Role of uncertainty quantification in validating physics simulation codes

Ken Hanson

T-16, Nuclear Physics; Theoretical Division
Los Alamos National Laboratory

Computational Methods in Transport Workshop
Lake Tahoe, September 9-14, 2006

This presentation available at
http://www.lanl.gov/home/kmh/

LA-UR-06-6221

Overview

- Validation and uncertainty quantification (UQ)
- Bayesian analysis
- Techniques for forward and inverse probability calculations
- Implications for simulation codes
- Examples
  - metal plasticity
  - neutron cross sections and criticality
  - inconsistent cross-section measurements
- Advanced Bayesian analysis

Validation

- Validation of physics simulation code –
  goal is to determine how well the code reproduces actual physical behavior in a specified application
- Uncertainty quantification (UQ) determines 'how well'

- Not mentioned, but important:
  - operating range of physical conditions
  - uncertainties in initial and boundary conditions of experiment
  - range of applicability
  - code user's experience and credentials

Bayesian analysis provides means for UQ

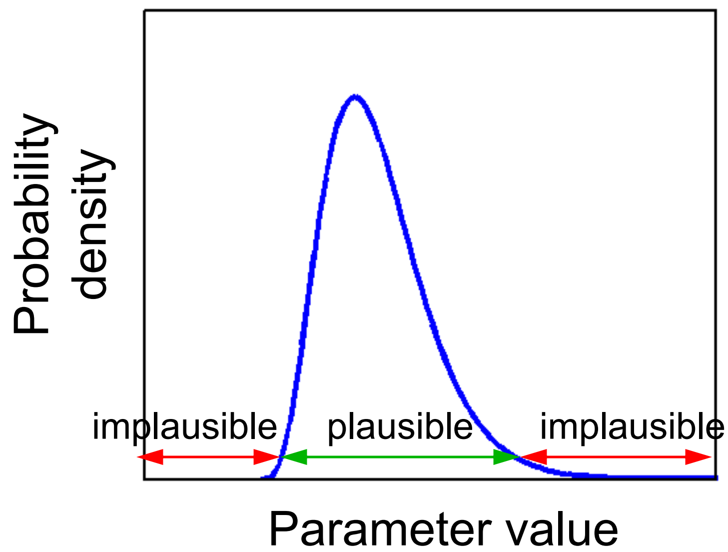
- Bayesian approach to analysis
  - focus is on uncertainties in parameters, as much as on their best (estimated) value
  - supports scientific method
  - model-based
  - experimental evidence should play decisive role
  - permits use of prior knowledge, e.g., previous experiments, modeling expertise, physics constraints

- Goal is to estimate
  - model parameters and their **uncertainties**
  - predictive accuracy of models

Uncertainties and probabilities

- Bayesian view – uncertainties in parameters are characterized by probability density functions (**pdf**)
- Probability interpreted as quantitative measure of "**degree of belief**"
- This interpretation is referred to as "subjective probability"
  - different for different people with different knowledge
  - changes with time
  - in science, we to try avoid bias, seek consensus
- Rules of classical probability theory apply
  - provides mathematical rigor and consistency

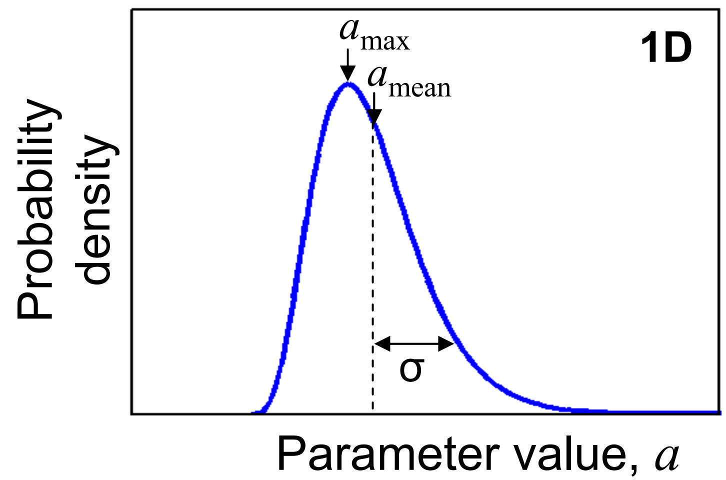
Probability density function



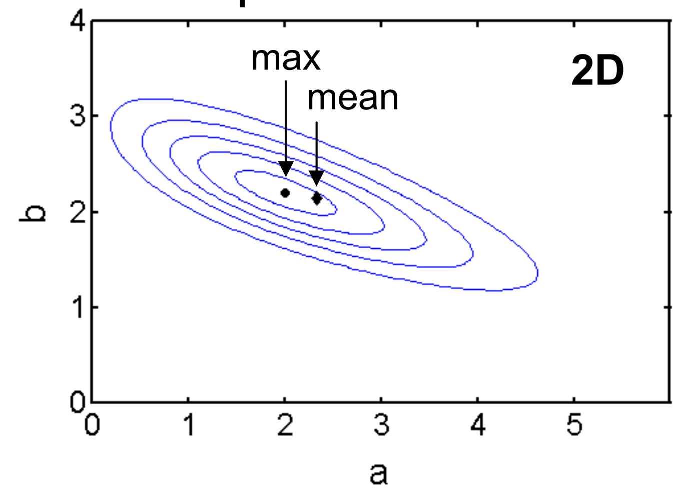
Parameter estimates and uncertainty

- Estimated value of parameter is often taken as
  - position of maximum (MAP) or
  - mean value (preferred estimator)
- Uncertainties characterized by rms deviation of pdf σ, called standard error; variance = σ^2
- In two or more dimensions, we must pay attention to
  - correlations
    - indicated by tilt in contour
  - marginalization over nuisance variables
    - project pdf onto variables of interest

Probability density function



pdf contours



Rules of probability

- Continuous variable x; $p(x)$ is a probability density function (**pdf**)
- **Normalization**: $\int p(x)dx = 1$
- Decomposition of **joint distribution** into conditional distribution:

$$p(x, y) = p(x \mid y)\, p(y)$$

where $p(x \mid y)$ is **conditional** pdf (probability of x given y)

  - if $p(x \mid y) = p(x)$, x is independent of y

- **Bayes law** follows:

$$p(y \mid x) = \frac{p(x \mid y)\, p(y)}{p(x)}$$

- **Marginalization**:

$$p(x) = \int p(x, y)\, dy = \int p(x \mid y)\, p(y)\, dy$$

is probability of x, without regard for y (nuisance parameter)

Rules of probability

- Change of variables: if **x** transformed into **z**, **z** = *f*(**x**), the pdf in terms of **z** is

$$p(\mathbf{z}) = \left|\mathbf{J}\right|^{-1} p(\mathbf{x})$$

where **J** is the Jacobian matrix for the transformation:

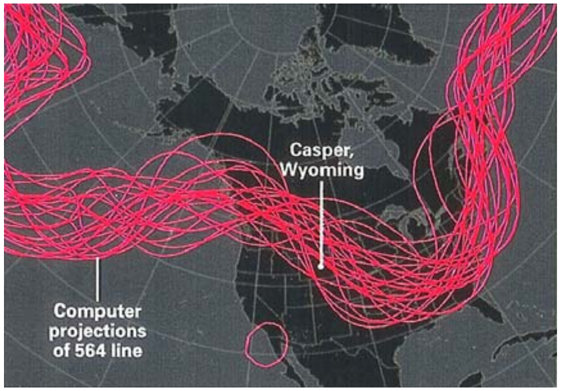
$$\mathbf{J} = \begin{pmatrix} \frac{\partial z_1}{\partial x_1} & \cdots & \frac{\partial z_3}{\partial x_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial z_1}{\partial x_3} & \cdots & \frac{\partial z_3}{\partial x_3} \end{pmatrix}$$

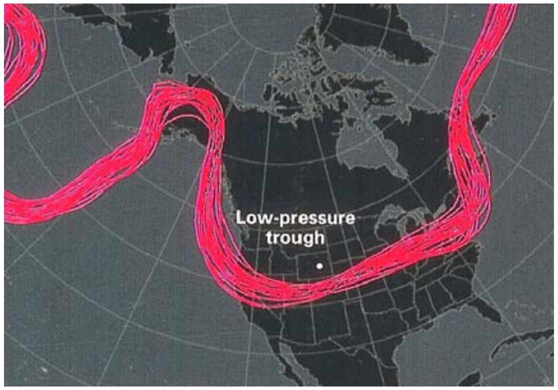
Visualizing uncertainties in weather forecasting

- Metrological forecast for Oct. 30, 2003 for Casper, Wyoming
- 22 predictions of 564 line (500 mb) obtained by varying input conditions; indicate plausible outcomes
- Density of lines conveys certainty/probability of winter storms

7 days ahead

564 line; predictive of winter storms





1 day ahead

4 days ahead

What happened? 20-inches of snow!

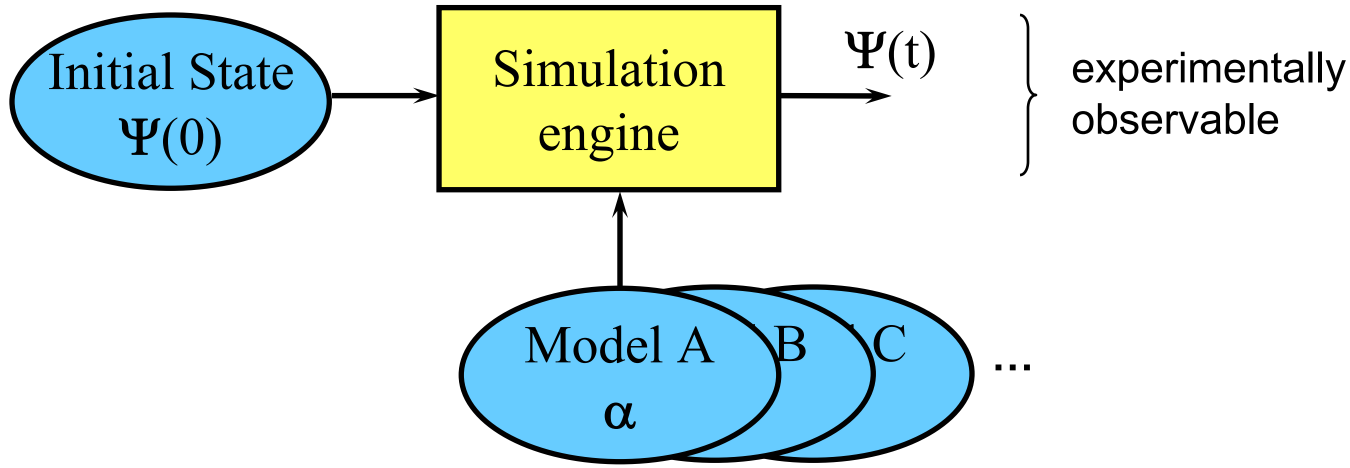
National Geographic, June 2005

Physics simulation codes

- Characteristics of simulation codes
  - complex computer codes
  - involve many submodels
    - each describes particular physical phenomenon
    - interactions possible
  - each simulation run is costly in time and computer resources

