LS-OPT® Overview and Preview of v4.2

Nielen Stander
LSTC, Livermore, CA

LS-OPT Infotag
Ingolstadt, BRD
October 20, 2010

Contents

- LS-OPT Goals
- Main features and Methodology
- Examples
- Job distribution
- Preview of Version 4.2

LS-OPT Goals

- Provide a design environment for LS-DYNA users with the following capabilities and features:
  - Design Improvement and Optimization

Stochastic

- Reliability-Based Optimization – include probability of failure
- Robust Design Optimization – maximize robustness
- Outlier Analysis – identify sources of variation

Current Dev.

- Network-based job scheduling – simplify job distribution
- Process Modeling – allow process flow
- Topology Optimization – conceptual design
- System Identification – calibrate materials/systems

Summary: Optimization

- **Metamodel-based Design Optimization**
  - Strategies
    - *Single Stage:* Fixed computational budget
    - *Sequential:* Maximize metamodel accuracy
    - *Sequential with Domain Reduction:* Converge to optimal region

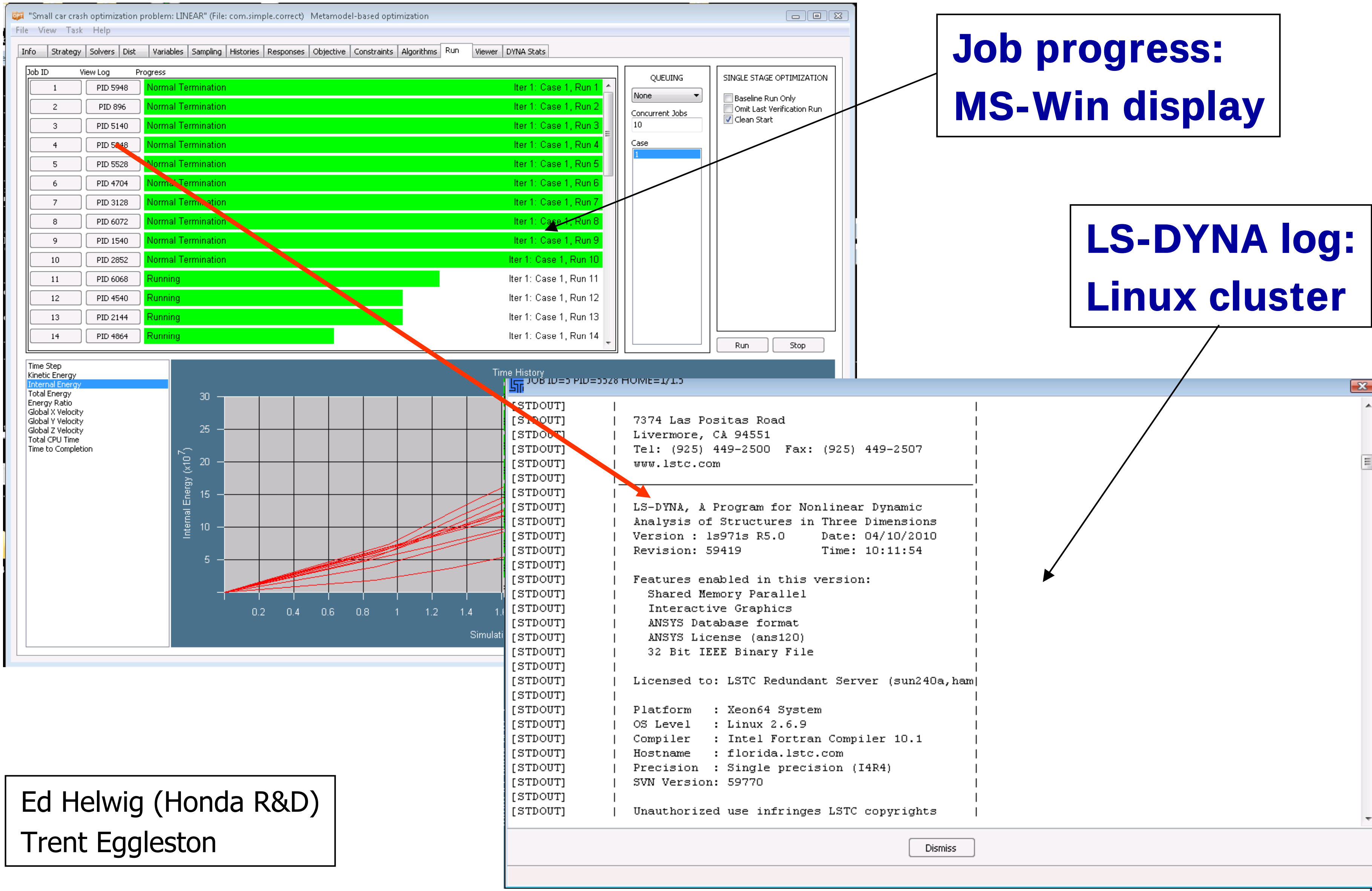
- **Direct Optimization**
  - Genetic Algorithm (GA)
  - Particle Swarm Optimization (PSO) (v4.2)

- **Multi-Objective Optimization (Direct or Metamodel)**
  - NSGA-II (Non-dominated sorting Genetic Algorithm)
  - SPEA-II (Strength Pareto Evolutionary Algorithm)
  - SMPSO (Speed-Constrained Multi-objective Particle Swarm) (v4.2)

LS-DYNA Integration

- Checking of Dyna keyword files (*DATABASE_)
- Importation of design parameters from Dyna keyword files (*PARAMETER_)
- Monitoring of Dyna progress
- Result extraction of most Dyna response types
- LS-DYNA history plots in Viewer
- D3plot compression (node and part selection)
- Outlier information to FE mesh (LS-PrePost display)
- LS-DYNA *CASE supported. Responses can be tied to a particular LS-DYNA Case
- *INCLUDE and *INCLUDE_PATH files automatically parsed, copied and/or transmitted to cluster

Job distribution: LSTCVM Secure job proxy server

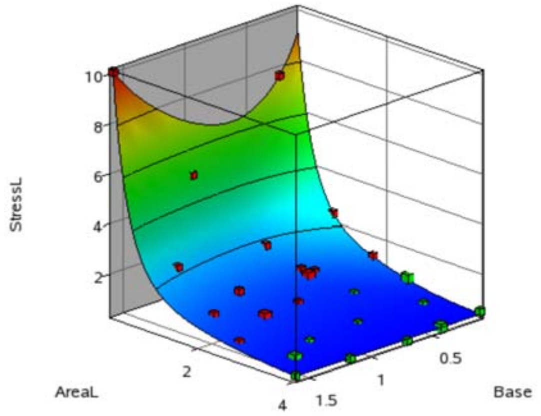


Job progress: MS-Win display

LS-DYNA log: Linux cluster

Ed Helwig (Honda R&D)
Trent Eggleston

Metamodeling

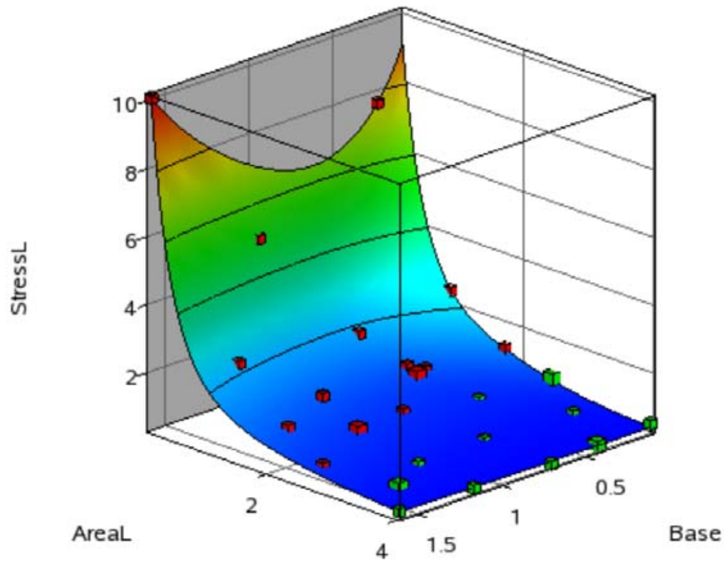


Metamodel: Approximating the design

Metamodeling

What is a metamodel ?

An *approximation* to the design response, usually a simple function of the design variables. Is used instead of actual simulations during design exploration hence also called *surrogate*.



Metamodel Applications

- Variable importance
  - Linear surface fit to produce gradients
  - Global Sensitivity Analysis using Sobol indices
- Optimization
  - Sequential metamodel construction/updating
  - Multiple objectives: Determine *Pareto*-optimal design set
- Reliability and Robustness
  - Probability of failure: Reliability
  - Standard deviation of a response: Robustness
- Outlier Analysis
  - Locate sources of variation and noise

Metamodel Types in LS-OPT

- Response Surface Methodology (RSM)
  - Polynomial-based
  - Typically regional approximation (especially linear)
- Feedforward Neural Networks (FF)
  - Simulation of a biological network, sigmoid basis function
  - Global approximation
- Radial Basis Function Networks (RBF)
  - Gaussian, Multi-quadric basis functions in a linear system
  - Global approximation
- Kriging
- User-defined
  - Dynamically linked (**.so, .dll**)

Sampling (schemes for point selection)

- Space Filling
  - Used with FFNN + RBFN
  - Max. Min. distance between
    - new points
    - new points + fixed points
  - Simulated Annealing

- *D*-Optimality
  - Used with polynomials

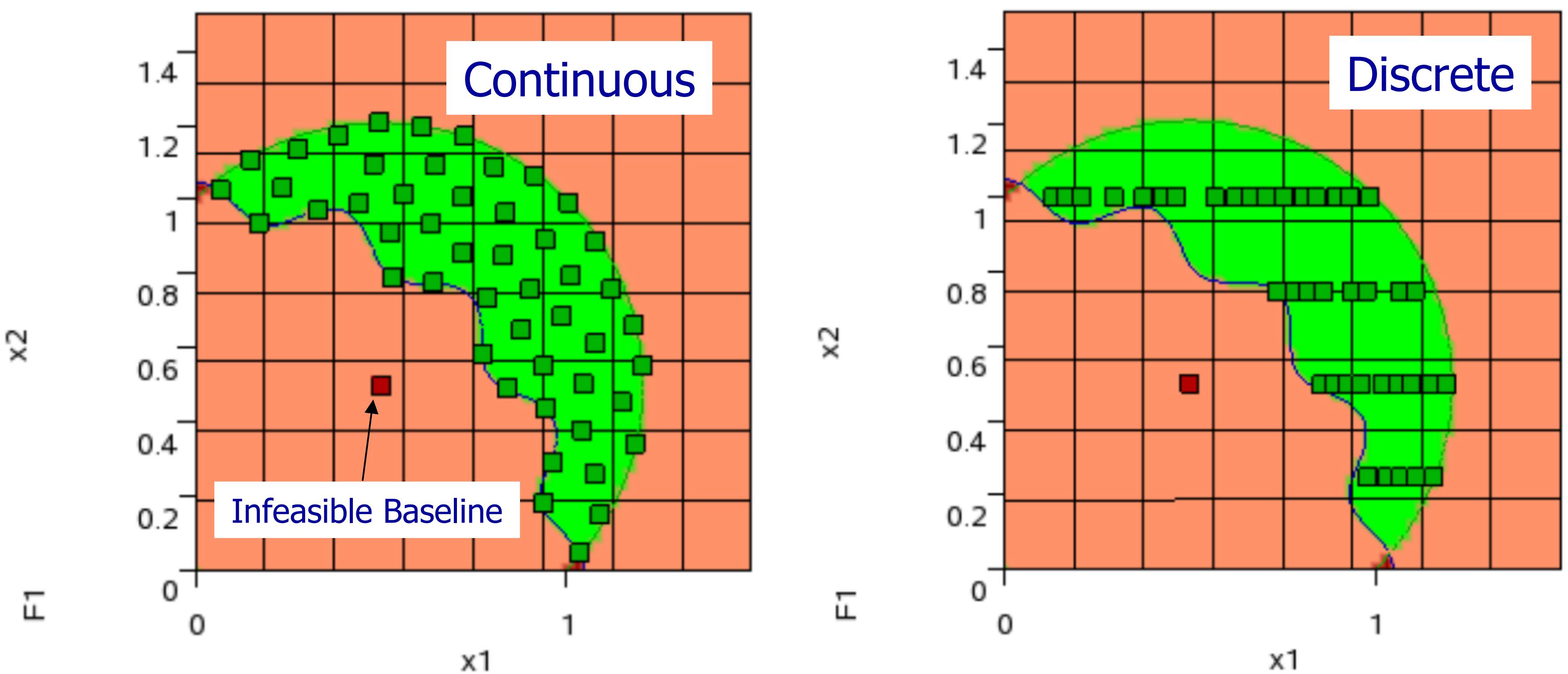
- Other types
  - Full factorial, Koshal, Central Composite, Latin Hypercube, User

Sampling in an irregular design space: Constrained Space Filling (v4.2)

$$\text{Max. Min. } \| \boldsymbol{x}_i - \boldsymbol{x}_j \| \text{ s.t. } g_j \le 0;\ j=1,\ldots,m$$

- TNK Example

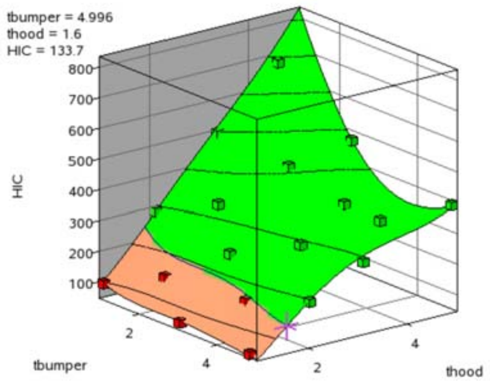
Constraints: $x_1^2 + x_2^2 - 1 - 0.1\cos(16\tan^{-1}(x_1/x_2)) \ge 0$; $(x_1 - 0.5)^2 + (x_2 - 0.5)^2 \le 0.5$



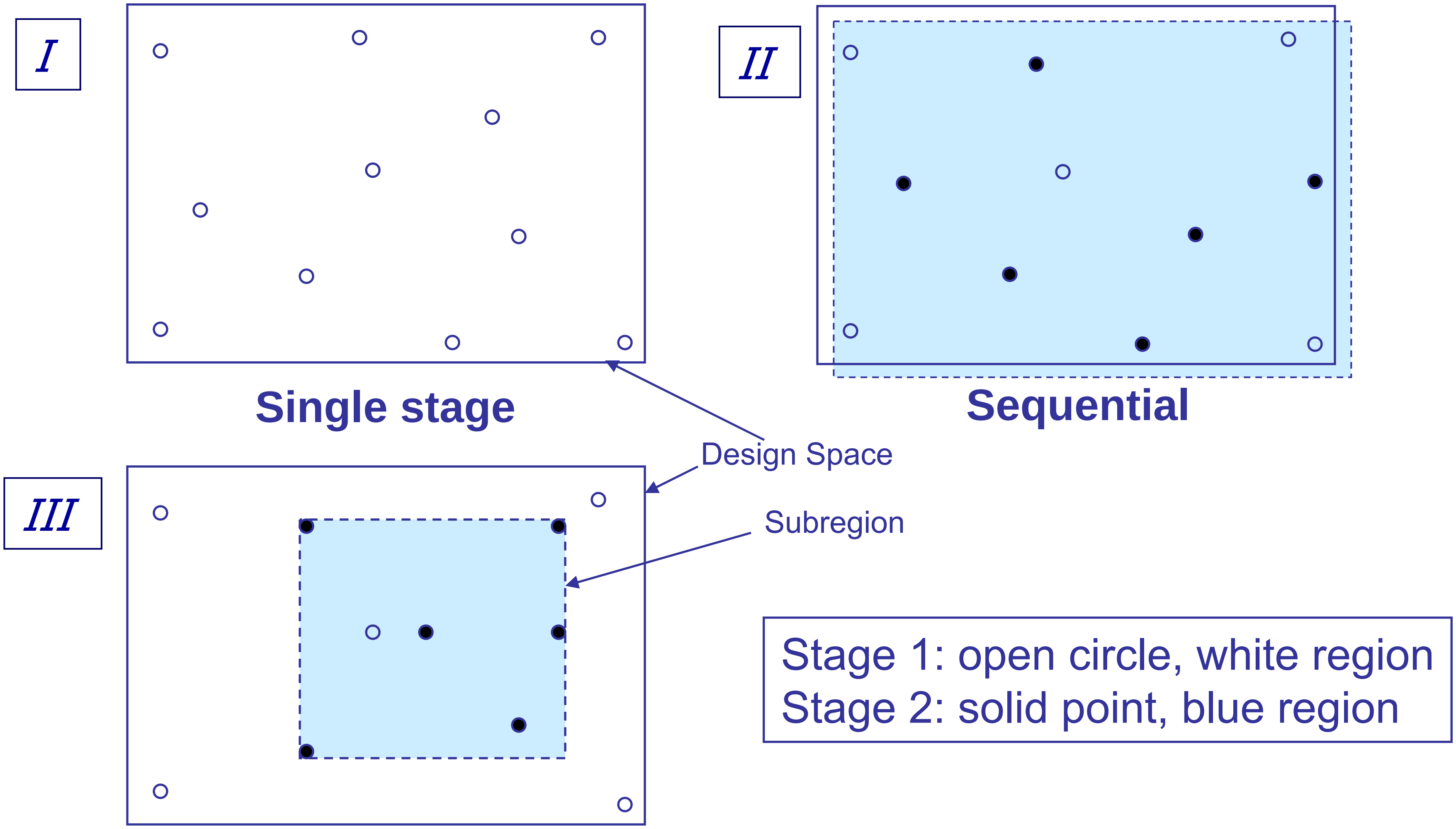
Metamodels: Summary

Response Surface Methodology (RSM)	Feedforward Neural Networks (FF)	Radial Basis Function Networks (RBF)
Polynomial basis functions	Simulation of a biological network. Sigmoid basis fns.	Local Gaussian or multi-quadric basis functions
Regional approximation, requires iterative domain reduction	Global approximation	Global approximation
Linear regression. Accuracy is limited by order of polynomial.	Nonlinear regression. Robustness requires committee (inner loop)	Linear regression within nonlinear loop. Cross-validation for high accuracy
Very fast	Very slow. Responses processed individually	Fast

