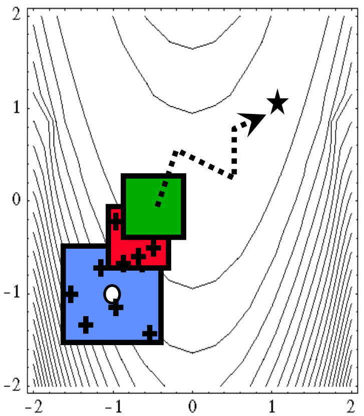
DAKOTA Advanced Topics: Hybrid and Advanced Algorithms

http://dakota.sandia.gov

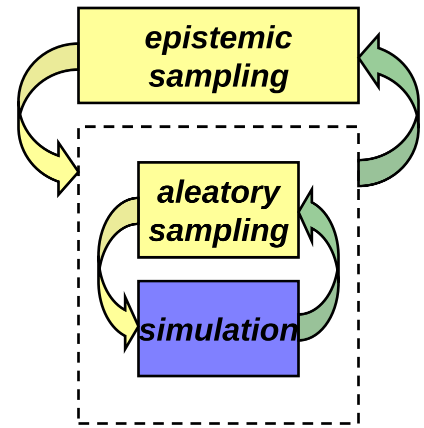
Opportunities for Mixing and Matching Methods



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

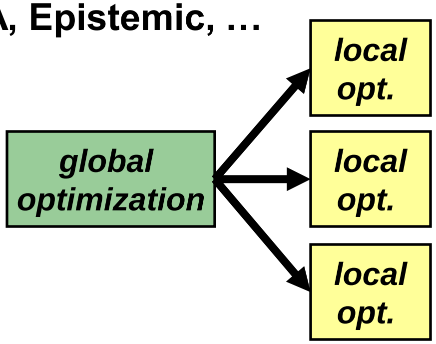
- opt within opt (multilevel opt & hierarchical MDO)
- UQ within UQ (second-order probability)
- UQ within opt (OUU) and NLS (MCUU)
- opt within UQ (uncertainty of optima)

with and without surrogate model indirection



Optimization

- Surrogate-based: data fit, multifidelity, ROM
- Mixed integer nonlinear programming (MINLP): PEBBL (parallel branch and bound)
- Optimization under uncertainty
  - TR-SBOUU, RBDO (Bi-level, Sequential)
  - MCUU, PC-BDO, EGO/EGRA, Epistemic, …
- Hybrids (e.g., global/local)
- Pareto set
- Multi-start
- Multilevel methods



Uncertainty

- Second order probability
- Uncertainty of optima

Nonlinear least squares

- Surrogate-based calibration
- Model calibration under uncertainty

Need to think of relationships between DAKOTA input blocks

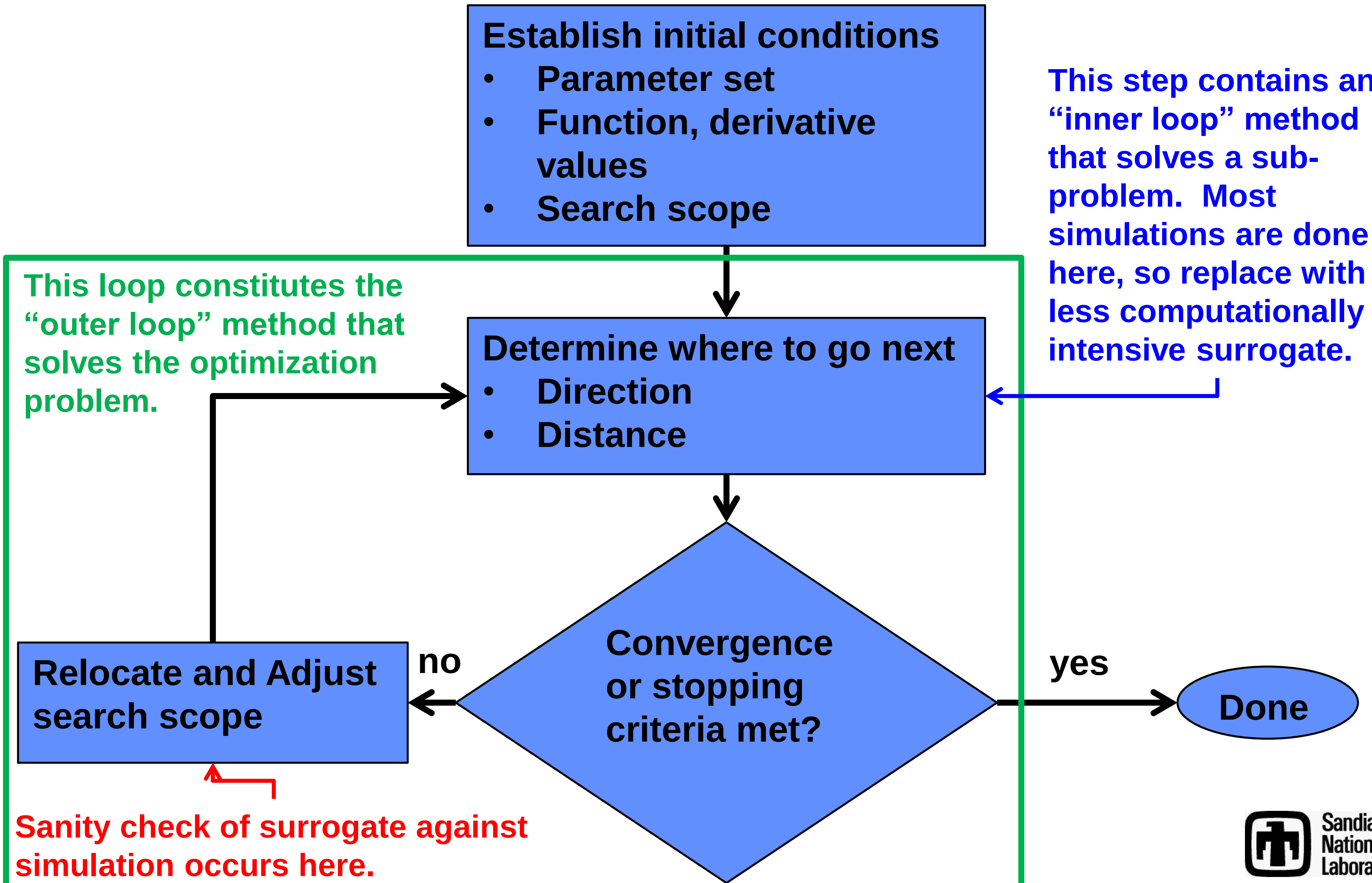
- Strategy
  - Consists of a method or set of methods
- Method
  - Operates on a model
- Model has
  - Variables/parameters
  - Responses
  - Interface

