DAKOTA 101

DAKOTA Overview

http://dakota.sandia.gov/

Learning goals: Understand:

- **How DAKOTA interfaces with a simulation (computational model)**
- **DAKOTA input file structure; corresponding DAKOTA abstractions**
- **How to run DAKOTA and JAGUAR to perform parameter studies**
- **DAKOTA framework benefits**

Training materials can be viewed at:
http://dakota.sandia.gov/training/2011/

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.

DAKOTA Team Introductions

Brian Adams
Project Lead
1441

Mike Eldred
Research Mgr.
1441

Sophia Lefantzi
Support Mgr.
8954

Jim Stewart
Business Mgr.
1441

Keith Dalbey
1441

Bill Bohnhoff
1341

Patty Hough
8954

Laura Swiler
1441

- **Ethan Chan (8954)**
- **John Eddy (6133)**
- **John Siirola (1465)**

DAKOTA in a Nutshell

Design and Analysis toolKit for Optimization and Terascale Applications includes a wide array of algorithm capabilities to support engineering transformation through advanced modeling and simulation.

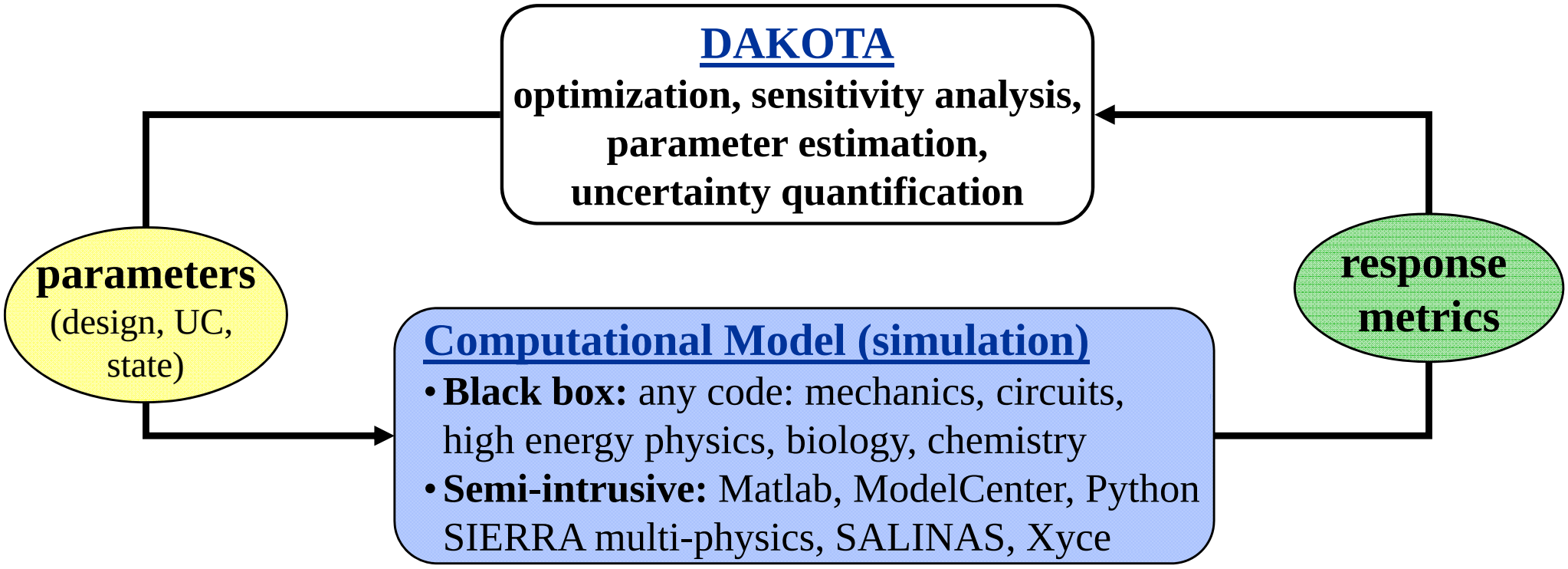
Adds value to simulation-based analysis by answering fundamental science and engineering questions:

- **What are the crucial factors/parameters and how do they affect key metrics? *(sensitivity)***
- **How safe, reliable, robust, or variable is my system? *(quantification of margins and uncertainty: QMU, UQ)***
- **What is the best performing design or control? *(optimization)***
- **What models and parameters best match experimental data? *(calibration)***

- ***All rely on iterative analysis with a computational model for the phenomenon of interest***

Automated Iterative Analysis

Automate typical "parameter variation" studies with advanced methods and a generic interface to your simulation