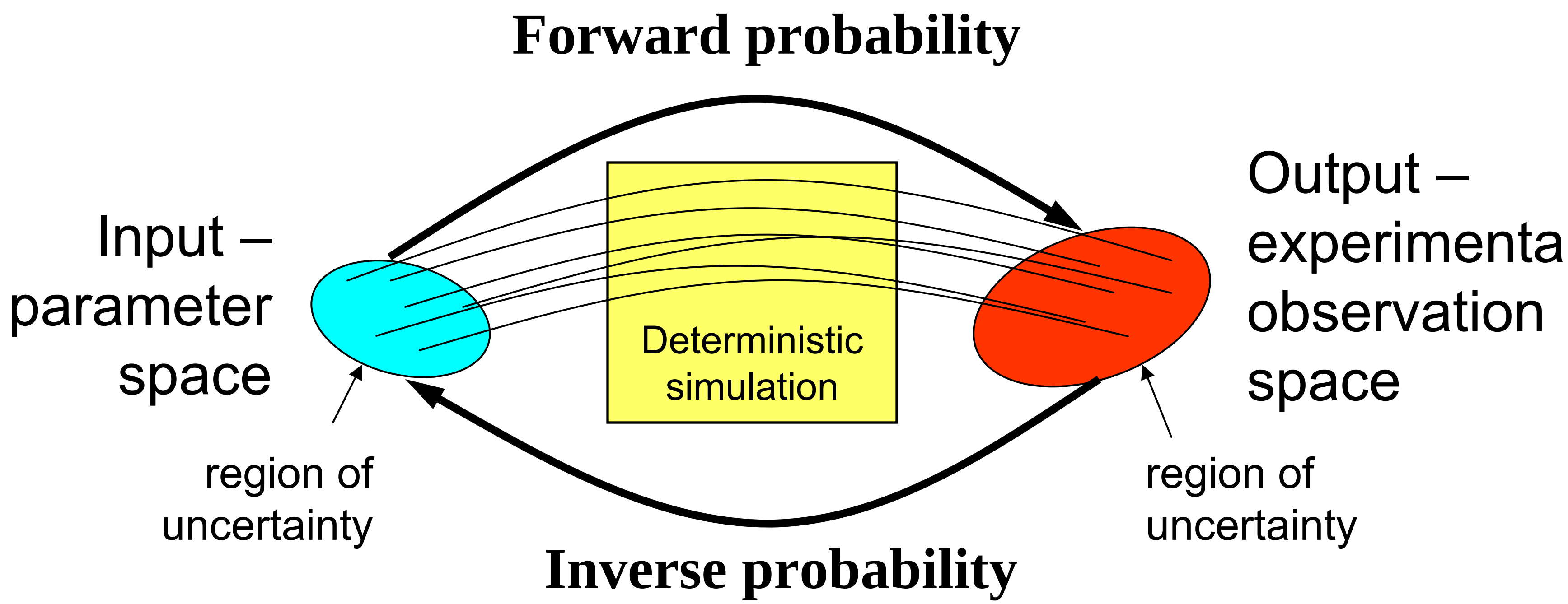
- May be difficult to quantify uncertainties and validate
  - number of simulation runs limited by cost or time
    $\Rightarrow$ restricts accuracy and depth of uncertainty assessment
  - some experiments can not be performed in a controlled and instrument way for intended application
    - meteor impact, tsunami, Big Bang

Schematic view of physics simulation code



- Simulation code predicts state of time-evolving system
  - $\Psi(t)$ = time-dependent state of system
- Requires as input
  - $\Psi(0)$ = initial state of system
  - description of physics behavior of each system component; e.g., physics model A with parameter vector α
- Simulation engine solves the dynamical equations (PDEs)

Uncertainties – forward and inverse probability



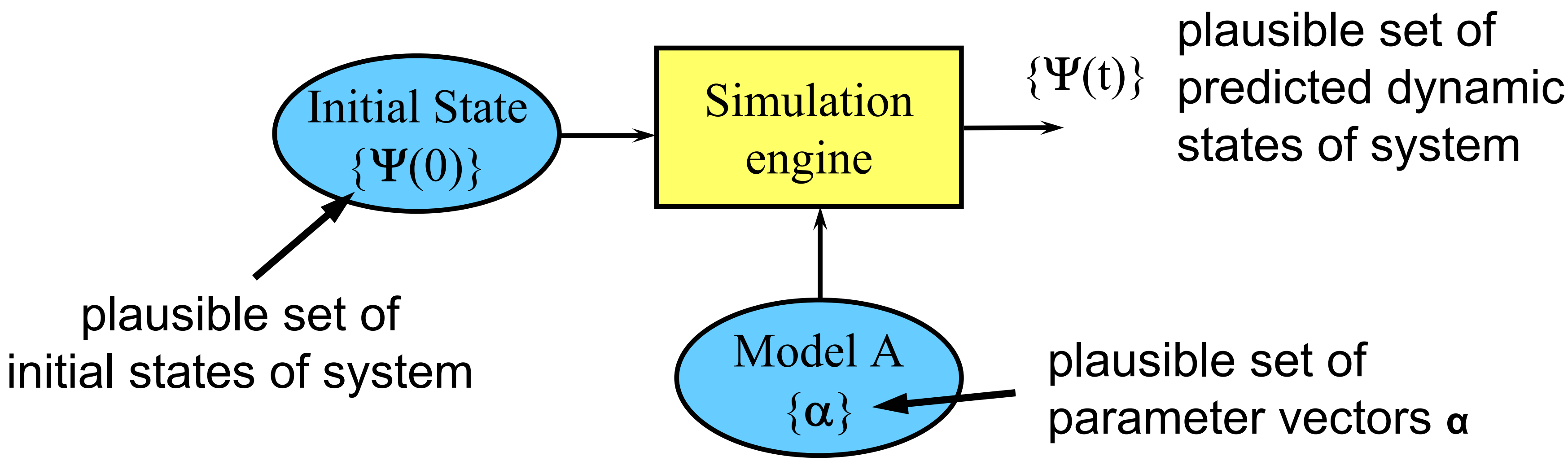
- Forward probability – propagate parameter uncertainties to uncertainties in observables
- Inverse probability - infer parameter uncertainties from uncertainties in observables

Techniques for calculating forward probability

Goal is to propagate uncertainties in parameters forward through simulation code

- Monte Carlo
  - use random samples from parameter uncertainty pdf to calculate corresponding outputs
- quasi-Monte Carlo
  - use well-ordered samples instead of random samples
- Sensitivity or functional analysis
  - characterize functional dependence of outputs on inputs
    - estimate proxy function to use in place of full simulation
  - often based on various strategies for generating sample patterns
  - differentiation of simulation code (or equations)

Forward probability using Monte Carlo



- Generate plausible predictions for known uncertainties in parameters and initial conditions
- Monte Carlo method
  - run simulation code for each random draw from pdf for α, $p(\alpha\,|.)$, and initial state, $p(\Psi(0)\,|.)$
  - simulation outputs represent plausible set of predictions, $\{\Psi(t)\}$
    - as a pdf, this is called the predictive distribution

Strategies for sensitivity analysis

Sensitivity analysis – many techniques are used to sample the functional dependence of simulation outputs relative to inputs

- One parameter at a time
  - finite differences – perturb each parameter (+Δa, ± Δa)
    - calculate first derivative (sensitivity); sometimes second derivatives
- Several parameters at a time
  - random sampling – basis of Monte Carlo calculation
  - quasi-random sampling – strive for even spacing – quasi-Monte Carlo
  - stratified random sampling – spread out evenly over domain
  - Latin Hypercube – even spacing in each parameter
- Differentiation of simulation code
  - automatic differentiation utilities produce auxiliary code based on simulation code; also can be done manually
  - solve differentiated physic equations

Techniques for calculating inverse probability

Goal is to infer parameter values and uncertainties whose simulation code outputs match experimental measurements – inference in Bayesian framework

- Maximum likelihood fitting (aka min. χ^2, least-squares, regression)
  - ▸ usually employs sensitivity analysis
- Markov Chain Monte Carlo (MCMC)
  - ▸ generate random walk, constrained by posterior pdf
  - ▸ many algorithms: Gibbs, Metropolis, hybrid, …
- Sensitivity or functional analysis
  - ▸ characterize functional dependence of outputs on inputs
    - estimate proxy function to use in place of full simulation
  - ▸ often based on various strategies for generating sample patterns
  - ▸ may also be based on differentiation of simulation code (or equations)

Bayesian inference from experimental data

- Bayes rule

$$p(\boldsymbol{a} \mid \boldsymbol{d}, I) = \frac{p(\boldsymbol{d} \mid \boldsymbol{a}, I)\, p(\boldsymbol{a} \mid I)}{p(\boldsymbol{d} \mid I)}$$

  - ***d*** is the vector of measured data values
    a is the vector of parameters for model that predicts the data
  - $p(\boldsymbol{d} \mid \boldsymbol{a}, I)$ is called the **likelihood** (probability of the data given the true model and its parameters)
  - $p(\boldsymbol{a} \mid I)$ is called the **prior** (on the parameters ***a***)
  - $p(\boldsymbol{a} \mid \boldsymbol{d}, I)$ is called the **posterior** – fully describes final uncertainty in the parameters
  - I stands for whatever **background information** we have
    about the situation and the model used, results from previous
    experiments, and our expertise
  - denominator provides normalization: $p(\boldsymbol{d}) = \int p(\boldsymbol{d} \mid \boldsymbol{a})\, p(\boldsymbol{a})\, d\boldsymbol{a}$

Auxiliary information – I

All relevant information about the situation may be brought to bear:

- Details of experiment
  - laboratory set up, experiment techniques, equipment used
  - potential for experimental technique to lead to mistakes
  - expertise of experimenters

  (the last two items: more subjective)
- Relationship between measurements and theoretical model
- History of kind of experiment
- Appropriate statistical models for likelihood and prior
- Experience and expertise

- We usually leave I out of our formulas, but keep it in mind

Likelihood

- Form of the likelihood $p(\boldsymbol{d} \mid \boldsymbol{a}, I)$ depends on how we model the uncertainties in the measurements $\boldsymbol{d}$
- If measurement uncertainties are independent, overall likelihood is product of individual likelihoods, $\Pi_i \, p(d_i \mid \boldsymbol{a}, I)$
- Choose pdf that appropriately describes uncertainties in data
  - Gaussian – good generic choice
  - Poisson – counting experiments
  - Binomial – binary measurements (coin toss …)
- Outliers exist
  - likelihood should have a long tail; large fluctuations are possible
- Systematic errors
  - caused by effects common to many (all) measurements
  - model by introducing variable that affects many (all) measurements; then marginalize it out

Priors

- Noncommittal prior (non-informative)
  - uniform pdf; $p(a)$ = const. when a is an offset parameter
  - uniform in log(a); $p(\log a)$ = const. when a is a scale parameter
  - choose pdf with maximum entropy, subject to known constraints
- Physical principles
  - some physical quantities can not be negative ⇒ $p(a) = 0$, when $a < 0$
  - invariance arguments, symmetries
- Previous experiments
  - use posterior from previous measurements for prior
  - Bayesian updating (Kalman)
- Expertise
  - elicit pdfs from experts in the field
  - elicitation, an established discipline, may be useful

Priors

- Conjugate priors
  - for many forms of likelihood, there exist companion priors that make it easy to integrate over the variables
  - these priors facilitate analytic solutions for posterior
  - for example, for the Poisson likelihood in n and λ, the conjugate prior is a Gamma distribution in λ with parameters α and β, which determine the position and width of the prior
  - conjugate priors can be useful and their parameters can often be chosen to create a prior close to what the analyst believes is correct
  - however, in the context of numerical solution of complicated overall models, they loose their appeal

Posterior

- Posterior $p(\boldsymbol{a} \mid \mathbf{d}, I)$
  - ▸ final result of a Bayesian analysis
  - ▸ summarizes our state of knowledge about parameters $\boldsymbol{a}$
  - ▸ it provides complete quantitative description of uncertainties
  - ▸ usually characterized in terms of an estimated value of the variables and their covariance
- Visualization
  - ▸ difficult to visualize directly because it is a density distribution of many variables (many dimensions)
  - ▸ Monte Carlo allows us to visualize the posterior through its effect on the model that has been used in the analysis (quasi-MC useful here)