Metamodel-based Optimization



Metamodel-based Optimization Strategies
Space-filling point selection

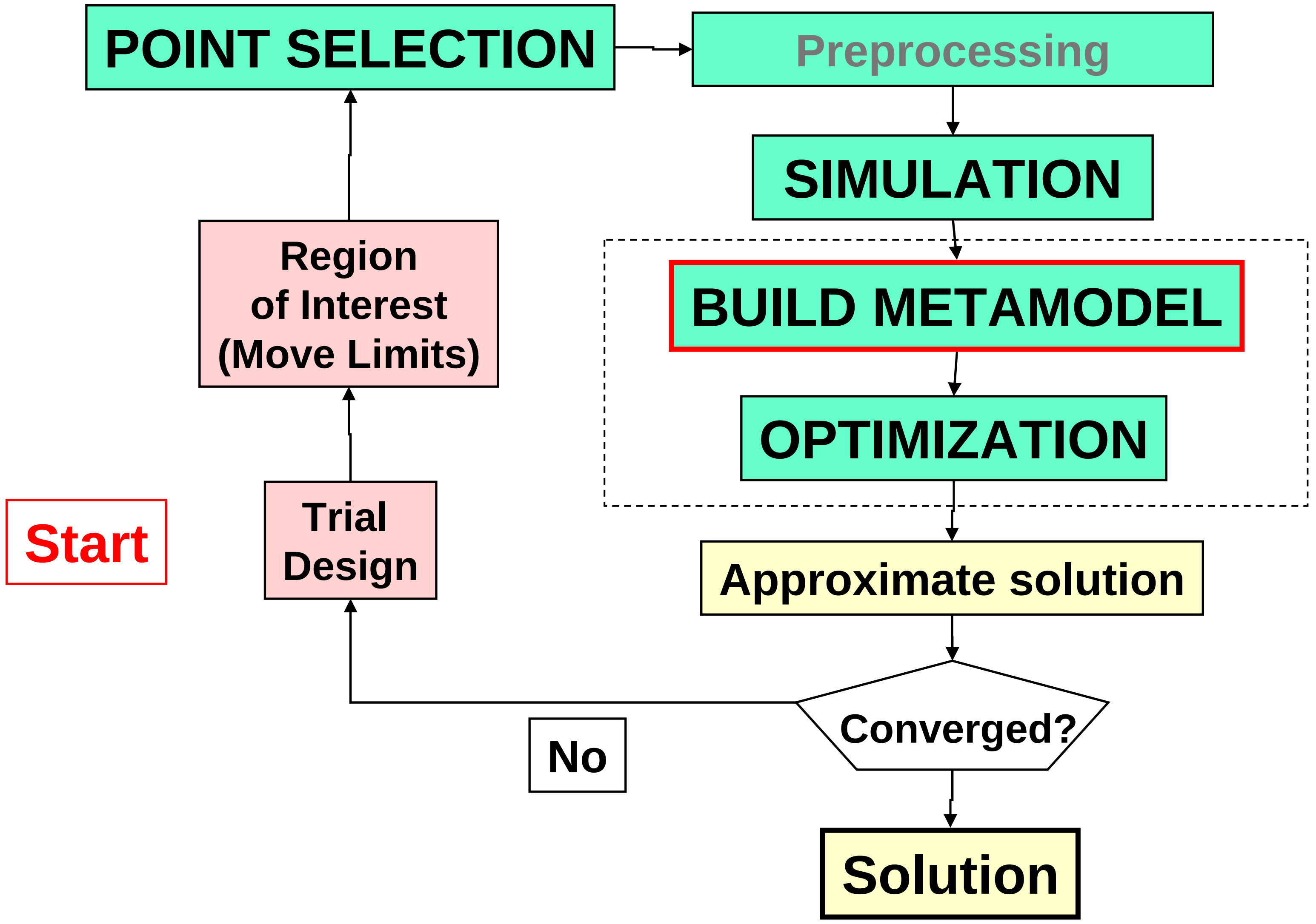


Sequential with domain reduction

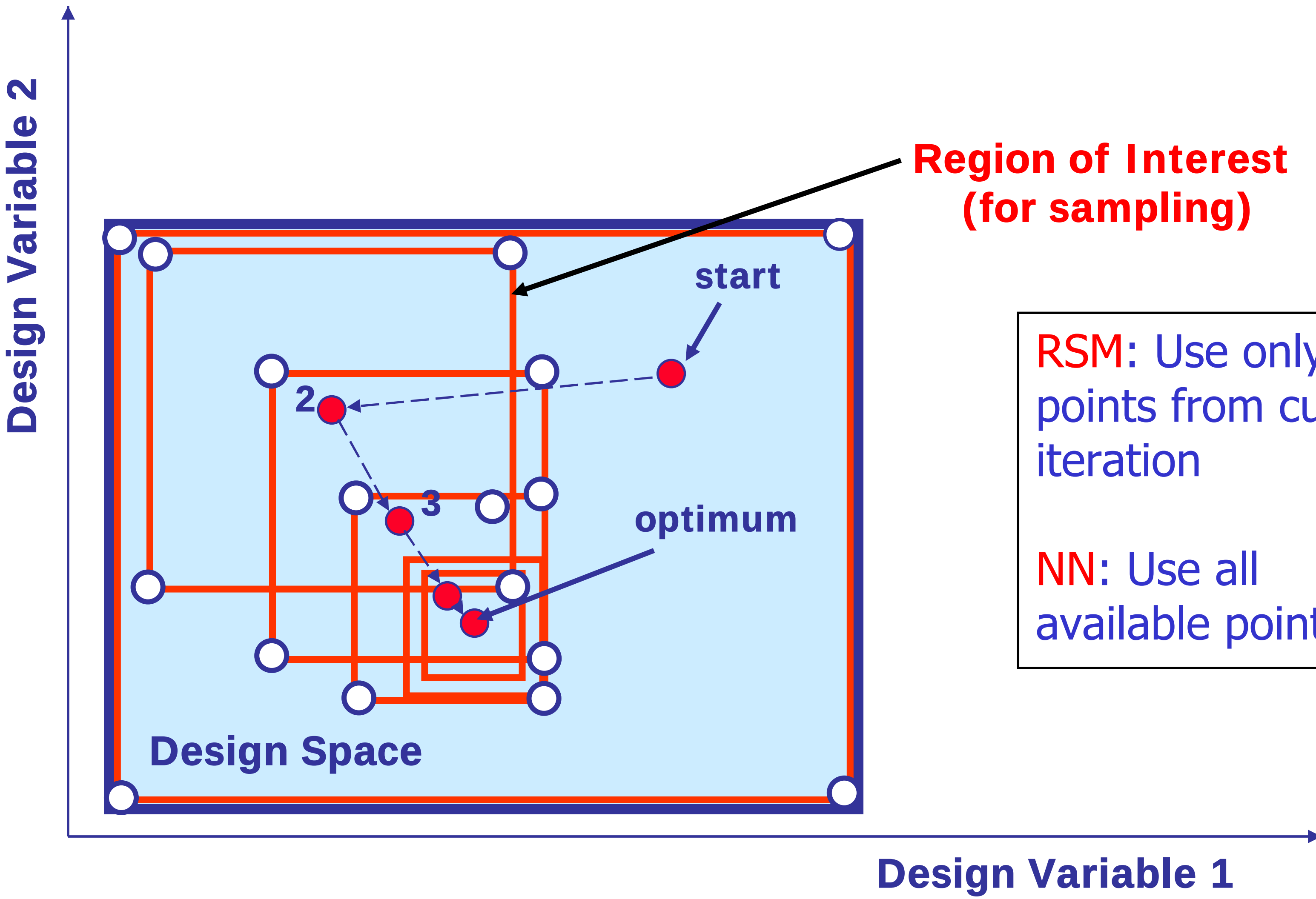
Optimization Strategies

- Single stage
  - Suitable for a **fixed computational budget**
  - All the points are determined in one stage, using Space Filling
  - Highly suitable to create a global metamodel
- Sequential
  - Suitable for **maximizing metamodel prediction accuracy** using a Stopping criterion
  - Add Space Filling points in each iteration
- Sequential with domain reduction
  - **Converges to a single optimum point** (single objective)
  - Domain reduction in each iteration: all points within a subregion
  - Ideal for system identification

Design Improvement Cycle
Simulation-based using Metamodel



Domain reduction: convergence



RSM: Use only points from current iteration

NN: Use all available points

Example (Domain reduction)

Crash model

30 000 elements

Intrusion = 552mm

Stage1Pulse = 14.34g

Stage2Pulse = 17.57g

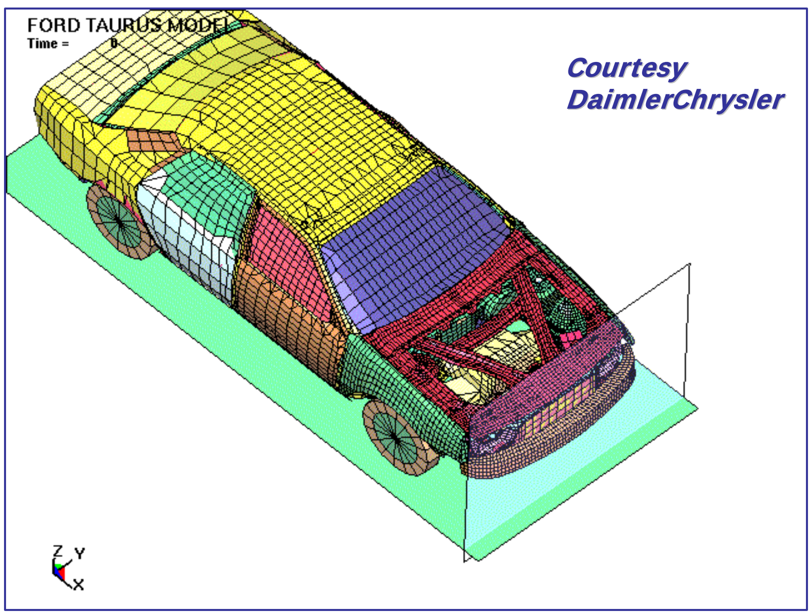
Stage3Pulse = 20.76g

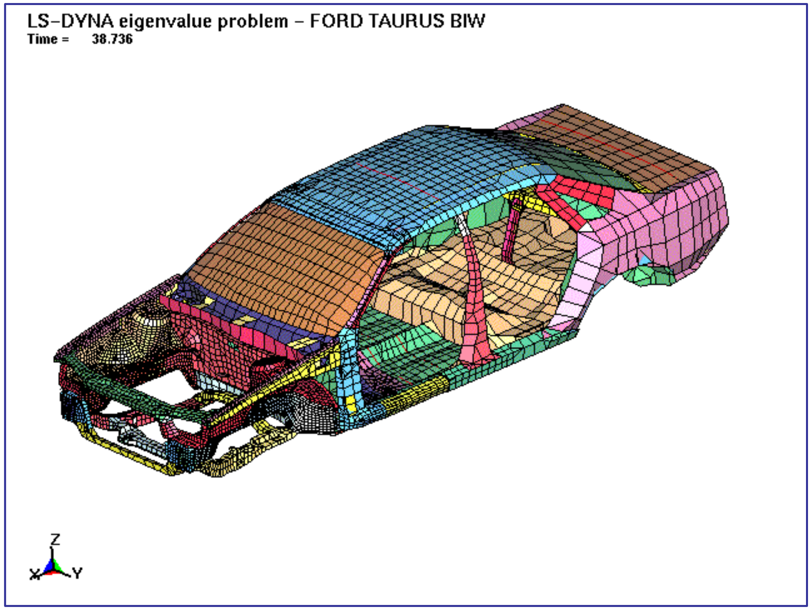
BIW model

18 000 elements

Torsional mode 1

Frequency = 38.7Hz





Design Formulation

Design Objective:

Minimize (Mass of components)

Design Constraints:

