DAKOTA 101

Optimization and Calibration

http://dakota.sandia.gov

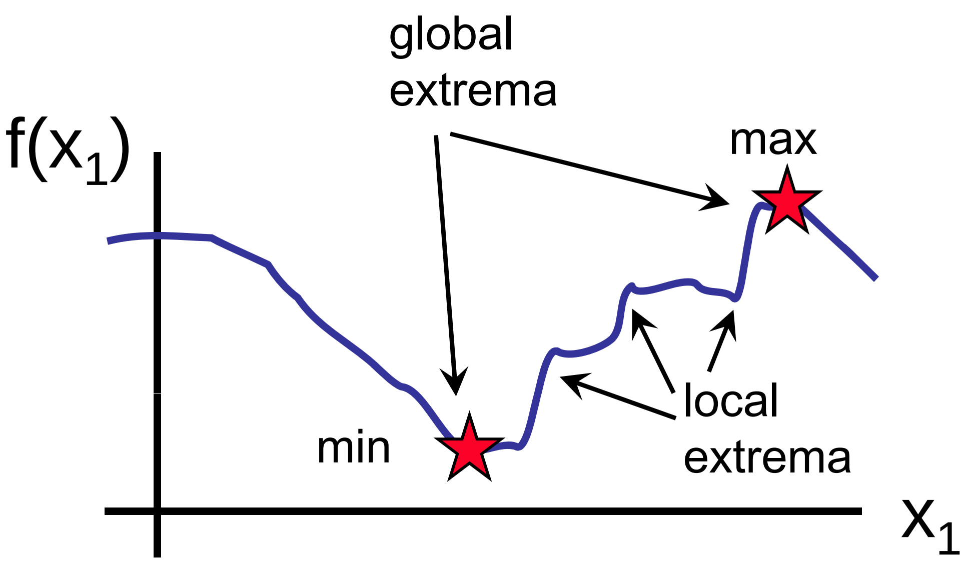
Learning Goals:

- Understand goals of optimization and solution approaches
- Use DAKOTA methods to design the cantilever beam
- Survey problem categories/considerations for method selection
- Know why calibration / parameter estimation problems and solutions are specialized optimization
- Calibrate cantilever beam model to synthetic data

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.

Optimization

- **GOAL: Vary parameters to extremize objectives, while satisfying constraints to find (or tune) the best design, estimate best parameters, analyze worst-case surety, e.g., determine:**
  - **delivery network maximizing profit / minimizing environ. impact**
  - **case geometry that minimizes drag and weight, yet is sufficiently strong and safe**
  - **material atomic configuration of minimum energy**
  - **fuel re-loading pattern yielding the smoothest nuclear reactor power distribution while maximizing output**



Some applications: local improvement suffices; others: must find global minimum at any cost

Typical Challenges for Simulation-based Optimization

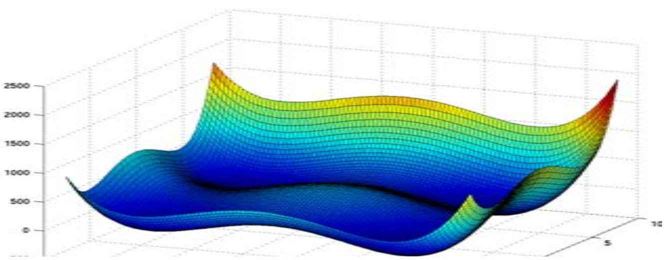
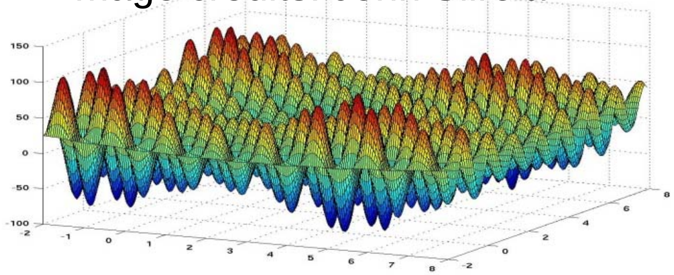
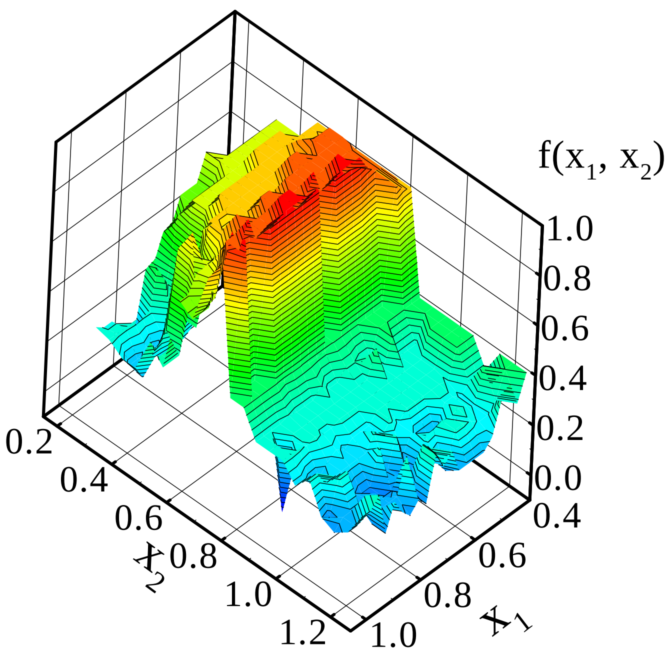


