WMT and Dakota

Mark Piper and Eric Hutton
CSDMS

Agenda

- Setup
- Uncertainty: why does it matter?
- Dakota
- Dakota in Python
- Dakota in WMT
- Summary, questions, and future work

Setup: requirements

- internet connection
- **beach** login and password
- ssh
- tutorial files:

https://github.com/mdpiper/dakota-tutorial

Setup: on beach

Connect to beach with ssh:

```
ssh [username]@beach.colorado.edu
```

Clone clinic files and configure environment:

```
git clone https://github.com/mdpiper/dakota-tutorial
cd dakota-tutorial
source scripts/setup-beach.sh
```

Uncertainty

particularly, knowledge and quantification of it,
are required to improve:

- accuracy,
- reliability, and
- accountability

in earth science models.

How can we combine knowledge of uncertainty in data and models to improve predictions?

Dakota

is

a toolkit that provides an interface between analysis codes and a library of iterative systems analysis methods.

Developer	Sandia National Labs
Current release	6.2 (May 2015)
License	LGPL
Website	https://dakota.sandia.gov

Dakota

helps answer questions such as:

What are the most influential parameters in my model? → sensitivity analysis

How robust and reliable is my model? → uncertainty quantification

What is the best performing design? → optimization

What model parameter values best match experimental data? → calibration

Dakota

has an extensive library of analysis techniques, including:

- parameter studies
- design of experiments
- design of computer experiments
- sampling (MC or LHS)
- local and global reliability
- adaptive sampling
- stochastic expansion
- epistemic methods
- gradient-based local and global optimization
- derivative-free local and global optimization
- nonlinear least squares
- surrogate models

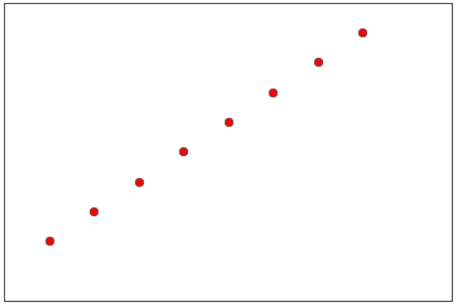
Parameter studies (PS)

are
one type of sensitivity analysis.

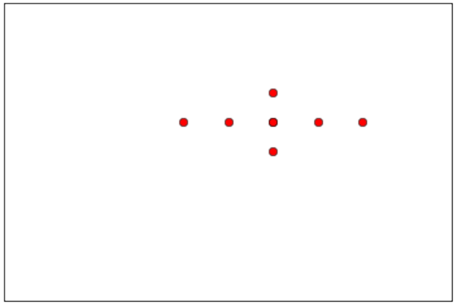
Parameter studies explore a model's response to the deterministic modification of its input parameter values.

PS methods used in this clinic:

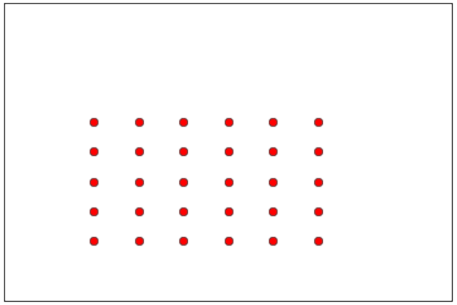
vector



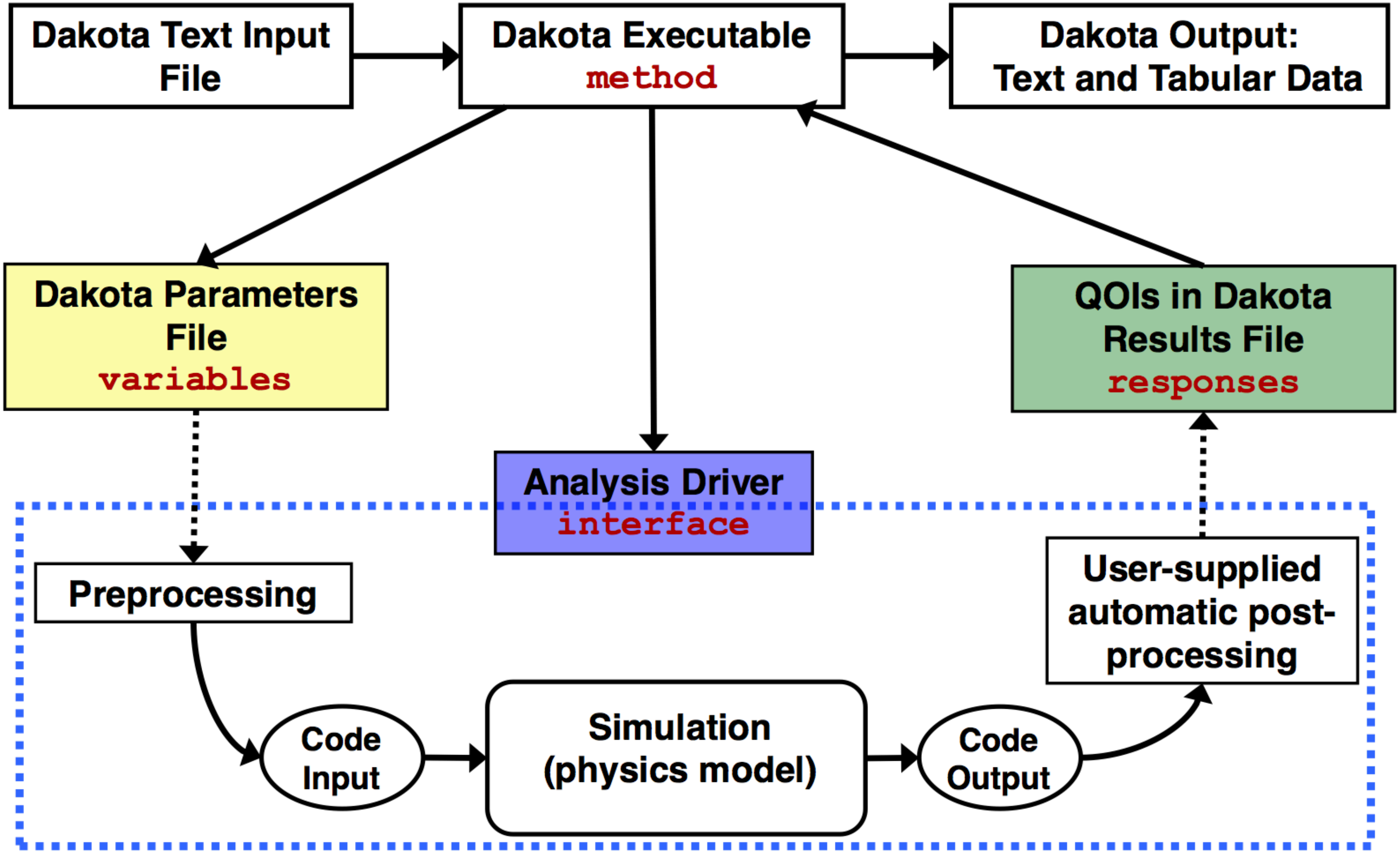
centered



multidim



Dakota execution model



(image courtesy J. Adam Stephens and Laura Swiler, SNL)

Example 1

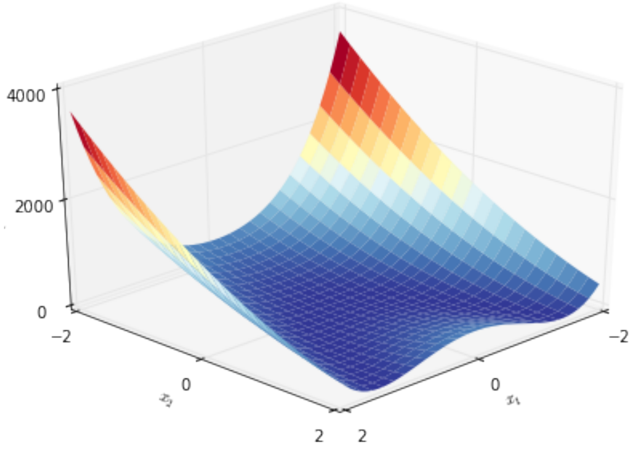
Summary: Use Dakota at the command line to minimize the Rosenbrock function