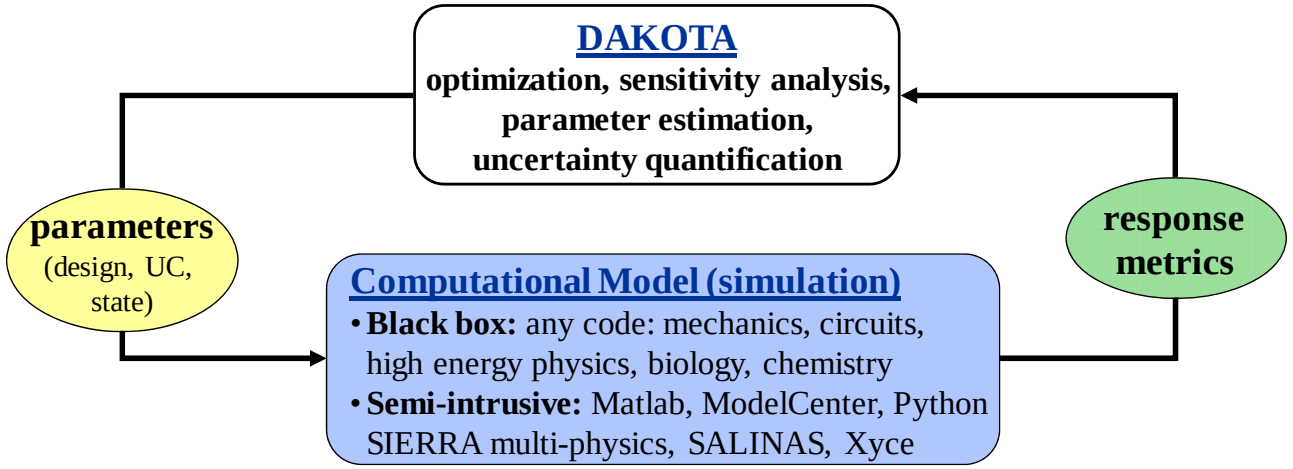
- **Can support experimental testing: examine many accident conditions with computer models, then physically test a few worst-case conditions.**

Key DAKOTA Capabilities

- Generic interface to simulations
- Time-tested and advanced algorithms to address non-smooth, discontinuous, multimodal, expensive, mixed variable, failure-prone
- Strategies to combine methods for advanced studies or improve efficiency with surrogates (meta-models)
- Mixed deterministic / probabilistic analysis
- Supports scalable parallel computations on clusters
- Object-oriented code; modern software quality practices
- Limited Windows interface (run via command prompt); however new graphical user interface. DART integration in progress.

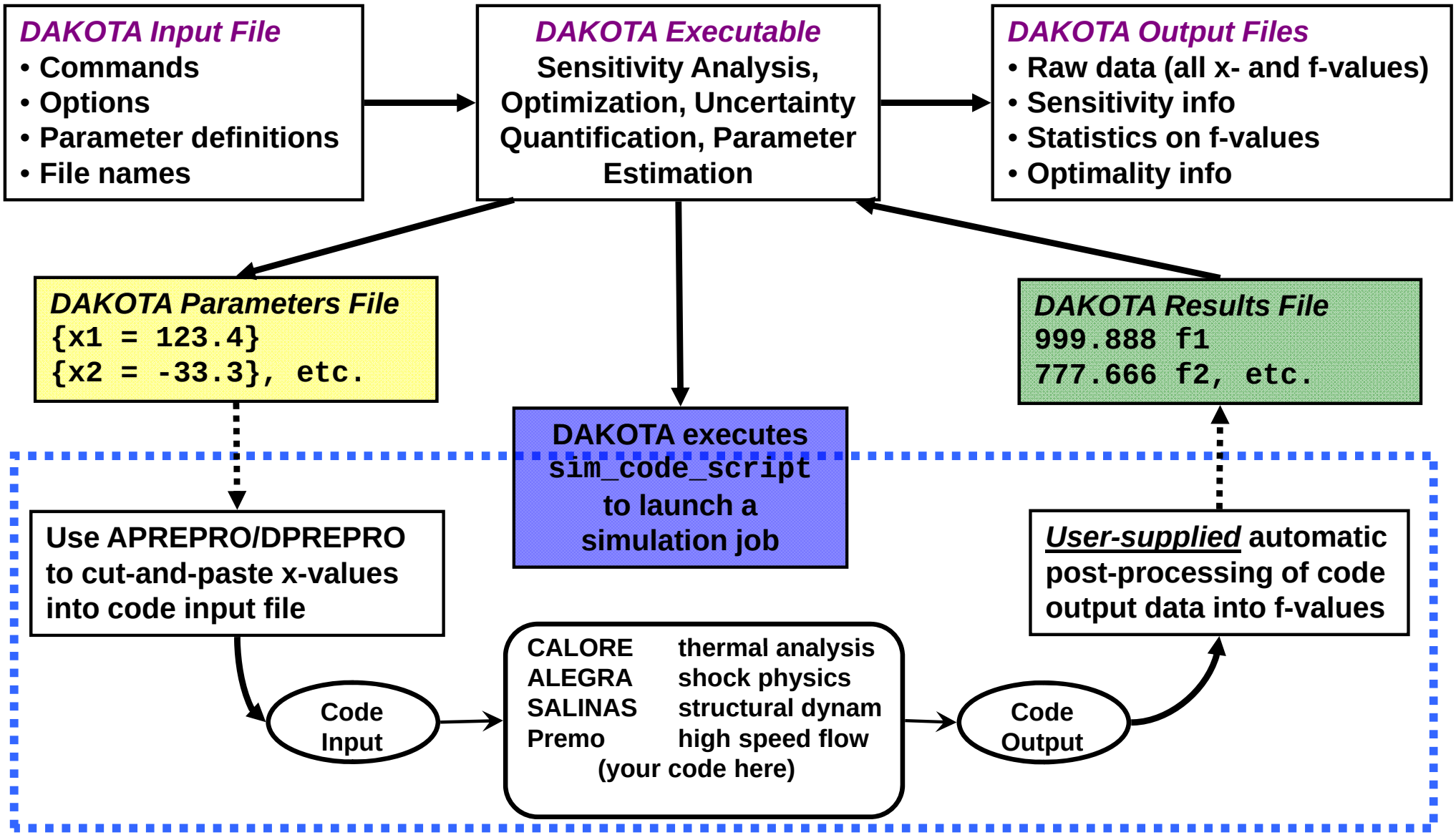
- Additional details: http://dakota.sandia.gov/
  - Extensive documentation, including a tutorial
  - Support resources: http://dakota.sandia.gov/resources.html
  - Software downloads: stable releases and nightly builds (freely available worldwide via GNU LGPL)