There may be more than one of these in a DAKOTA input file.

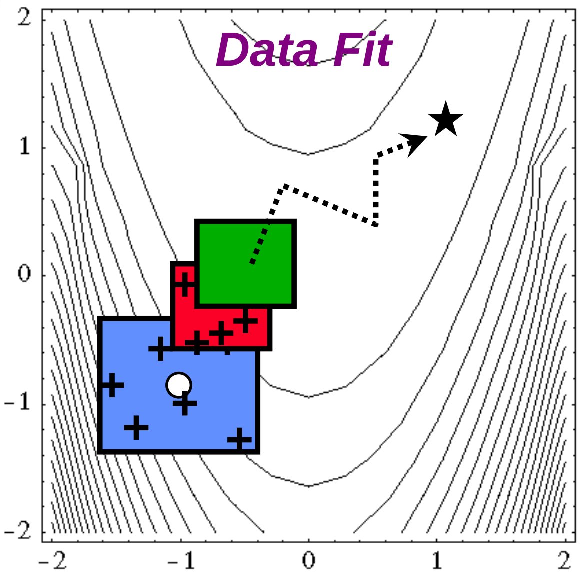
Additionally, methods and models may be “layered”

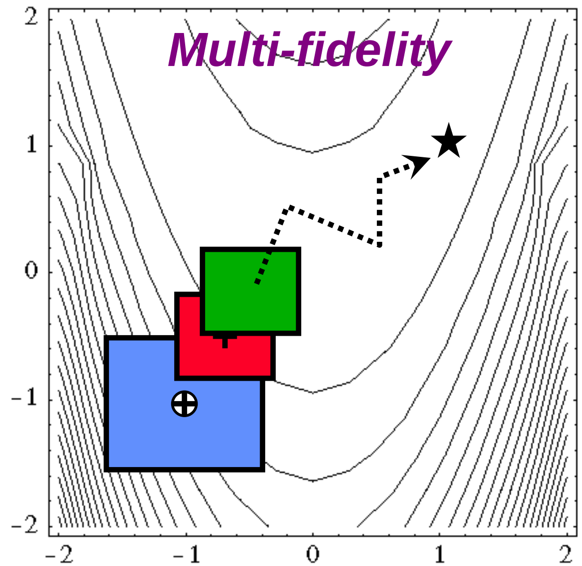
- Methods
  - May need to specify a method to solve a sub-problem
- Models
  - Hierarchical surrogate allows user-specified models of differing fidelity
  - Data-fit surrogate constructs a response surface of the user-supplied high-fidelity model
  - Nested allows for “splitting” model parameters into multiple sets for nested analyses

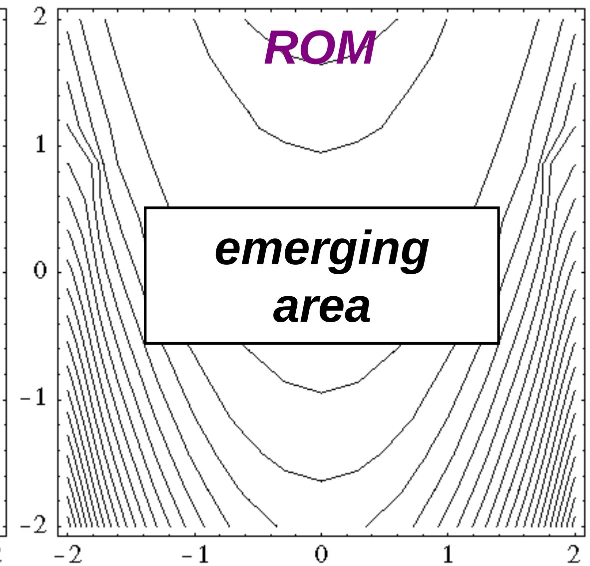
Structure of surrogate-based (or multi-fidelity) optimization