Intrusion < 552.38mm

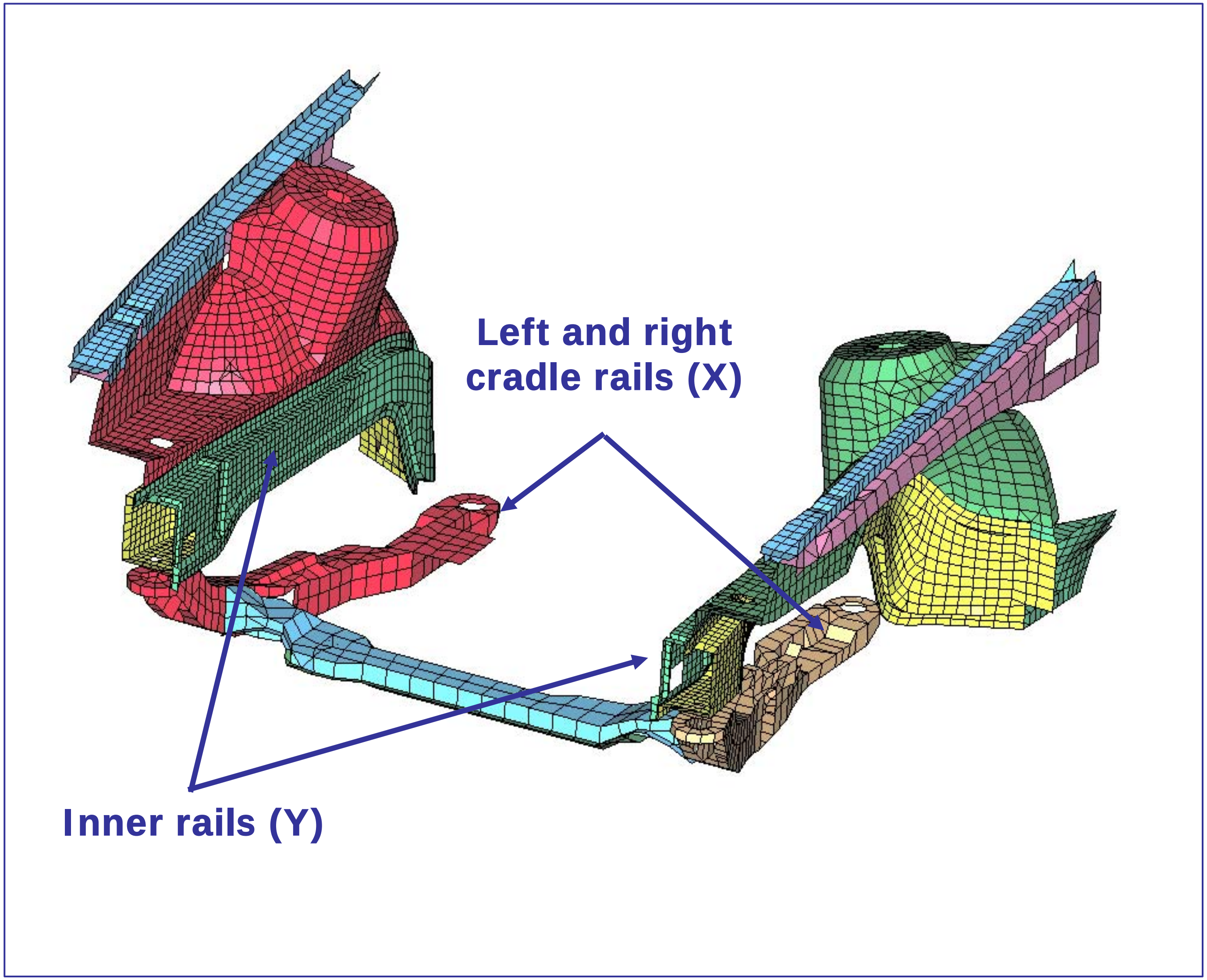
Stage1Pulse > 14.58g

Stage2Pulse > 17.47g

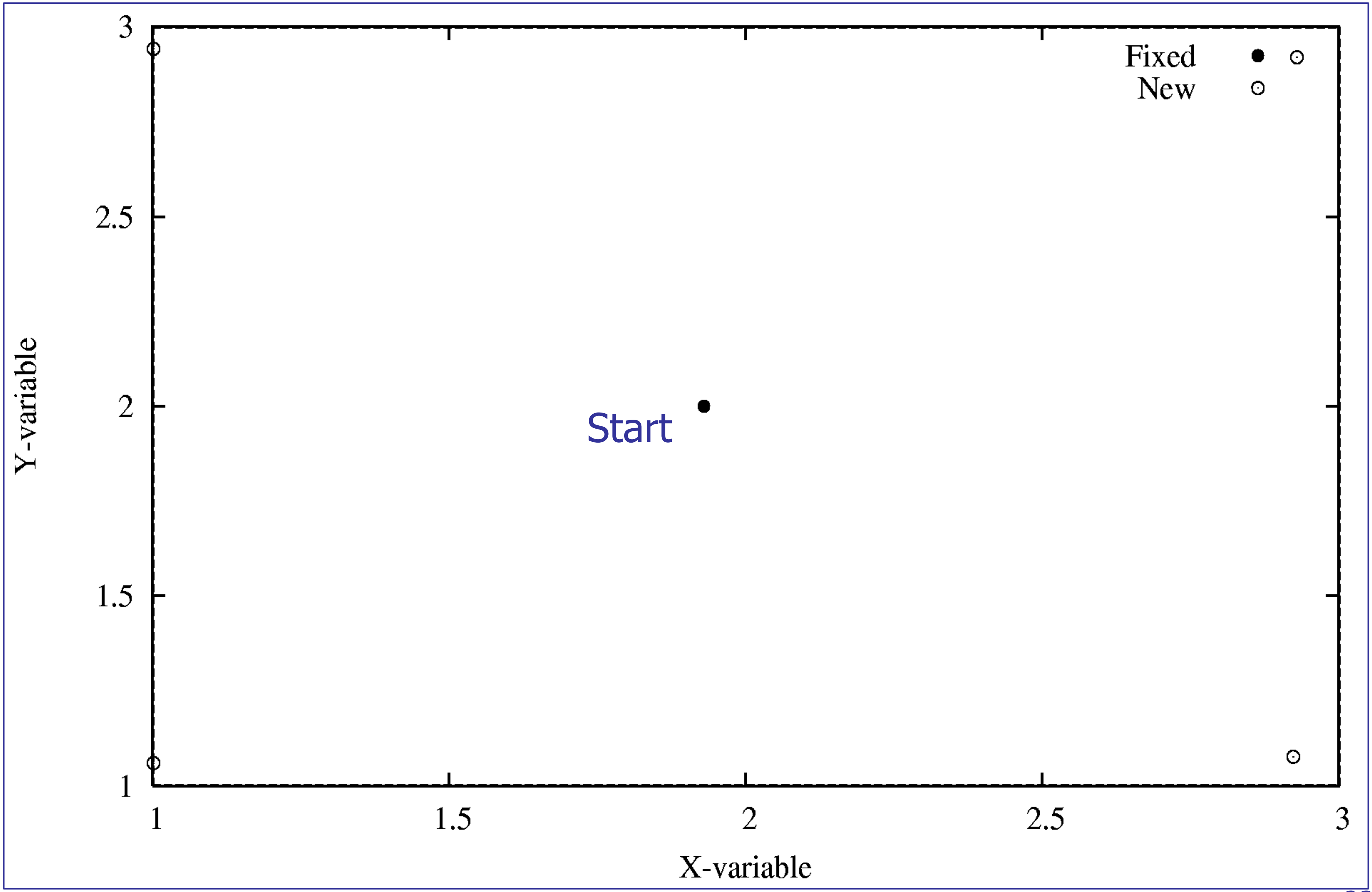
Stage3Pulse > 20.59g

41.38Hz < Torsional mode 1 frequency < 42.38Hz

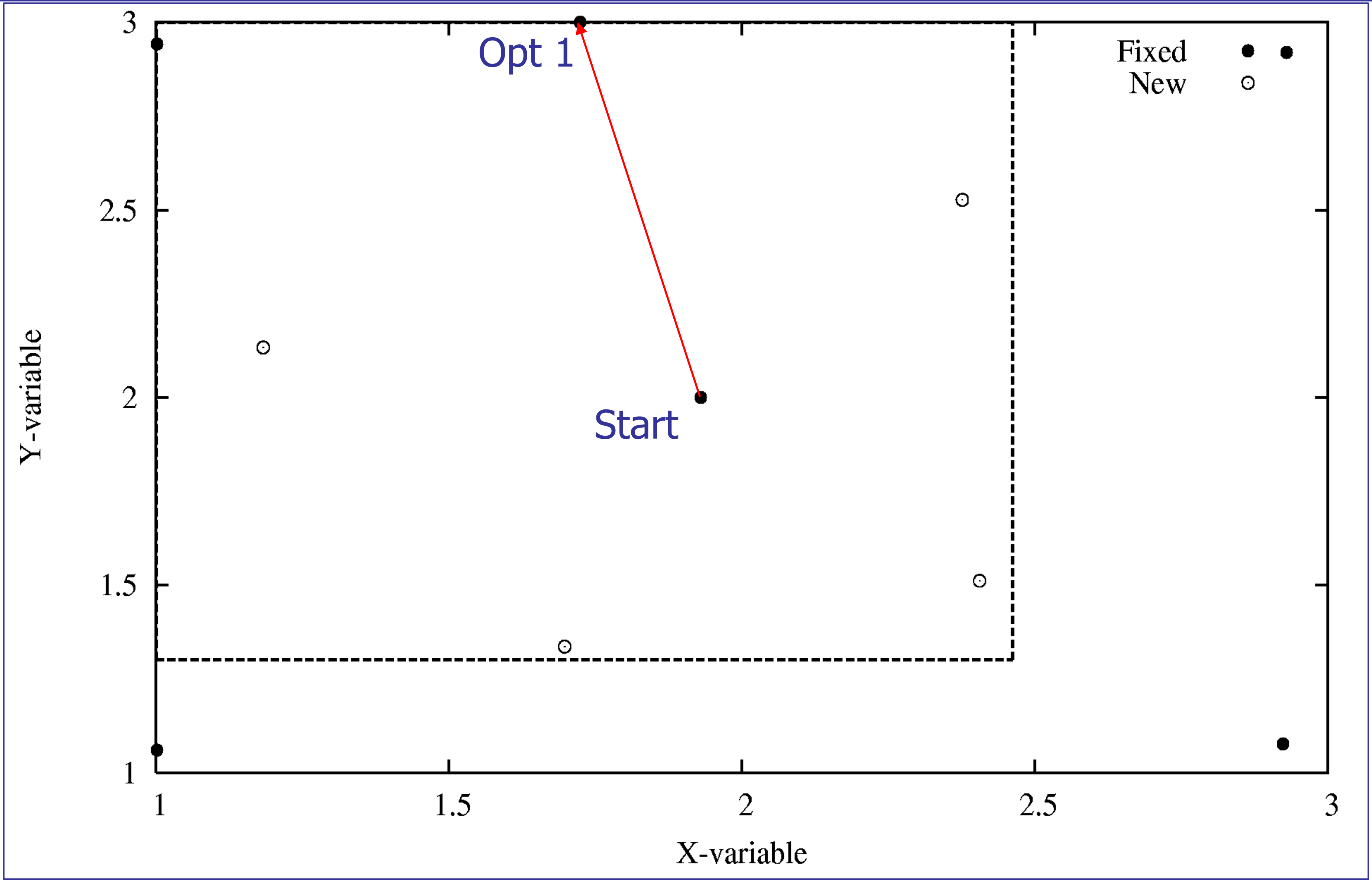
Two Design Variables



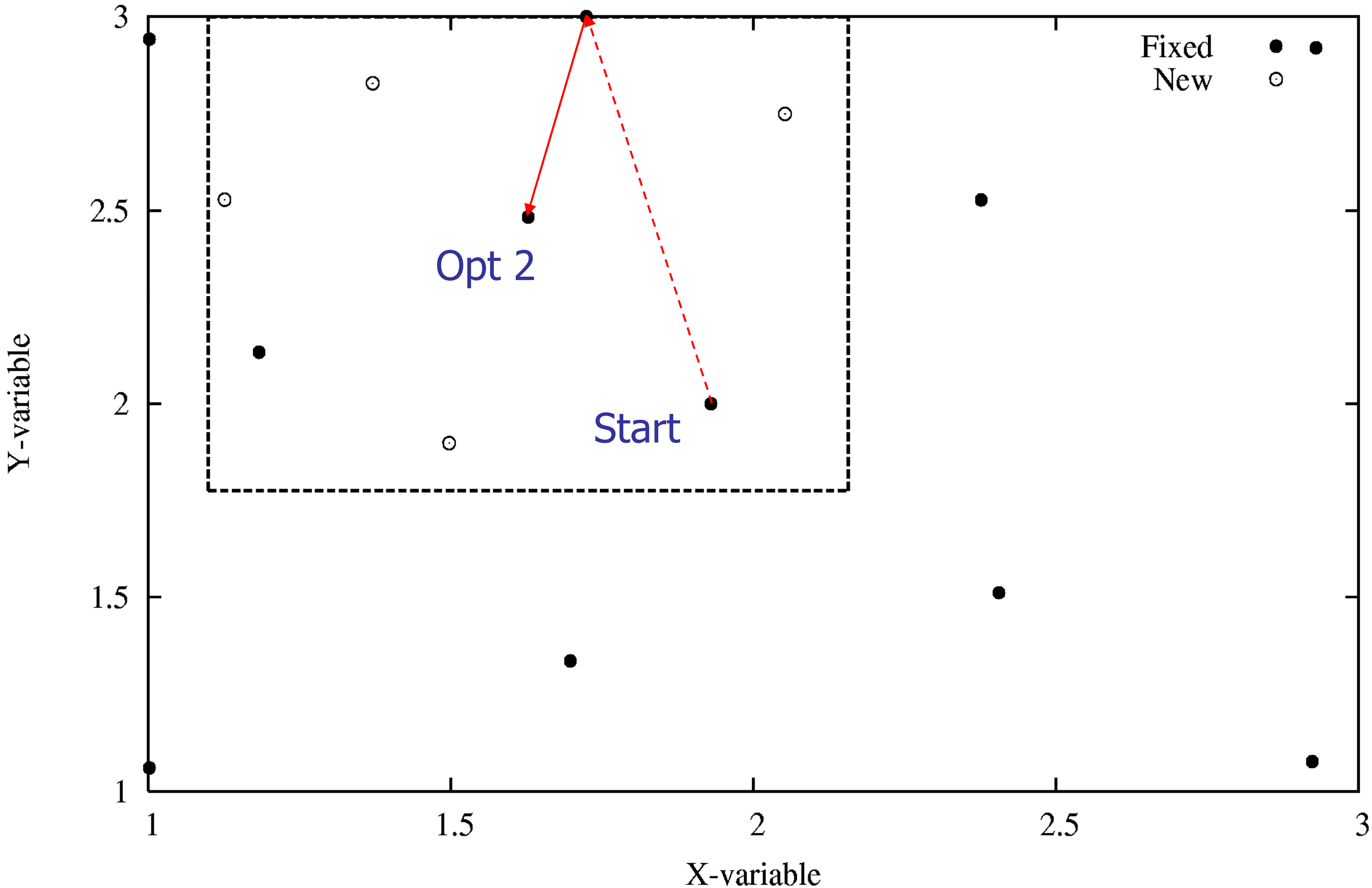
Sampling: Space Filling Method
Iteration 1



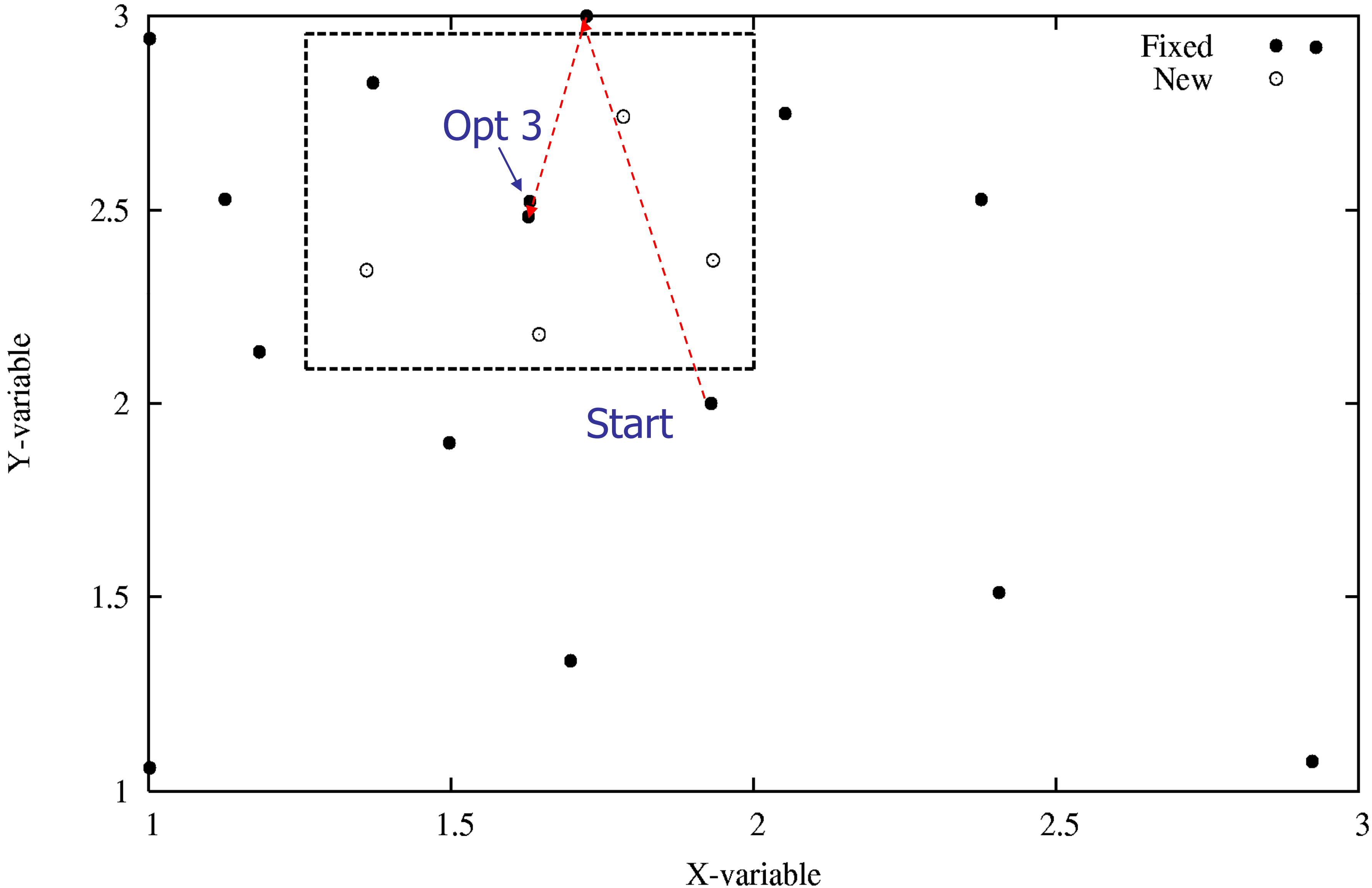
Sampling: Space Filling Method
Iteration 2



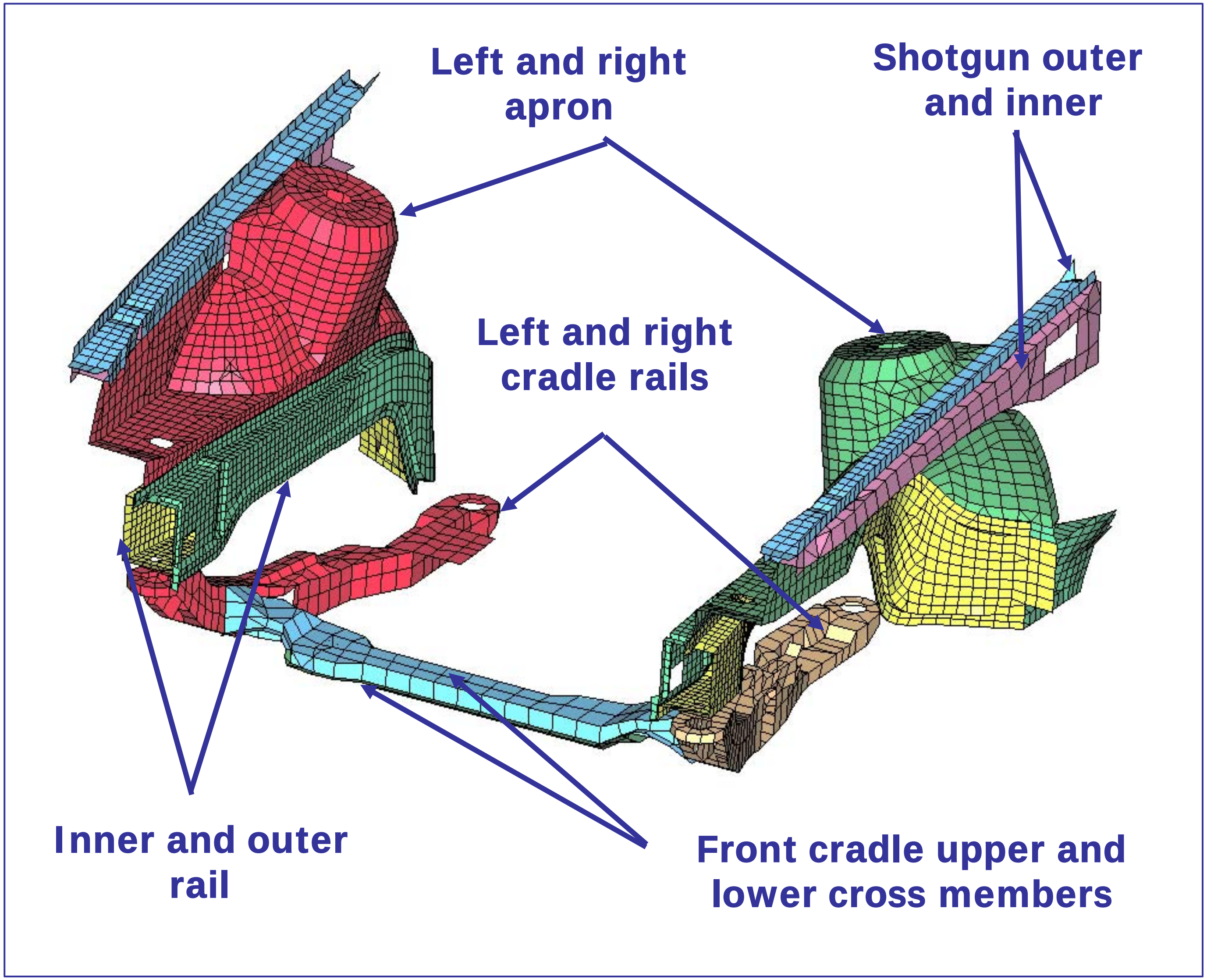
Sampling: Space Filling Method
Iteration 3



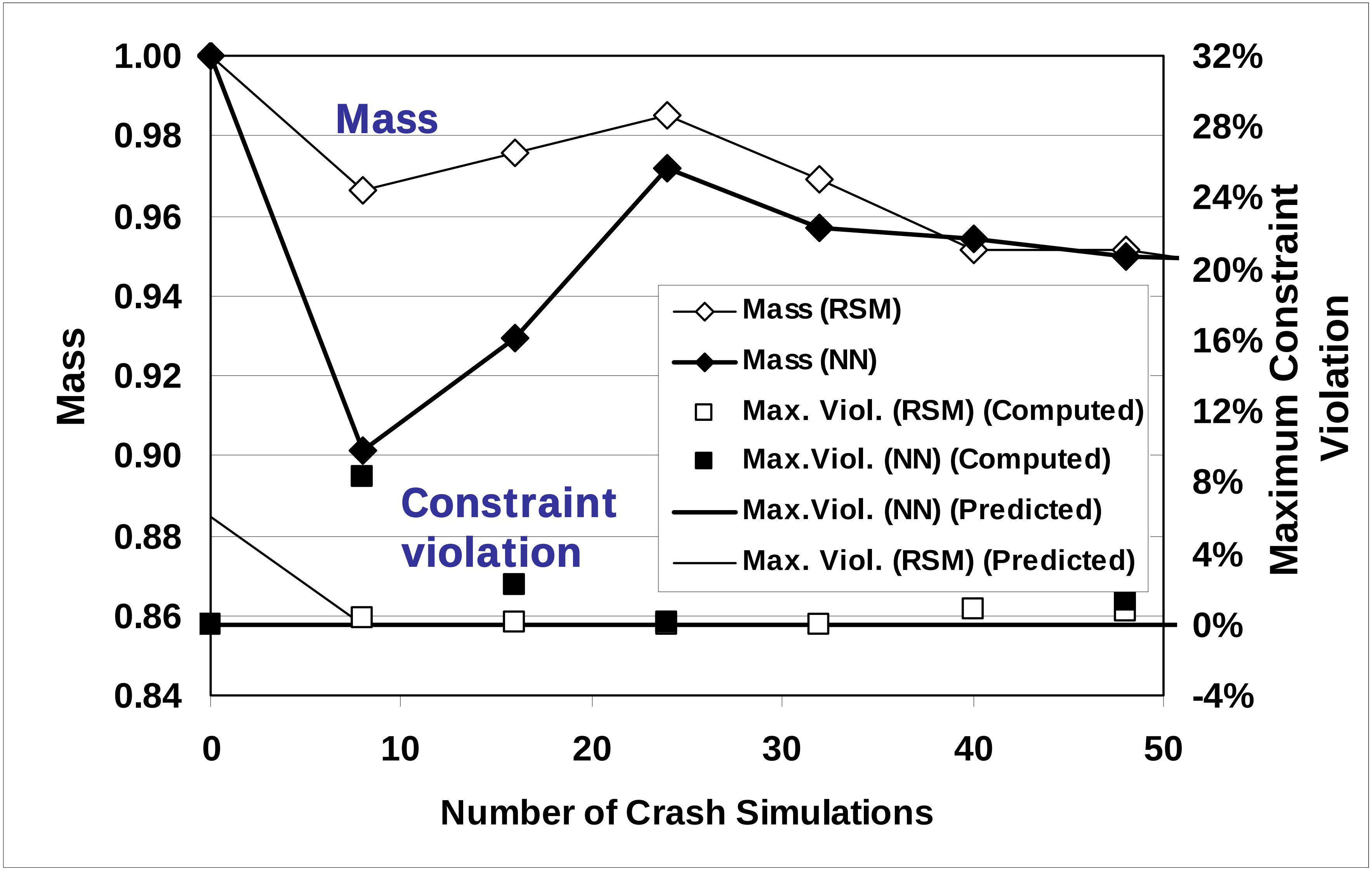
Sampling: Space Filling Method
Iteration 4



Domain reduction: more variables



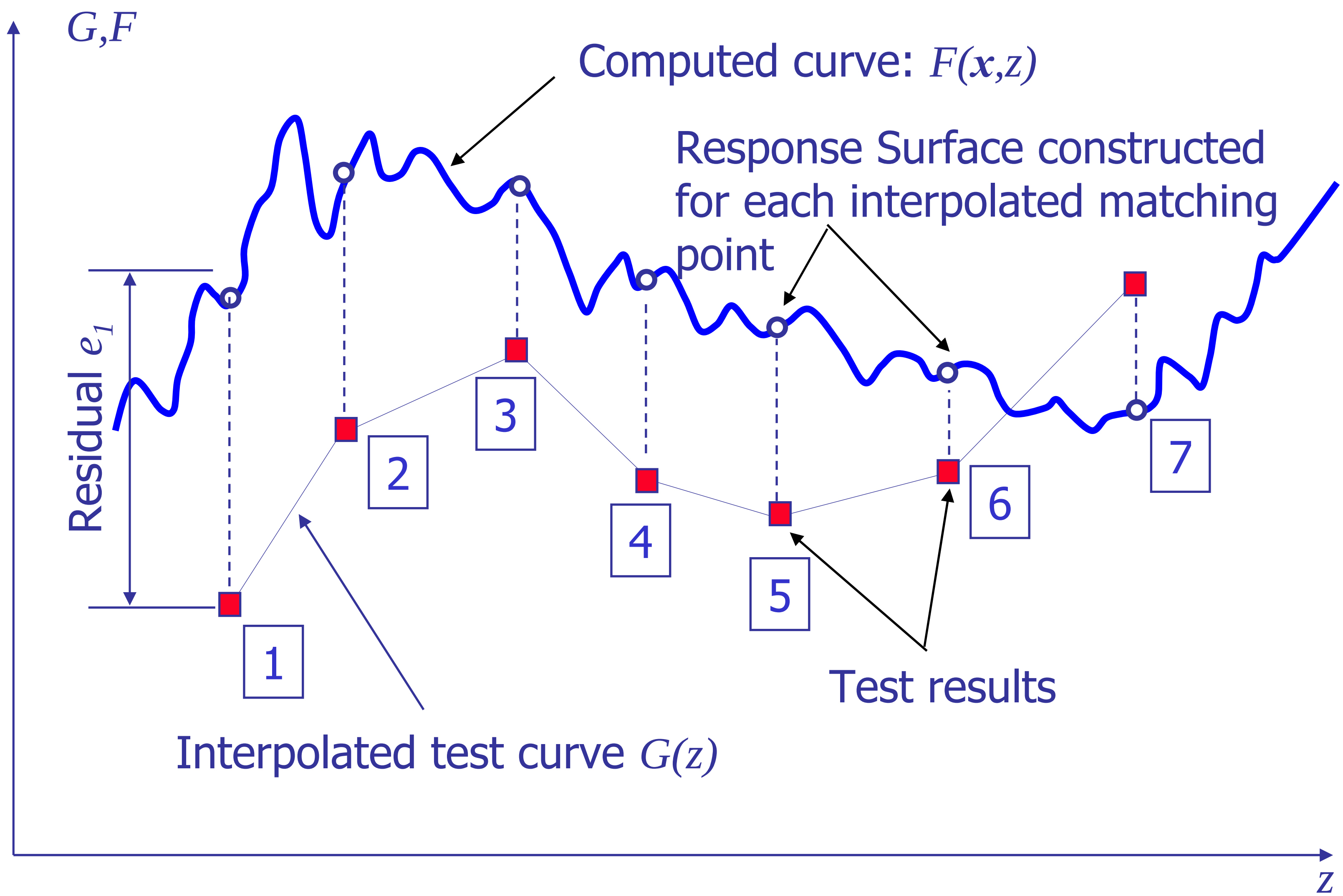
Metamodel Comparison



Parameter Identification

- Used for calibrating material or system properties
- Methodology uses minimization of the differences between test and computed results
- Strategy
  - History-based Mean Squared Error
    - The target values can be specified in a history file and imported as a history. A single function computes the MSE

History-based Parameter Identification
Test points + Computed curve



History-based Parameter Identification
Mean squared error

Weight (Importance of error)

Response Surface Value

Test Value

Residual

$$\frac{1}{P}\sum_{p=1}^{P} W_i\left(\frac{F_i(\boldsymbol{x})-G_i}{s_i}\right)^2=\frac{1}{P}\sum_{p=1}^{P} W_i\left(\frac{e_i(\boldsymbol{x})}{s_i}\right)^2$$

Number of points

Residual Scale factor
(Normalization of error)

Variables (material
or system constants)