Image credits: John Siirola





In science and engineering problems of interest, we typically have:

- **no explicit function for $f(x_1,x_2)$**
  - **can't leverage algebraic structure**
- **limited number of evaluations/samples**
  - **expensive to evaluate $f(x_1,x_2)$ (long runtime even on many processors)**
  - **simulation may fail (hidden constraints)**
- **noisy / non-smooth**
  - **can't reliably estimate derivatives**
- **local extrema, non-convex**
  - **globally optimal solutions challenging**

Considerable research has been done and implemented in DAKOTA to mitigate these issues.

Sample Optimization Approaches

Gradient Descent

("go downhill")

- **efficient/scalable for smooth problems; descent direction from derivative**
- **requires analytic or numerical derivatives**
- **local convergence**

Pattern Search

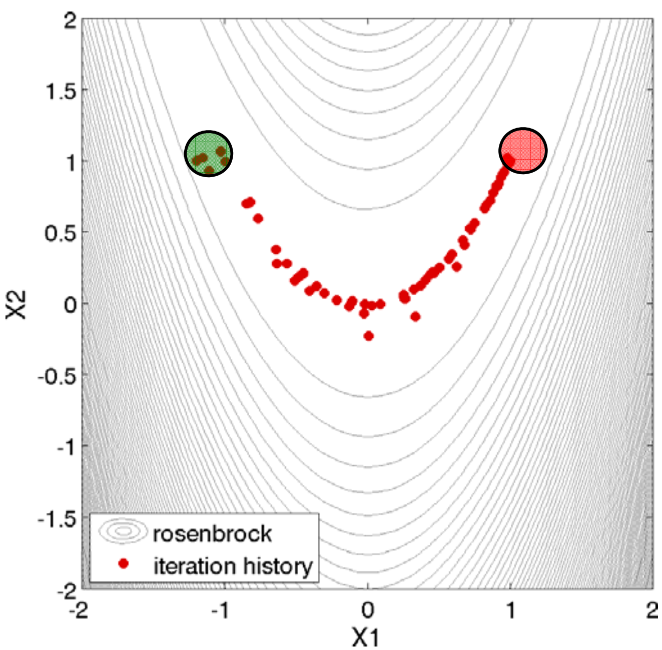
(derivative-free local)

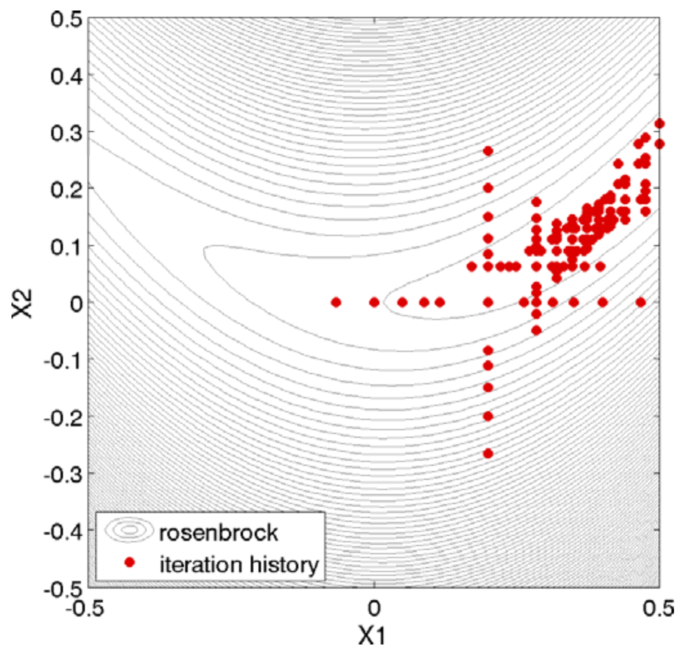
- **stencil expansion/ contraction-based**
- **good for noisy; unreliable or expensive derivatives**
- **converge to local minimum**