Trust Region Surrogate-Based Minimization







Data fit surrogates

- Global: polynomials, splines, neural network, Kriging, RBFs
- Local: 1st/2nd-order Taylor

Data fits in SBO

- Smoothing: extract global trend
- DACE: limited # design vars
- Must balance local consistency with global accuracy

Multifidelity surrogates:

- Coarser discretizations, looser conv. tols., reduced element order
- Omitted physics: e.g., Euler CFD, panel methods

Multifidelity SBO

- HF scale better w/ des. vars.
- Requires smooth LF model
- May require design mapping
- Correction quality is crucial

ROM surrogates:

- Spectral decomposition
- POD/PCA w/ SVD
- KL/PCE (random fields, stochastic processes)

ROMs in SBO

- Key issue: parametrize (extended or spanning ROM)
- Otherwise like data fit case

Many Types of Data-Fit Surrogates

Polynomials are accurate in small regions and smooth noisy data.

linear

$$\hat{f}(\mathbf{x}) \approx c_0 + \sum_{i=1}^{n} c_i x_i$$

quadratic

$$\hat{f}(\mathbf{x}) \approx c_0 + \sum_{i=1}^{n} c_i x_i + \sum_{i=1}^{n} \sum_{j \geq i}^{n} c_{ij} x_i x_j$$

cubic

$$\hat{f}(\mathbf{x}) \approx c_0 + \sum_{i=1}^{n} c_i x_i + \sum_{i=1}^{n} \sum_{j \geq i}^{n} c_{ij} x_i x_j + \sum_{i=1}^{n} \sum_{j \geq i}^{n} \sum_{k \geq j}^{n} c_{ijk} x_i x_j x_k$$

Splines can represent complex multi-modal surfaces and smooth noisy data.

$$\hat{f}(\mathbf{x}) = \sum_{m=1}^{M} a_m B_m(\mathbf{x})$$

truncated power basis functions

Gaussian processes are good predictors of mean and variance but can suffer from ill conditioning.

$$\hat{f}(\underline{x}) \approx \underline{g}(\underline{x})^T \underline{\beta} + \underline{r}(\underline{x})^T \underline{\underline{R}}^{-1}(\underline{f} - \underline{\underline{G}}\,\underline{\beta})$$

trend correlation

Correction terms can be applied to surrogates for improved accuracy.

additive

$$\hat{f_{hi_\alpha}}(\mathbf{x}) = f_{lo}(\mathbf{x}) + \alpha(\mathbf{x})$$

multiplicative

$$\hat{f_{hi_\beta}}(\mathbf{x}) = f_{lo}(\mathbf{x})\beta(\mathbf{x})$$

convex combination

$$\hat{f_{hi_\gamma}}(\mathbf{x}) = \gamma \hat{f_{hi_\alpha}}(\mathbf{x}) + (1-\gamma)\hat{f_{hi_\beta}}(\mathbf{x})$$

Sandia National Laboratories

Method input for surrogate-based (or multi-fidelity) optimization

Need to identify relationships between blocks by using pointers and IDs.

```
strategy,
  single_method
  method_pointer = 'SBLO'

method,
  id_method = 'SBLO'
  surrogate_based_local
  model_pointer = 'SURROGATE'
  approx_method_pointer = 'NLP'
  max_iterations = 50
  trust_region
    initial_size = 0.10
    contraction_factor = 0.5
    expansion_factor   = 1.50

method,
  id_method = 'NLP'
  conmin_mfd
    max_iterations = 50
    convergence_tolerance = 1e-4
```

Strategy block points to the "outer loop" method.

This is the "outer loop" method. It requires an "inner loop" method and a (surrogate) model on which the inner loop will operate. It also has configuration parameters.

This is the "inner loop" method.

Method: surrogate-based optimization

- **method: optimizer for surrogate sub-problem**
- **model: data fit or multi-fidelity surrogate**

Model input for data-fit surrogate-based optimization

Need to identify relationships between blocks by using pointers and IDs.

```
model,
  id_model = 'SURROGATE'
  responses_pointer = 'SURROGATE_RESP'
  surrogate global
    dace_method_pointer = 'SAMPLING'
    polynomial quadratic

method,
  id_method = 'SAMPLING'
  model_pointer = 'TRUTH'
  dace lhs
    seed = 12345
    samples = 10

model,
  id_model = 'TRUTH'
  single
    interface_pointer = 'TRUE_FN'
    responses_pointer = 'TRUE_RESP'
```

"surrogate global" or "surrogate local" specifies a data-fit surrogate. A sampling method is needed to collect data, and the type of response surface must be specified. This type of model has no interface associated with it.

This is the sampling method used to collect data for surrogate construction. It operates on the high-fidelity model.

This is the high-fidelity model, with variables, responses, and an interface. Variables, responses, and interface blocks look the same as in DAKOTA 101.

Model: data-fit surrogate

↳ **method: sampler for data collection**

↳ **model: V,R,I for high-fidelity simulation**

Same model structure applies when doing surrogate-based UQ.