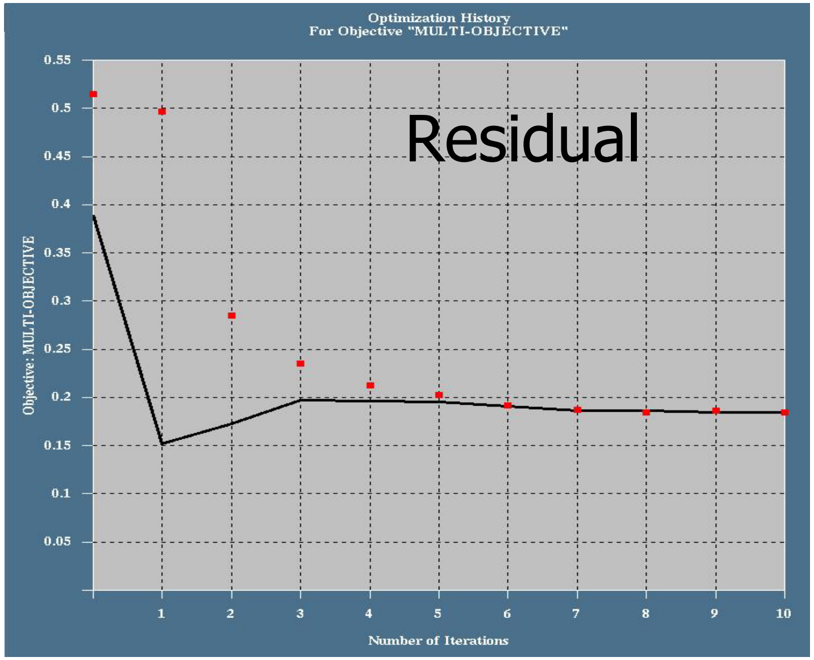
Material Identification: Concrete Mat 159

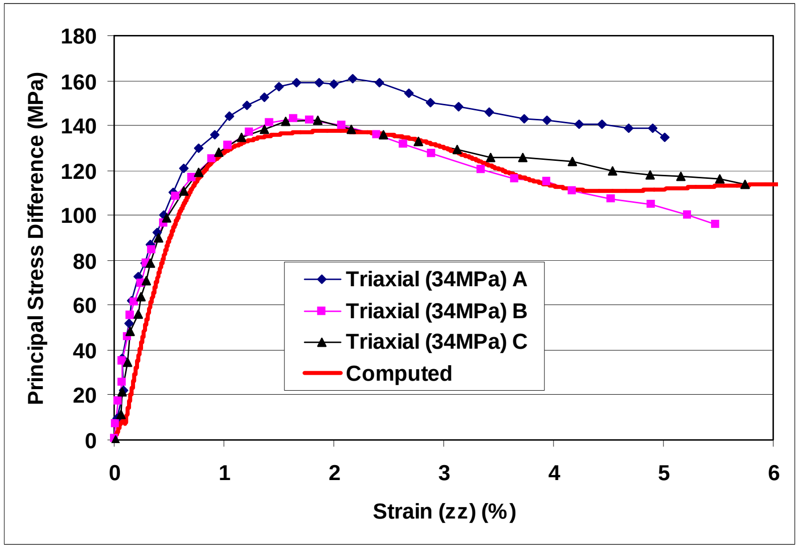
11 parameters, 9 test types, 20 test sets

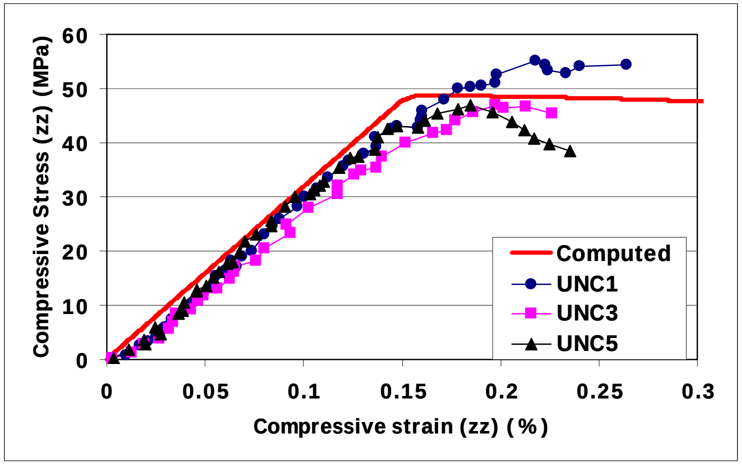
Par.	C00 UNC	T00 DP	PRS ISO-comp	UNX UNX	C07 TXC7	C14 TXC14	C20 TXC20	C34 TXC34	C69 TXC69
G	●	●	●	●					
K	●	●	●	●					
R				●	●	●	●	●	●
X_0			●						
W			●	●					
D_1			●						
D_2			●						
θ					●	●	●	●	●
λ					●	●	●	●	●
β					●	●	●	●	●
η					●	●	●	●	●

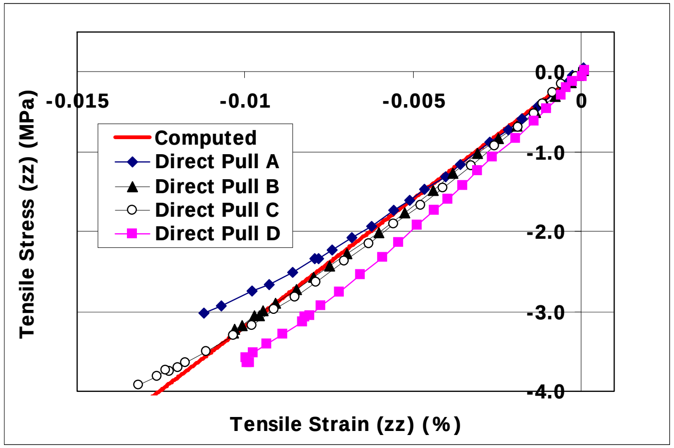
Multiple cases, shared variables

Material Identification:
Optimization (10 iterations): Stress vs. Strain Results



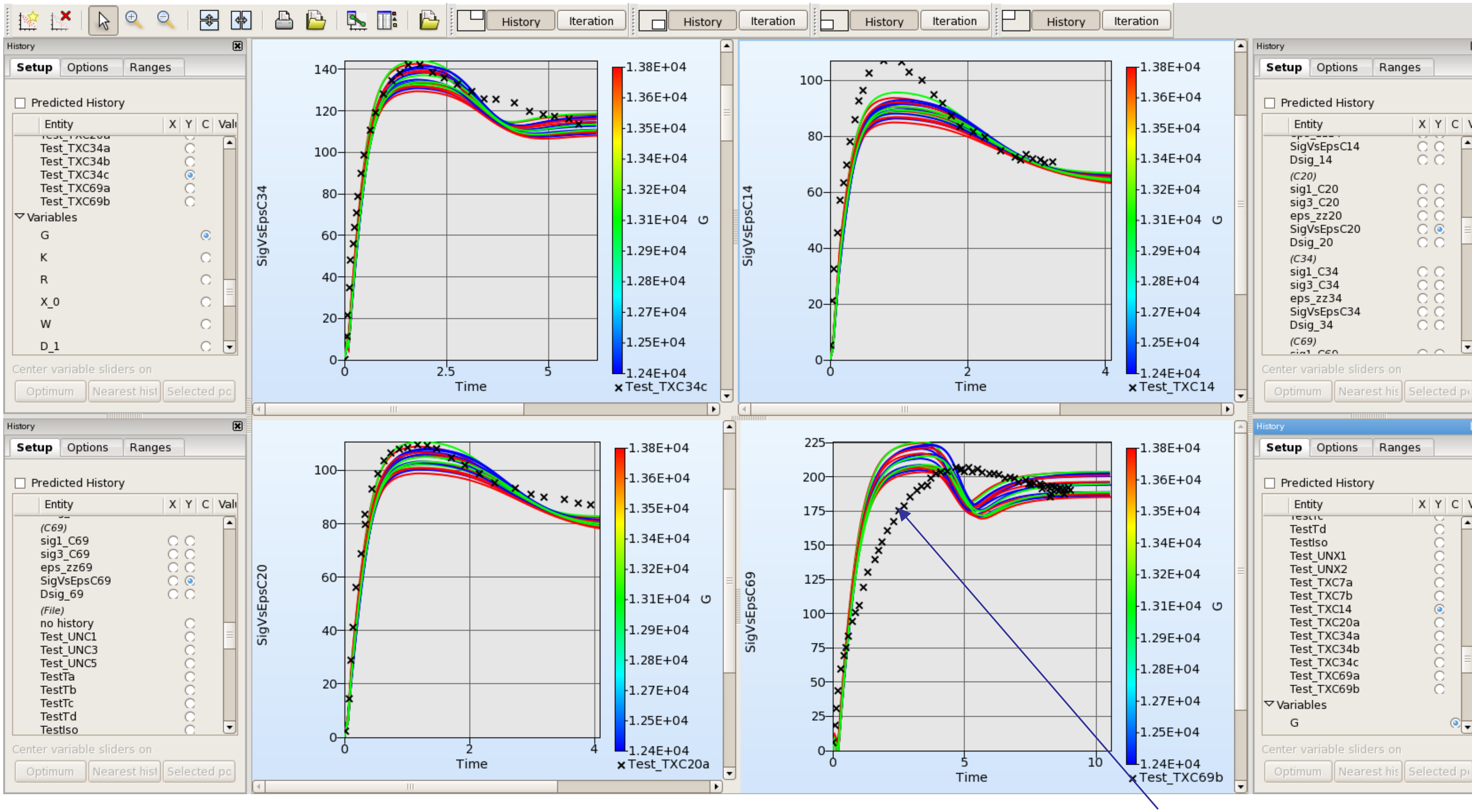






Viewer: Computed Histories

◈ View comparative histories at all design points

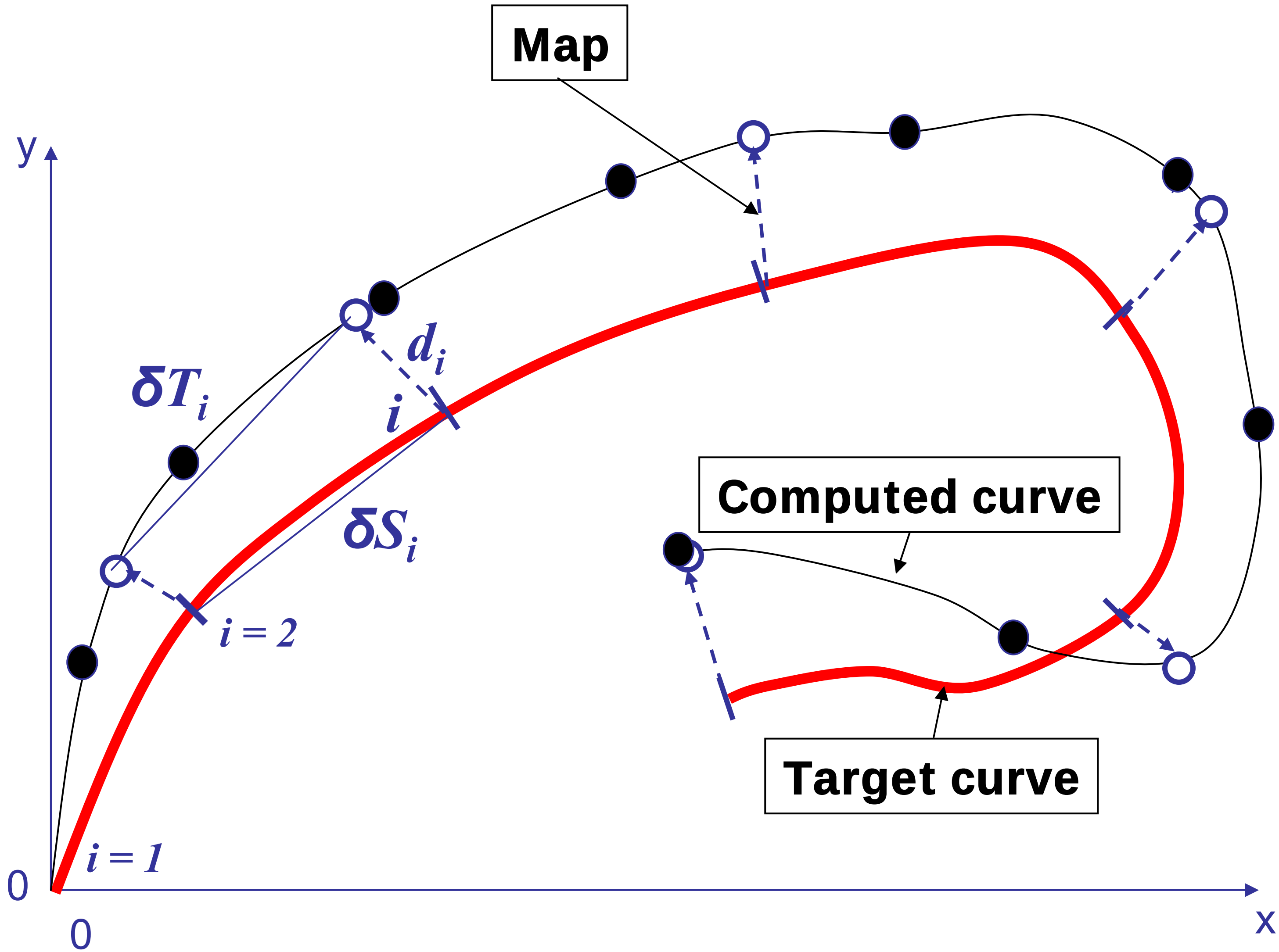


Test data

Example: Parameter Identification

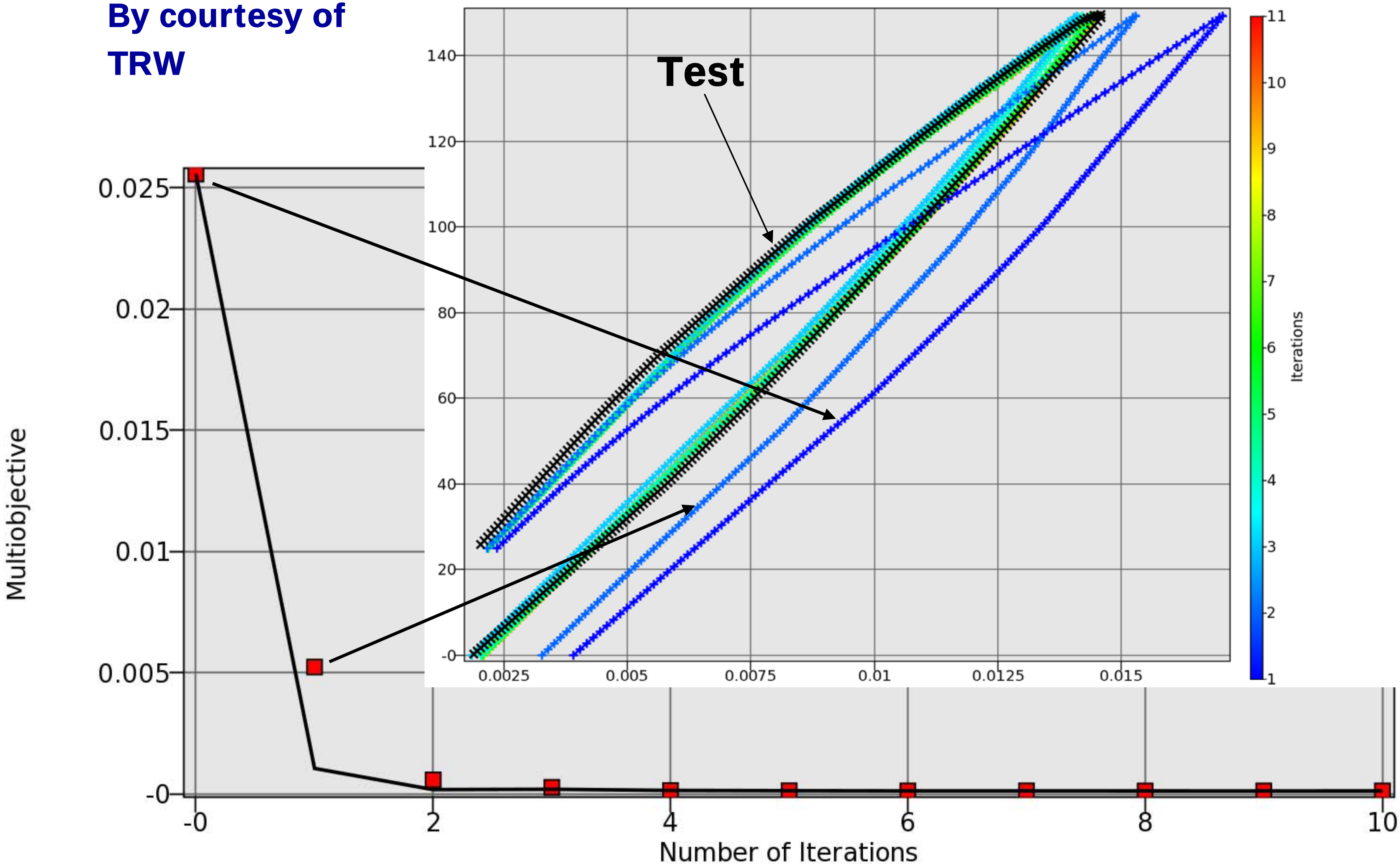
- Used for calibrating material or system properties
- *Min.*
  Difference between *test* and *computed* results
- Use Sequential Response Surface Method (Linear)

Mapped Curve Matching (v4.2)



Curve Matching (v4.2)

By courtesy of TRW



Discrete Optimization

- Discrete variables can have only distinct values, e.g. { 1.0, 2.0, 2.5, 4.5 }
- Discrete and continuous variables can be used together
- Discrete sampling can be combined with discrete optimization

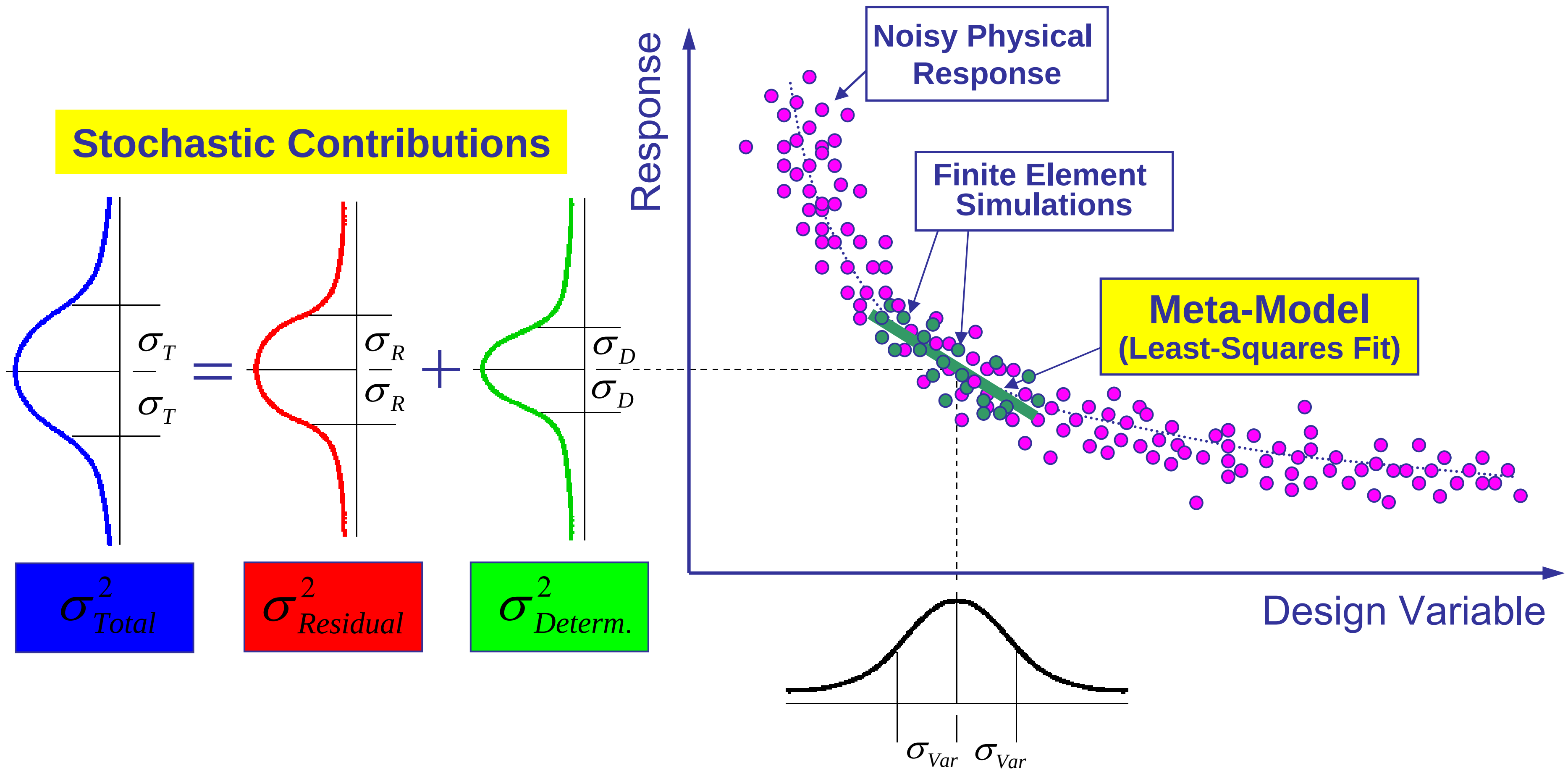
Reliability/Robust Design

Summary: Probabilistic Analysis

◈ Reliability and Robustness

- Reliability:
  - Calculate probability of failure
- Robust Design:
  - Standard Deviation of response
  - Consistent product performance
- Reliability-based Design Optimization (RBDO)
  - Incorporates Reliability and Robustness into design improvement
- Identify sources of uncertainty in the FE models: Outlier Analysis