Overall DAKOTA 101 Goals



- **DAKOTA 101 will focus on setting up various kinds of studies to drive this iterative flow**
- **We'll use JAGUAR GUI or text file editing to create input files, with command line run**
- **For this class we'll substitute test functions from the DAKOTA distribution for the simulation**
- ***DAKOTA Integration focuses on connecting to your actual application***

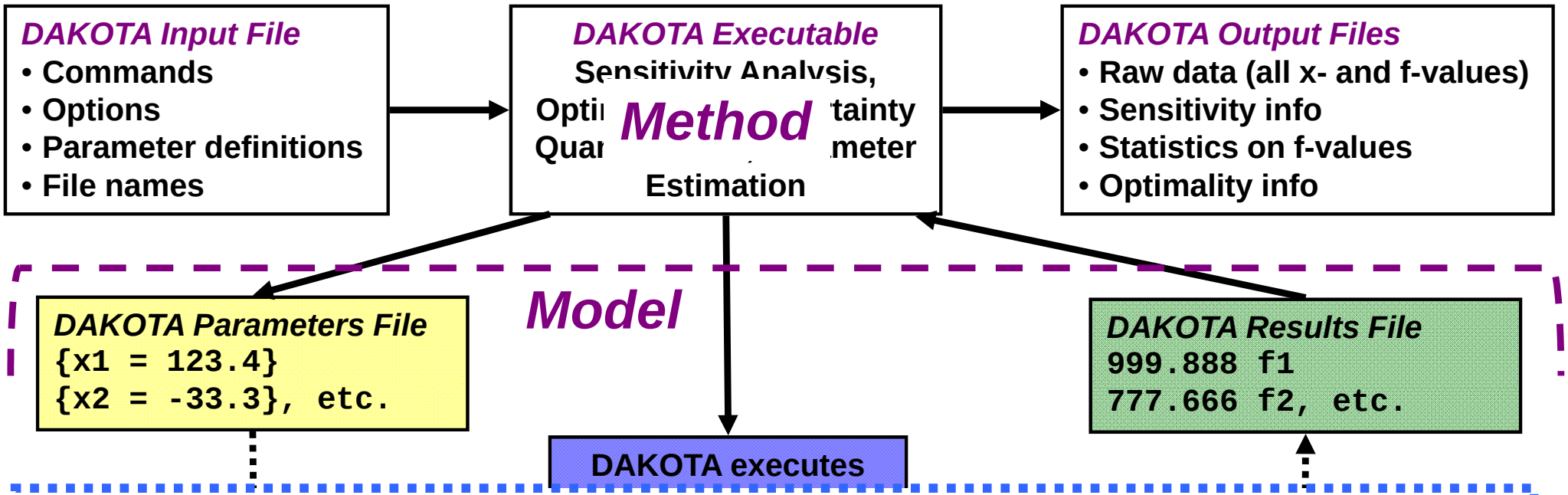