Sandia National Laboratories

Model input for multi-fidelity optimization

Need to identify relationships between blocks by using pointers and IDs.

```
model,
  id_model = 'SURROGATE'
  surrogate hierarchical
    low_fidelity_model  = 'LOFI'
    high_fidelity_model = 'HIFI'


model,
  id_model = 'LOFI'
  single
    interface_pointer = 'LOFI_FN'

model,
  id_model = 'HIFI'
  single
    interface_pointer = 'HIFI_FN'
```

"surrogate hierarchical" specifies a multi-fidelity optimization with user-provided models. Both high and low fidelity models must be identified. There is no interface associated with this type of model.

This is the low-fidelity model, with variables, responses, and an interface.

This is the high-fidelity model, with variables, responses, and an interface. Note that pointers to variables and responses are not needed if both models use the same ones. Variables, responses, and interface blocks look the same as in DAKOTA 101.

Model: multi-fidelity surrogate

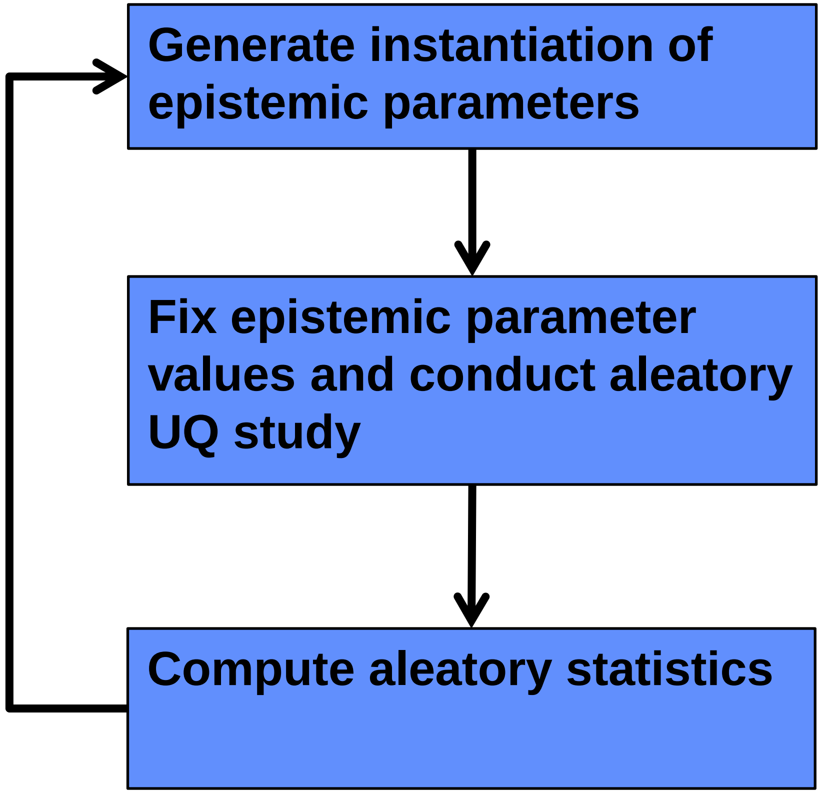
→ **model: V,R,I for low-fidelity simulation**

→ **model: V,R,I for high-fidelity simulation**

Same model structure applies when doing multi-fidelity UQ.

Sandia National Laboratories

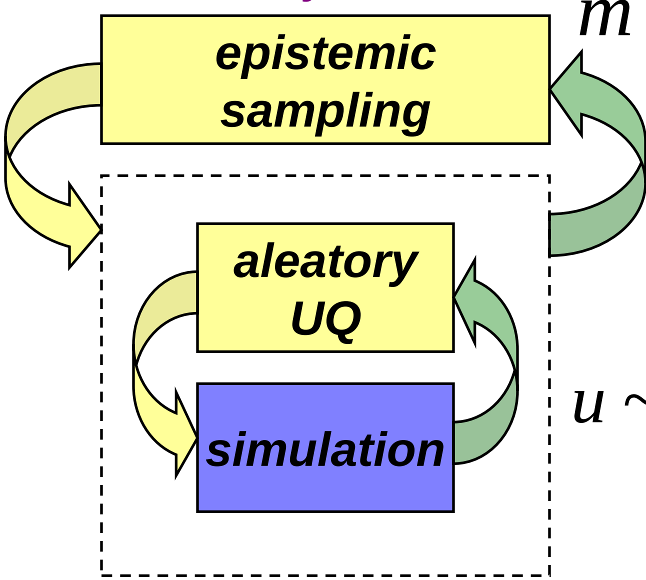
Structure of mixed (or nested) uncertainty quantification