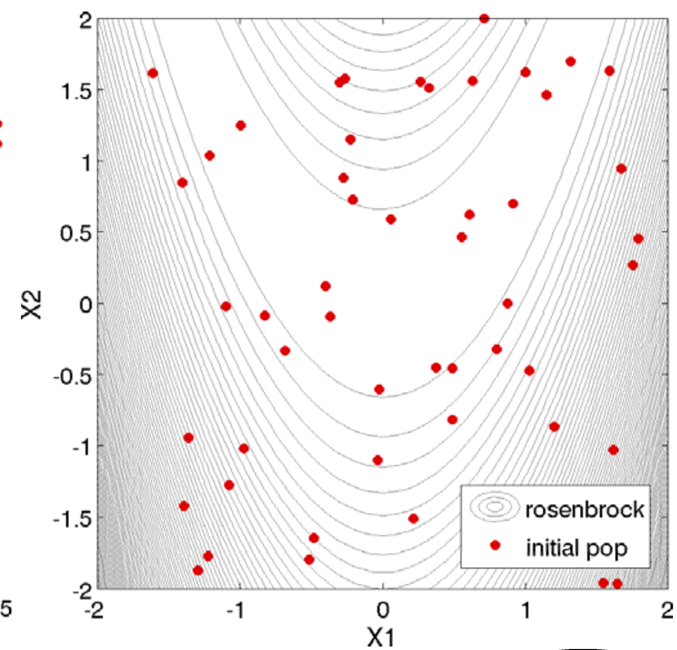
Genetic Algorithm

(derivative-free global)

- **well-dispersed initial population**
- **balance global vs. local search with genetic fitness principles**
- **typically costly**







Sandia National Laboratories

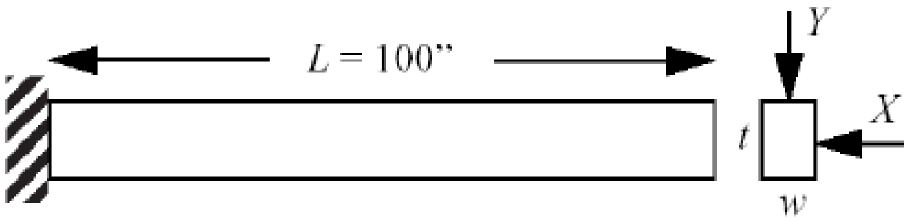
Problem Formulation: Objectives and Constraints

Application information with which to configure the solver:

Minimize:	$f(x_1, \ldots, x_N)$	***Objective function(s)****
Subject to:	$g_{LB} \leq g(x) \leq g_{UB}$	***Nonlinear inequality constraints***
	$h(x) = h_E$	***Nonlinear equality constraints***
		(Metrics above are typically implicit: computed by/extracted from a simulation code)
		(Algebraic metrics below are typically specified directly to an optimization solver)
	$A_I x \leq b_I$	***Linear inequality constraints***
	$A_E x = b_E$	***Linear equality constraints***
	$x_{LB} \leq x \leq x_{UB}$	***Bound constraints***