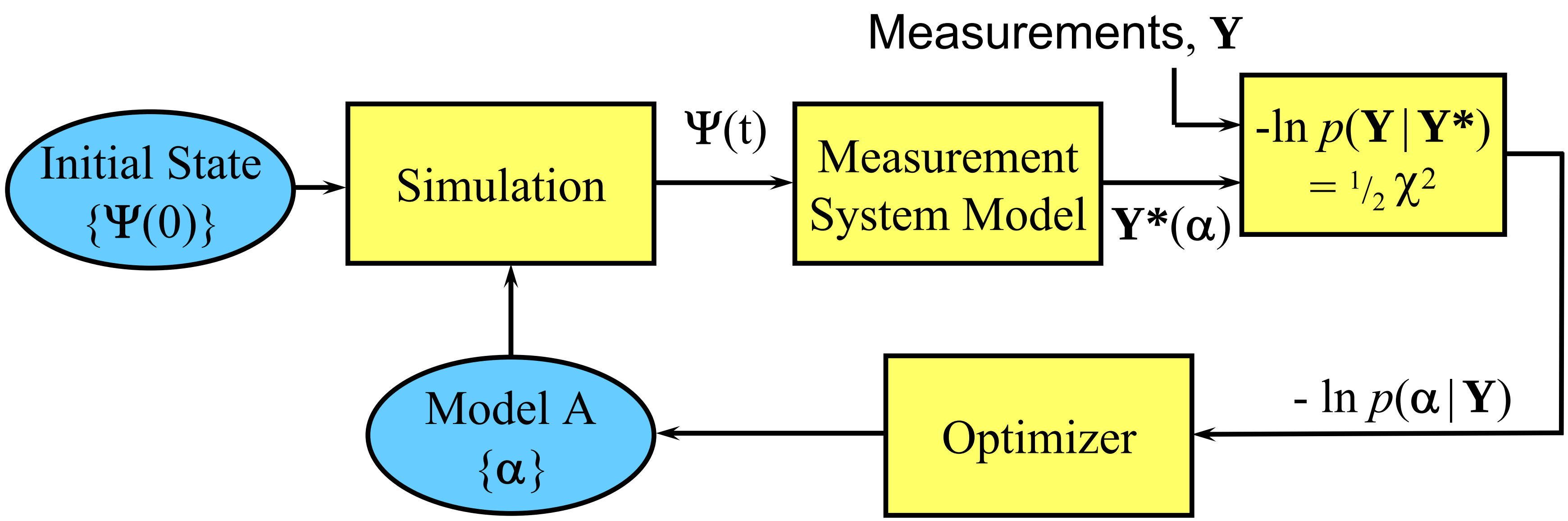
The likelihood and chi-squared

- Assuming the uncertainty in each measurement d_i is **Gaussian distributed** with zero mean and variance σ_i^2, and the uncertainties are statistically **independent**, the likelihood is

$$p(\boldsymbol{d} \mid \boldsymbol{a}) \propto \exp\left\{-\tfrac{1}{2}\sum_i \left[\frac{[d_i - y_i(\boldsymbol{a})]^2}{\sigma_i^2}\right]\right\} \propto \exp(-\tfrac{1}{2}\chi^2)$$

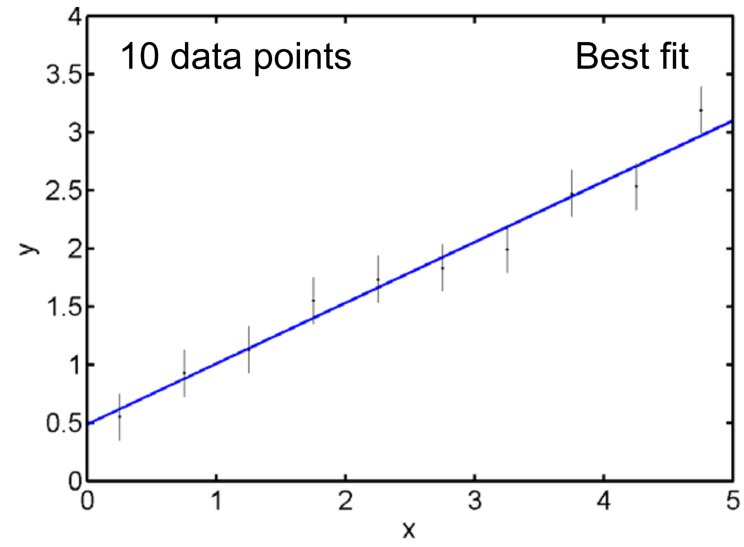
  - where y_i is the value predicted for parameter set $\boldsymbol{a}$
- For a non-informative **uniform prior**,
  - posterior $p(\boldsymbol{a} \mid \boldsymbol{d})$ is proportional to the likelihood $p(\boldsymbol{d} \mid \boldsymbol{a})$, and
  - **maximum likelihood** solution same as **maximum likelihood**; equivalent to **minimum chi squared (**or **least squares**)
- Estimated parameters $\boldsymbol{a}$ and their uncertainties are given by the dependence on $\boldsymbol{a}$ of posterior $p(\boldsymbol{a} \mid \boldsymbol{d})$
  → usually used to approximate posterior with a Gaussian

Parameter estimation - maximum likelihood



- Optimizer adjusts parameters α to minimize $-\ln p(\mathbf{Y}\,|\mathbf{Y^*}(\alpha))$
- Result is maximum likelihood estimate for α (also known as minimum-chi-squared solution)
- Optimization process is accelerated by using gradients
  - differentiation of code efficiently calculates gradients of forward calc.

Fit linear function to data – minimum χ^2



- Linear model: $y = a + bx$
- Simulate 10 data points, $\sigma_y = 0.2$
  exact values: $a = 0.5 \quad b = 0.5$
- Determine parameters, intercept a and slope b, by minimizing chi-squared (standard least-squares analysis)
- Result: $\chi^2_{\min} = 4.04 \quad p = 0.775$

$$\hat{a} = 0.484 \qquad \sigma_a = 0.127$$

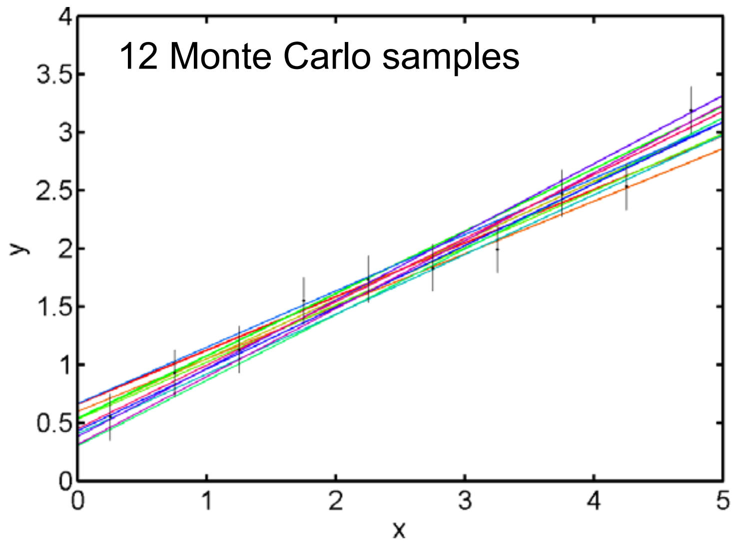
$$\hat{b} = 0.523 \qquad \sigma_b = 0.044$$

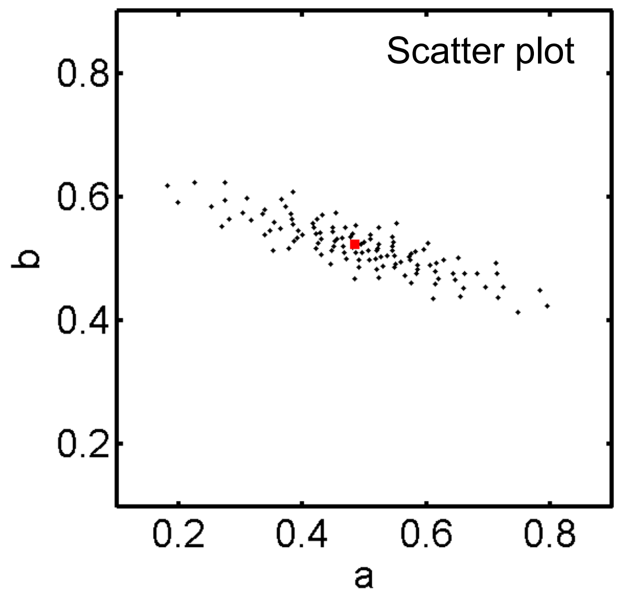
$$\mathbf{R} = \begin{bmatrix} 1 & -0.867 \\ -0.867 & 1 \end{bmatrix}$$

- Strong correlations between parameters a and b

Linear fit – uncertainty visualization

- Uncertainties in parameters are represented by Gaussian pdf in 2-D parameter space
  - correlations evidenced by tilt in scatter plot
  - points are random samples from pdf
- Should focus on implied uncertainties in physical domain
  - model realizations drawn from parameter uncertainty pdf
  - these appear plausible – called **model checking**
  - this comparison to the original data confirms model adequacy
  - called **predictive distribution**



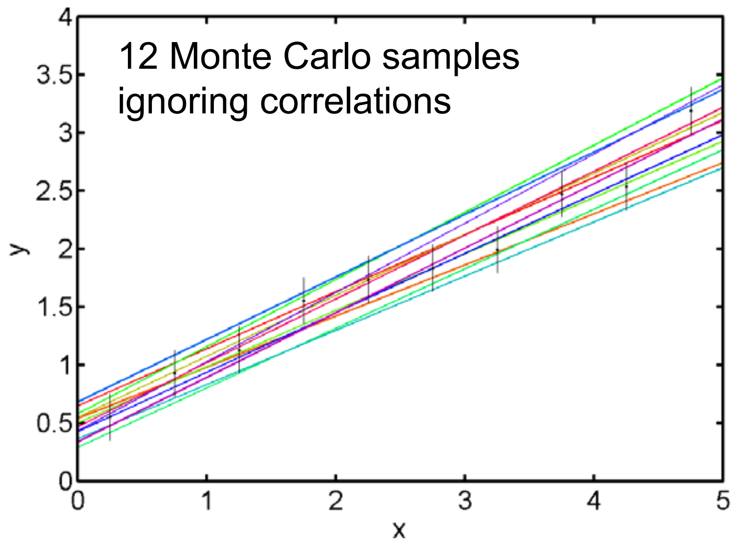


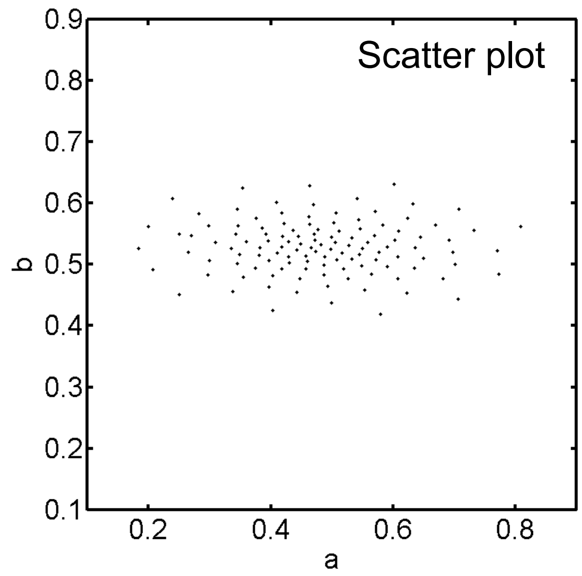
Linear fit – correlations are important

- Plots show what happens if off-diagonal terms of covariance matrix are ignored
- Correlation matrix is