DAKOTA Execution & Info Flow



DAKOTA Application Interfacing Class

Sandia National Laboratories

DAKOTA Execution & Info Flow



Algebraic test function which "reads" parameters and writes results, e.g.,

`rosenbrock`
`text_book`
`cantilever`

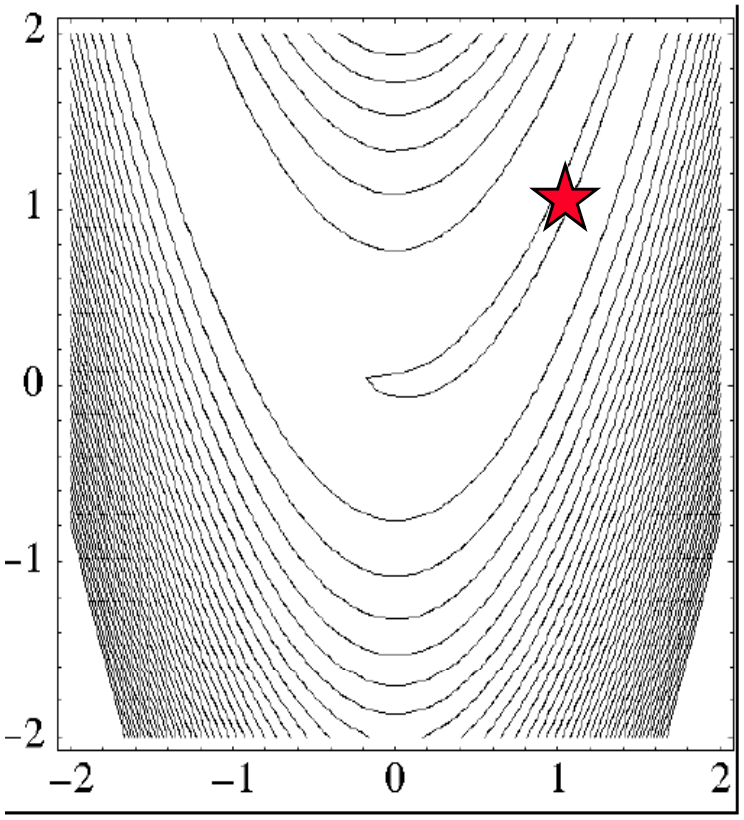
Application Stand-in: Rosenbrock "Banana" Function

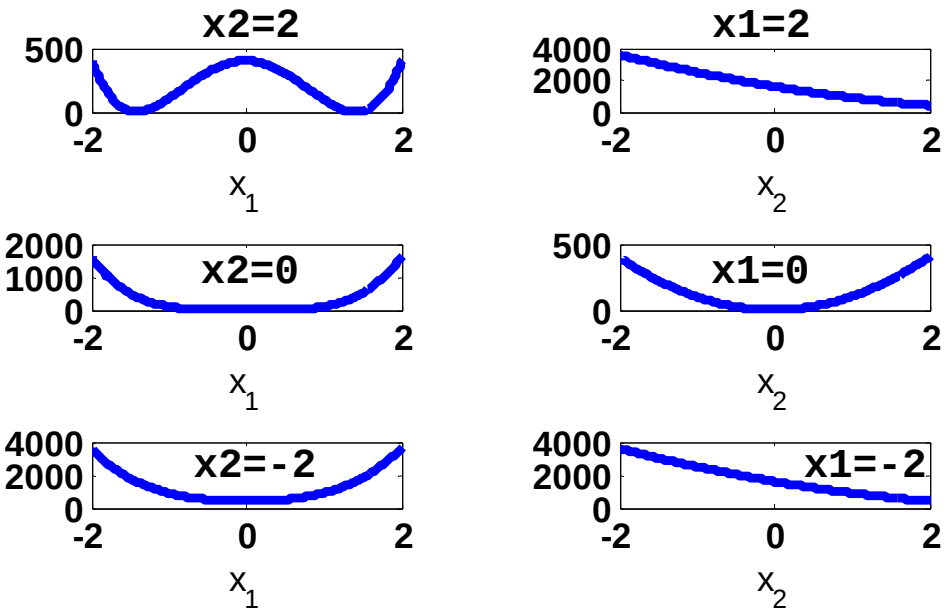
$f(x1,x2) = 100*(x2-x1*x1)^2 + (1-x1)^2$