Epistemic UQ: Nested ("Second-order")Approaches

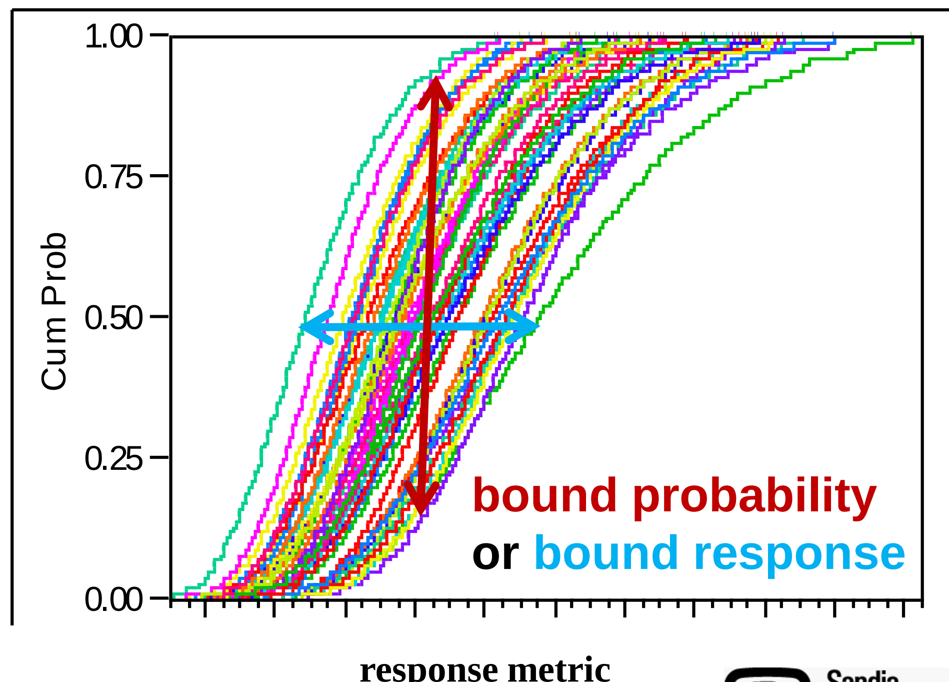
- **Propagate over epistemic and aleatory uncertainty, e.g., UQ with bounds on the mean of a normal distribution (hyper-parameters)**
- **Typical in regulatory analyses (e.g., NRC. WIPP)**
- **Outer loop: epistemic (interval) variables, inner loop UQ over aleatory (probability) variables; *potentially costly, not conservative***
- ***If treating epistemic as uniform, do not analyze probabilistically!***

50 outer loop samples:
50 aleatory CDF traces



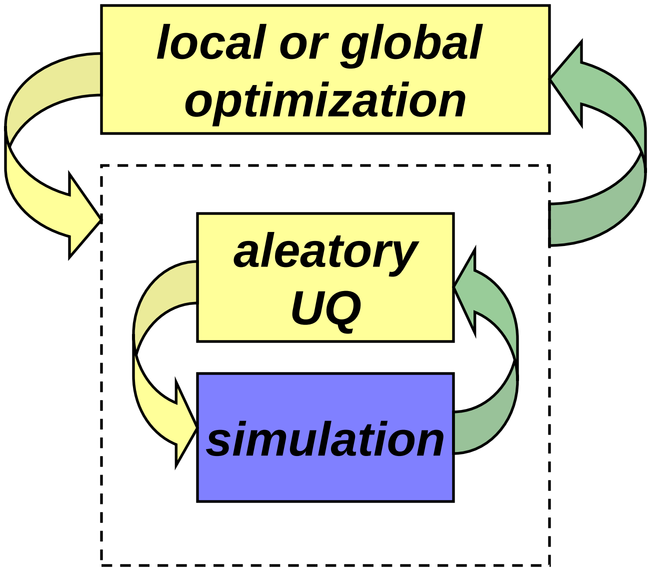
$$m \in [L, U]$$

$$u \sim N(m, \sigma)$$



"Envelope" of CDF traces represents response epistemic uncertainty

Interval Estimation Approach (Probability Bounds Analysis)



- ***Propagate intervals through simulation code***
- **Outer loop: determine interval on statistics, e.g., mean, variance**
  - **global optimization problem: find max/min of statistic of interest, given bound constrained interval variables**
  - **use EGO to solve 2 optimization problems with essentially one Gaussian process surrogate**
- **Inner loop: Use sampling, PCE, etc., to determine the CDFs or moments with respect to the aleatory variables**

$$\min_{u_E} \; f_{STAT}(u_A \mid u_E)$$
$$u_{LB} \le u_E \le u_{UB}$$
$$u_A \sim F(u_A; u_E)$$

$$\max_{u_E} \; f_{STAT}(u_A \mid u_E)$$
$$u_{LB} \le u_E \le u_{UB}$$
$$u_A \sim F(u_A; u_E)$$

Method and Model input for nested uncertainty quantification

Need to identify relationships between blocks by using pointers and IDs.

```
strategy,
  single_method
    method_pointer = 'PSTUDY'
method,
  id_method = 'PSTUDY'
  model_pointer = 'PS_M'
  centered_parameter_study
    step_vector = 1 1
    steps_per_variable = 2
model,
  id_model = 'PS_M'
  nested
    variables_pointer  = 'PS_V'
    sub_method_pointer = 'ALEATORY'
    responses_pointer  = 'PS_R'
    primary_variable_mapping   = 'X1' 'X3'
    secondary_variable_mapping = 'mean' 'mean'
    primary_response_mapping   = 1. 0. 0. 0. 0. 0. 0. 0.
    secondary_response_mapping = 0. 0. 0. 0. 1. 0. 0. 0.
                                 0. 0. 0. 0. 0. 0. 0. 1.
method,
  id_method = 'ALEATORY'
  model_pointer = 'ALEAT_M'
  sampling samples = 50 seed = 12347
    num_response_levels = 0 1 1
    response_levels = 10000. 10000.
    compute reliabilities
    complementary distribution
model,
  id_model = 'ALEAT_M'
  single
    variables_pointer = 'ALEAT_V'
    interface_pointer = 'ALEAT_I'
    responses_pointer = 'ALEAT_R'
```

Strategy block points to the "outer loop" method, epistemic UQ in this case.

