



Nondestructive Evaluation Sciences Branch

Enabling Enhanced Validation of Non-Destructive Evaluation Computational Models and Simulations

William C. Schneck, III Ph.D.

NASA Langley Research Center

Nondestructive Evaluation Sciences Branch

william.c.schneck.@nasa.gov



Introduction

- Computer simulations of physical processes are increasingly used in the development, design, deployment, and life-cycle maintenance of many engineering systems
 - This is especially true in aeronautics
 - Aerodynamics (Computational Fluid Dynamics)
 - Aerostructures (Computational Structures and Mechanics)
 - Non-Destructive Evaluation (NDE) is a multidisciplinary field developing technologies to evaluate material system states to detect and characterize defects and degradation in materials
 - NDE capabilities need to keep pace with increasingly sophisticated material systems
 - Flaw characteristics become more complex and subtle
 - Experimental exploration of those flaws becomes cost and time prohibitive
 - Computer simulations enable timely and cost-effective exploration of highly dimensional problems
 - These simulations must be validated to assure reliable use for relevant NDE applications
- Objective: briefly cover Verification and Validation (V&V) and provide an NDE perspective towards the development of V&V processes and guidelines for NDE simulations.



Motivation

Nondestructive Evaluation Sciences Branch

- NASA missions require safety assurance of critical parts
 - NDE is a critical field to ensuring flight system safety
- Some current key questions are:
 - Can we design for inspection?
 - Can we detect and characterize part flaws?
- Current efforts at NASA involve modeling POD for friction stir welds (FSW) in the Space Launch System
 - FSW can have very complex flaws entrained in the weld



- Design for Inspection:
 - Designer must be able to anticipate inspection behavior
- Detect/Characterize part flaws:
 - Models enable more exploration of parametric spaces to better understand flaw effects on inspection phenomena