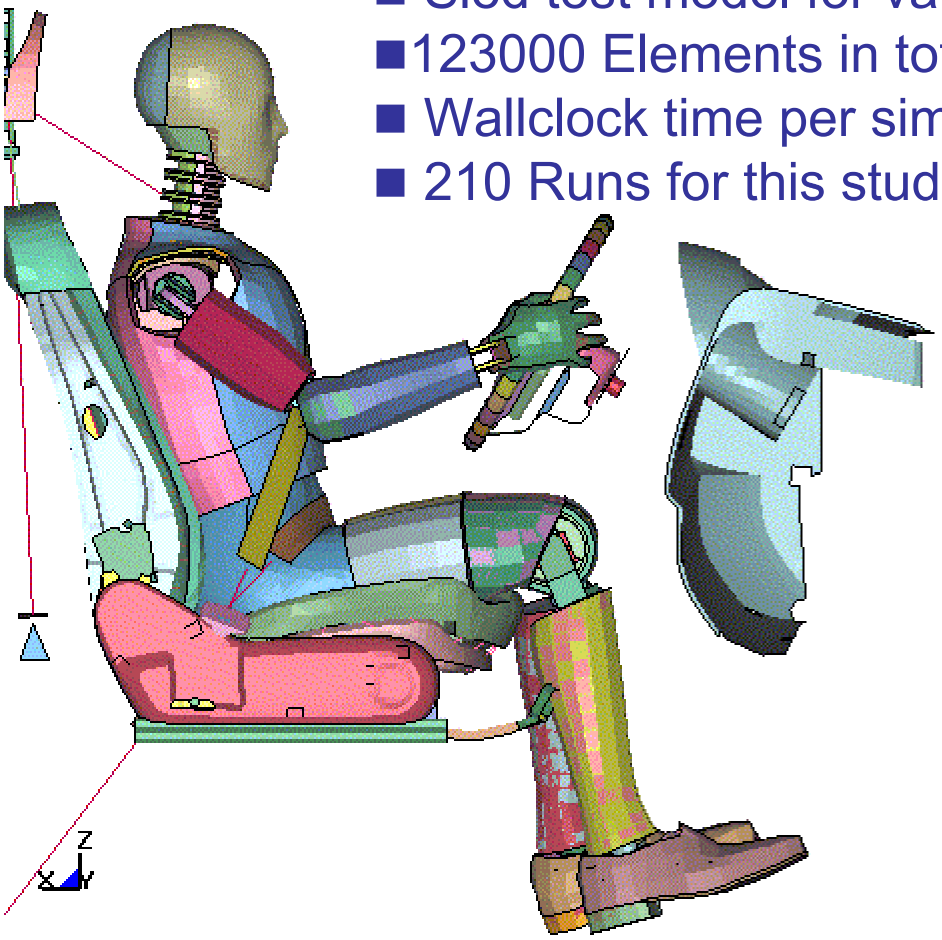
Meta-Modeling and Stochastic Contributions



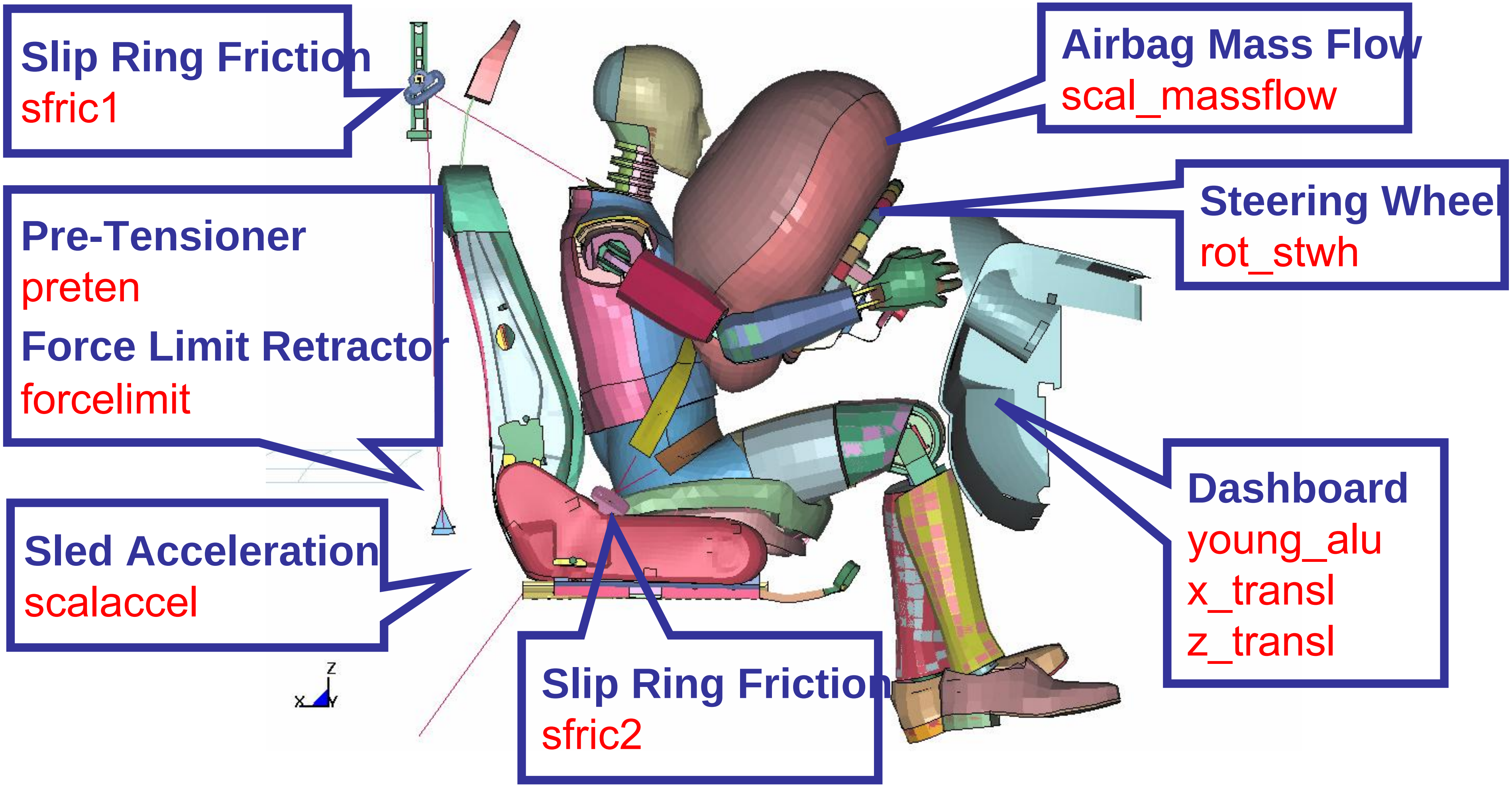
Günther, F., Müllerschön, H., Roux, W.J.
LS-DYNA International Users Conference, 2004

Occupant Simulation Model

- Sled test model for validation of occupant simulation
- 123000 Elements in total (Beams/Shells/Solids)
- Wallclock time per simulation: 9.5 hours on 16 cpus
- 210 Runs for this study

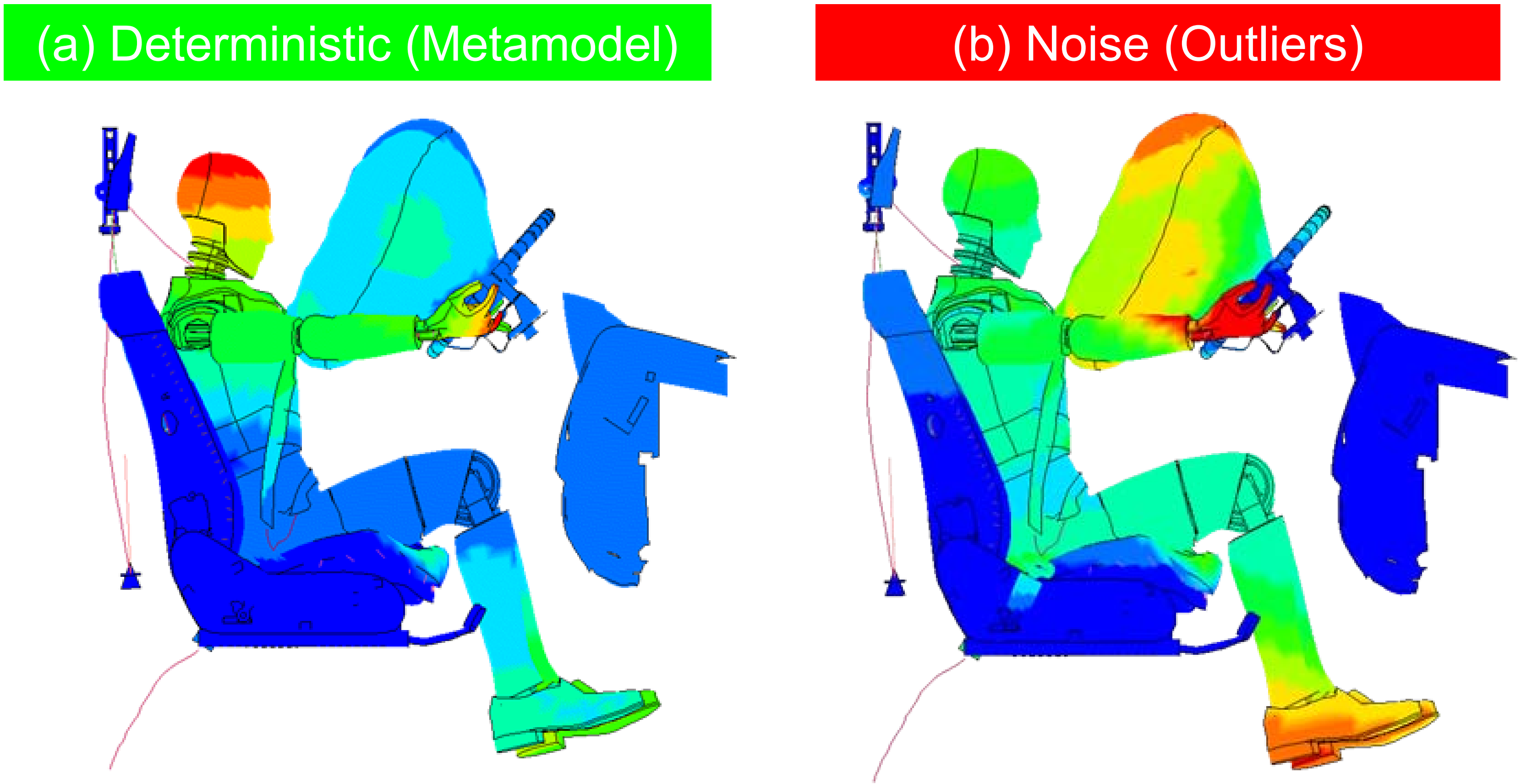


Uncertainty
Design Variables

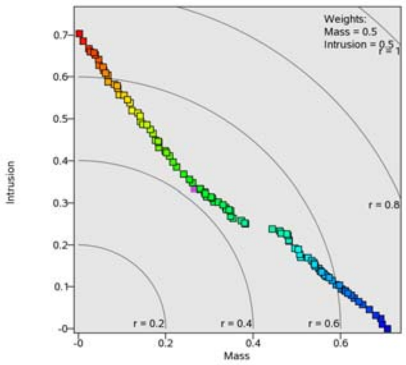


Displacement Statistics: Metamodel vs. Outliers

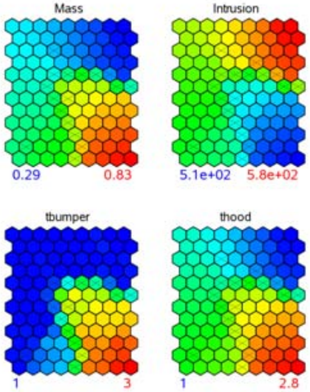
Standard deviation of x-displacements of each node (120 runs)



Günther, F., Müllerschön, H., Roux, W.J.
LS-DYNA International Users Conference, 2004



Multi-objective Optimization



Multi-objective Optimization

- Most engineering problems deal with multiple objectives e.g., cost, weight, safety, efficiency etc.
- Often conflicting requirements e.g., weight vs. efficiency
- **No single optimal solution!**
- **Strategies**
  - **Direct Simulation**
    - Higher cost
  - **Metamodel-based**
    - Accuracy depends on quality of metamodel
    - Can use sequential updating of metamodel to improve accuracy

Validation of GA Using Benchmarks

Unconstrained test problems

$$\underset{\vec{X}}{Minimize} \quad f_1(\vec{X}) = x_1; \quad f_2(\vec{X}) = g(\vec{X})h\left(g(\vec{X}), f_1(\vec{X})\right),$$

$$\textbf{ZDT2}: g(\vec{X}) = 1 + 9(N-1)^{-1}\sum_{i=2}^{N} x_i;$$

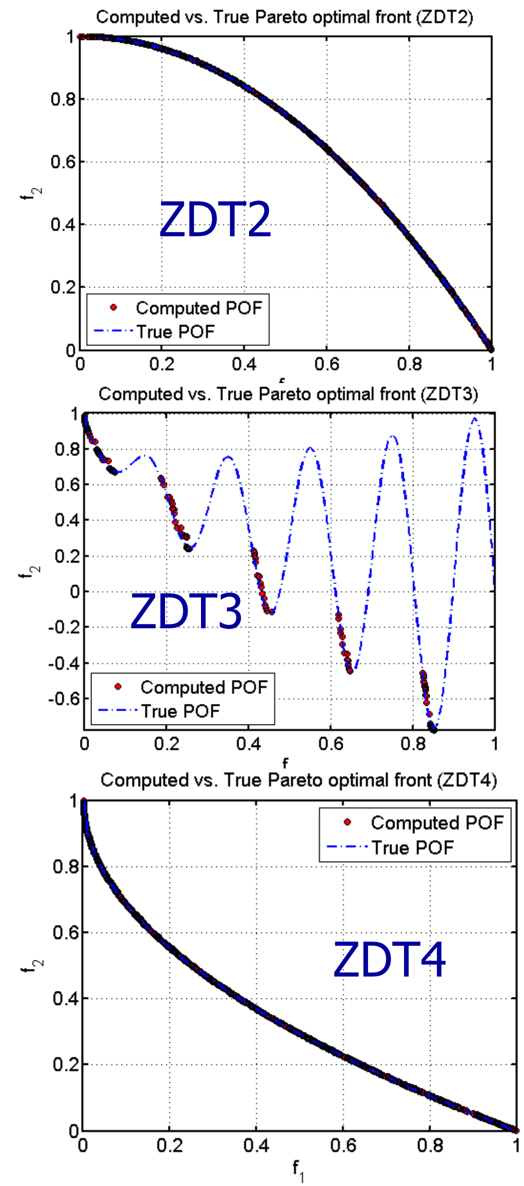
$$h(\vec{X}) = 1 - (f_1/g)^2; x_i \in [0,1].$$

$$\textbf{ZDT3}: g(\vec{X}) = 1 + 9(N-1)^{-1}\sum_{i=2}^{N} x_i;$$

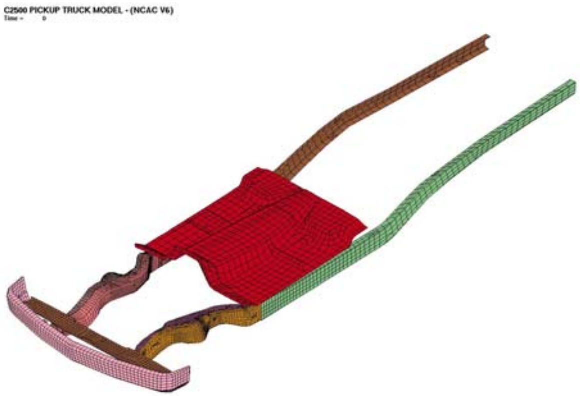
$$h(\vec{X}) = 1 - \sqrt{f_1/g} - (f_1/g)\sin(10\pi f_1); x_i \in [0,1].$$

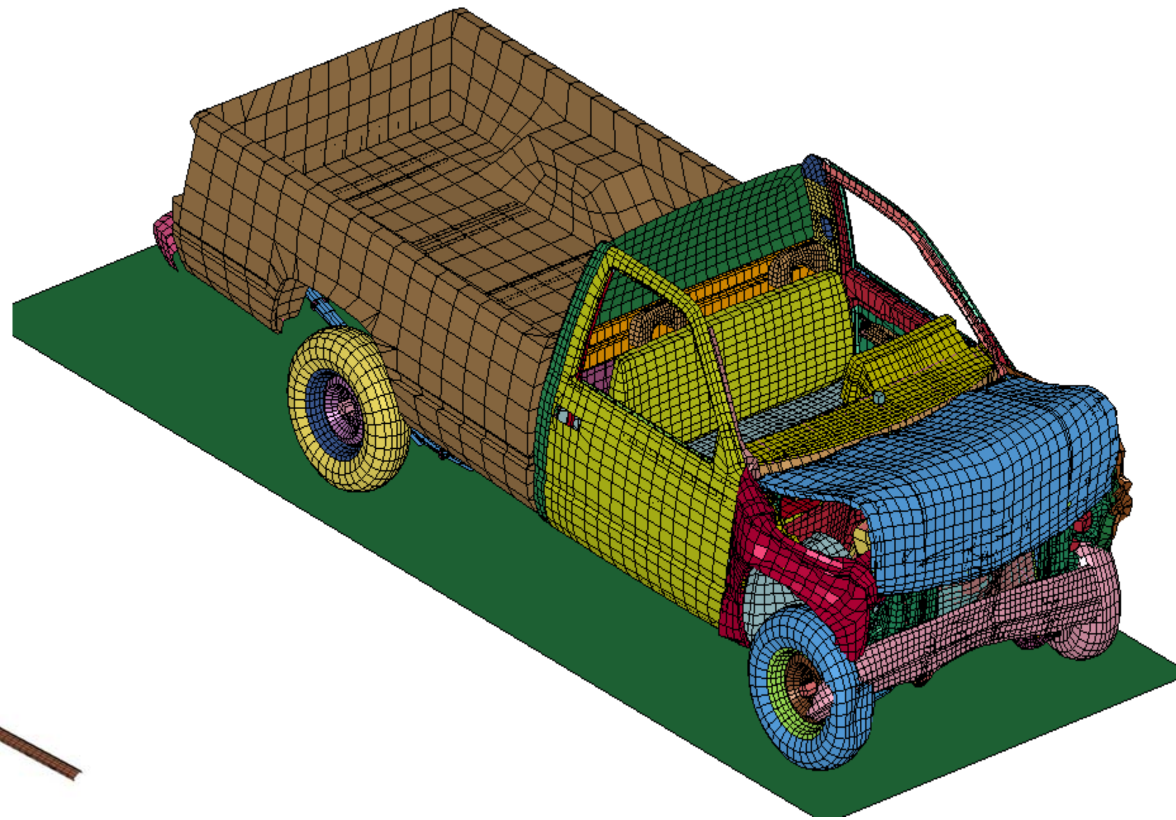
$$\textbf{ZDT4}: g(\vec{X}) = 1 + 10(N-1) + \sum_{i=2}^{N}(x_i^2 - 10\cos(4\pi x_i));$$

$$h(\vec{X}) = 1 - \sqrt{f_1/g}; x_1 \in [0,1]; x_{2,3,\ldots N} \in [-5,5].$$



Metamodel-based MOO/Robust design





Thickness design variables

Design criteria

Minimize

- Mass
- Acceleration
- Standard deviation of the intrusion (robustness)