The epistemic UQ method is the "outer loop" method and operates on the epistemic model parameters only.

The nested model "splits" the parameter set. The subset specified here is varied by the epistemic UQ method. The rest are deferred to the aleatory study specified by the sub_method_pointer. Variable mappings define context of the epistemic parameters in the aleatory study. Response mappings define which aleatory statistics for each simulation response are of interest. There is no interface associated with this type of model.

The aleatory UQ method is the "inner loop" method and operates on the epistemic model parameters only but inserts values passed through by the epistemic "outer loop".

This is the high-fidelity model, with variables, responses, and an interface. Variables, responses, and interface blocks look the same as in DAKOTA 101.

Method: epistemic uncertainty quantification

↳ Model: nested

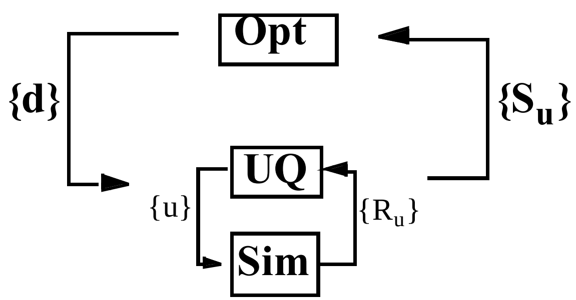
↳ method: aleatory uncertainty quantification

↳ model: V,R,I for high-fidelity simulation

Same method and model structure applies when doing optimization under uncertainty.

Optimization Under Uncertainty

Rather than design and then post-process to evaluate uncertainty…
actively design optimize while accounting for uncertainty/reliability metrics $s_u(d)$, e.g., mean, variance, reliability, probability:



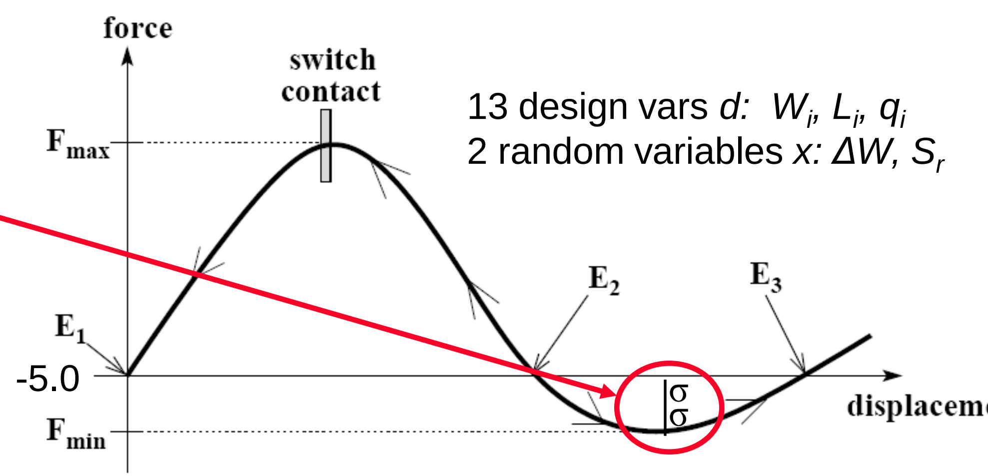
(nested paradigm)

$$
\begin{aligned}
\min \quad & f(d) + W\, s_u(d) \\
\text{s.t.} \quad & g_l \le g(d) \le g_u \\
& h(d) = h_t \\
& d_l \le d \le d_u \\
& a_l \le A_i\, s_u(d) \le a_u \\
& A_e\, s_u(d) = a_t
\end{aligned}
$$

Bistable switch problem formulation (Reliability-Based Design Optimization):