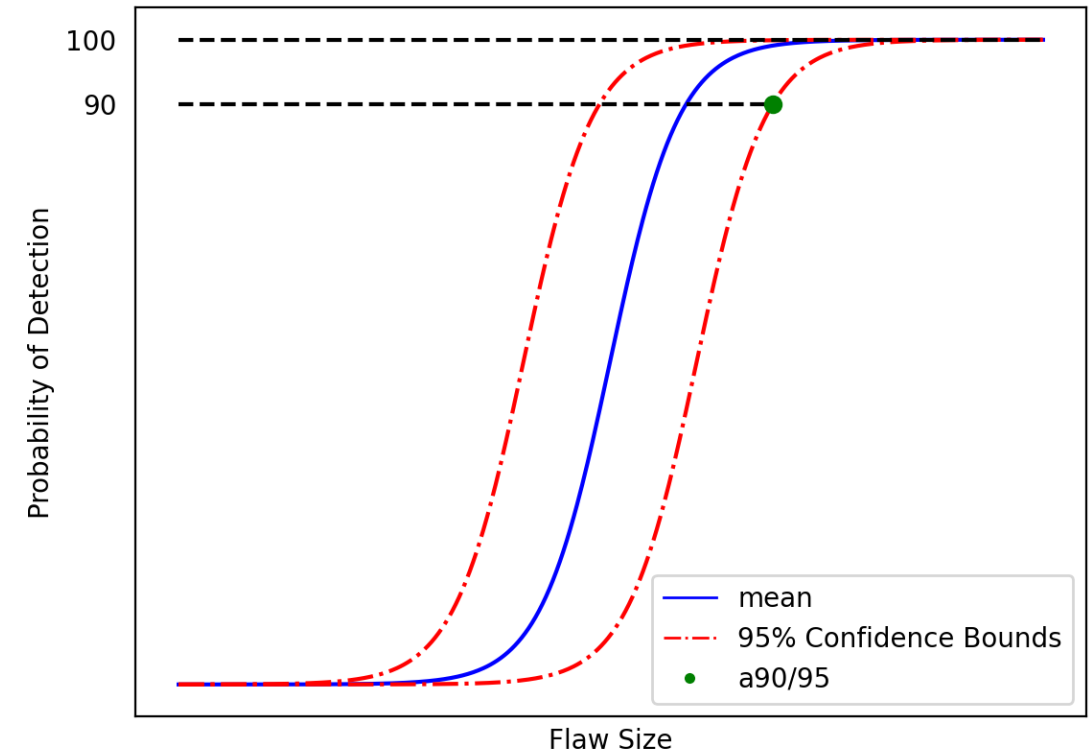




Probability of Detection

Nondestructive Evaluation Sciences Branch

- Probability of Detection (POD) is the likelihood of an inspection system (including the operator) detecting a given flaw
- POD uses either Hit/Miss data or Signal/Response (a vs $\hat{a}$) data
- Sources of signal variability:
 - Crack-to-crack variations (usually the most prominent)
 - Inspector-to-inspector variations
 - System-to-system variations





Model-Assisted Probability of Detection

Nondestructive Evaluation Sciences Branch

- As materials and their associated damage modalities become more complex, achieving sufficient experimental coverage becomes prohibitively expensive
 - Validated models can be used to backfill the data to construct meaningful POD curves
- This requires a robust Verification and Validation framework to determine model acceptability for any given application!



Definitions

- Model
 - *The conceptual, mathematical, and numerical representations of the physical phenomena needed to represent specific real-world conditions and scenarios*
 - “How do I think this system works?”
- Verification
 - *The process of determining that a model implementation accurately represents the developer’s conceptual description of the model and the solution to the model*
 - “Did I solve the model correctly?”
- Validation
 - *The process of determining the degree to which a model is an accurate representation of the real world from the **perspective of the intended use of the model** [emphasis mine]*
 - “Did I solve the correct model?”



Verification and Validation

Nondestructive Evaluation Sciences Branch

Verification

- Ask: Is the model solving correctly?
- Does not:
 - Determine if this is the correct model
 - Determine if the model matches reality
- Two major tasks:
 - Confirm software operational correctness
 - Confirm algorithmic correctness

Validation

- Ask: Does the model agree with reality?
- Does not:
 - Assert predictive capability
 - Prove correctness
- Major task:
 - Raise level of trust in model to acceptable levels for use



Verification

Nondestructive Evaluation Sciences Branch

- Objective:
 - Generate evidence that the computer model accurately solves the conceptual mathematical model
- Additionally determine:
 - Numerical stability and any stability criteria
 - Numerical convergence
 - Model result uncertainties (uncertainty quantification)
- Techniques
 - Compare analytical solutions with model results
 - Method of manufactured solutions
 - Stability analysis
 - Von Neumann analysis
 - For any software development:
Software tests
 - Most verification tasks can be wrapped into software tests
 - Unit testing frameworks and automatic testing
 - Most git tools (e.g. GitHub or GitLab) provide some means to run automatic testing

Verification tests that the code is correct and that it accurately reproduces the desired mathematical behaviors:

it does not confirm that the model correctly reproduces reality.



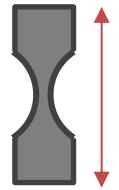
Uncertainty Quantification

Nondestructive Evaluation Sciences Branch

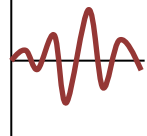
Model Parameters

Physical

Material properties, uniformity, domain dimensions



Forcing functions, excitations (timing and magnitudes)



Numerical

Precision

0.00

Grid Convergence/grid errors



Discretization order





Validation

- “All models are wrong; some are useful”
- Validation is a process that gathers evidence that the model does/does not represent reality
 - In general, no mathematical proof of validation exists; only accrued evidence that it is valid
 - The fidelity of the validation required depends on:
 - Project mission objectives
 - Anticipated operational envelope
 - Any safety and mission assurance requirements
 - Project budget and timeline concerns
- “Does my code match Reality?”
 - Comparison with laboratory testing is preferred
 - Cross-validation with analytical or other previously validated model can be acceptable where resources are constrained and safety is not a concern