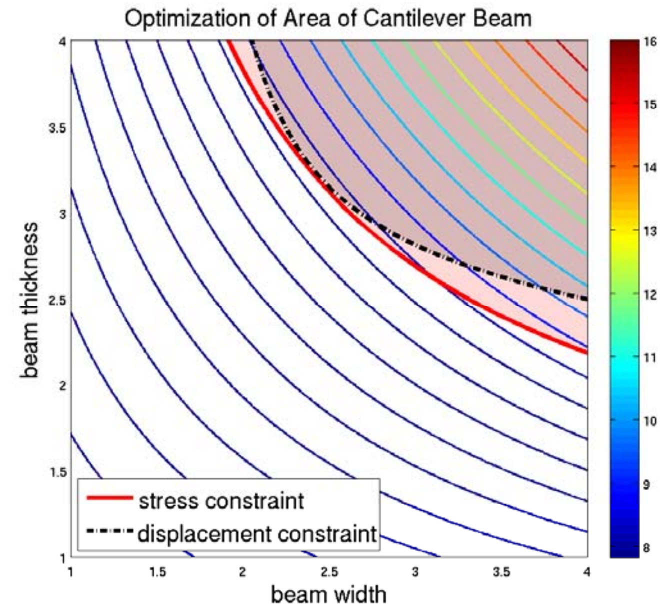
***** In practice, multiple f-values can comprise the objective function ("multi-objective optimization"), and there can be multiple constraints of each type.***

Deterministic Optimization for Cantilever Beam



$$stress = \frac{600}{wt^2} Y + \frac{600}{w^2 t} X - R \le 0$$

$$displacement = \frac{4L^3}{Ewt} \sqrt{\left(\frac{Y}{t^2}\right)^2 + \left(\frac{X}{w^2}\right)^2} - D_0 \le 0$$



- **Create DAKOTA study to minimize area subject to constraints**
  `1.0 ≤ beam_width ≤ 4.0, 1.0 ≤ beam_thickness ≤ 4.0, stress ≤ 0, displacement ≤ 0`
- **Use nominal (state variables): R=40000, E=2.9e7, X=500, Y=1000**
- **Use CONMIN MFD method (could modify or borrow from template Optimization Local Constrained GradientBased)**
- **Responses: 1 objective (area), 2 nonlinear inequality constraints**
- **Try analytic vs. numerical gradients**
- **Compare to Asynchronous Pattern Search, Coliny EA**

Potential Solution: Cantilever Optimization

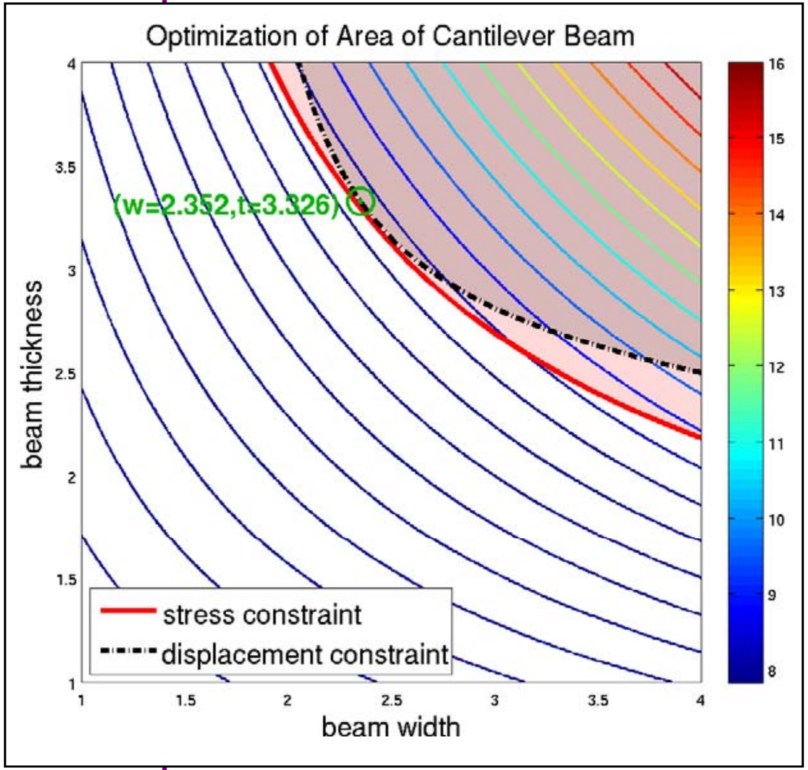
```
# extraexamples/cantilever_optimization.in
# Perform deterministic optimization with uncertainties at nominal

method
  conmin_mfd

variables
  continuous_design = 2
    upper_bounds    4.0                   4.0
    initial_point   2.5                   2.5
    lower_bounds    1.0                   1.0
    descriptors     'beam_width'          'beam_thickness'
  # Fix at nominal
  continuous_state = 4
    initial_state   40000      2.9e7      500        1000
    descriptors     'R'        'E'        'X'        'Y'

interface
  direct
    analysis_driver = 'mod_cantilever'

responses
  num_objective_functions = 1
  num_nonlinear_inequality_constraints = 2
    descriptors = 'area' 'stress' 'displacement'
  analytic_gradients
  no_hessians
```



Note: given uncertainty in *R*, this design might not satisfy constraints.