simultaneously reliable and robust designs

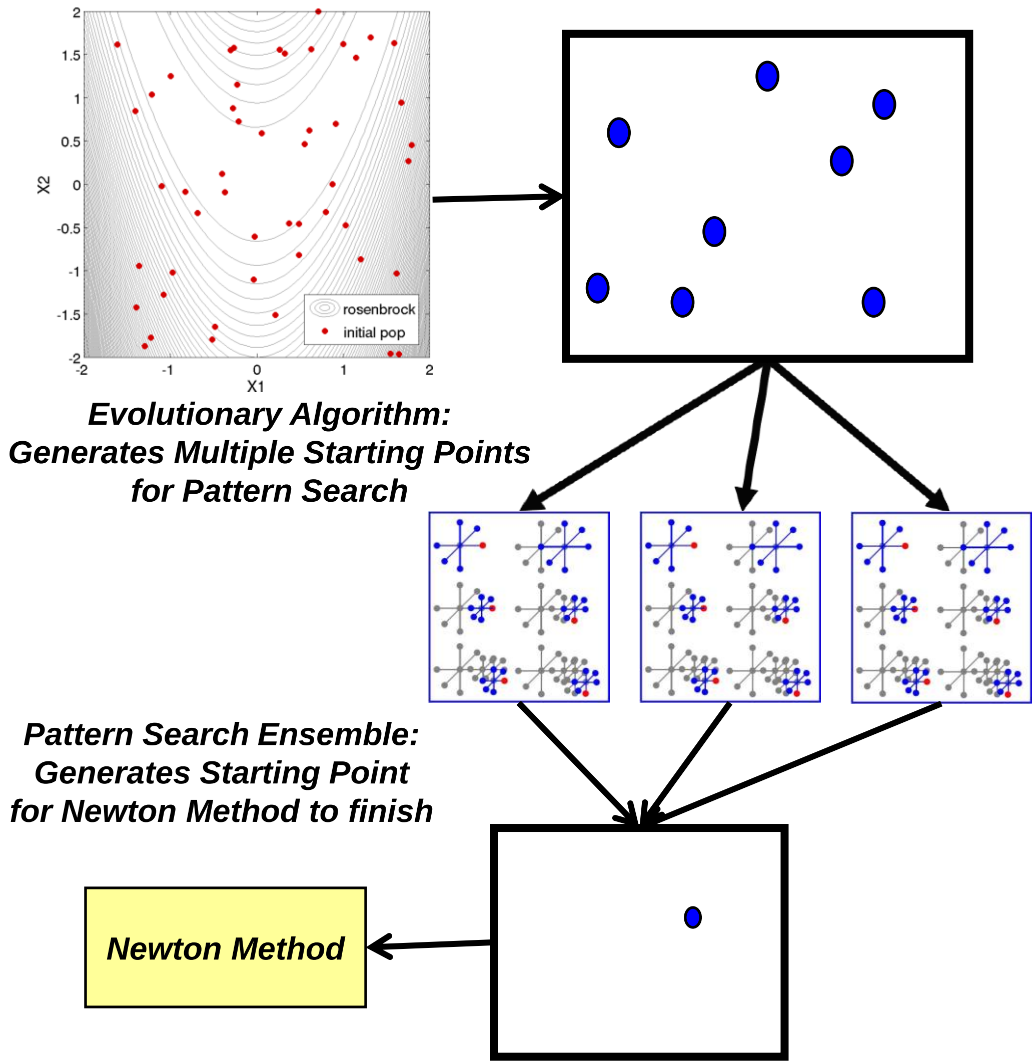
$$
\begin{array}{lrcll}
\text{max} & & & \mathrm{E}\left[F_{min}(\mathbf{d},\mathbf{x})\right] & \\
\text{s.t.} & 2 & \le & \beta_{ccdf}(\mathbf{d}) & \\
 & 50 & \le & \mathrm{E}\left[F_{max}(\mathbf{d},\mathbf{x})\right] & \le \; 150 \\
 & & & \mathrm{E}\left[E_2(\mathbf{d},\mathbf{x})\right] & \le \; 8 \\
 & & & \mathrm{E}\left[S_{max}(\mathbf{d},\mathbf{x})\right] & \le \; 3000
\end{array}
$$



13 design vars *d*: W_i, L_i, q_i
2 random variables *x*: ΔW, S_r

Hybrid Optimization

```
strategy,
  graphics
  hybrid sequential
    method_list = 'GA' 'PS' 'NLP'
method,
  id_method = 'GA'
  model_pointer = 'M1'
  coliny_ea
    seed = 1234
    population_size = 10
    verbose output
method,
  id_method = 'PS'
  model_pointer = 'M1'
  coliny_pattern_search stochastic
    seed = 1234
    initial_delta = 0.1
    threshold_delta = 1.e-4
    solution_accuracy = 1.e-10
    exploratory_moves basic_pattern
    verbose output
method,
  id_method = 'NLP'
  model_pointer = 'M2'
  optpp_newton
    gradient_tolerance = 1.e-12
    convergence_tolerance = 1.e-15
    verbose output
model,
  id_model = 'M1'
  single
    variables_pointer = 'V1'
    interface_pointer = 'I1'
    responses_pointer = 'R1'
model,
  id_model = 'M2'
  single
    variables_pointer = 'V1'
    interface_pointer = 'I1'
    responses_pointer = 'R2'
variables,
  id_variables = 'V1'
  continuous_design = 2
    initial_point     0.6     0.7
    upper_bounds      5.8     2.9
    lower_bounds      0.5    -2.9
    descriptors       'x1'    'x2'
interface,
  id_interface = 'I1'
  direct
    analysis_driver=   'text_book'
responses,
  id_responses = 'R1'
  num_objective_functions = 1
  no_gradients
  no_hessians
responses,
  id_responses = 'R2'
  num_objective_functions = 1
  analytic_gradients
  analytic_hessians
```



