$

Nilpotent rule!

Then $f(z) = f(x_0) + f'(x_0)\,x_1\,\epsilon + \dfrac{f''(x_0)}{2} x_1^2\,\epsilon^2 + O(\epsilon^3)$

$$f(z) = f(x_0) + f'(x_0)\,x_1\,\epsilon$$

Exact! Not an approximation, but the derivative must be known

Example: sine function. (Yu and Blair, 2013)

$$\sin(z) = \sin(x_0) + \cos(x_0)\,x_1\,\epsilon$$

UTSA UNIVERSIDAD EAFIT

Elementary functions for multiduals

$$f(z) = \sum_{k=0}^{\infty} \frac{f^{(n)}(a)}{n!}(z-a)^n$$

Also based on the Taylor series expansion.

If $z \in \mathbb{D}_2 , \quad z = x_0 + x_1\,\epsilon_1 + x_2\,\epsilon_2 + x_3\,\epsilon_1\epsilon_2 , \quad a = x_0 ,$

Then $f(z) = f(x_0) + f'(x_0)\,x_1\,\epsilon_1 + f'(x_0)\,x_2\,\epsilon_2 + [f'(x_0)\,x_3 \; + f''(x_0)\,x_1 x_2]\,\epsilon_1\epsilon_2$

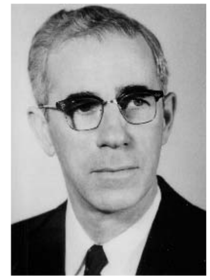
Example: sine function (Fike and Alonso, 2011).

$$\sin(z) = \sin(x_0) + \cos(x_0)\,x_1\,\epsilon_1 + \cos(x_0)\,x_2\,\epsilon_2 + [\cos(x_0)\,x_3 \; - \sin(x_0)\,x_1 x_2]\,\epsilon_1\epsilon_2$$

UTSA UNIVERSIDAD EAFIT

Elementary functions for multicomplex

- In this case Taylor series approach is not exact.
- However, multicomplex can be represented by real matrices called Cauchy Riemann matrices.
- They are related through algebra isomorphism (G. Baley Price, 1991).



Procedure: (G. Lantoine, 2010)

1. Convert multicomplex to matrix form.
2. Apply matrix function.
3. Convert matrix result to multicomplex.

UTSA UNIVERSIDAD EAFIT 10/26

4. PLENARY DISCUSSION THEMES

Following the discussions from the four sessions of presentations, 12 topic areas for further discussion were identified:

1. Terminology (i.e. multi-dual versus hyper dual);
2. Multivariate expansions;
3. Computer memory related issues for calculating PROMs;
4. Polynomial expansion functions for the PROM formulations;
5. “Killer” applications;
6. Moving nodes or changing the number of nodes within a high fidelity model;
7. Mixed sets of elements within a high fidelity model;
8. Element independent formulations;
9. Selection of model points for the PROM formulations;
10. How do we know the valid extent of parameterization?
11. Is it possible to reduce or remove terms in the multivariate expansions?
12. Nonlinear materials/models – how can we cross ideas from structural dynamics to solid mechanics to materials?

Several other questions included:

- What problems will break the dual/complex calculations? (None found so far)
- Is mesh refinement necessary for higher order derivatives? For third order? For fifth order? (There is some evidence that this may be the case)
- Does the order of the element matter to the formulations?

4.1 Terminology

During discussion of the PROM methodologies, it became clear that there is a need for consensus in terms of nomenclature. Two different sets of terminology were used to refer to the same concept: **hyper dual numbers** and **multi dual numbers**. At issue is that **hyper complex numbers** is a previously defined term that includes each family of generalized complex numbers (complex, double, dual, etc.). Thus, the use of hyper dual could be confusing. However, hyper dual is currently used by a number of researchers, whereas the suggested alternative, multi dual, is a new term that has not been adopted outside of a limited number of research groups. No consensus on a path forward has been reached, but due to the wide use of hyper dual to mean higher order dual numbers, it is most likely that this terminology will persist. The use of multi dual is acceptable provided that it is accompanied by a statement such as “also known as hyper dual numbers [9].” In order to help alleviate confusion in the future, Jeffrey Fike and Andres Aguirre have been tasked with creating a Wikipedia article on the subject.

4.2 Multivariate Expansions

One of the greatest challenges for extending the use and applicability of PROMs is developing efficient, multivariate expansions. As the number of parameterized variables increase, the number of terms in the expansion significantly increases. Thus, an open question is “How should the increase of terms for multivariate, higher order expansions be managed?” Related to the eleventh theme, one suggestion was the development of an algorithm to assess the necessary terms (i.e. calculating the sensitivities of the derivatives is automatic as higher order derivatives is taken, once a sensitivity goes to zero, no other terms are needed in that branch of derivatives).

Central to this, is that multivariate expansions convolute two separate problems: parameterizations of variables of interest, and formulations of ROMs. Both problems represent significant research challenges that will require significant innovation for advancement.

4.3 Nonlinear Models

For nonlinear structures, there is no clear definition of mode shapes or superposition. This results in a challenge for PROM formulations as they are based on modal reductions from structural dynamics. Thus, the difficulty of defining a PROM from a local calculation is that the extent of validity is expected to be too small to be useful (e.g. consider using nonlinear strain information calculated about one design point for predicting how geometric changes might affect a system). Consequently, the meta-modeling approach seems attractive for studying nonlinear systems as this approach is able to capture global influences of parameters instead of just local.