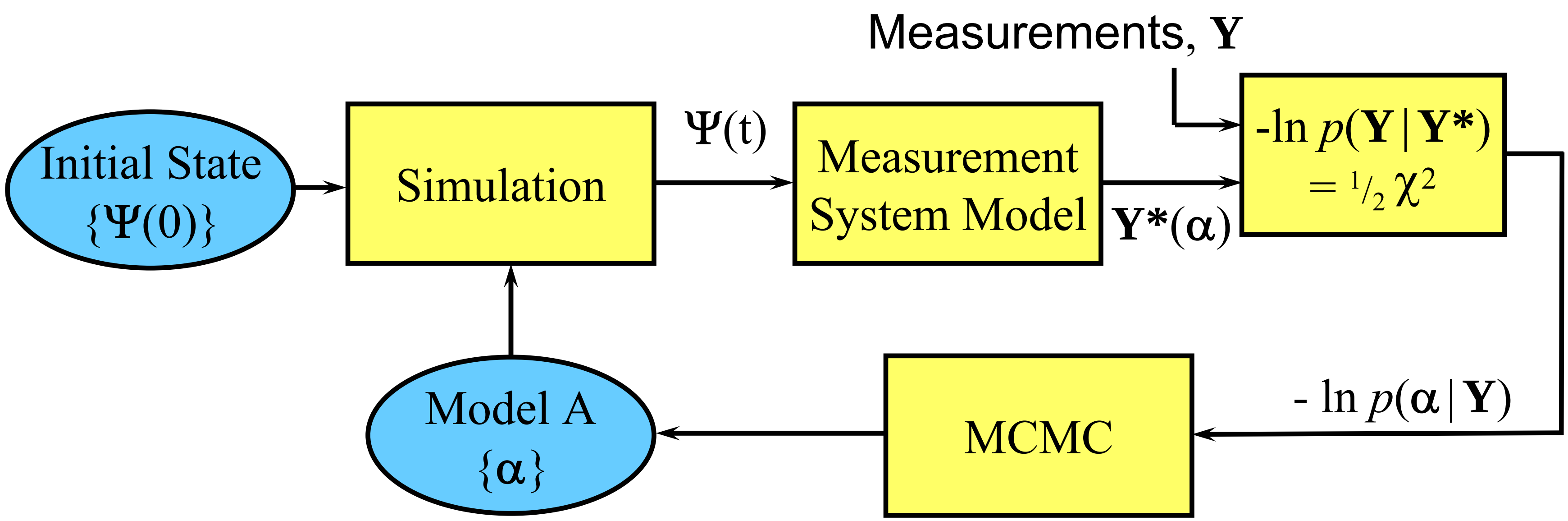
$$\mathbf{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

- Model realizations show much wider dispersion than consistent with uncertainties in data
- No tilt in scatter plot – uncorrelated
- Correlations are important !





Inverse probability using MCMC



- Markov Chain Monte Carlo (MCMC) algorithm generates a random sequence of parameters that sample posterior probability $p(\boldsymbol{\alpha} \mid \mathbf{Y})$, yielding plausible a set of parameters $\{\boldsymbol{\alpha}\}$.
- MCMC algorithm based on values of $p(\boldsymbol{\alpha} \mid \mathbf{Y})$ calculated for random trial samples of $\boldsymbol{\alpha}$
- MCMC can be used for posteriors with arbitrary functional forms

MCMC - problem statement

- Parameter space of n dimensions represented by vector $\mathbf{x}$
- Given an "arbitrary" **target** probability density function (pdf), $q(\mathbf{x})$, draw a set of samples $\{\mathbf{x}_k\}$ from it
- Only requirement typically is that, given $\mathbf{x}$, one be able to evaluate $Cq(\mathbf{x})$, where C is an unknown constant, that is, $q(\mathbf{x})$ need not be normalized

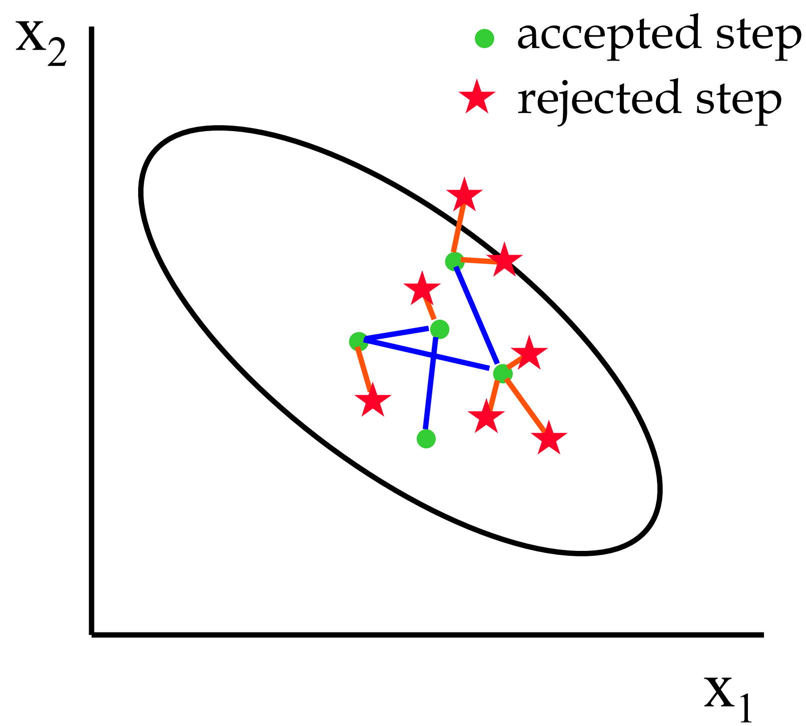
- It all started with seminal paper (from LANL):
  - N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equations of state calculations by fast computing machine," *J. Chem. Phys.* **21**, pp. 1087–1091 (1953)
    - MANIAC: 5 KB RAM, 100 KHz, 1 KHz multiply, 50 KB disc

Markov Chain Monte Carlo

Generates sequence of random samples from an arbitrary probability density function

- Metropolis algorithm:
  - draw trial step from symmetric pdf, i.e., $t(\Delta \mathbf{x}) = t(-\Delta \mathbf{x})$
  - accept or reject trial step
  - simple and generally applicable
  - relies only on calculation of target pdf for any **x**

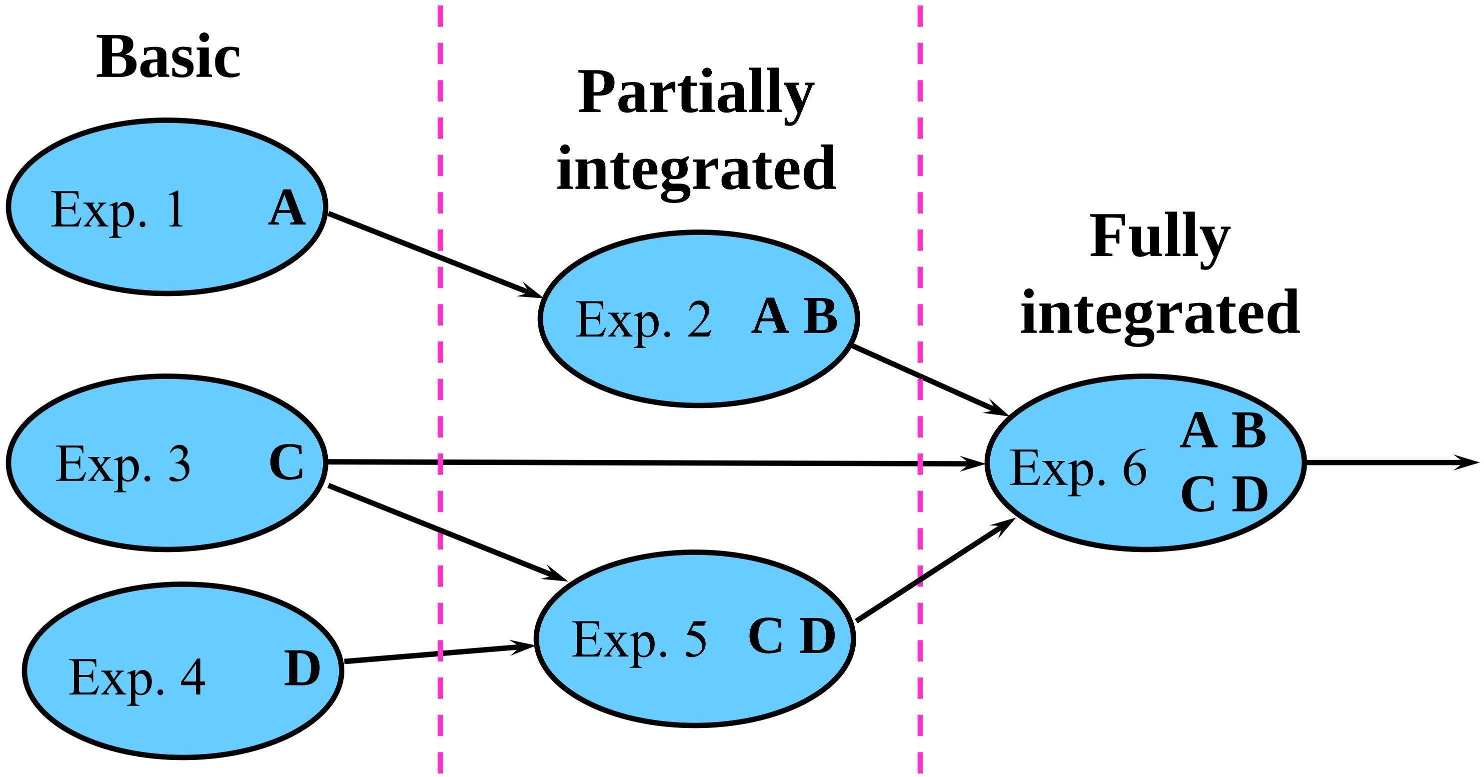
Probability(x_1, x_2) = q(**x**)



Uncertainty quantification for simulation codes

- Goal is to develop an uncertainty model for the simulation code by comparing it to experimental measurements
  - determine and quantify sources of uncertainty
  - uncover potential inconsistencies of submodels with experiments
  - possibly introduce additional submodels, as required
  - deal with **model error** (discrepancy with measurements)
- Recursive process
  - aim is to develop submodels that are consistent with all experiments (within uncertainties)
  - a **hierarchy of experiments** helps substantiate submodels over wide range of physical conditions and accumulate information
  - each experiment potentially advances our understanding

Linked analyses of hierarchy of experiments

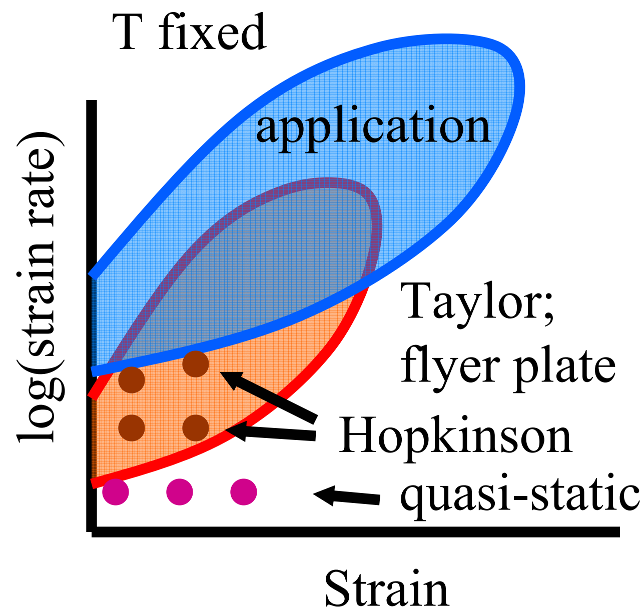


- Information flow in analyses of series of experiments
- Bayesian calibration –
  - analysis of each experiment updates model parameters (A, B, C, etc.) and their uncertainties, consistent with previous analyses
  - information about models accumulates