Maximize

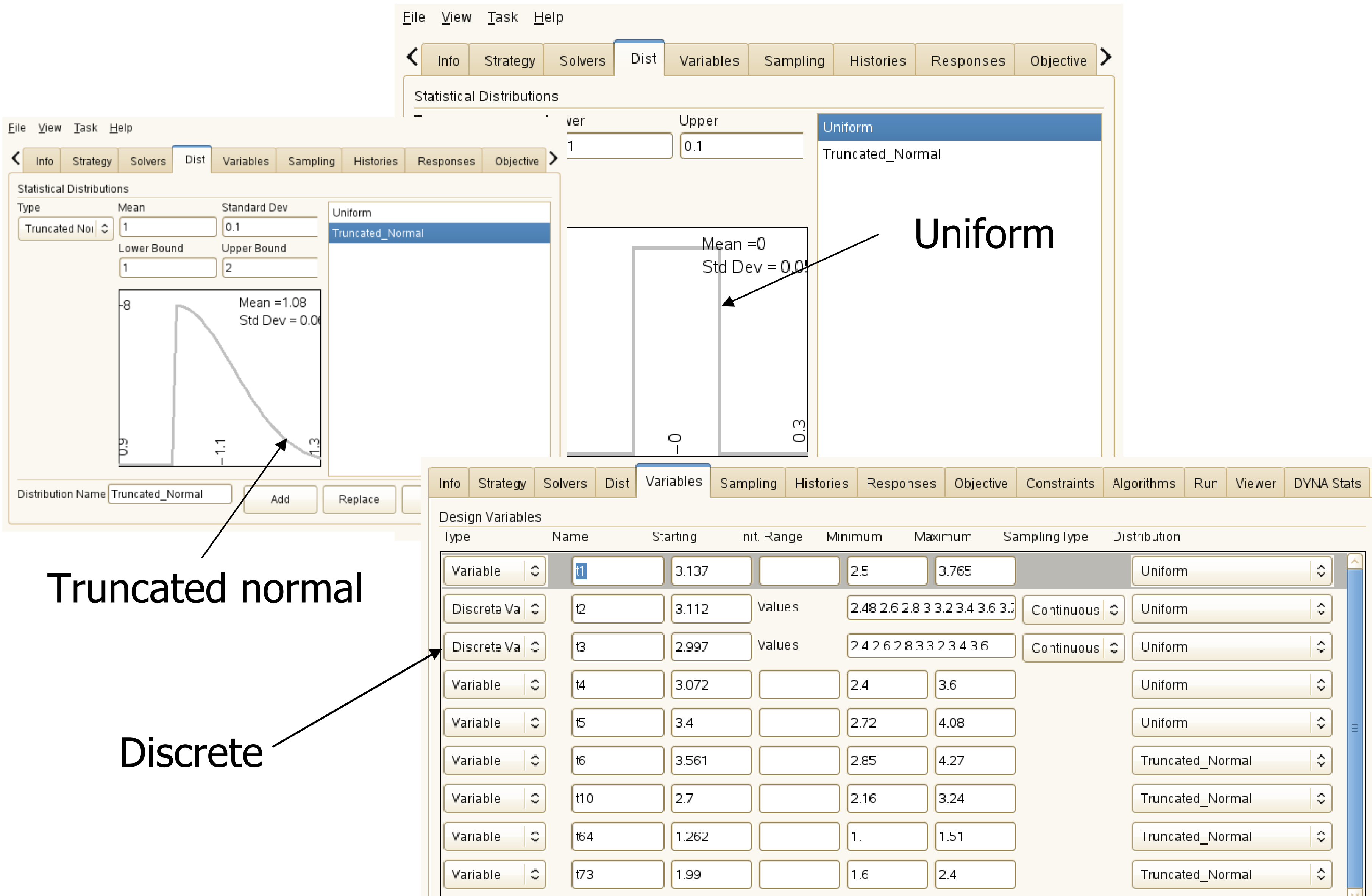
- Intrusion
- Time to zero velocity

9 stochastic thickness variables of main crash members.
2 discrete + 7 continuous

Intrusion	<	721
Stage 1 pulse	<	7.5g
Stage 2 pulse	<	20.2g
Stage 3 pulse	<	24.5g

Probability of failure < 0.15

Stochastic input



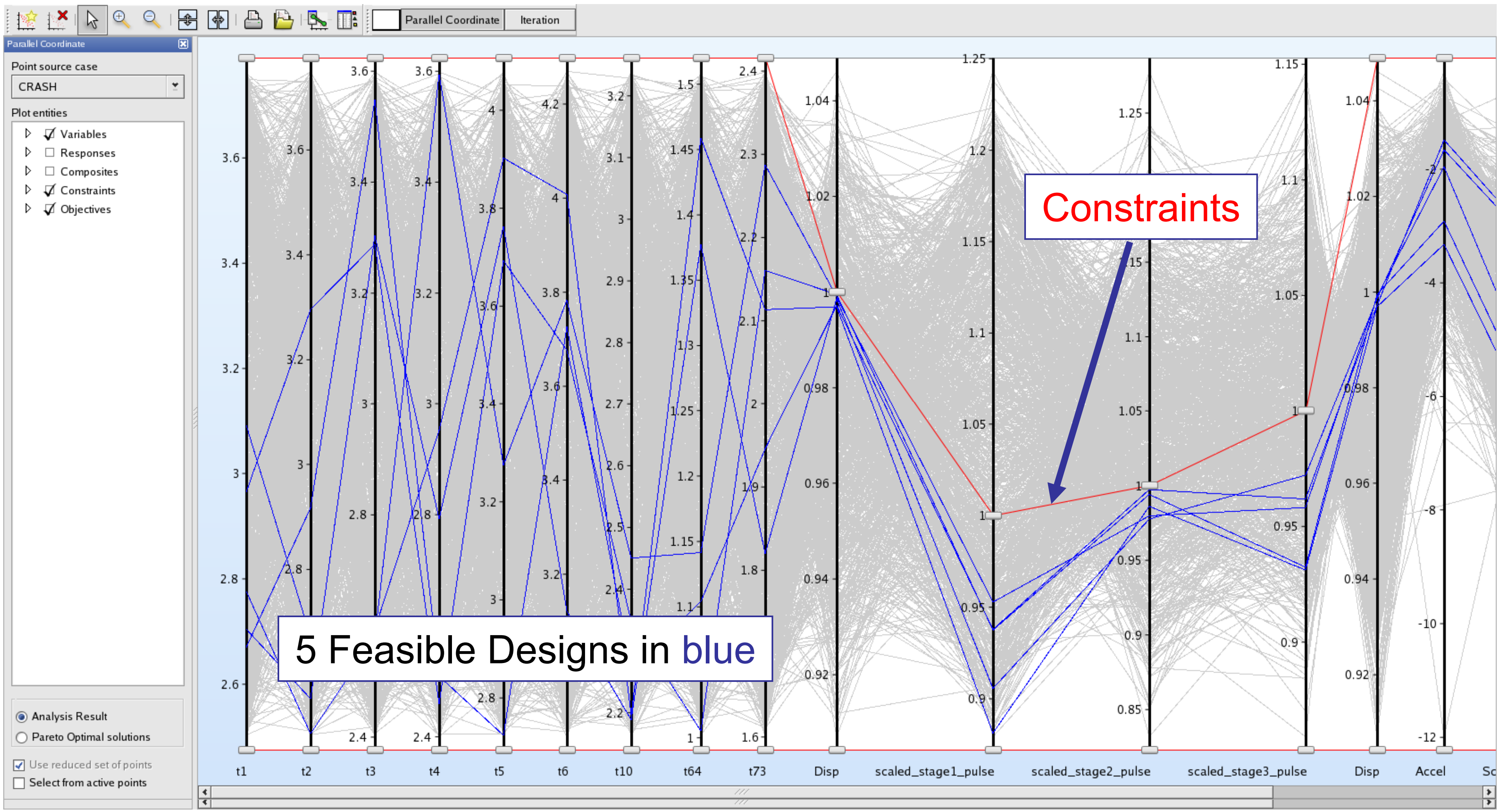
Simulation statistics

- 640-core HP XC cluster (Intel Xeon 5365 80 nodes of 2 quad-core)*
- Queuing through LSF
- Total of **1000** crash runs

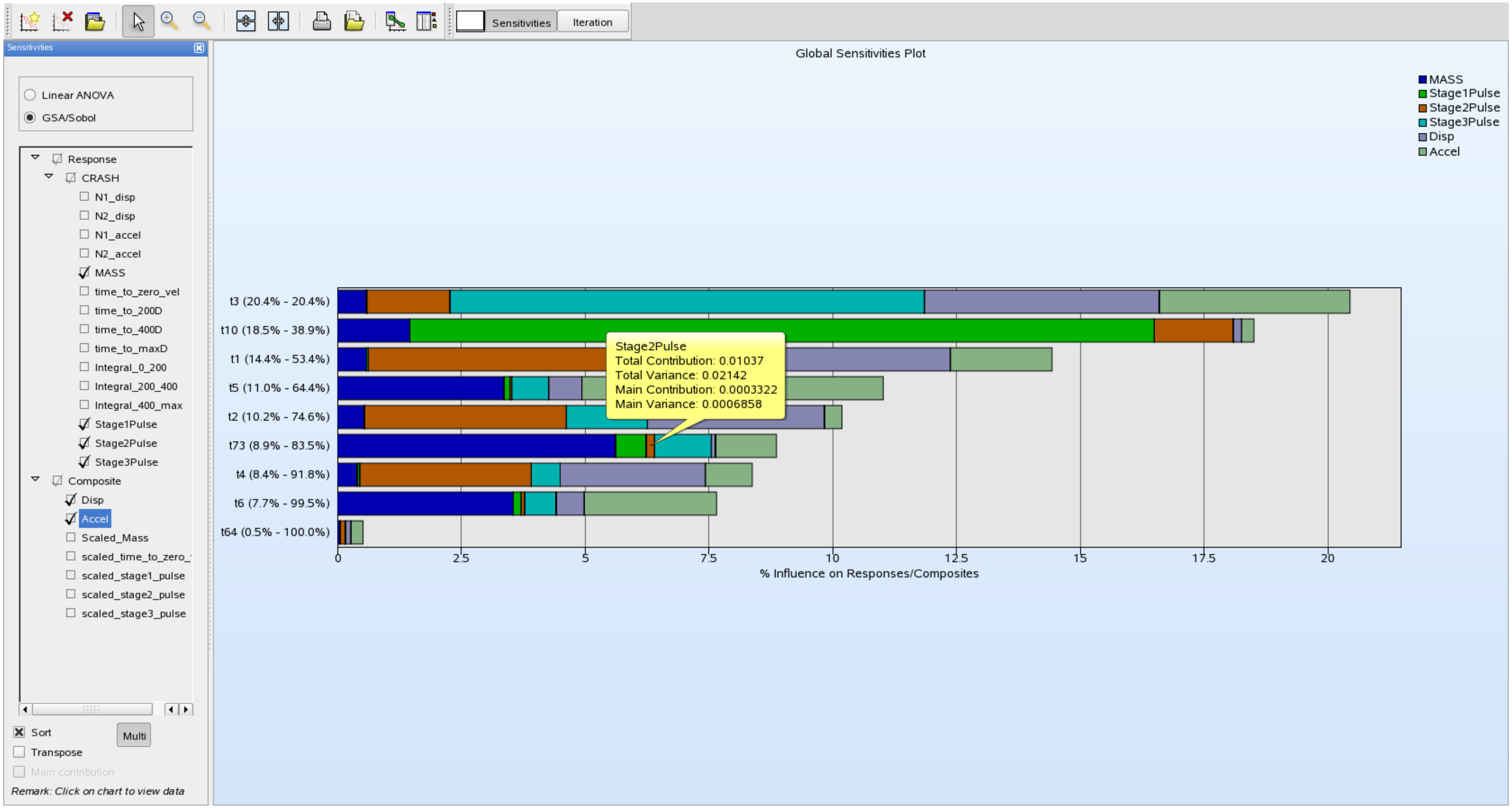
- Strategy: Single stage run
- Sampling scheme: *Space Filling* (MinMax distance) using 1000 points
- Metamodel: *Radial Basis Function Network*
- Optimization solver: *NSGA-II* to find *Pareto Optimal Frontier*

** In collaboration with Yih-Yih Lin Hewlett-Packard Company*

Parallel Coordinate Plot: 1000 Simulations

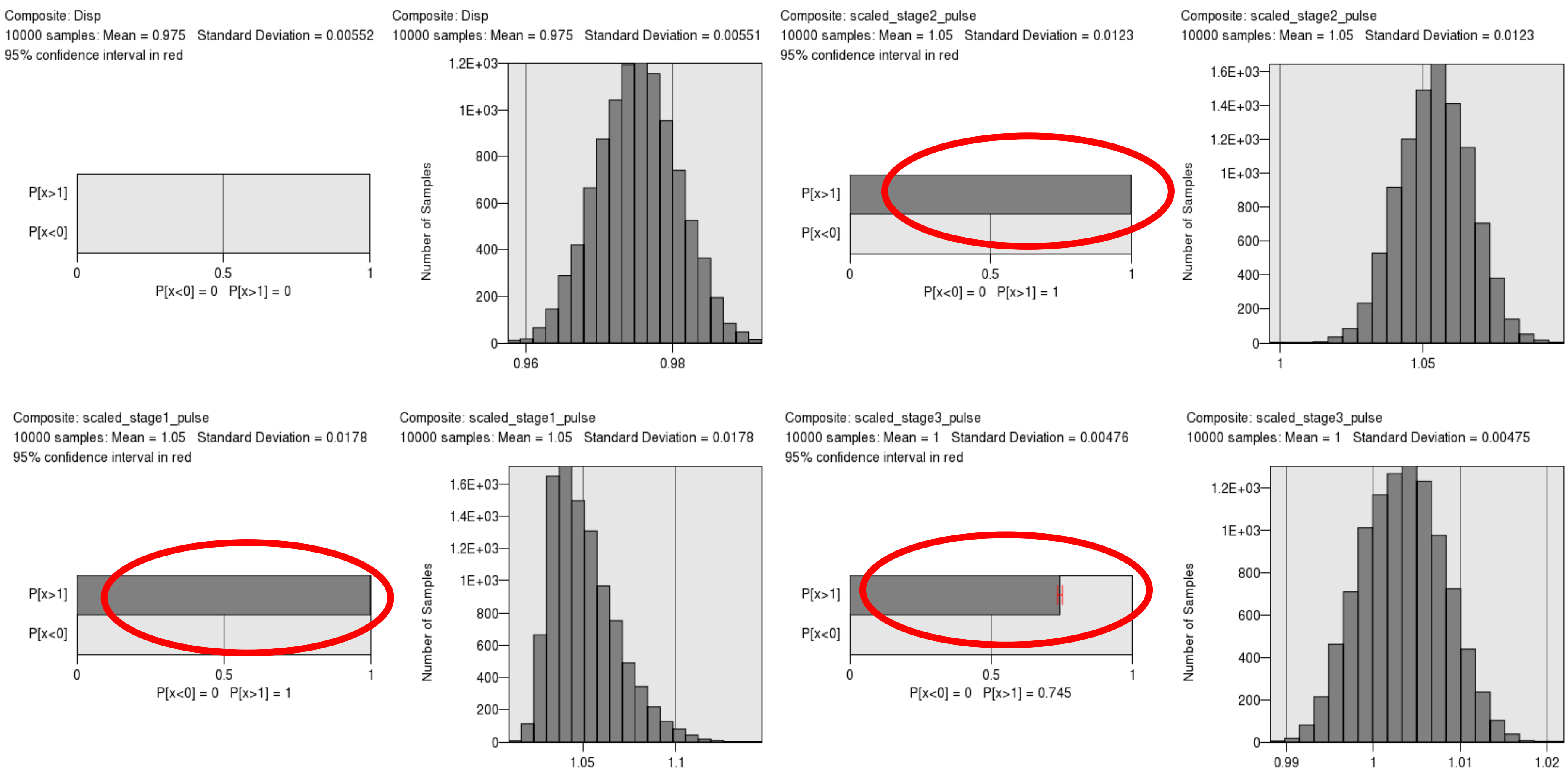


Global Sensitivity Analysis (Sobol indices)



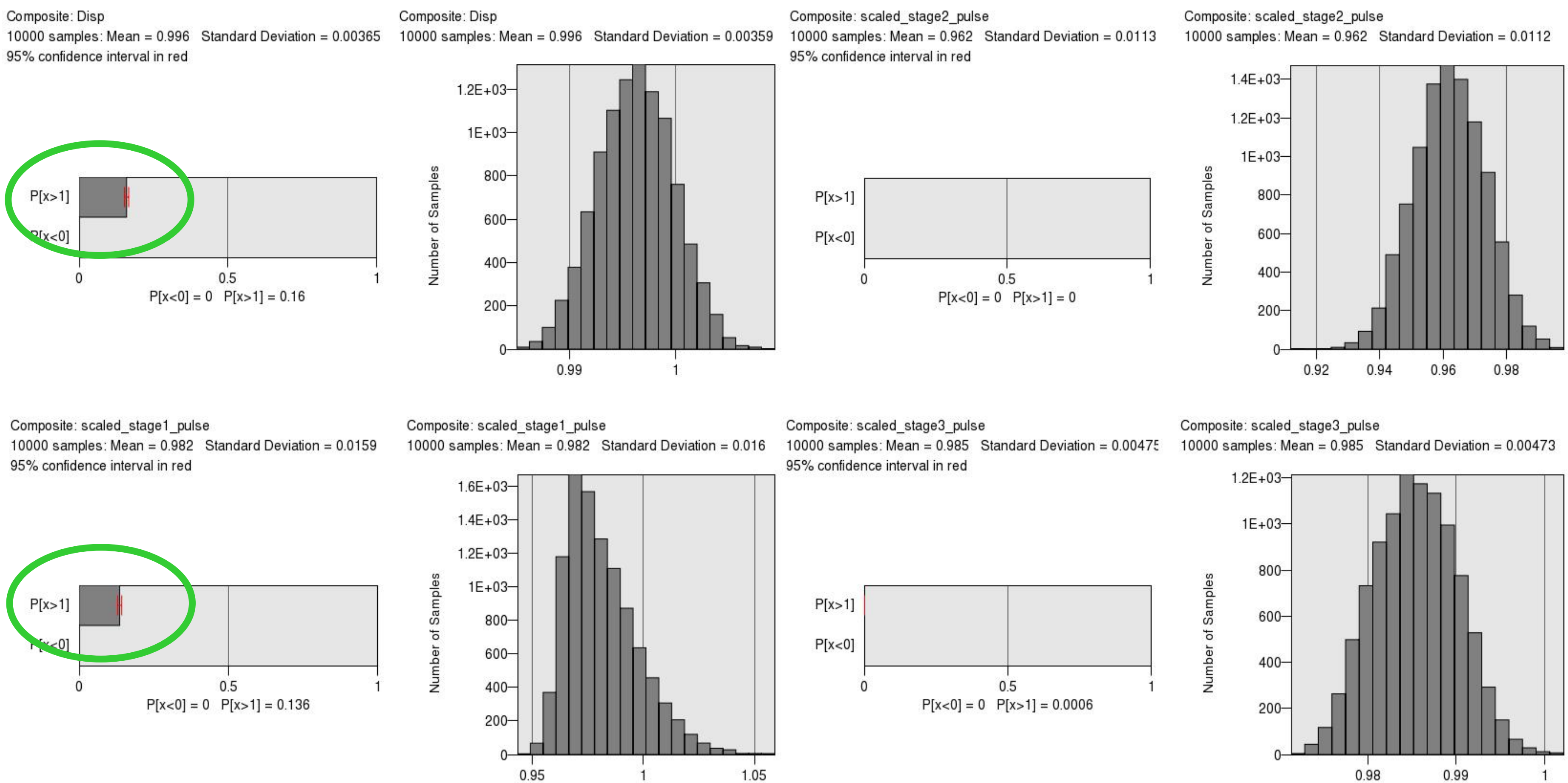
Probability distributions of constraint values

Starting Design (infeasible)