4.4 Meta-Modeling

With the attractiveness of meta-modeling for extending the validity of PROMs and addressing concerns raised from the nonlinear modeling standpoint, several questions arose:

- How can it be determined when a new design point needs to be included in the meta-model expansion?
- How should the design points for the meta-model expansion be chosen? (e.g. Gaussian points, stochastic reduced order models, etc.)
- How should new design points be incorporated?
- How should the parameterization be optimally constructed? (e.g. splines)

With regards to this last question, some insight comes from the NX-PROM work: a more accurate parameterization is able to be achieved by using an element-dependent expansion function. Thus, the optimal parameterization may depend on physical information and the finite difference formulation.

4.5 Microstructure Parameterization

One area that is promising for extensions of parameterized modeling is developing methodologies for representing microstructures. With the maturation of multi-scale modeling approaches, such as highlighted in the talks given by G. Castelluccio and J. Brown, can the meta-modeling approach or other PROM concepts be extended to representative volume elements in order to improve the understanding of the link between material properties and physical processes? Another way to view this question is: “Are there intuition based material modeling approaches that can be replaced with a rigorous approach?” An example might be knowing both information about a granular structure and some uncertainty quantification for it, what is the optimal macroscale material model – isotropic, anisotropic, etc.? One particular area of applicability might be the optimization of composites.

4.6 Multidisciplinary Collaborations

In the workshop, there were two distinct populations of researchers: solid mechanics and structural dynamics. There needs to be a greater level of collaboration between these two communities as they are closely related. The concepts of ROMs in structural dynamics and multi-scale modeling in solid mechanics are, to some extent, inter-related in terms of ultimate goals and the necessity for mathematical methods to reduce the system equations. One example of an area that is well posed for collaboration is developing reduced order models for materials. That is, developing a methodology to investigate specific material models in structural dynamics ROMs (such as anisotropic, viscoelastic, etc.); this would result in being able to answer questions such as “If a material was welded in one direction versus another, how does that affect the dynamic response?” i.e. how does the grain structure or microstructure affect the dynamic response or system processes?

REFERENCES

[1] S. Hong, B. Epureanu and M. Castanier, "Joining of Components of Complex Structures for Improved Dynamic Response," *Journal of Sound and Vibration,* vol. 331, pp. 4285-4298, 2012.

[2] S. Hong, B. Epureanu and M. Castanier, "Next-Generation Parametric Recuded-Order Models," *Mechanical Systems and Signal Processing,* vol. 37, pp. 403-421, 2013.

[3] D. Kammer and S. Nimityongskul, "Propagation of Uncertainty in Test-Analysis Correlation of Substructured Spacecraft," *Journal of Sound and Vibration,* vol. 330, pp. 1211-1224, 2011.

[4] D. Kammer and D. Krattiger, "Propagation of Uncertainty in Substructured Spacecraft Using Frequency Response," *AIAA Journal,* vol. 51, pp. 353-361, 2013.

[5] F. Momin, H. Millwater, R. Osborn and M. Enright, "A Non-Intrusive Method to Add Finite Element-Based Random Variables to a Probabilisitic Design Code," *Finite Elements in Analysis and Design,* vol. 46, pp. 280-2877, 2010.

[6] A. Voorhees, H. Millwater and R. Bagley, "Complex Variable Methods for Shape Sensitivity of Finite Element Models," *Finite Elements in Analysis and Design,* vol. 47, pp. 1146-1156, 2011.

[7] M. Brake, J. Fike and S. Topping, "Parameterized Reduced Order Models from a Single Mesh Using Hyper-Dual Numbers," *Journal of Sound and Vibration,* vol. 371, pp. 370-392, 2016.

[8] J. A. Brown and J. E. Bishop, "Quantifying the Impact of Material-Model Error on Macroscale Quantities-of-Interest Using Multiscale a Posteriori Error-Estimation Techniques," *MRS Advances,* In Press.

[9] J. Fike and J. Alonso, "The Development of Hyper-Dual Numbers for Exact Second-Derivative Calculations," in *49th AIAA Aerospace Sciences Meeting,* Orlando, FL, 2011.

DISTRIBUTION

1	Andres Aguirre aaguirr2@eafit.edu.co	Universidad EAFIT
1	Arturo Montoya Arturo.Montoya@utsa.edu	University of Texas, San Antonio
1	Bogdan Epureanu epureanu@umich.edu	University of Michigan
1	Brenton Taft brenton.taft@us.af.mil	Air Force Research Laboratories
1	Daniel Kammer daniel.kammer@wisc.edu	University of Wisconsin, Madison
1	Derek Hengeveld dhengeveld@loadpath.com	Air Force Research Laboratories
1	Harry Millwater harry.millwater@utsa.edu	University of Texas, San Antonio
1	Jau-Ching Lu jauching@umich.edu	University of Michigan
1	Kevin Irick kevin.irick.1.cttr@us.af.mil	Air Force Research Laboratories
1	Laura Mainini lmainini@mit.edu	Massachusetts Institute of Technology
1	Manuel Garcia mgarcia@eafit.edu.co	Universidad EAFIT
1	Matthew Castanier matthew.p.castanier.civ@mail.mil	US Army TARDEC

1	MS0557	Benjamin R. Pacini	1522 (electronic copy)
1	MS0828	Org 1544	1544 (electronic copy)
1	MS0840	Group 1550	1550 (electronic copy)
1	MS0845	Org 1542	1542 (electronic copy)
1	MS1064	Brian A. Robins	1515 (electronic copy)
1	MS0899	Technical Library	9536 (electronic copy)