Hierarchy of experiments – metal plasticity

Suppose application is high-speed projectile impacting plate

- Basic characterization experiments – measure stress-strain relationship at specific stain and strain rate
  - quasi-static tests – low strain rates
  - Hopkinson bar – medium strain rates
- Partially integrated experiment
  - Taylor test – cylinder impact into wall
  - flyer plate expt. – plate impacted
- Fully integrated experiments
  - mimic application as closely as possible
  - may involve extrapolation of operating range, introducing additional uncertainty
  - integrated experiments can help reduce model uncertainties in their operating range; may expose model deficiencies

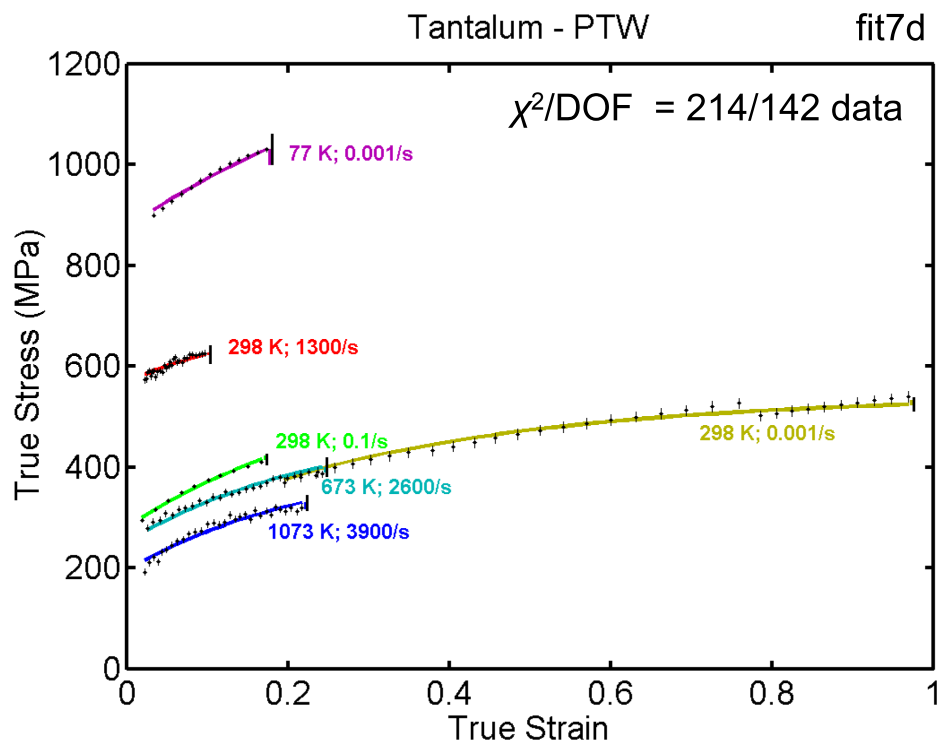


Fit PTW model to stress-strain measurements

Quasi-static and Hopkinson bar measurements for Tantalum

- Preston-Tonks-Wallace (PTW) model describes stress-strain relations in dynamic plastic deformation of metal
- Measurement standard errors carefully assessed
- **Systematic uncertainty** (3%) in offset for each data set; accounts for specimen variability
- Fit 7 PTW + 6 offset parameters
- Result of fitting process is
  - parameter values and their standard errors
  - correlation matrix

} define Gaussian posterior



PTW curves include adiabatic heating effect for high strain rates

†data supplied by S-R Chen, MST-8

PTW parameters and their uncertainties

Parameters +/- rms error:

$\theta = 0.0080 \pm 0.0004$

$\kappa = 0.68 \pm 0.06$

$-\ln(\gamma) = 11.5 \pm 0.8$

$y_0 = 0.0092 \pm 0.0005$

$y_\infty = 0.00147 \pm 0.00011$

$s_0 = 0.0176 \pm 0.0032$

$s_\infty = 0.00358 \pm 0.00018$

Minimum chi-squared fit yields estimated PTW parms. and rms errors, as well as correlation coefficients, which are crucially important!

Fixed parms:

$p = 4$

$y_1 = 0.012$

$y_2 = 0.4$

$\beta = 0.23$

$\alpha_p = 0.48$

$G_0 = 722$ MPa

$T_{melt} = 3290$ ºK

$\rho = 16.6$ g/cm^2

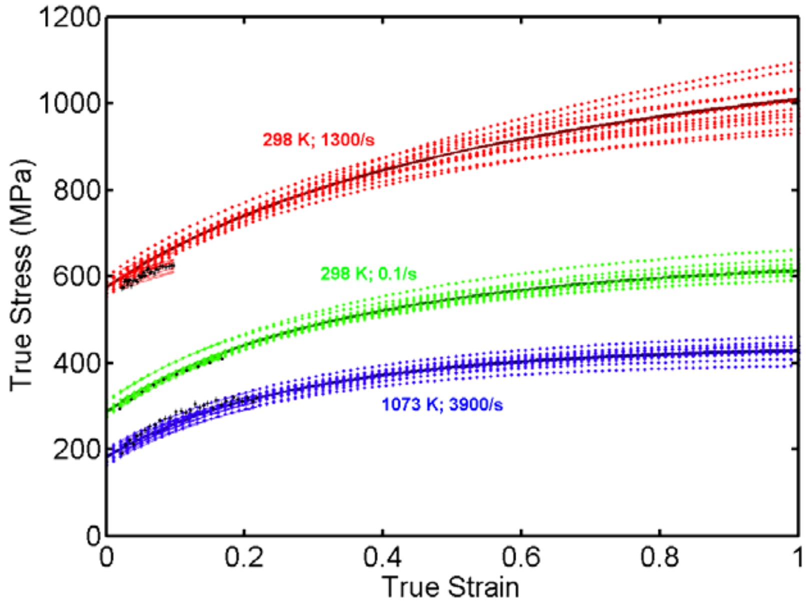
Correlation coefficients

	θ	κ	$-\ln(\gamma)$	y_0	y_∞	s_0	s_∞
θ	1	-0.180	-0.108	-0.113	-0.283	-0.817	0.211
κ	-0.180	1	0.716	0.596	0.644	0.292	0.580
$-\ln(\gamma)$	-0.108	0.716	1	0.046	0.111	0.105	0.171
y_0	-0.113	0.596	0.046	1	0.502	0.282	0.477
y_∞	-0.283	0.644	0.111	0.502	1	0.350	0.640
s_0	-0.817	0.292	0.105	0.282	0.350	1	-0.278
s_∞	0.211	0.580	0.171	0.477	0.640	-0.278	1

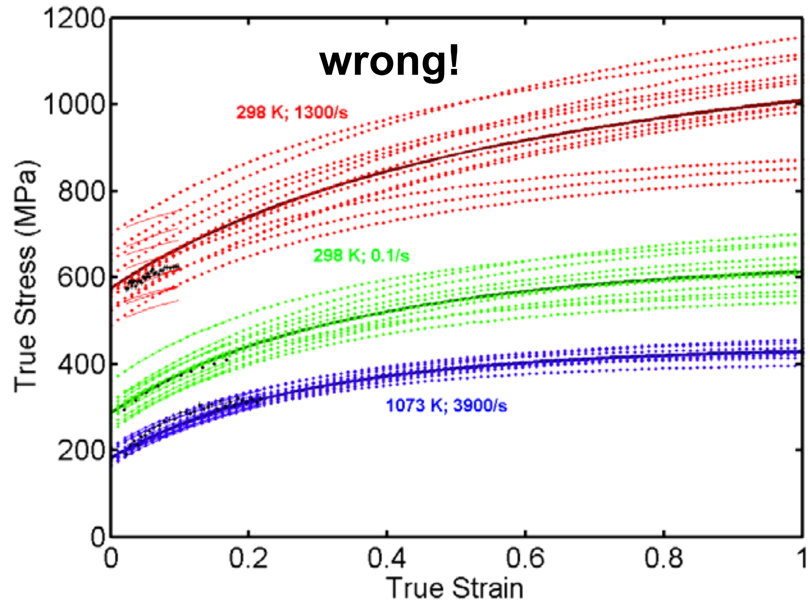
Visualization of uncertainties in model

- Uncertainties visualized by displaying (quasi) Monte Carlo draws from uncertainty distribution
  - done **correctly** with full covariance matrix (left)
  - done **incorrectly** by neglecting off-diagonal terms in covariance matrix (right)

MC with correlations

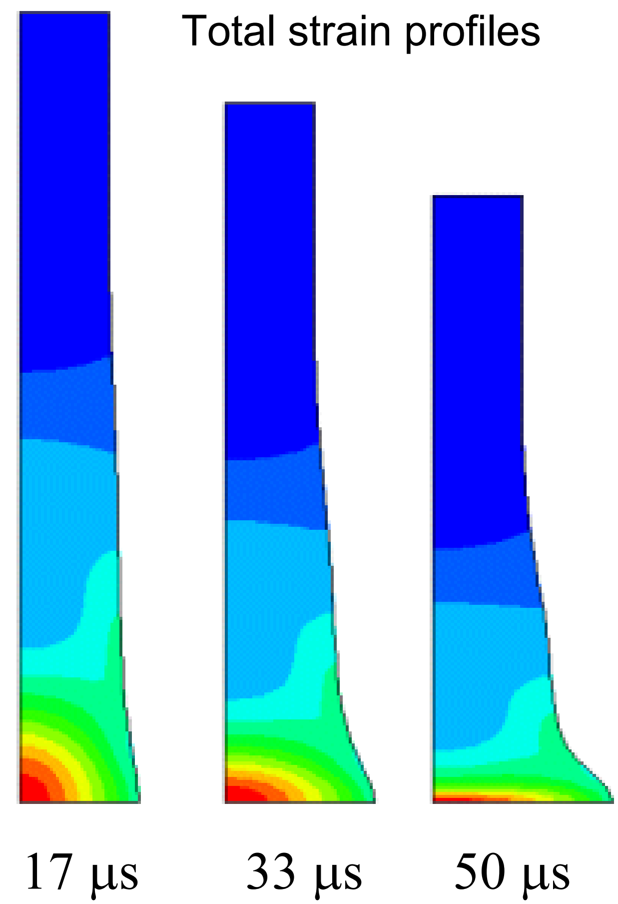


MC without correlations



Taylor test simulations

- Simulate Taylor impact test – steel cylinder impacting rigid wall
- For impact velocity = 350 m/s, effective total strain reaches 250%
- Submodels required:
  - dynamical equations
  - equation of state (EOS): $T(p, \rho)$
  - material plasticity behavior
  - at very high impact speeds
    - material fracture, break up
    - melting
    - liquid behavior

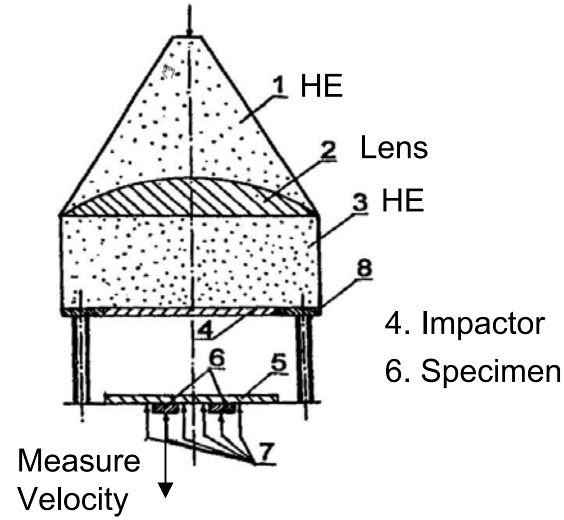


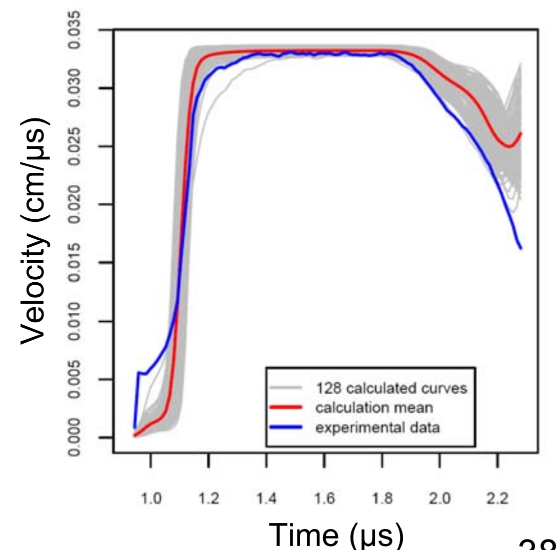
Simulation by Abaqus (FEM code)
High-strength steel cylinder
5 mm dia, 38 mm long