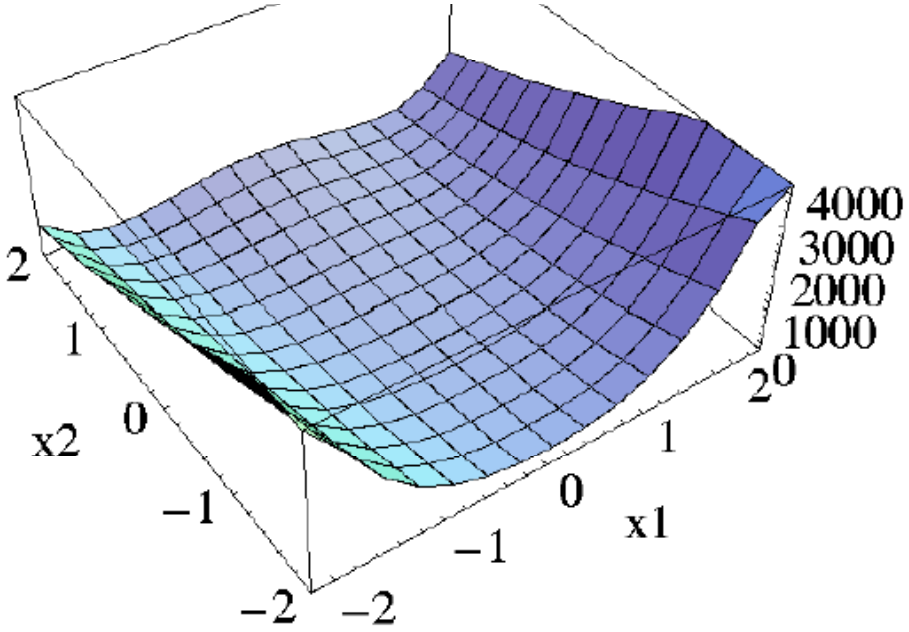
$-2 \leq x1 \leq 2$

$-2 \leq x2 \leq 2$

Minimum: $f(x1,x2) = f(1,1) = 0.0$







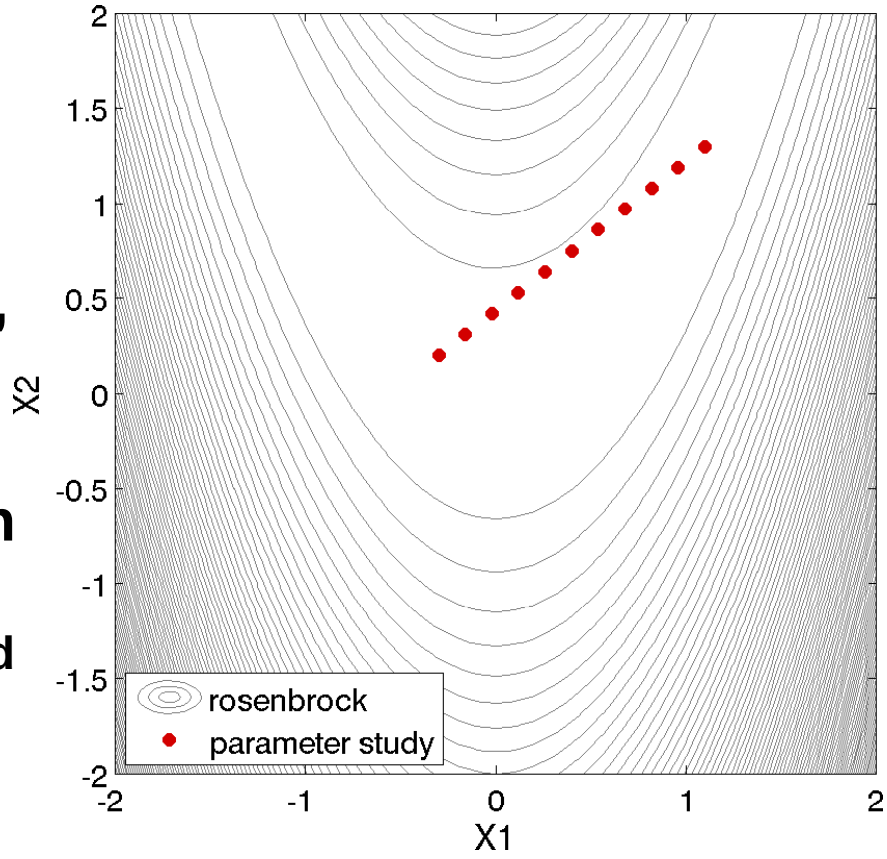
Sandia National Laboratories

Exploring Parameter-Response Relationship: Vector Parameter Study

- **GOAL: assess global trend of f(x1, x2) along a line in parameter space**

- **Example: 11 equally-spaced samples along a vector in the *x1-x2* parameter space (based on start point, end point, number of samples)**

- **Not especially useful with N=2, but can be when N>2**
  - **With large steps, provides some global trend info on function values**
  - **With small steps, provides some local trend info on f-values (quasi-derivatives)**



See User's Manual Section 2.4.1.2

Sandia National Laboratories

Exploring Rosenbrock: Running a Vector Parameter Study

- If using JAGUAR:
  - Start JAGUAR (`Jaguar` command) and set the DAKOTA executable location in Window > Preferences > Jaguar
  - Create a new file from template: "Parameter Study Vector"
- If using a text editor, copy to a location of your choice: `Dakota/examples/tutorial/dakota_rosenbrock_vector.in`
- Discuss the sections of this DAKOTA input file
- Verify the example runs (running in JAGUAR can be slower):
  - Use JAGUAR "Execute Problem" tab: "Check" and "Run"; or
  - Run `dakota -input myvector.in` from the command line (try `-check`)
- Use either editor to modify the input (vary start or end point, number steps, etc.); discuss observations
- See DAKOTA reference manual locally or at http://dakota.sandia.gov/documentation.html