Sandia National Laboratories

Considerations: Choosing an Optimization Method

Key considerations (see DAKOTA User's Manual "Usage Guidelines")

- Trend and smoothness (perform local and global sensitivity analysis)
- Simulation expense
- Constraint types present; single or multi-objective
- Goal: local optimization (improvement) or global optimization (best possible)
- Variable types present (real, integer, categorical)
- Any special structure, e.g., quadratic objective, highly linearly constrained

Unconstrained or bound-constrained problems

- Smooth and cheap: nearly any method but gradient-based will be fastest
- Smooth and expensive: gradient-based methods
- Nonsmooth and cheap: non-gradient methods such as pattern search (local opt), genetic algorithms (global opt), DIRECT (global opt), or surrogate-based optimization (quasi local/global opt)
- Nonsmooth and expensive: surrogate-based optimization (SBO)*

Nonlinearly-constrained problems

- Smooth and cheap: gradient-based methods, though direct search works too
- Smooth and expensive: gradient-based methods
- Nonsmooth and cheap: non-gradient methods w/ penalty functions, SBO
- Nonsmooth and expensive: SBO

Extra Examples: Optimization Problems and Methods

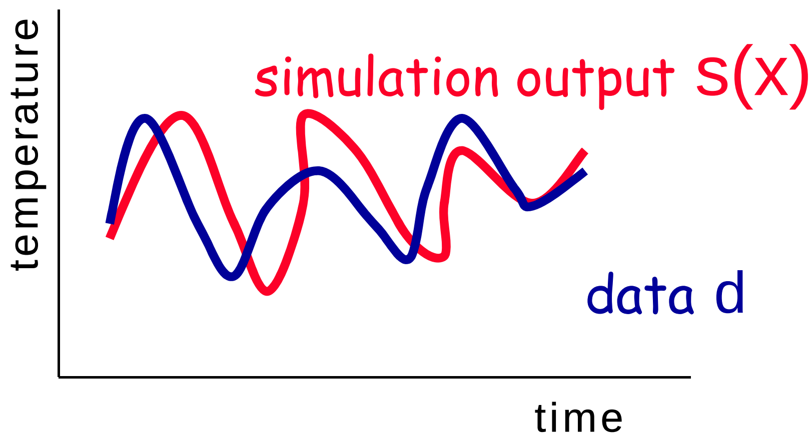
- **Constrained**
  - Minimize an objective given constraints
  - Exercise: See template Optimization Local Constrained GradientBased
- **Multi-start local**
  - Provide multiple starting points to a local optimizer to find multiple local minima
  - Exercise: See template Optimization Local MultiStart
- **Global**
  - Find the global extreme value
  - Exercise: See template Optimization Global Evolutionary Algorithm
- **Multi-objective**
  - Optimize across multiple competing objectives
  - Exercise: See template Optimization Local MultiObjective (modify to use optpp_q_newton method)
- **Surrogate-based/multifidelity**
  - Reduce the computational cost (i.e., number of function evaluations) of optimization
  - Exercise: See template Optimization Global Surrogate
- **Hybrid**
  - Use multiple optimization methods to solve a single problem
  - Exercise: See template Optimization Hybrid Textbook Example

Special Optimization Case: Calibration (Parameter Estimation)

- **Calibration: Adjust model parameters x to maximize agreement with a set of experimental data.**
- **A.K.A. parameter estimation, parameter identification, systems identification, nonlinear least-squares, inverse problem.**

Minimize

$$\boxed{f(x) = \sum_{i=1}^{n} (\underbrace{s_i(x)}_{\text{simulation output that depends on x}} - \underbrace{d_i}_{\text{given data}})^2}$$



Why use calibration?

- Tune a model to experimental or trusted simulation data to
  - ensure sufficient simulation code predictive capability
  - decrease the amount of info lost due to using a model instead of the “truth” (minimize discrepancy)
  - gain understanding of design space
  - find parameters yielding improved model robustness

- Calibration is not validation! Separate data should be used to assess whether a calibrated model is valid.