Flyer-plate experiment

- Flyer plate impacts specimen – measure velocity on back surface
  - ▸ aim is to make specimen spall
- Simulation code uses PTW model to predict velocity
- Plot compares flyer-plate measurements with generous range of predictions
- Challenge: PTW model consistent with flyer-plate and calibration experiments
- Submodels required:
  - ▸ dynamical equations
  - ▸ equation of state (EOS): $T(p, \rho)$
  - ▸ material plasticity behavior (PTW)
  - ▸ material fracture

Experimental set up



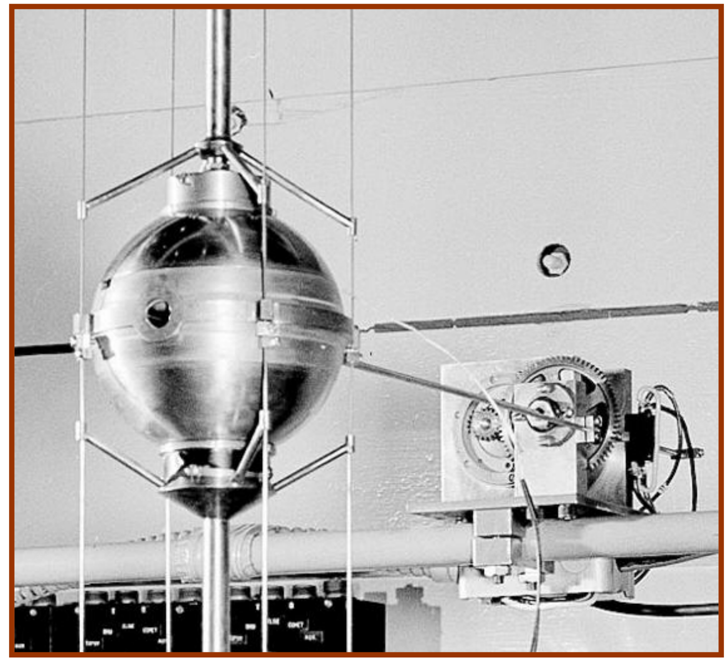


†plot from B. Williams et al.

JEZEBEL – criticality experiment

- JEZEBEL experiment (1950-60)
  - fissile material ^{239}Pu
  - measure neutron multiplication as function of separation of two hemispheres of fissile material
  - summarize criticality with neutron multiplication factor, $k_{eff} = 0.9980 \pm 0.0019$ for a specific geometry
  - **very accurate** measurement
- Our goal – use highly accurate JEZEBEL measurement to improve our knowledge of ^{239}Pu cross sections

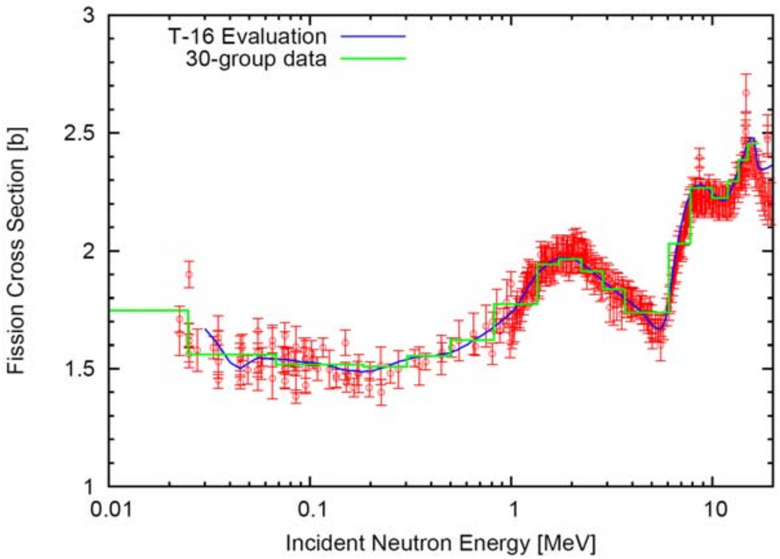
JEZEBEL set up



Neutron cross sections for ^{239}Pu

- Plot shows
  - measured fission cross sections for neutrons on ^{239}Pu (red data points)
  - inferred cross sections (blue line)
  - weighted average in 30 energy bins (groups) for PARTISN calculation (green histogram)
- PARITSN code simulates neutron transport based on multigroup, discrete-ordinates method
- We use PARTISN and JEZEBEL to
  - update cross sections to improve their accuracy (inference)
  - predict uncertainties after update (forward prop.)

measured
^{239}Pu fission cross sections

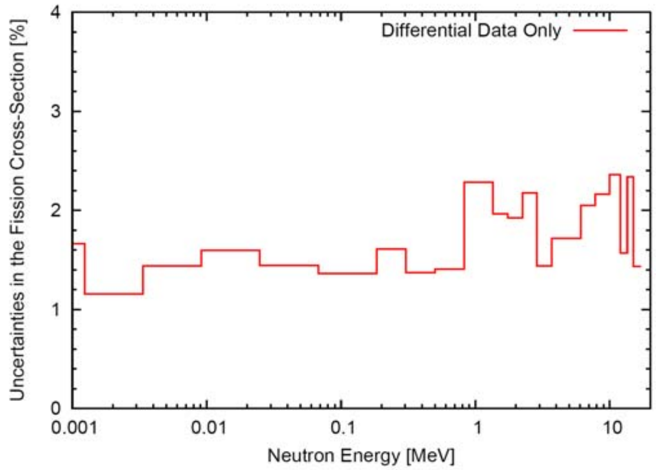


Neutron cross sections - uncertainties

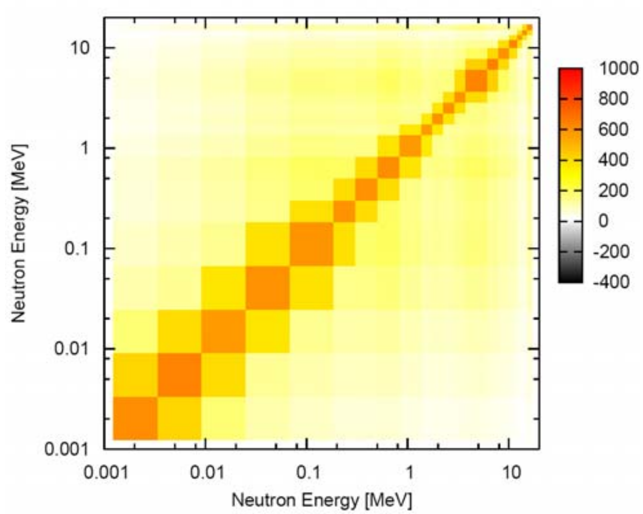
- Analysis of measured cross sections yields a set of evaluated ^{239}Pu cross sections

- Uncertainties in evaluated cross sections are ~ 1.4-2.4 %
- Covariance matrix important
- Strong positive correlations caused by normalization uncertainties in each experiment

standard error in cross sections



correlation matrix

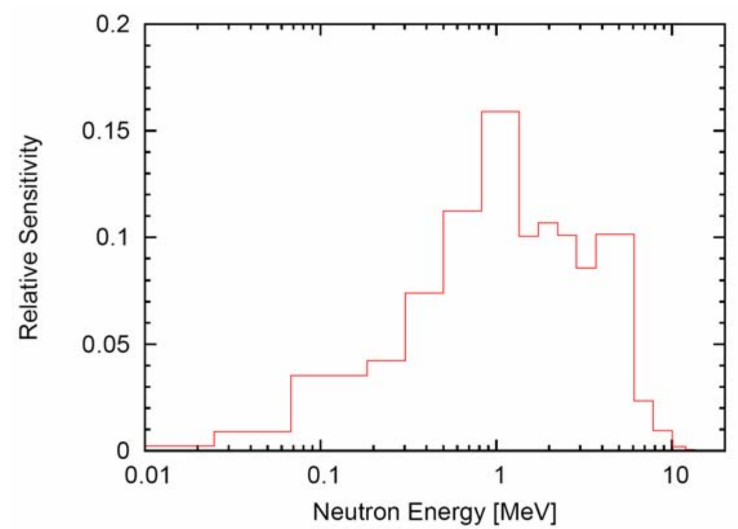


JEZEBEL – sensitivity analysis

- PARTISN code relates k_{eff} to neutron cross sections
- Sensitivity of k_{eff} to cross sections found by perturbing cross section in each energy bin by 1% and observing increase in k_{eff}
- Observe that 1% increase in all cross sections results in 1% increase in k_{eff} , as expected

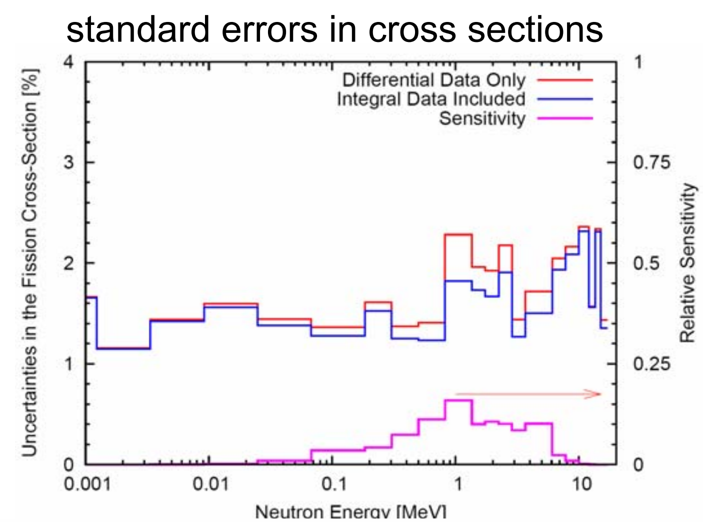
- In real applications, one often does not have this sensitivity vector, so Monte Carlo used to propagate uncertainties

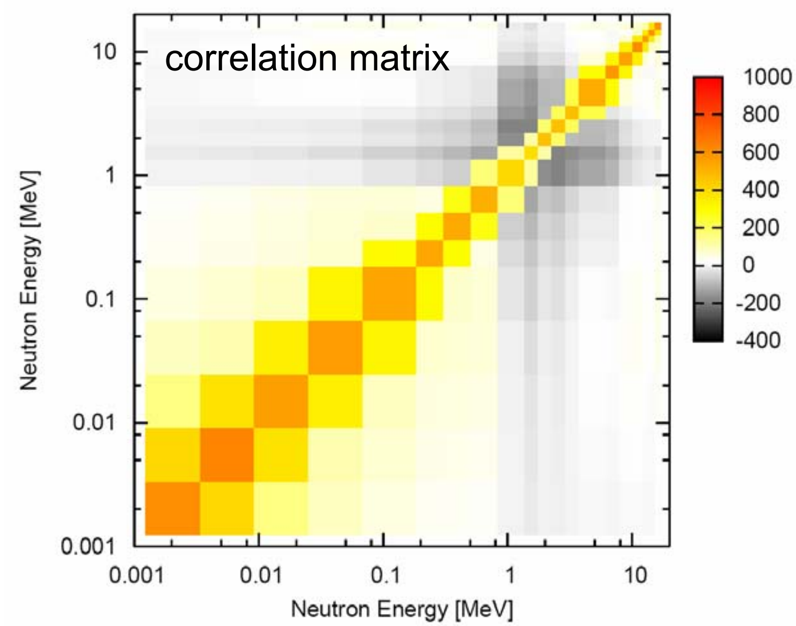
k_{eff} sensitivity to cross sections



Cross sections updated using JEZEBEL

- Plot shows uncertainties in cross sections before and after incorporating JEZEBEL measurement
- Individual uncertainties modestly reduced
  - follows energy dependence of sensitivity
- Correlation matrix is significantly altered
- Strong negative correlations are introduced by integral constraint of matching JEZEBEL's k_{eff}
- What are uncertainties in new PARTISN simulation?