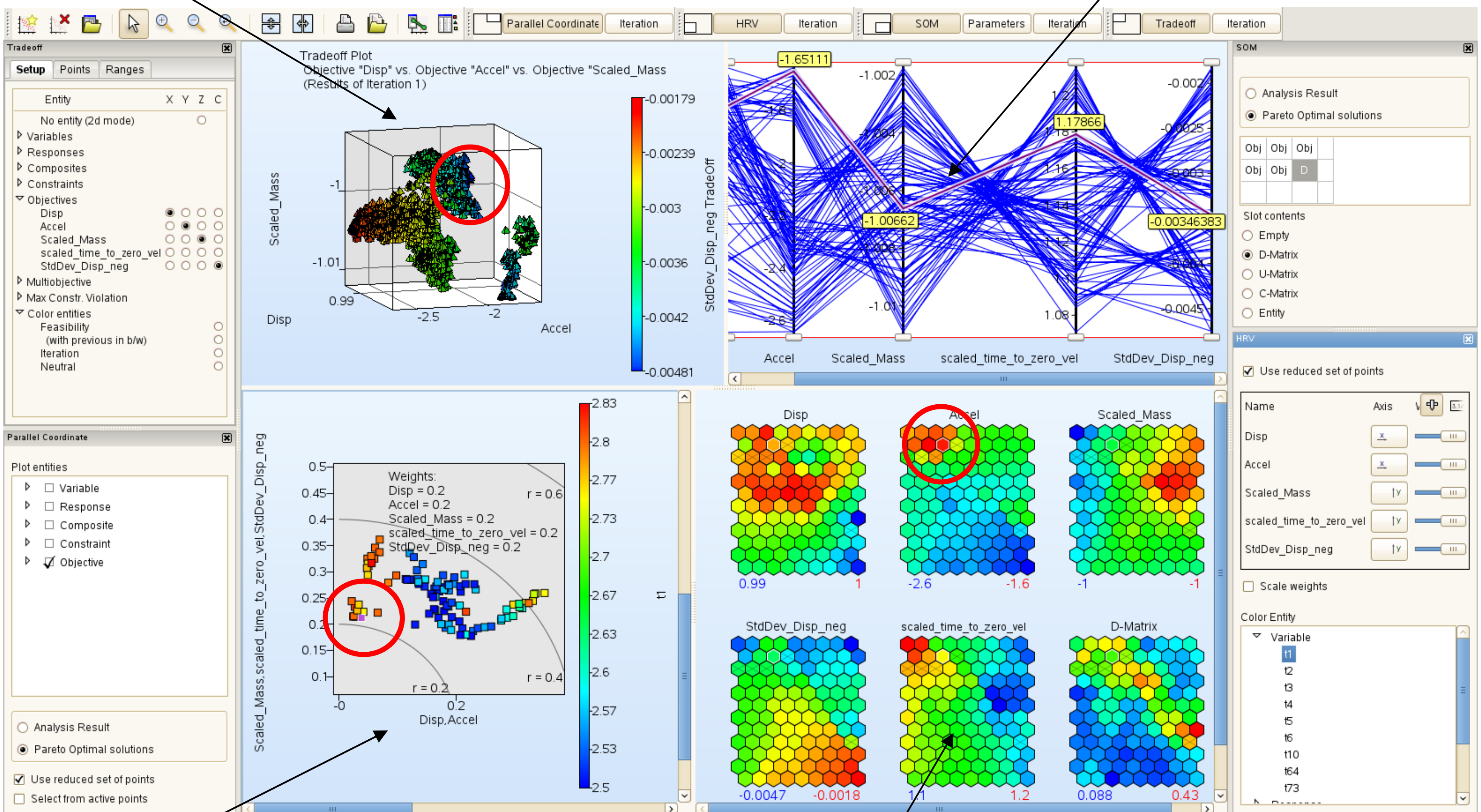


Probability distributions of constraint values

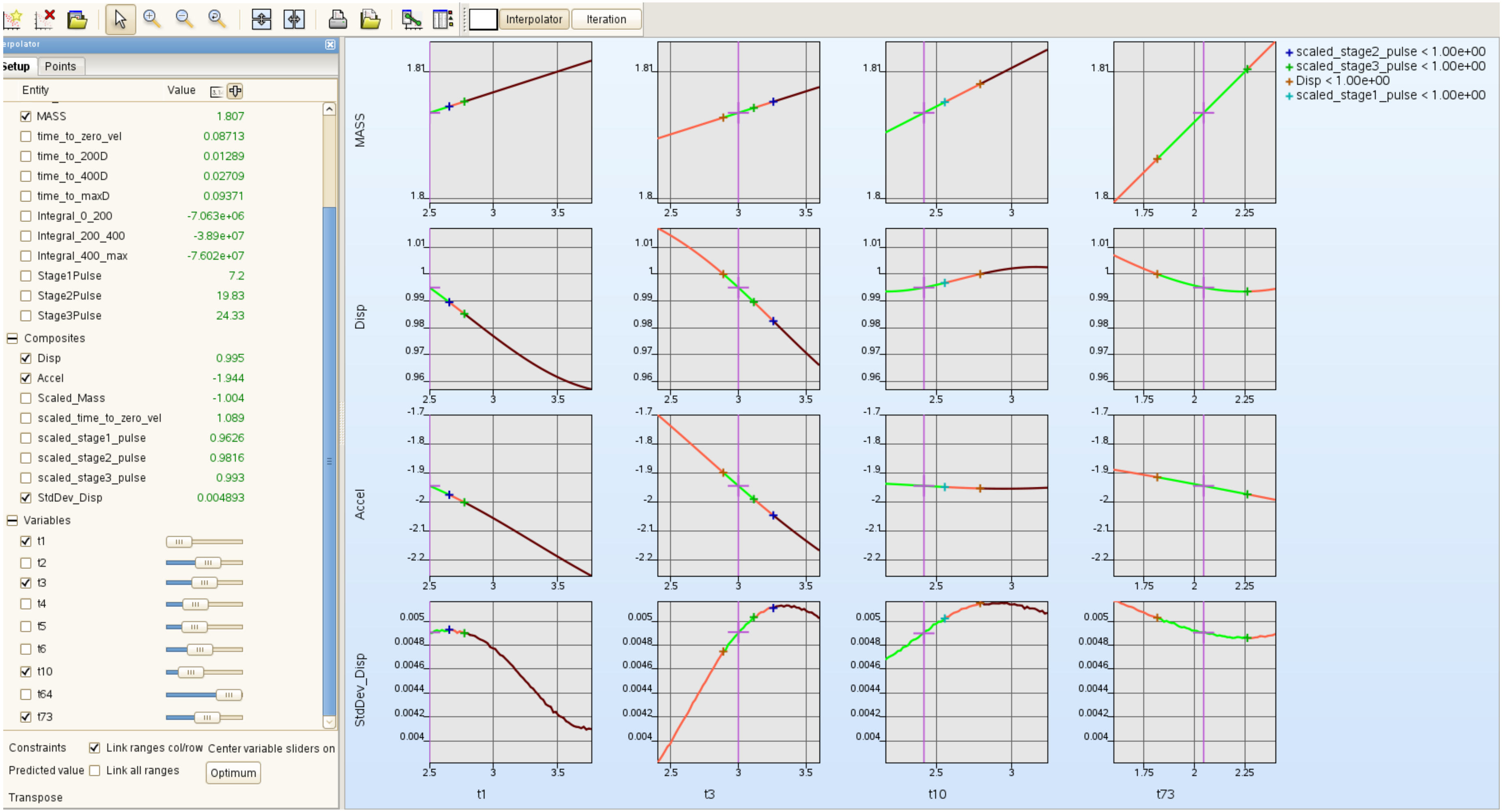
Optimal Design (equal weights)



Integrated Pareto Front Exploration

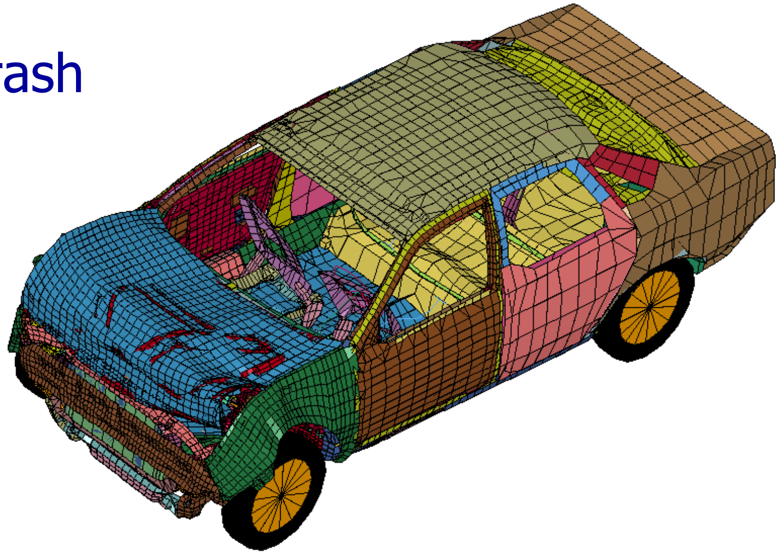


2D sections of the design response



Example: Direct MDO/MOO

Crash

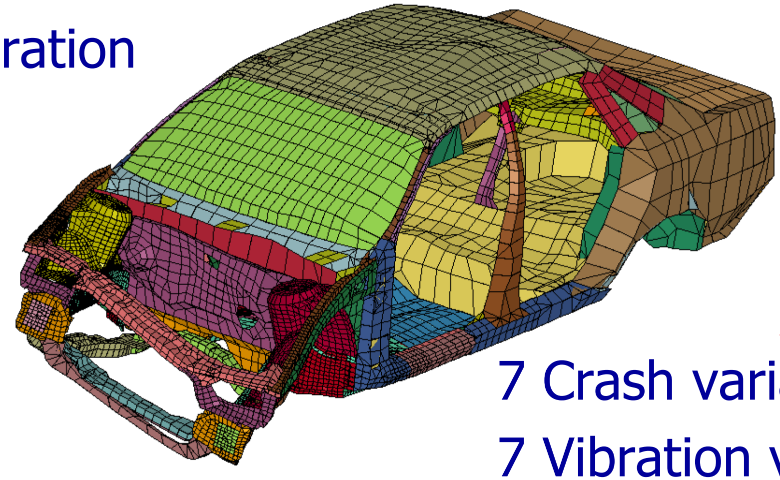


Min. (**Mass**, **Intrusion**)

Subject to:

- Intrusion ≤ 551mm
- Stage 1 pulse > 14.5*g*
- Stage 2 pulse > 17.6*g*
- Stage 3 pulse > 20.7*g*
- 41.38Hz ≤ *freq* ≤ 42.38Hz

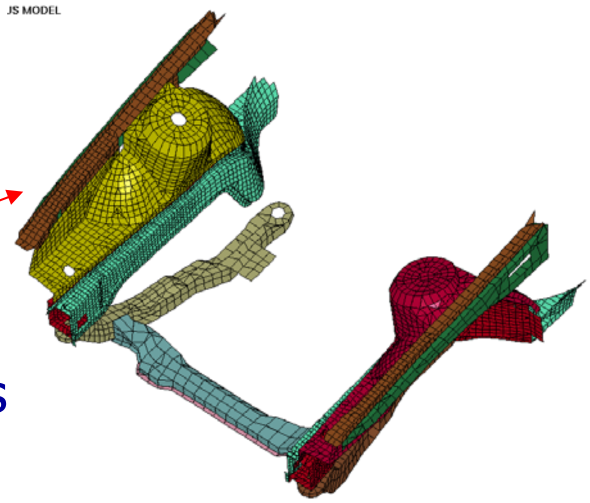
Vibration



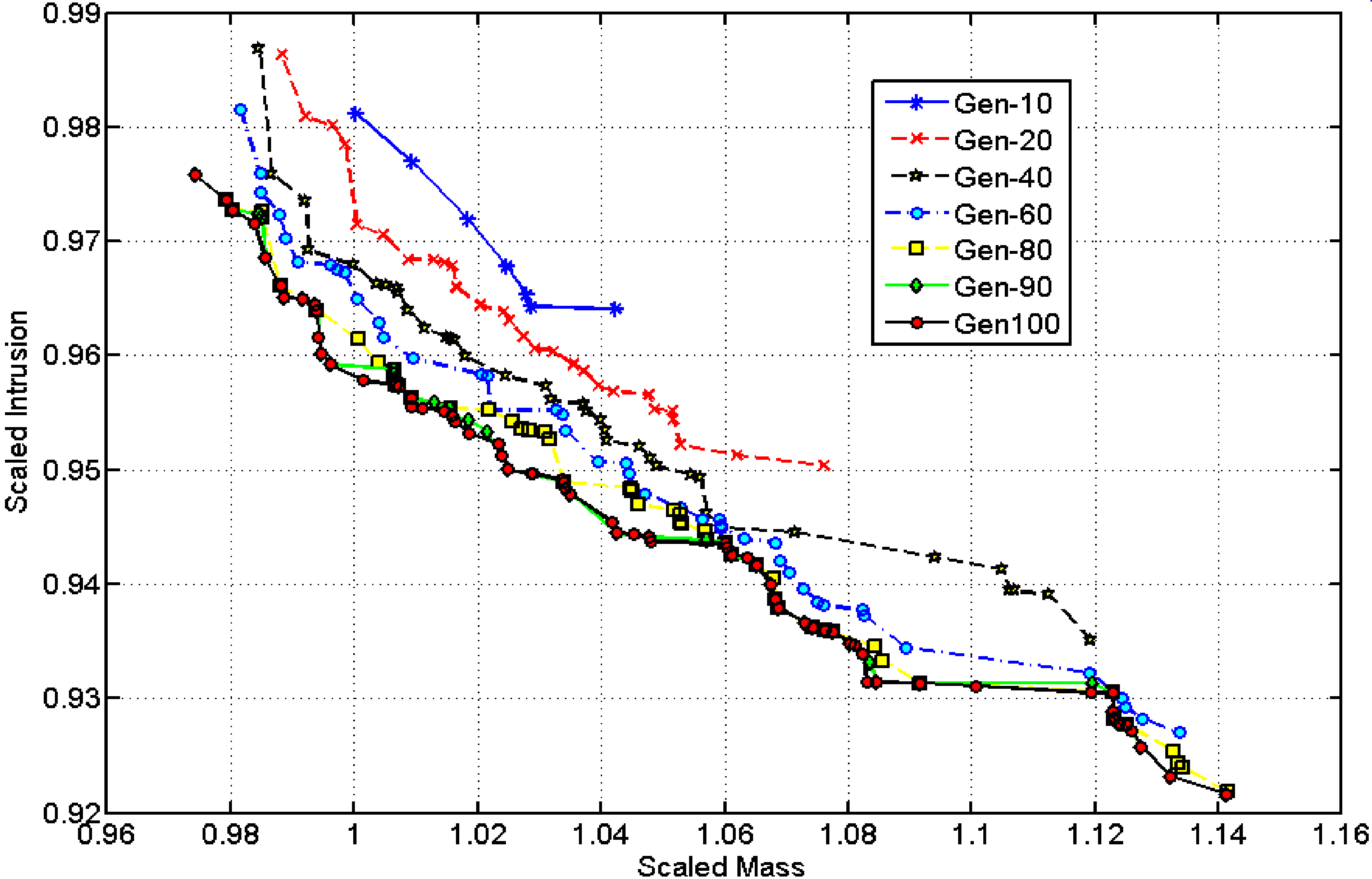
7 Crash variables

7 Vibration variables

(2 discrete)



Direct MDO/MOO: Pareto Optimal Front History

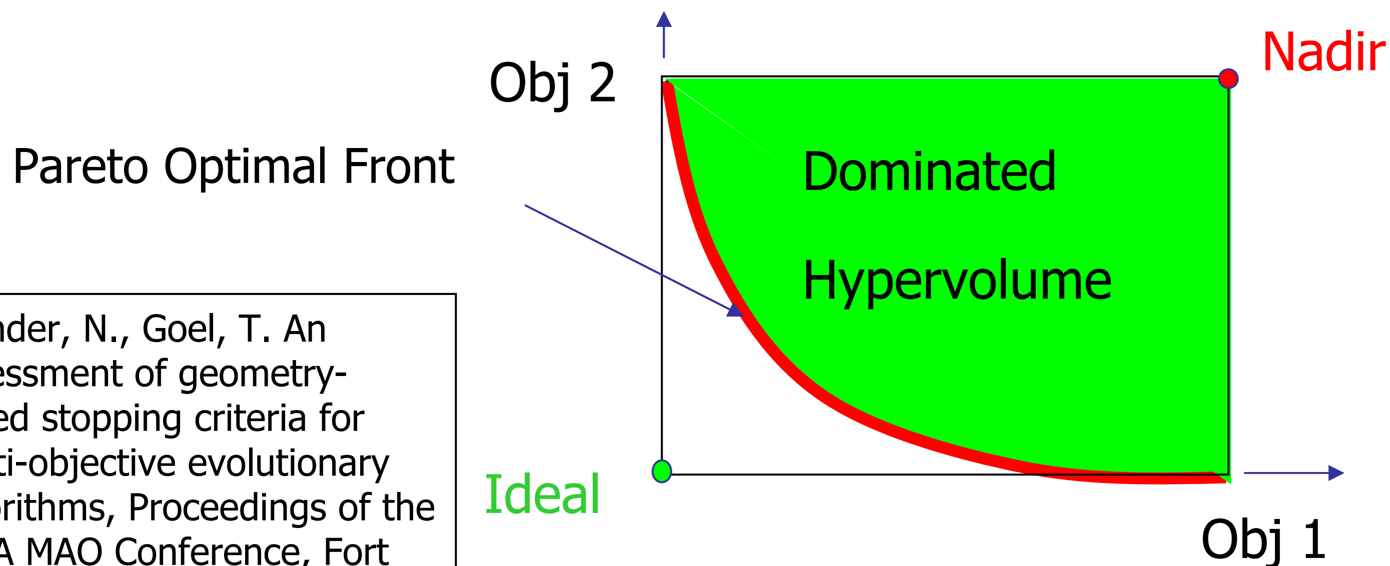


Li G, Goel T, Stander N, Assessing the convergence properties of NSGA-II for direct crashworthiness optimization, 10th International LS-Dyna Conference, Jun 8-10, 2008, Detroit, MI.

Direct MOO Convergence Metrics (v4.2)

Dominated Hypervolume

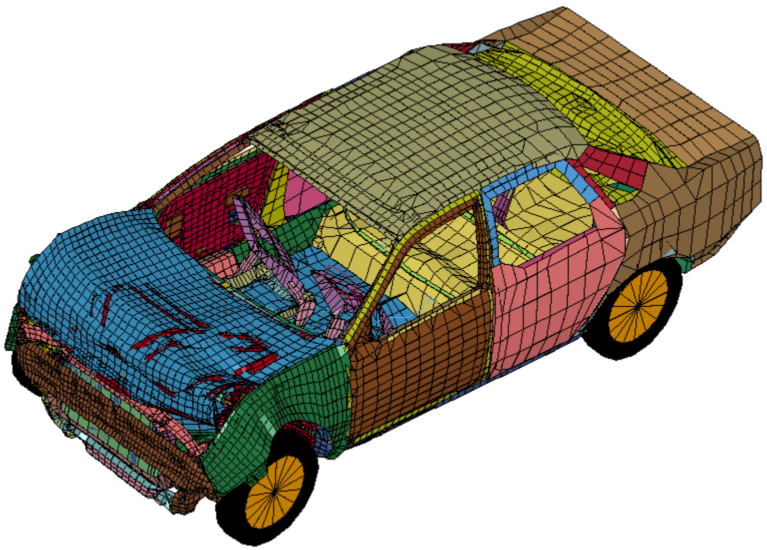
- Measure the volume of the dominated portion of the objective space with respect to a reference point.
- In this study use the Nadir vector as reference point
- Hypercube between Ideal vector and Nadir vector is normalized
- HSO (Hypervolume by slicing objectives) algorithm used to compute volume efficiently. While *et al* (2005).