Input File:
Vector Parameter Study

```
# DAKOTA INPUT FILE - dakota_rosenbrock_vector.in

strategy,
          single_method
            graphics,tabular_graphics_data

method,
          vector_parameter_study
            final_point = 1.1  1.3
            num_steps = 10

model,
          single

variables,
          continuous_design = 2
            initial_point   -0.3      0.2
            descriptors      'x1'     "x2"

interface,
          direct
            analysis_driver = 'rosenbrock'

responses,
          num_objective_functions = 1
          no_gradients
          no_hessians
```

JAGUAR Guides Creation of Dakota Input Deck

6 potential sections (2 optional, 4 required):

Define Problem

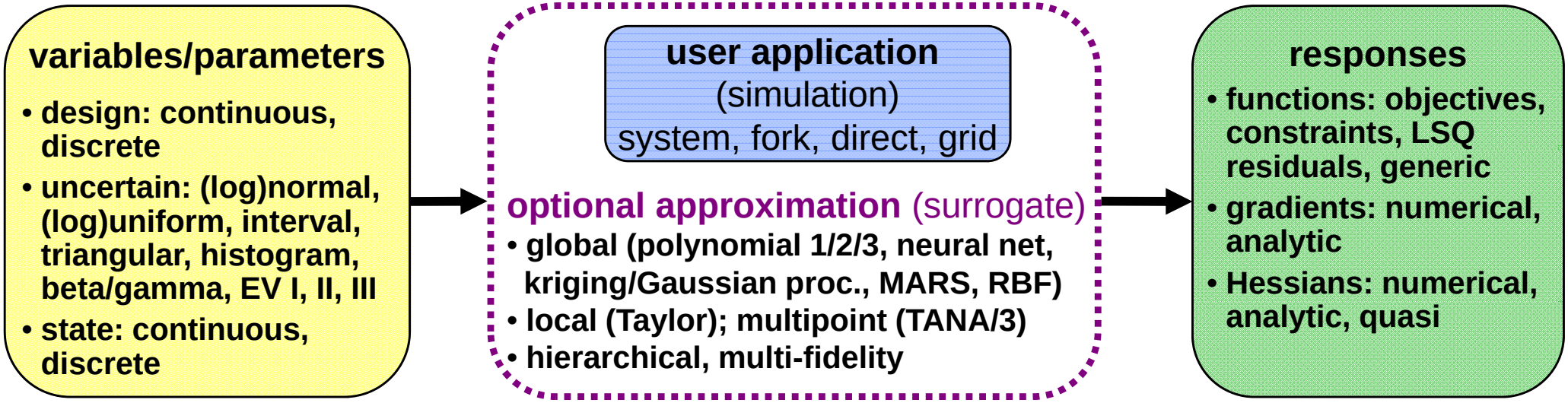
- Model *(optional):* single, surrogate (global, local, hierarchical), nested
- Variables *(required):* design, uncertain, and state variables; continuous/discrete
- Interface *(required):* system call, fork, or direct; specify parallel options
- Responses *(required):* number of responses/constraints, gradients, Hessian

Define Flow

- Strategy *(optional): coordination of methods* single_method, hybrid, multi_start, pareto_set
- Method *(required):* parameter studies, nondeterministic methods, optimization methods

Optional Info: Flexibility with Models

DAKOTA models map inputs to response metrics of interest:



integrate parameters into application inputs

extract relevant metrics

For all DAKOTA studies, must specify the:

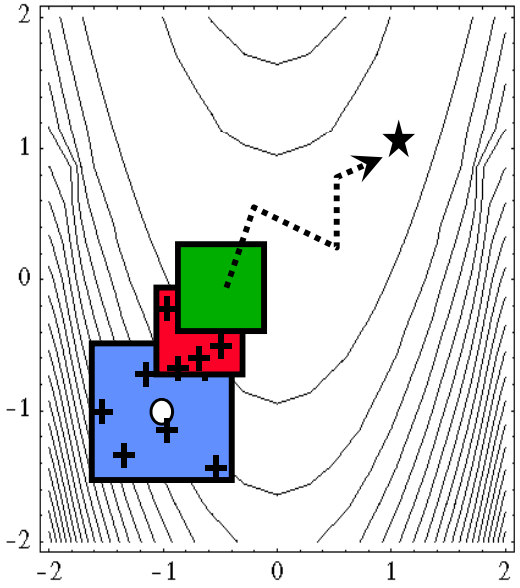
- **variables of interest (types vary by study),**
- **interface that evaluates them**
- **responses it produces**