Multi-Objective Optimization

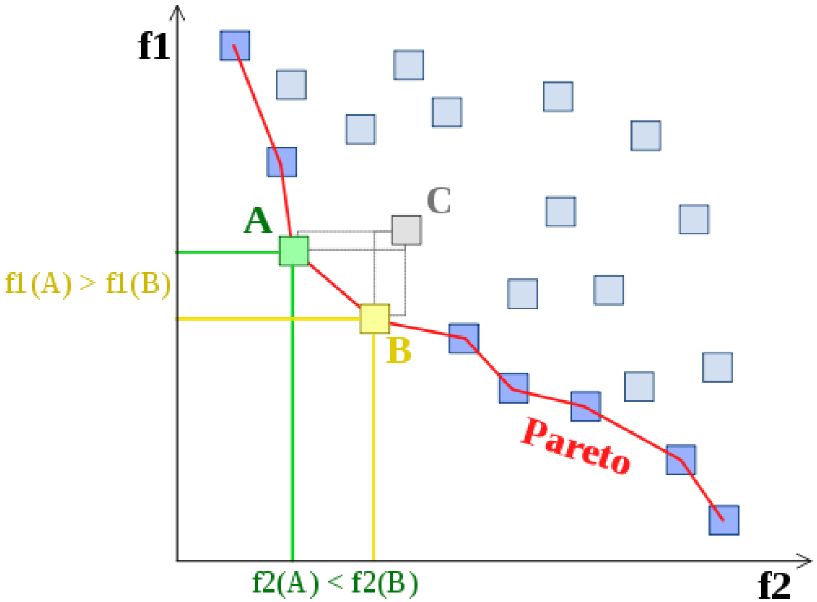


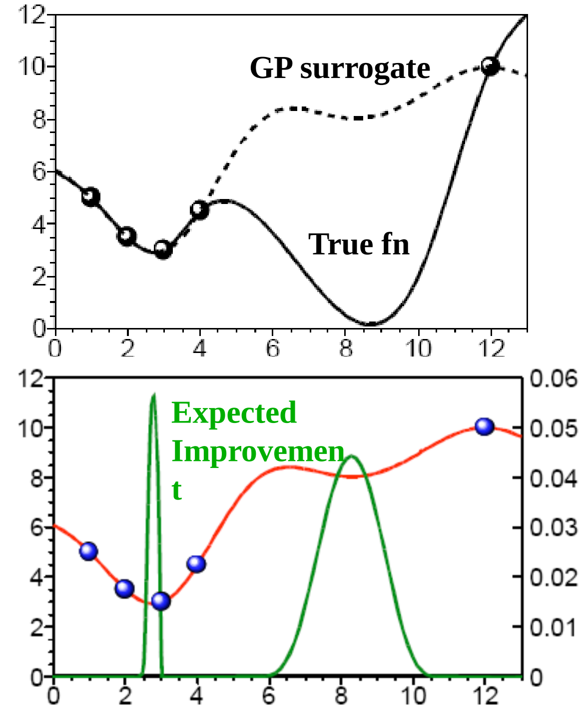
Image from http://en.wikipedia.org/wiki/Pareto_efficiency

May want tradeoffs between multiple objectives.

```
strategy,
  single_method
  tabular_graphics_data
method,
  optpp_q_newton
    output verbose
    convergence_tolerance = 1.e-8
variables,
  continuous_design = 2
    initial_point    0.9    1.1
    upper_bounds     5.8    2.9
    lower_bounds     0.5   -2.9
    descriptors      'x1'   'x2'
interface,
  system asynchronous
    analysis_driver=  'text_book'
responses,
  num_objective_functions = 3
  multi_objective_weights = .7 .2 .1
  analytic_gradients
  no_hessians
```

Efficient Global Optimization

- **Technique due to Jones, Schonlau, Welch**
- **Build global Gaussian process approximation to initial sample**
- **Balance global exploration (add points with high predicted variance) with local optimality (promising minima) via an “expected improvement function”**



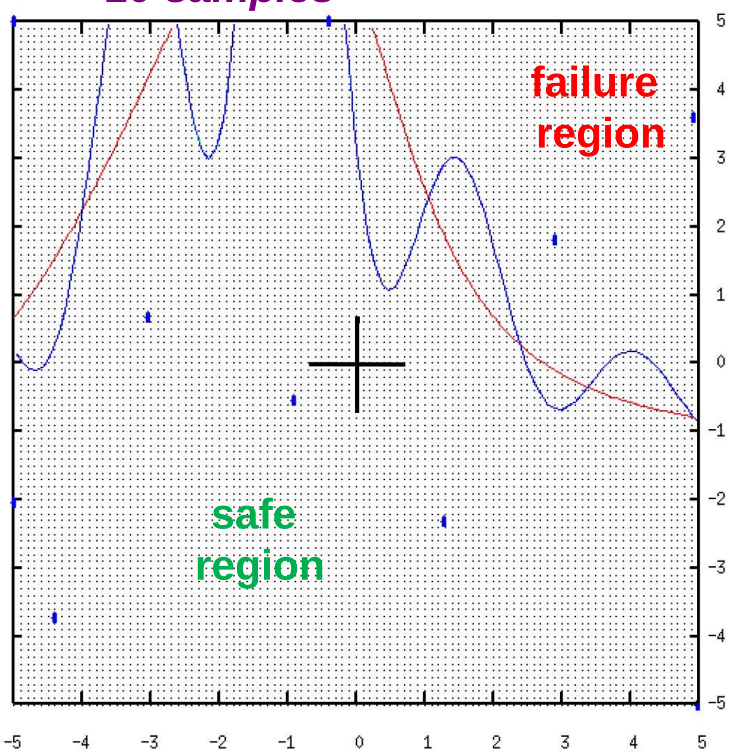
From Jones, Schonlau, Welch, 1998

Efficient Global Reliability Analysis: GP Surrogate + MMAIS (B.J. Bichon)

- Apply an EGO-like method to the equality-constrained optimization problem
- In EGRA, an expected feasibility function balances exploration with local search near the failure boundary to refine the GP
- Cost competitive with best MPP search methods, yet better probability of failure estimates; addresses nonlinear and multimodal challenges

Gaussian process model (level curves) of reliability limit state with

10 samples



28 samples