Validation Ladder

- Approach from a “bottom-up” perspective
 - Start from unit problems, and work up
 - Use these laboratory data to validate the model

UNIT CASES:

- Simple systems usually with one key feature
- Maximized laboratory control
 - Dedicated V&V testing; no competing interests
- Simple system minimizes testing uncertainties
- Simple system maximizes sensor access for improved testing characterization

BENCHMARK CASES:

- Canonical systems usually with reduced key feature set (e.g. simplified but representative geometries)
- Maximized laboratory control
 - Dedicated V&V testing; no competing interests
- Increased sophistication may result in increased testing uncertainties

SUBSYSTEM CASES:

- Real component with feature set, but isolated from the rest of the full system
- Possibly reduced laboratory control
 - Expensive/limited production quantity hardware may make dedicated V&V testing intractable
 - Reduced access may make experimental characterization harder

COMPLETE SYSTEM:

- Real component in real application environment (possibly still installed, etc)
- Reduced laboratory control
 - Dedicated V&V testing unlikely due to costs
 - Reduced access may make experimental characterization harder even beyond the Subsystem testing



Nondestructive Evaluation Sciences Branch

Experimental Error Analysis and Design of Experiments Considerations

- Consider not just instrument error
- Part-to-part variations:
 - Material properties
 - Geometry
- Inspector/Experimentalist variations
 - Variations in skill/technique
 - “Tuning” of instrument controls