Purpose: Show the Dakota calling syntax, as well as the structure of input, output and data files

Method: Vector parameter study

Rosenbrock function

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$



IPython Notebook

Example 2

Summary: Use Dakota at the command line to explore the effects of temperature and precipitation on suspended sediment load in HydroTrend

Purpose: Show an example of an analysis driver

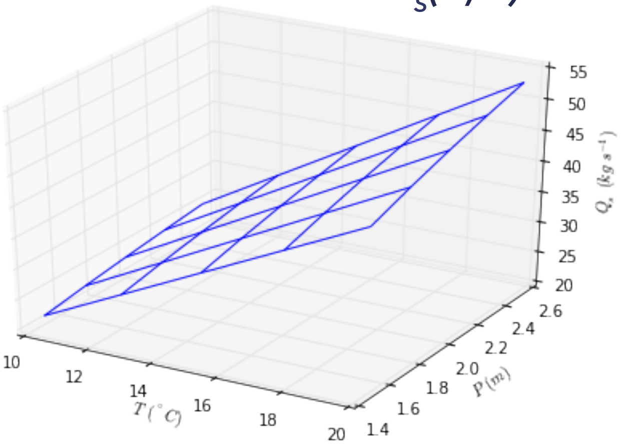
Method: Multidim parameter study

HydroTrend sediment load model

$$Q_s = \omega B Q^{0.31} A^{0.5} R T$$

IPython Notebook

$\overline{Q}_s(T,P)$



Dakota in Python

CSDMS is currently developing
a Python interface for calling Dakota.

Why?

- Simplifies the setup of Dakota experiments
- First step in writing a BMI for Dakota

Developer	CSDMS
Current release	0.1 (May 2015)
License	MIT
Website	https://github.com/csdms/dakota

Dakota in Python

- Features:
  - ☑ writes Dakota input file
  - ☑ implements generic analysis driver
  - ☑ communicates with Dakota through auxiliary configuration file
- Drawbacks:
  - □ only vector, centered, and multidim parameter studies are supported (currently)
  - □ HydroTrend is only supported model (currently)
  - □ restricted to basic NumPy statistics on responses
  - □ not coupleable

Dakota in Python

Basic workflow:

```
from dakota.core import Dakota
d = Dakota(method='vector_parameter_study')
d.write_configuration_file()
d.write_input_file()
d.run()
```

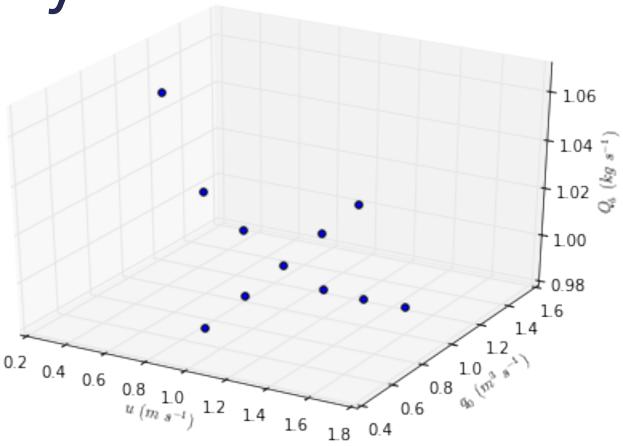
Example 3

Summary: Call Dakota from Python to explore how varying mean river velocity and constant annual base flow affect median daily bedload values in HydroTrend

Purpose: Show the syntax of configuring and running Dakota through the Python interface

Method: Centered parameter study

$<Q_b(u,q_0)>$



IPython Notebook

Dakota in WMT

CSDMS is currently developing
an interface for Dakota in WMT.