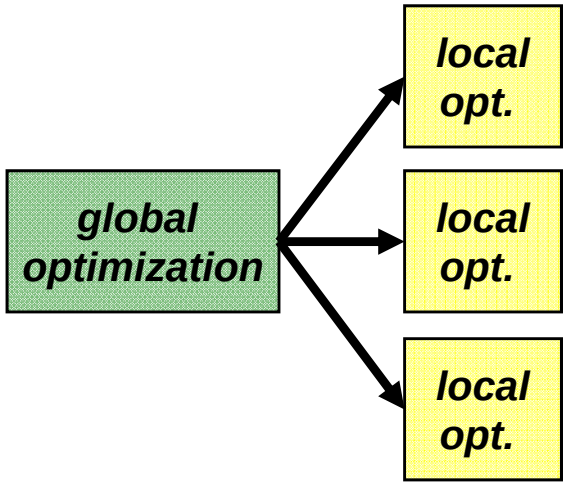
Sandia National Laboratories

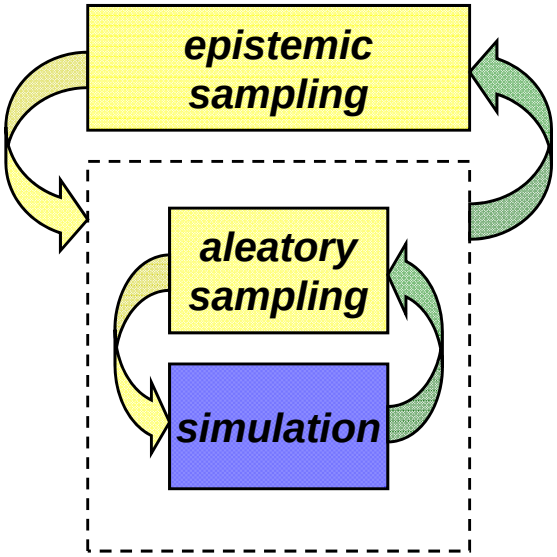
Optional Info: Strategies (and advanced/multi-component methods)

***Strategies** (nesting, layering, sequencing and recasting facilities) combine methods to enable advanced studies:*

- **Combine optimization/calibration with uncertainty quantification**
- ***Surrogate-based approaches***
- ***Hybrid optimization***
- **Nested UQ**







Sandia National Laboratories

Exploring Rosenbrock: Running a Vector Parameter Study

- If using JAGUAR:
  - Start JAGUAR (`Jaguar` command) and set the DAKOTA executable location in Window > Preferences > Jaguar
  - Create a new file from template: "Parameter Study Vector"
- If using a text editor, copy to a location of your choice: `Dakota/examples/tutorial/dakota_rosenbrock_vector.in`

- Discuss the sections of this DAKOTA input file

- **Verify the example runs (running in JAGUAR can be slower):**
  - **Use JAGUAR "Execute Problem" tab: "Check" and "Run"; or**
  - **Run `dakota -input myvector.in` from the command line (try `-check`)**
- **Use either editor to modify the input (vary start or end point, number steps, etc.); discuss observations**

- **See DAKOTA reference manual locally or at http://dakota.sandia.gov/documentation.html**

Sandia National Laboratories

Dakota Execution and Output

- DAKOTA is most commonly run from a UNIX or Windows command prompt

- Capture output: input variable and response information for each function evaluation; method-specific info

```
>> dakota –i my_run.in –o my_run.out
```

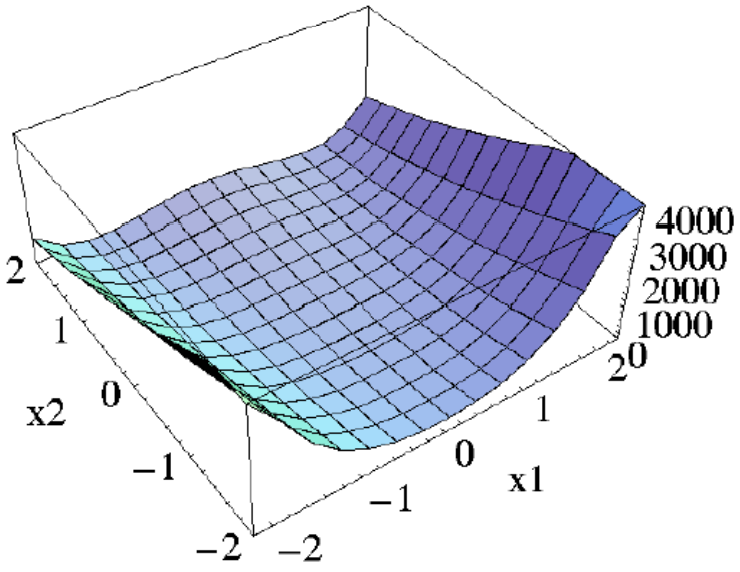
- If strategy includes `tabular_graphics_data`, generates tabular listing of inputs and outputs, called `dakota_tabular.dat`. Useful for Excel, Matlab, or other package import.