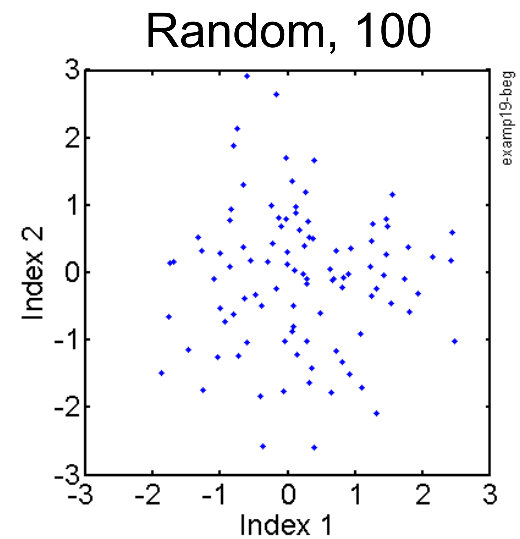
Uncertainty in subsequent simulations

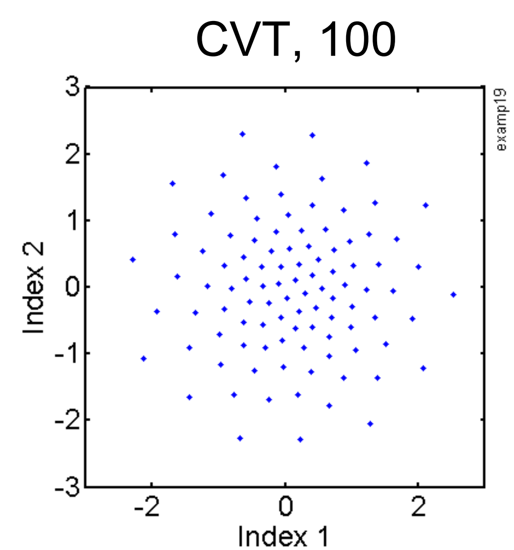
- Intent is to use updated cross sections in new calculations, expecting a reduction in uncertainties in calculated k_{eff}
- Need to estimate the uncertainty in k_{eff} calculated for new scenarios
- For this demonstration, we do calculation for JEZEBEL

- Forward propagation of uncertainties
  - standard approach is to use random Monte Carlo
  - we try using quasi-Monte Carlo to "predict" k_{eff}
    - qMC point sets obtained using Centroidal Voronoi Tessellation
    - result: mean and rms deviation of k_{eff} are better determined than with random MC

CVT for 2D Gaussian distribution

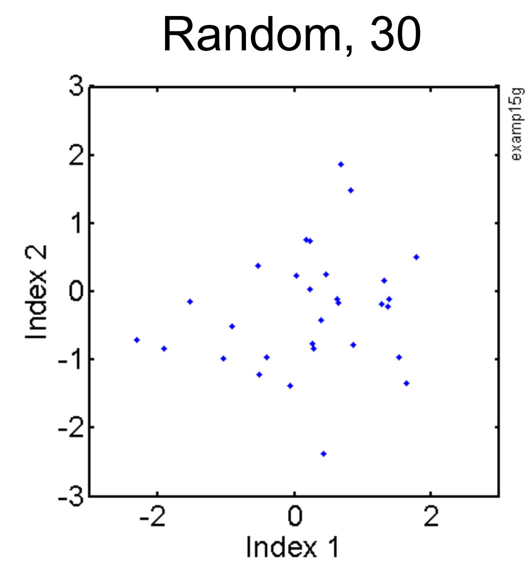
- Centroidal Voronoi Tessellation (CVT)
  - when generating points of Voronoi cells match the cells' centroids
  - easy to produce CVT point sets in high dimensions using Monte Carlo
- Plots show starting random point set and final CVT set for 2D unit-variance Gaussian
- CVT points more evenly distributed; regular pattern
  - better integration accuracy than random
- Propose using CVT for forward propagation of uncertainties for better accuracy



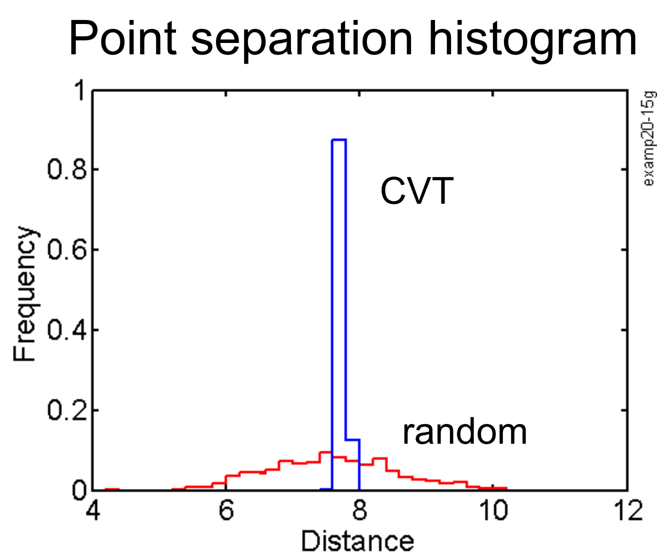


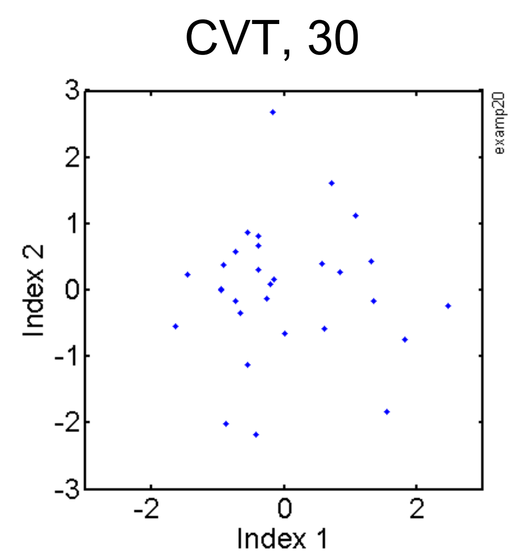
CVT: 30 points in 30 dimensions

- 30D unit-variance Gaussian distribution
- Projected onto 2D plane, CVT result doesn't look much different than random sample set
- However, CVT points are uniformly distributed in 30D, while random points are not



All points are nearest neighbors!

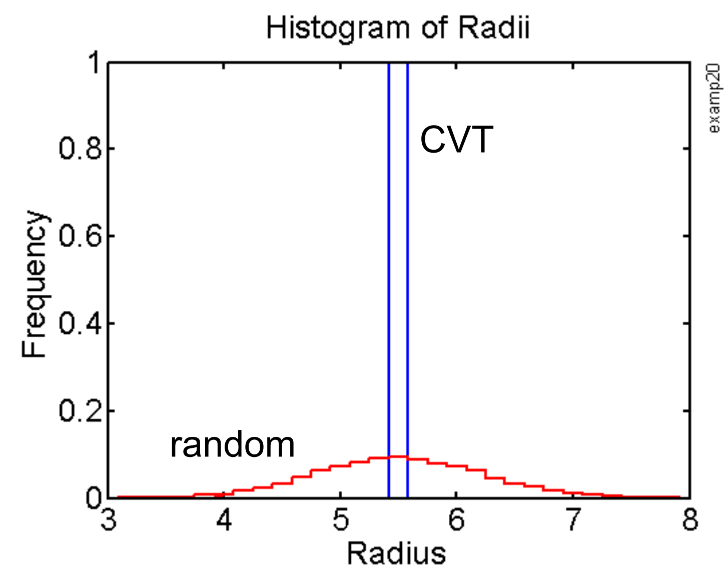




CVT radial distribution: 30 points in 30D

- All 30 CVT points in 30D are at the same radius
  - lie on the surface of a hypersphere
- As seen in last slide, the inter-point distances for CVT are essentially identical
  - regular point pattern (unique?)
- Rotation is only degree of freedom between different realizations of CVT
- One can generate new CVT patterns by randomly rotating an existing one

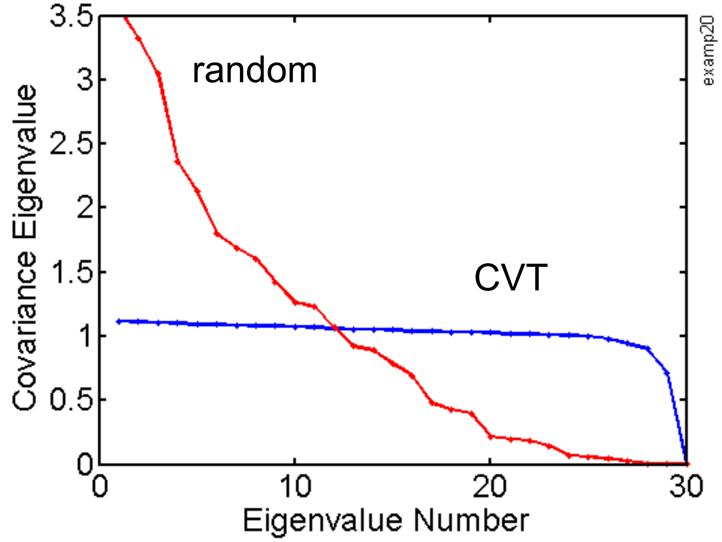
n = 30; d = 30



Covariance analysis of 30 CVT points in 30D

- CVT applied to 30 points in 30 dimensions yields an evenly distributed set of points
  - all at same radius
  - all equally spaced
- Eigenanalysis of covariance matrix of point set yields the covariance spectrum
- Conclude
  - CVT spectrum is much more uniform than for random set
  - variance of projection of points is same in almost all directions
- Last eigenvalue is zero; rank = 29
  - 31 points needed to fully sample 30D behavior

n = 30; d = 30



Accuracy of predicted k_{eff} and its uncertainty

- Check accuracy of predicted mean and standard deviation of k_{eff} for JEZEBEL, based on 30 samples, random and CVT
  - exact value known from sensitivity and linear model used
- Conclude – CVT is very accurate, for both mean and rms dev.
  - random samples yield 15% accuracy for k_{eff} std. dev

Results from 1000 sample sets; 'rot' indicates single sample set randomly rotated to obtain each new one

	est. mean k_{eff}		est. std. dev. k_{eff}	
	avg.	rms dev.	avg.	rms dev.
random	0.99788	0.00037	0.00191	0.00028
random-rot	0.99824	0.00010	0.00218	0.00010
CVT-rot	0.99796	0.00001	0.00197	0.00002
exact-linear	0.99796	-	0.00195	-

Further advantages of Bayesian analysis

- Bayesian method helps us cope with the difficulties commonly encountered in data analysis
  - systematic uncertainties
  - inconsistent data
  - outliers
  - uncertainties in stated uncertainties
  - model checking –
    does model agree with experimental evidence?
  - model selection – which model is best?
    - between two models, which is best supported by data?
    - how many spline knots should be used to fit data?