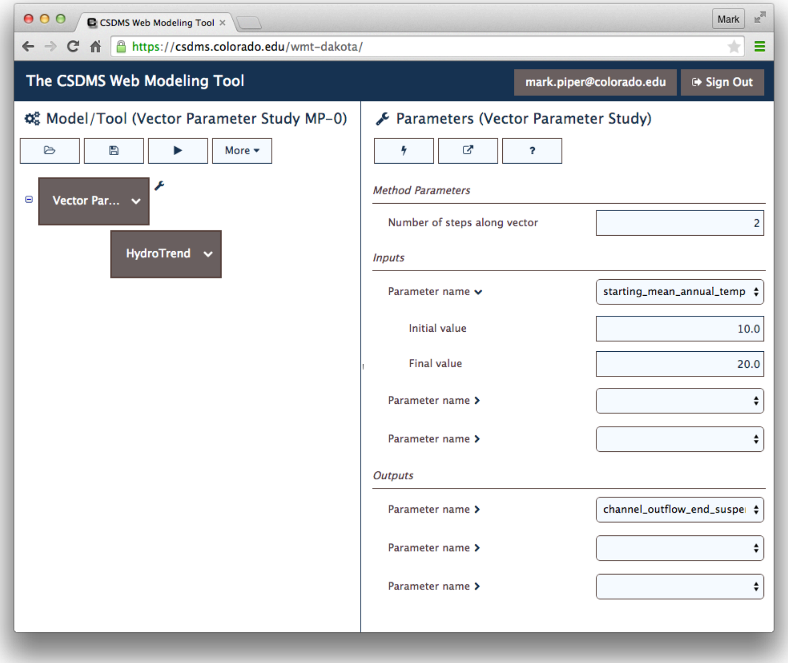
Try it at:
https://csdms.colorado.edu/wmt-dakota

Development prototype

☑ It works!

□ It's limited:

- only parameter studies
- only HydroTrend
- limited responses
- not coupleable


CSDMS Web Modeling Tool
Mark
https://csdms.colorado.edu/wmt-dakota/
The CSDMS Web Modeling Tool
mark.piper@colorado.edu
Sign Out
Model/Tool (Vector Parameter Study MP-0)
More
Vector Par...
HydroTrend
Parameters (Vector Parameter Study)
?

Method Parameters

Number of steps along vector: 2

Inputs

Parameter name: starting_mean_annual_temp

Initial value: 10.0

Final value: 20.0

Parameter name:

Parameter name:

Outputs

Parameter name: channel_outflow_end_suspe

Parameter name:

Parameter name:

WMT login

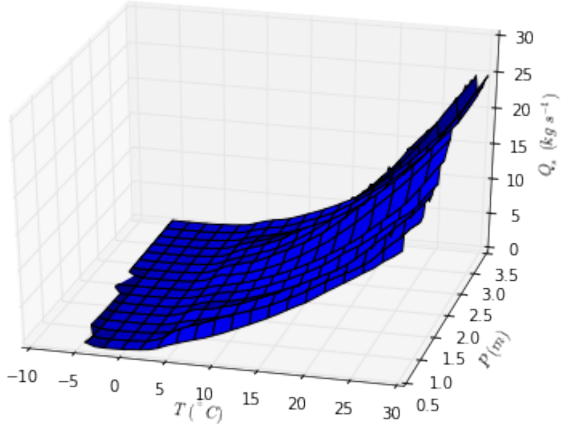
Example 4

Summary: Use Dakota in WMT to explore the effects of varying temperature and precipitation on suspended sediment load in HydroTrend

Purpose: Show how Dakota can be called through WMT

Method: Multidim parameter study

$\overline{Q_s}(T,P)$



This 40 x 30 multidim parameter study uses 1271 calls to HydroTrend!

IPython Notebook

Summary

CSDMS has developed prototype Dakota interfaces for Python and for WMT.

- □ Only a few Dakota analysis methods have been implemented
- □ Only one CSDMS model is supported
- □ Responses returned to Dakota are limited
- □ They're not coupleable in the CSDMS framework
- ☑ They work
- ☑ They make Dakota easier to use

Future work

- Bring the Python and Dakota prototypes into the CSDMS framework
  - easier to add components
  - faster execution
- Add Dakota analysis methods
- Write documentation
- Explore Dakota library API

Thank you!

Mark Piper and Eric Hutton
CSDMS

Explore!

- Please take a few minutes to explore how Dakota works in Python and in WMT.
- Would you use these interfaces? We’d appreciate your feedback!