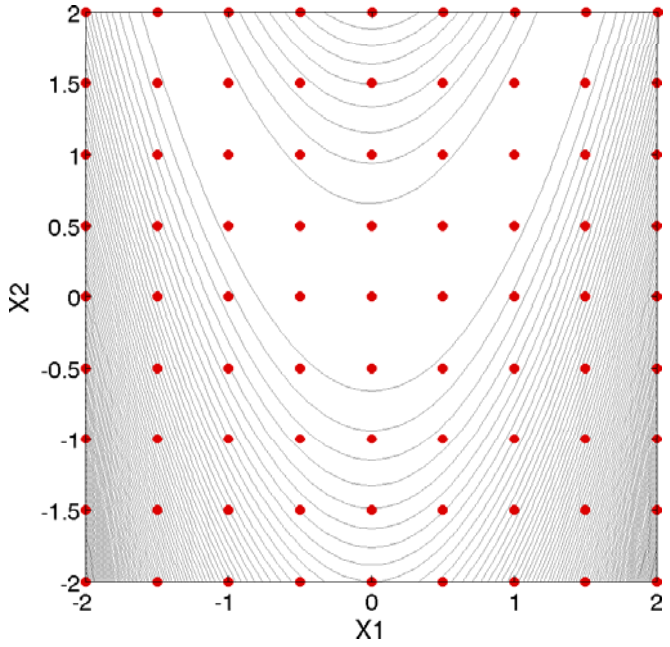
- *Other command-line options*

```
>> dakota –help
```

Exercise: Multi-dimensional Parameter Study

- **Goal: understand how response f(x1,x2) varies with respect to the inputs x1 and x2 on a grid of points.**
- ***Exercise: create or modify and run an input file to evaluate the Rosenbrock computational model at a grid of points over [-1.5, 2.5] using the multi-dimensional parameter study method***
- **Try 9 points in one dimension, 6 in the other**
- **See method and variable commands in DAKOTA reference manual**
- **What parts of the file did you have to change?**





Example: uniform grid over [-2.0, 2.0]

Sandia National Laboratories

Input File: Multi-dimensional Parameter Study

```
# rosenbrock_grid.in

strategy,
          single_method
            graphics,tabular_graphics_data

method,
          multidim_parameter_study
            partitions = 8 5

model,
          single

variables,
          continuous_design = 2
            lower_bounds    -1.5     -1.5
            upper_bounds     2.5      2.5
            descriptors      'x1'     "x2"

interface,
          direct
            analysis_driver = 'rosenbrock'

responses,
          num_objective_functions = 1
          no_gradients
          no_hessians
```

DAKOTA 101: Coming Up

Method Tour

- Sensitivity Analysis
- Uncertainty Quantification
- Optimization and Calibration
- Advanced Topics per class interest and time

Additional Topics
(revisit later today per class interest)

General features

- Restart
- Evaluation cache
- Utilities in dakota_restart_util
- Tabular graphics data
- Failure capturing: abort, retry, recover, ignore
- Constraint specification: linear, nonlinear; equality, inequality
- Input/output scaling
- Matlab interface

Approximation methods

- Global data fit surrogate methods (polynomials, MARS, Kriging, etc.)
- Local surrogate methods (Taylor series, multipoint)
- Hierarchical: high/low fidelity models
- Corrections

Strategies/Advanced approaches

- Nested models: OUU
- Multi-objective (Pareto) optimization
- Multistart; multi-level hybrid
- Surrogate-based optimization (variety of constraint handling approaches): trust region; EGO/EGRA
- Reliability-based design optimization
- Advanced UQ topics: polynomial chaos, second-order probability, Dempster-Shafer, surrogate-based UQ
- AMPL: for analytic problems / algebraic mappings

Parallel capabilities: message passing, asynchronous local, hybrid

- Asynchronous evaluations
- Dakota parallel, application serial
- Dakota serial, application parallel
- Multi-level parallel: concurrent iteration, concurrent function evaluations, concurrent analyses,
- multiprocessor simulations

Getting Started with DAKOTA

- **Access a Sandia installation: `module avail dakota`**
  AMECH (CA), CEE (ESHPC/SCICO, NM), Computer clusters (both)
  or download (see Analyst Home Page or DAKOTA webpage)

- **Supported on Linux, Solaris, AIX (purple), Mac OS X, Windows (no MinGW or Cygwin install required), Redstorm**

- **Key resource: http://dakota.sandia.gov**
  - **Extensive documentation (user, reference, developer)**
  - **Support mailing lists / archives**
  - **Software downloads: releases and nightly stable & VOTD builds (freely available worldwide via GNU GPL)**

- **User's Manual, Chapter 2: Tutorial with example input files**

- **Support:**
  - **dakota-users@software.sandia.gov**
    (DAKOTA team and internal/external user community)
  - **dakota-help@sandia.gov**
    (for SNL-specific or issues involving proprietary information)