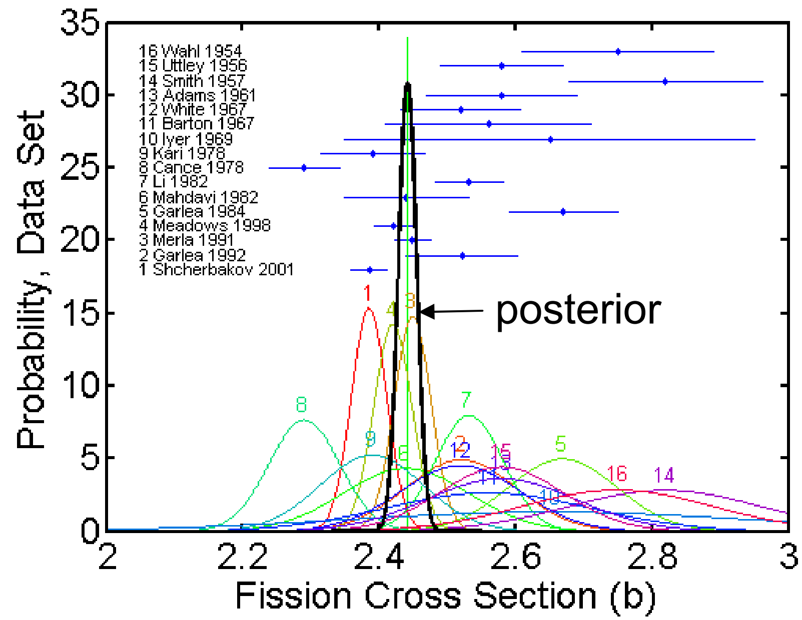
^{239}Pu cross sections – coping with outliers

- Gaussian likelihood (min χ^2) fit yields
  - $\chi^2 = 44.7$, $p = 0.009\%$ for 15 DOF
    2.441 ± 0.013 b
  - implausibly small uncertainty, given that three smallest uncertainties ≈ 0.027 b
- Each datum reduces the standard error of result, even if it does not agree with it!
  - consequence of Gaussian likelihood

$$\sigma^{-2} = \sum_{i=1}^{n} \sigma_i^{-2}$$

  - independent of where data lie!

Summary of measurements of ^{239}Pu cross section at 14.7 MeV



Gaussian: 2.441 ± 0.013 b

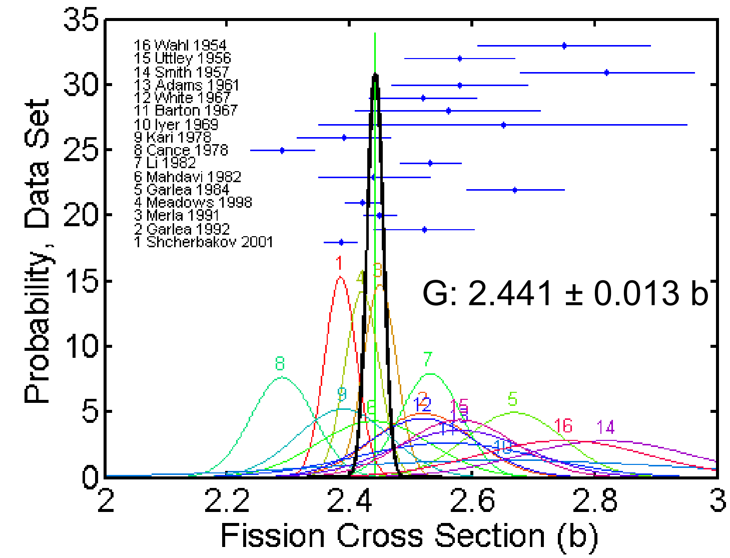
^{239}Pu cross sections – outlier-tolerant likelihood

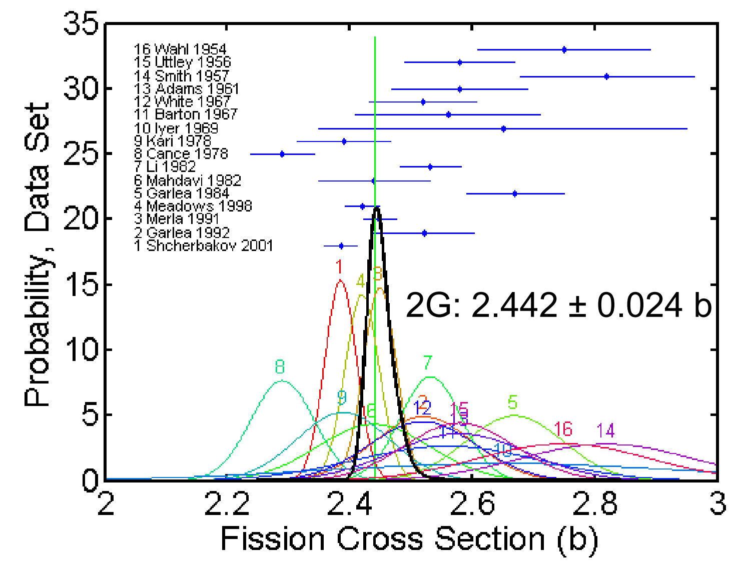
- Long-tailed likelihood for each datum used in Bayesian analysis to account for outliers
- Two-Gaussian likelihood has right properties

$$(1-\beta)\exp\left\{-\frac{(x-m)^2}{2\sigma^2}\right\}+\frac{\beta}{\gamma}\exp\left\{-\frac{(x-m)^2}{2\gamma^2\sigma^2}\right\}$$

  - where $\beta = 0.01$ and $\gamma = 10$

- With 2-Gaussian likelihood
  2.442 ± 0.024 b
  whereas Gaussian yields
  2.441 ± 0.013 b
  - 2G gives almost same mean value but more conservative standard error





Model selection – higher order inference

- Bayes rule for posterior for parameters

$$p(\boldsymbol{a} \mid \boldsymbol{d}, \mathrm{M}) = \frac{p(\boldsymbol{d} \mid \boldsymbol{a}, \mathrm{M})\, p(\boldsymbol{a} \mid \mathrm{M})}{p(\boldsymbol{d} \mid \mathrm{M})}$$

  - $\boldsymbol{d}$ represents measurements
  - $\boldsymbol{a}$ represents parameters for model M
  - $p(\boldsymbol{d} \mid \boldsymbol{a}, \mathrm{M})$ is the **likelihood**
  - $p(\boldsymbol{a} \mid \mathrm{M})$ is the **prior**
  - $p(\boldsymbol{a} \mid \boldsymbol{d}, \mathrm{M})$ is called the **posterior**
  - denominator provides normalization:

$$p(\boldsymbol{d} \mid \mathrm{M}) = \int p(\boldsymbol{d} \mid \boldsymbol{a}, \mathrm{M})\, p(\boldsymbol{a}, \mathrm{M})\, d\boldsymbol{a}$$

  - inference about parameters does not require knowing this integral

Bayesian model selection

- Bayes rule for probability of model M

$$p(\mathrm{M}\,|\,\boldsymbol{d}) \propto p(\boldsymbol{d}\,|\,\mathrm{M})\, p(\mathrm{M})$$
$$= p(\mathrm{M})\int p(\boldsymbol{d}\,|\,\boldsymbol{a},\mathrm{M})\, p(\boldsymbol{a},\mathrm{M})\, d\boldsymbol{a}$$

- Odds ratio between two models

$$\frac{p(\mathrm{M}_1\,|\,\boldsymbol{d})}{p(\mathrm{M}_2\,|\,\boldsymbol{d})} = \frac{\int p(\boldsymbol{d}\,|\,\boldsymbol{a},\mathrm{M}_1)\, p(\boldsymbol{a},\mathrm{M}_1)\, d\boldsymbol{a}}{\int p(\boldsymbol{d}\,|\,\boldsymbol{a},\mathrm{M}_2)\, p(\boldsymbol{a},\mathrm{M}_2)\, d\boldsymbol{a}} \times \frac{p(\mathrm{M}_1)}{p(\mathrm{M}_2)}$$

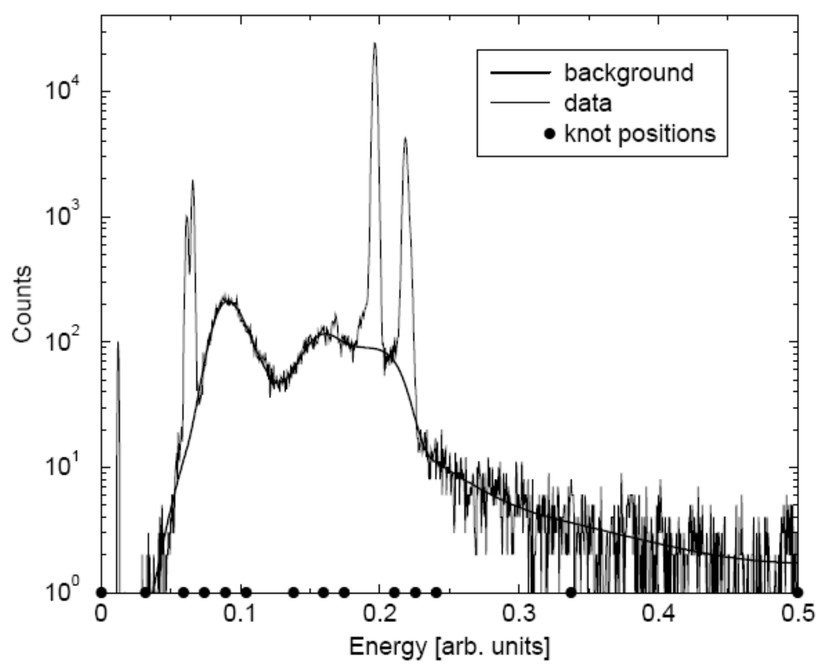
Posterior odds = Bayes factor x Prior odds

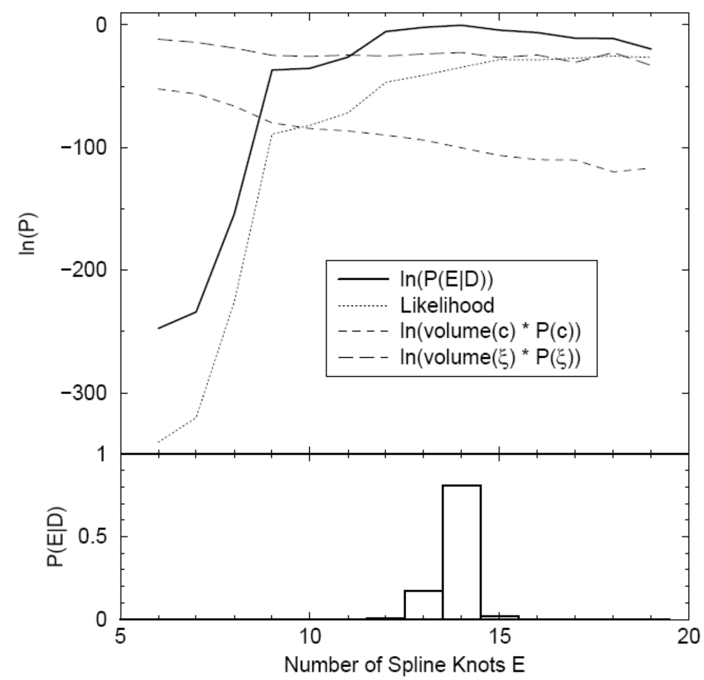
- ‣ integrals over volume of data likelihood times prior
  - may be difficult to evaluate
  - doable under Gaussian assumption with estimate of covariance matrix
- ‣ choice of prior odds important

- May be used to select best model to represent data, including
  - ‣ polynomial order, number of spline knots

Background estimation in spectral data

- Problem: estimate background for PIXE spectrum
- Approach is based on assuming background is smooth and treating resonances as outlying data
- Fully Bayesian calculation using MCMC to estimate spline parameters, their knot positions, and **number of knots**





from Fischer et al., Phys. Rev. E **61**, 1152 (2000)

Summary

- Uncertainty quantification is fundamental to validation
- Bayesian analysis provides valuable tools for UQ
- Variety of techniques are available (or being developed) for validating simulation codes simulation codes
- Hierarchical approach to conducting UQ is suggested for physics simulation codes