- Calibration (inverse) problems often suffer from non-uniqueness or lack of identifiability of some parameters

Nonlinear Least Squares

- **Calibration problems are often formulated to minimize the two norm of the error between the model and data: *minimize***

$$f(x) = \frac{1}{2} r(x)^T r(x) = \frac{1}{2}\left[s(x) - d\right]^T \left[s(x) - d\right] = \frac{1}{2}\sum_{i=1}^{n}\left(s_i(x) - d_i\right)^2$$

- **Any optimizer can be applied to the sum of squared residuals *f(x)***
- **Variance-weighted and Bayesian calibration also popular**
- **A specialized class of local derivative-based optimization algorithms exploit least squares structure for efficient solution without second derivative information**

- **Example: `osborne1` analytic test problem, with *i = 1,…,33*:**

$$r_i(x) = \underbrace{\left(x_1 + x_2 e^{t_i x_4} + x_3 e^{t_i x_5}\right)}_{\text{model/simulation}} - \underbrace{d_i}_{\text{data}}; \quad t_i = -10(i-1)$$

Exercise: Calibrate Cantilever to Experimental Data

- **Calibrate design variables *E, w, t* to data from all 3 responses**
- ***X, Y, R* fixed (state) at nominal values**
- **Use NL2SOL or OPT++ Gauss-Newton**
- **Key DAKOTA specs:**
  - **num_least_squares_terms = 3**
  - **no constraints**
  - **least_squares_datafile**
- **Possible template: Calib. Local Data File**

DATA	clean	with error
area	7.5	7.772
stress	2667	2658
displacement	0.309	0.320

```
cantilever_clean.dat
cantilever_witherror.dat
```

- ***For least-squares methods, application normally must return residuals $r_i(x) = s_i(x) - d_i$ to DAKOTA***
- ***Here we return the usual area, stress, displacement and specify a datafile and DAKOTA computes the residuals***

- **How do the calibrated parameter values differ with clean vs. noisy data?**
- **How do the confidence intervals differ?**
- **Bonus: experiment with scaling variables, responses**

Sandia National Laboratories

Potential Solution: Cantilever Least-Squares

```
# Calibrate to area, stress, and displacement data generated with
# E = 2.85e7, w = 2.5, t = 3.0

method
  nl2sol
    convergence_tolerance = 1.0e-6

variables
  continuous_design = 3
    upper_bounds  3.1e7 10.0 10.0
    initial_point 2.9e7 4.0  4.0
    lower_bounds  2.7e7 1.0  1.0
    descriptors   'E' 'beam_width' 'beam_thickness'
  # Fix at nominal
  continuous_state = 3
    initial_state 40000 500 1000
    descriptors 'R' 'X' 'Y'

interface
  direct
    analysis_driver = 'mod_cantilever'

responses
  num_least_squares_terms = 3
#    least_squares_data_file = 'cantilever_clean.dat'
    least_squares_data_file = 'cantilever_witherror.dat'
    descriptors = 'area' 'stress' 'displacement'
  analytic_gradients
  no_hessians
```

```
CIs without error:
E: [ 2.850e+07, 2.850e+07 ]
w: [ 2.500e+00, 2.500e+00 ]
t: [ 3.000e+00, 3.000e+00 ]
```

```
CIs with error:
E: [ 1.992e+07, 4.190e+07 ]
w: [ 1.962e+00, 3.918e+00 ]
t: [ 1.954e+00, 3.309e+00 ]
```

Summary:
DAKOTA Optimization Methods

Gradient-based methods

(DAKOTA will compute finite difference gradients and FD/quasi-Hessians if necessary)

- *DOT (various constrained)*
- CONMIN (FRCG, MFD)
- *NPSOL (SQP)*
- *NLPQL (SQP)*
- OPT++ (CG, Newton)

Calibration (least-squares)

- NL2SOL (GN + QH)
- *NLSSOL (SQP)*
- OPT++ (Gauss-Newton)

Derivative-free methods

- COLINY (PS, APPS, Solis-Wets, COBYLA2, EAs, DIRECT, *NGSA-II, MILocal*)
- JEGA (single/multi-obj GAs)
- EGO (efficient global opt via Gaussian Process models)
- DIRECT (Gablonsky)
- OPT++ (parallel direct search)
- *TMF*

Pareto, Hybrid, Multi-start, Surrogate-based local and global