Stander, N., Goel, T. An assessment of geometry-based stopping criteria for multi-objective evolutionary algorithms, Proceedings of the AIAA MAO Conference, Fort Worth, Texas, Sep. 2010

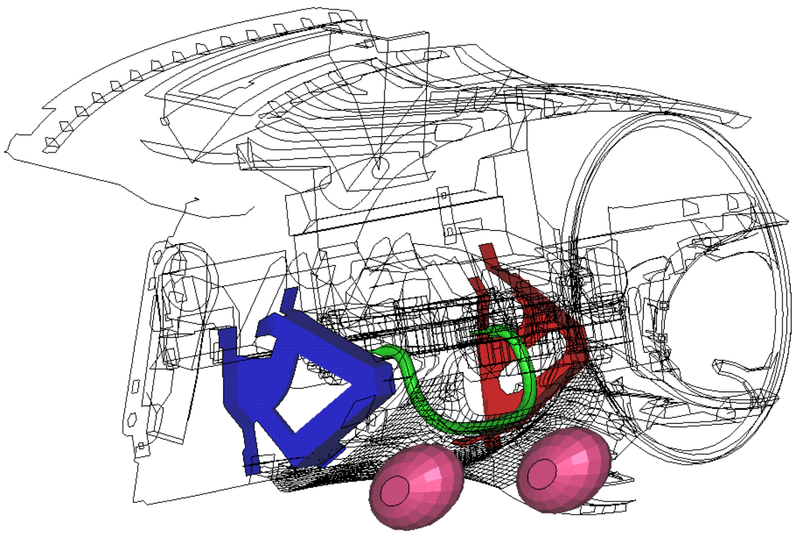
Direct MOO Convergence: Test Problems

MDO frontal crash/vibration



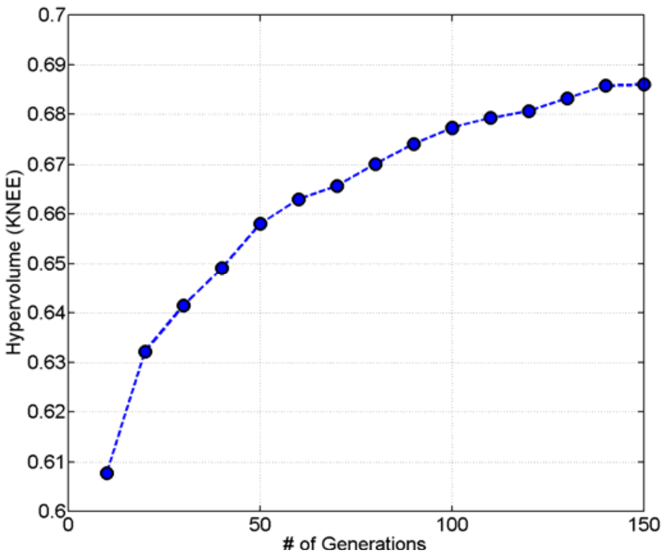
Frontal crash of a NHTSA vehicle
7 variables, 6 responses
30K+ elements
18K+ elements for modal analysis
90ms crash

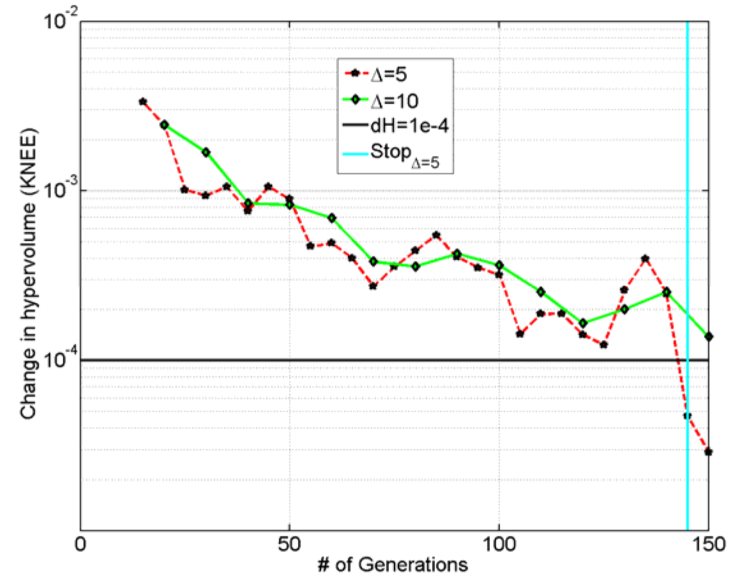
Knee impact

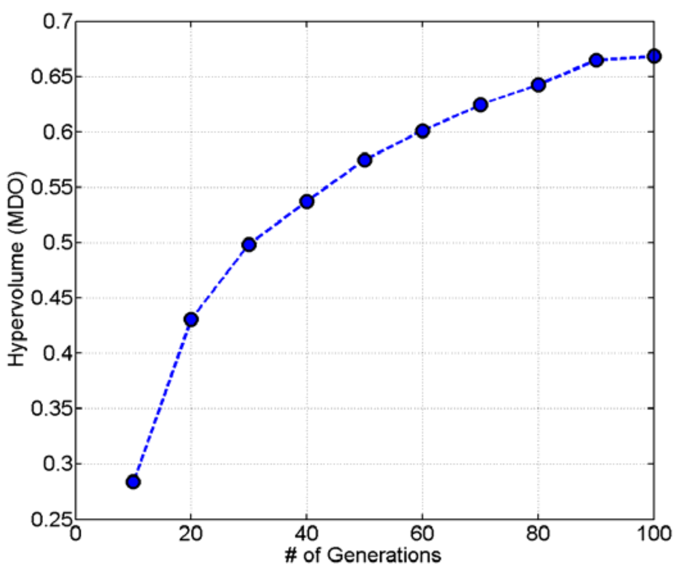


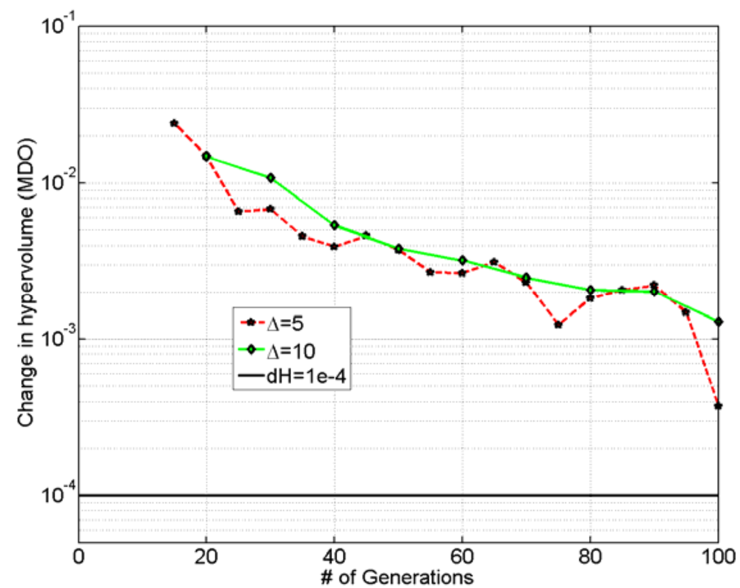
Automotive panel impact with knee
11 variables, 7 responses
25K+ elements
400 ms crash

Hypervolume and Change in hypervolume









Injury Criteria

- HIC (Head Injury Criterion)
- VC (Viscous Criterion)
- Chest Compression
- A3ms (Acceleration level for 3ms)
- Clip3m
- Clip3m (3nodes)
- Deformation/intrusion in local coordinates (to be merged into v4.1)
- MOC (Total Moment about Occipital Condyle)*
- Nij (Normalized Neck Injury Criterion)*
- NIC (Neck Injury Criterion)*
- Nkm (Neck criteria)*
- LNL (Lower Neck Load)*
- TTI (Thoracic Trauma Index)*
- TI (Tibia Index)*
- MTO (Total Moment)*

** Available in Version 4.2*

Reference: Crash Analysis Criteria Description, Arbeitskreis Meßdatenverarbeitung Fahrzeugsicherheit, Mai 2008

Frequency/Mode Tracking

- NASTRAN Frequency with Mode tracking
  - Modal Assurance Criterion (MAC)
    - Use correlation coefficient to match eigenvector
  - Orthogonality criterion:

$$\max_i [(\varphi_r^T \mathbf{M}_r)\varphi_i]$$

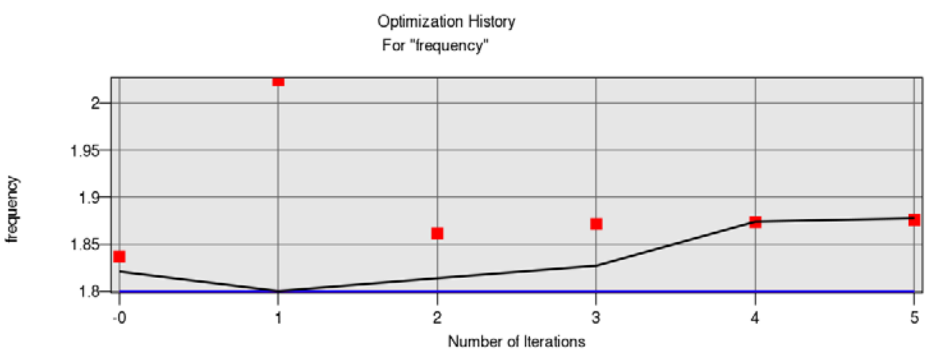
r the reference mode

- Industry tested in a multidisciplinary automotive setting

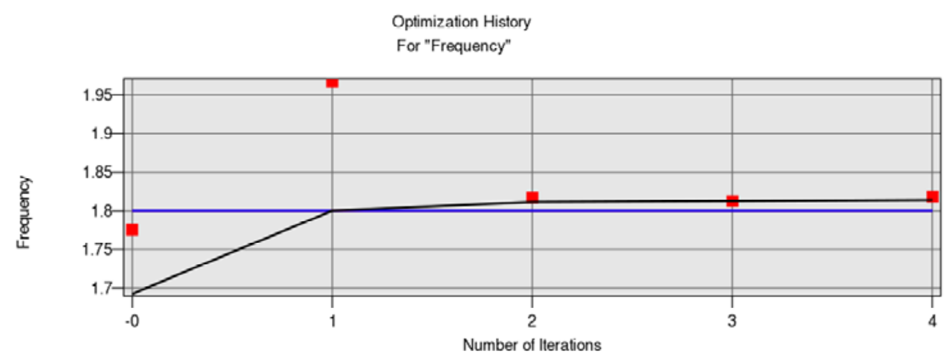
Frequency/Mode Tracking

Modes corresponding to a twisting frequency

NASTRAN



LS-DYNA



Pre/Postprocessors

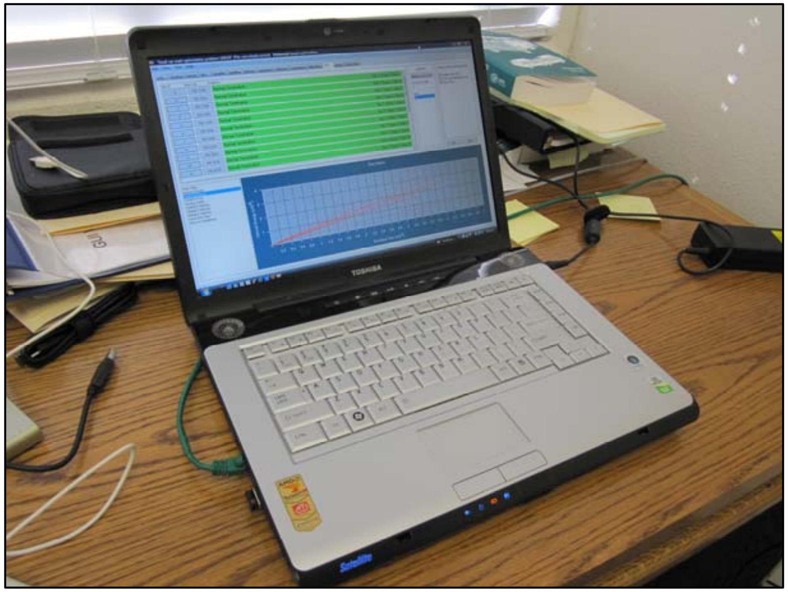
- Morphing
  - ANSA (BETA CAE Systems SA)
  - DEP Meshworks (v4.2)
- Post-processing
  - MetaPOST (BETA CAE Systems SA)
  - GenEx (LS-OPT)
    - Generic Text file result extraction
    - Vector/history extraction (v4.2)

Simulation job distribution with LSTCVM

LSTCVM Secure Proxy Server

Linux

MS Windows



Secure connection to cluster: **LSTCVM** Proxy Server

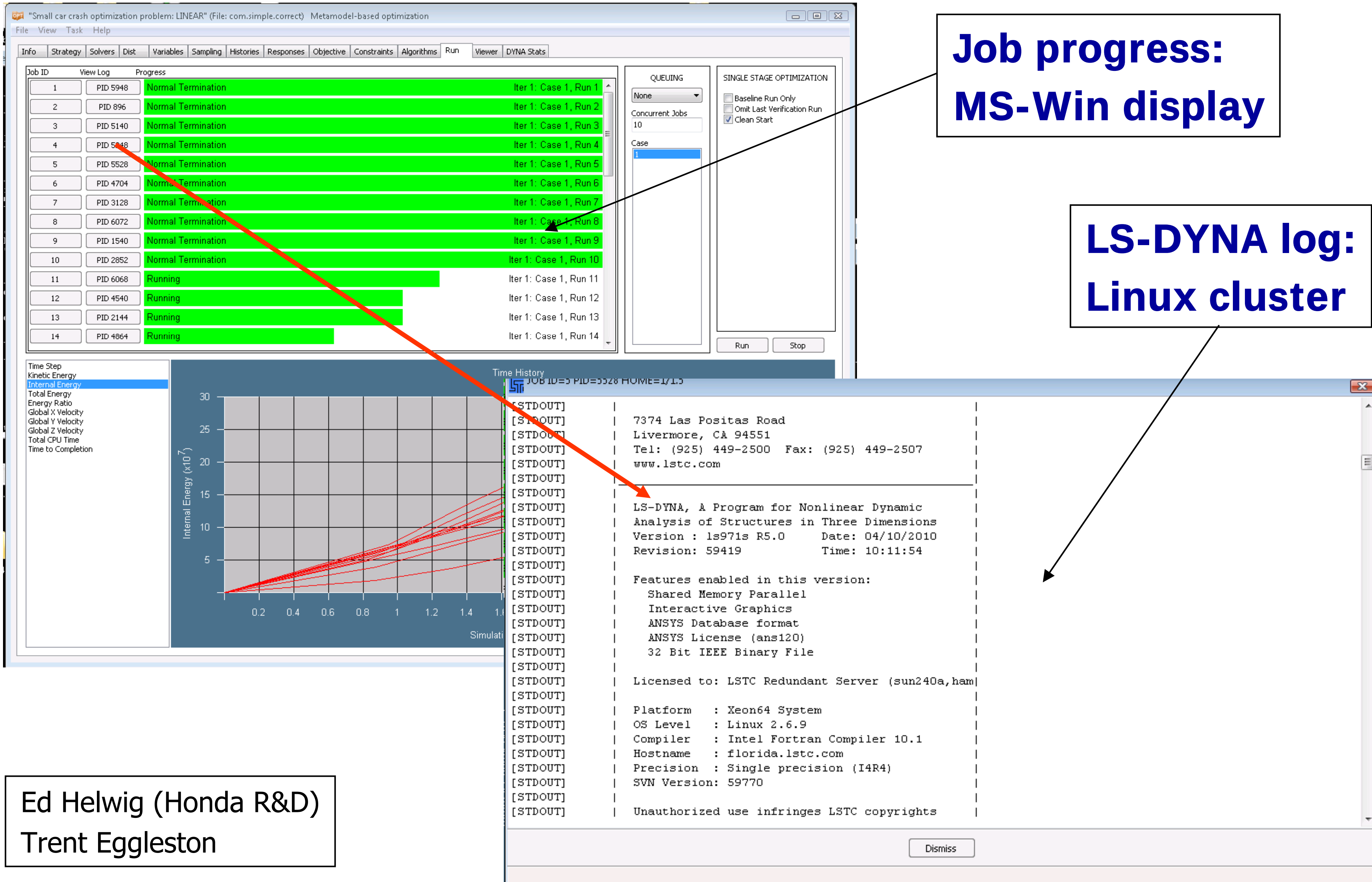
- Popular execution mode: LS-OPT on Windows controlling/monitoring LS-DYNA on a Linux cluster
- LSTCVM avoids security risks associated with rsh/ssh
  - Administrator sets up restrictions:
    - allowable commands
    - allowable locations
    - allowable users
    - no interactivity
  - No login required → no passwords transmitted
- File system can be shared or not (latter requires LS-OPT/*wrapper* executable for transmission)
- Interfaces to queuing systems
- Available with v4.1 (current production version)

Ed Helwig (Honda R&D)
Trent Eggleston

LSTCVM: Secure job proxy

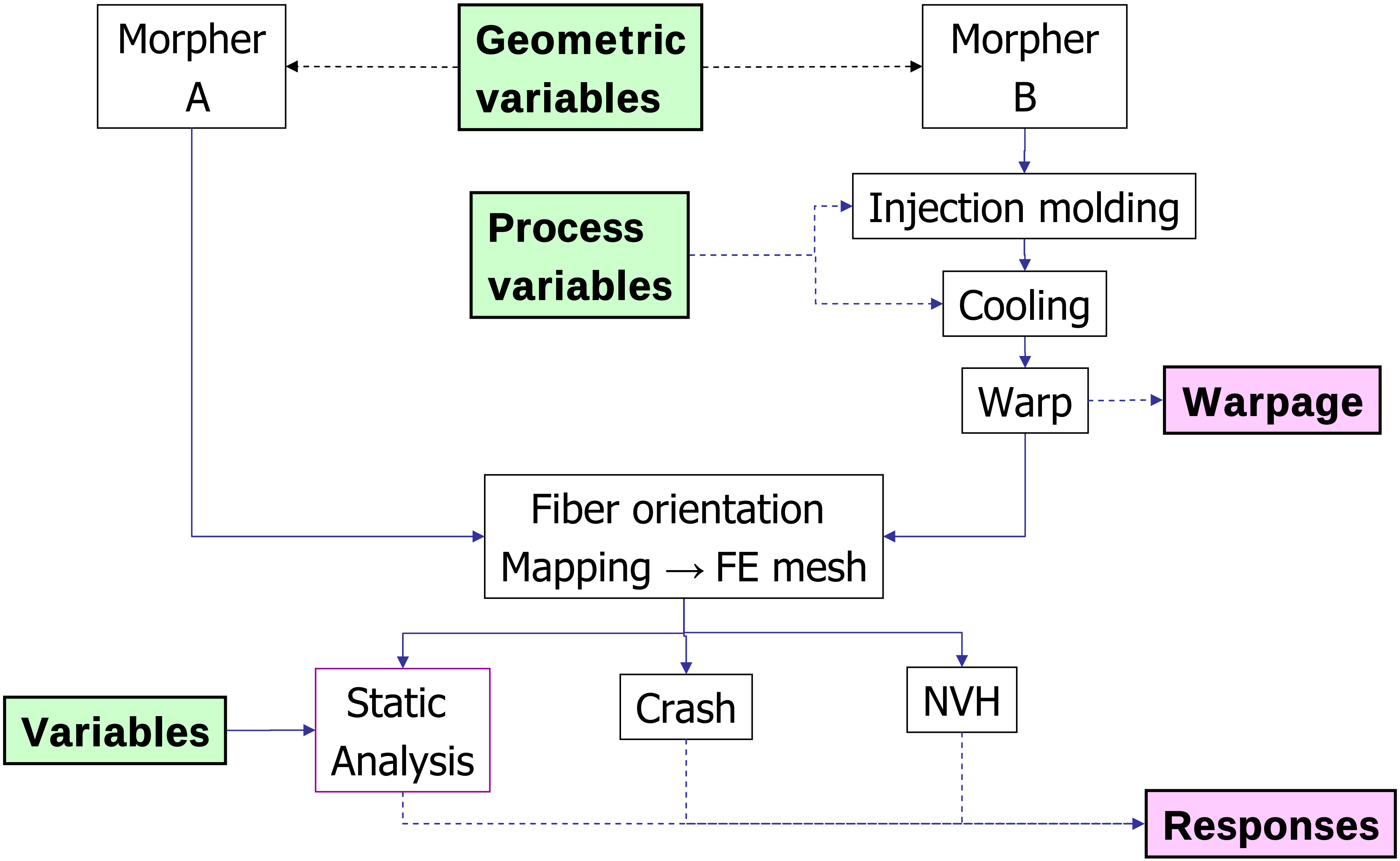
Job progress: MS-Win display

LS-DYNA log: Linux cluster



Ed Helwig (Honda R&D)
Trent Eggleston

Process Modeling (v4.2, v5.0)



Outlook: Process modeling features

- File handling
  - Copying, Moving, Saving, Deleting, Renaming
- Job scheduling
  - Load balancing – allow concurrent jobs where possible
- Enhanced usability
  - Stepping capability, enhancements to repair feature
- Backward compatibility

Process Modeling

- V4.2 : Limited functionality (process definition, load balancing, file handling) based on an extension of the current GUI – Spring 2011
- V5.0 : Redesigned GUI – Winter 2011

Thank you for your attention!