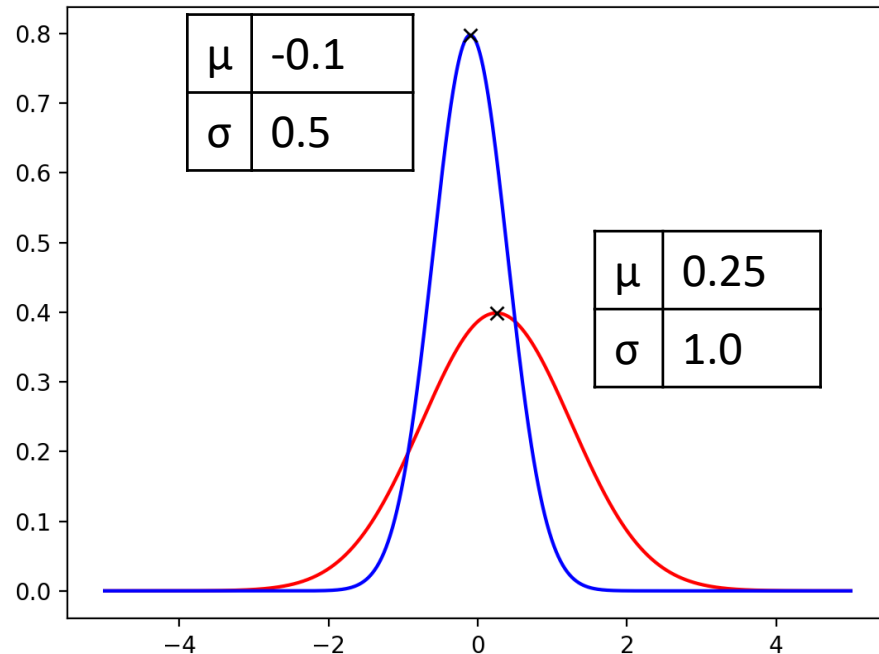


How Do Model Data and Test Data Compare

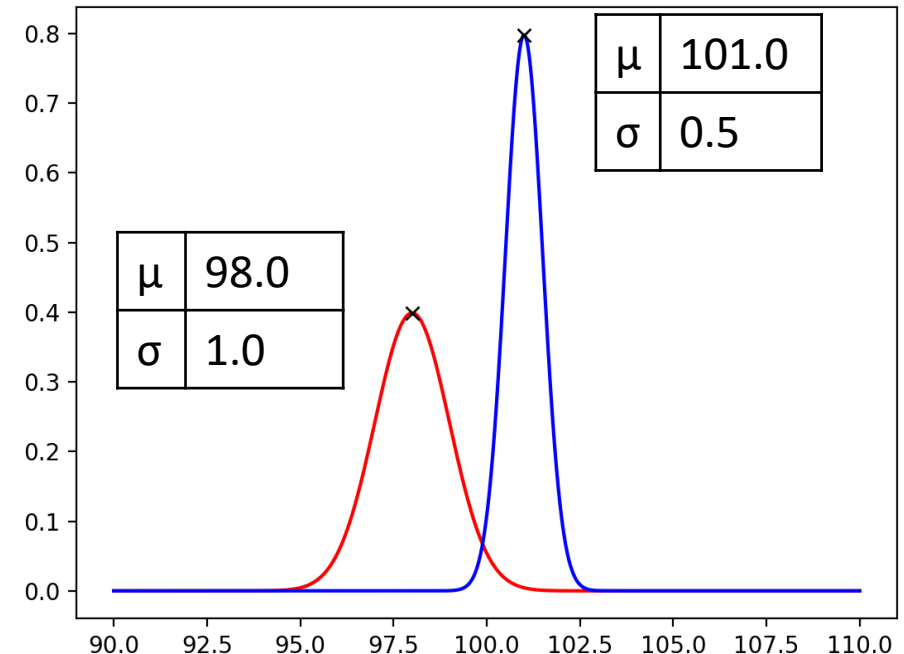
Nondestructive Evaluation Sciences Branch

- Percent difference between values may carry insufficient information about the agreement
 - Values near zero can have large relative differences
 - Values far from zero can have small relative differences
 - Consider the distribution of probable values

overlapped distributions, difference = 140 %



overlapped distributions, difference = -3.06122 %





Idea: Validation Repository

Nondestructive Evaluation Sciences Branch

- Validation requires significant amounts of quality data
- Modeling projects frequently lack the resources (e.g., time, budget, facilities, or expertise) to execute a sufficiently high-quality test campaign
- Other parallel efforts produce high quality data relevant to the modeling efforts
- Future non-model work may also find that access to the data is useful (and avoids unnecessary replication of work)
- Proposal: develop an archival system that enables future scientific or engineering teams to access validation-quality data
 - Data sets should be complete with metadata necessary to replicate the experiment in its entirety
 - Digitally or physically
 - Has knock-on benefits of better scientific stewardship
 - Enables wider dissemination and use of high-quality experimental data
 - There are historical analogues of varying levels of success



Summary

- Model verification and validation is a situational endeavor
 - There is no blanket statement of “This model is validated.”
 - Agreement between model results and laboratory results can be complicated
 - Consider the distributions of likely values, not just nominal
 - Uncertainty quantification becomes key
 - Algorithm verification should be performed using methods such as Method of Manufactured Solutions
 - Algorithm stability limits should be explored using methods such as von Neumann stability analysis
 - If custom developed code, much of the verification activities can be rolled into testing suites
- A repository of open-access experimental data could enhance model development for NDE applications



Nondestructive Evaluation Sciences Branch

Backup



How Do Model Data and Test Data Compare

Nondestructive Evaluation Sciences Branch

- Bayes' Factor

$$B = \frac{\Pr(y_D|H_0)}{\Pr(y_D|H_1)}$$

B = Bayes' factor

H_0 = hypothesis (model)

H_1 = alternative hypothesis

y_D = observed data

If $B > 1$; the data support the model over the alternative.
The larger the Bayes' factor, the stronger the support

overlapped distributions, difference = 140 %

