Fusing models and data to achieve efficient design, optimization, and uncertainty quantification

KAREN E. WILLCOX
Science at Extreme Scales: Where Big Data Meets Large-Scale Computing Tutorials
Institute for Pure and Applied Mathematics
September 18, 2018
1. Why use multifidelity modeling?
2. What is multifidelity modeling?
3. Multifidelity optimization (MFO)
4. Multifidelity Monte Carlo (MFMC)
5. Multifidelity Importance Sampling (MFIS)
6. Multi Information Source Optimization (MISO)
7. Conclusions
1. Motivation

Why use multifidelity modeling?
Why use a multifidelity formulation?

Reduced model (approximate) — Full model ("truth")
Why use a multifidelity formulation?

Reduced model (approximate)

Computationally cheap(er)

Full model ("truth")

Computationally expensive
Why use a multifidelity formulation?

- Replace full model with reduced model and solve \{opt, UQ, inverse\}
- Propagate error estimates on forward predictions to determine error in \{opt, UQ, inverse\} solutions (may be non-trivial)
Why use a multifidelity formulation?

Reduced model (approximate)

Certified?
- no

- Replace full model with reduced model and solve {opt, UQ, inverse}
- Hope for the best

Full model ("truth")
Why use a multifidelity formulation?

Reduced model (approximate)

Full model ("truth")

Certified?

no

- Use a multifidelity formulation that invokes both the reduced model and the full model
- Trade computational cost for the ability to place guarantees on the solution of {opt, UQ, inverse}
Why use a multifidelity formulation?

Reduction model
(approximate)

Certified?

no

Full model
(“truth”)

• Use a multifidelity formulation that invokes both the reduced model and the full model

• Trade computational cost for the ability to place guarantees on the solution of {opt, UQ, inverse}

• Certify the solution of {opt, UQ, inverse} even in the absence of guarantees on the reduced model itself
2. Multifidelity models and multifidelity methods

“All models are wrong, but some are useful.”

George Box, 1979
Decisions are informed by multiple sources of information.

Analysis and design typically begin with low-fidelity models and progressively incorporate higher fidelity tools.

Many information sources available: multifidelity models, historical data, operational data, experimental data, expert opinions.

Physics-based models

Historical data

Surrogate & reduced models

Experts

Experimental data

Telling us different things about the system: the collective information is greater than the individual parts.

Critical to get the right information early in the decision process.
Multifidelity models can come in different forms

- Covering a range of different resolutions, scales, modeling assumptions, etc.

- Simplified physics, loosened tolerance, coarse grid, data-fit, projection-based reduced models, etc.
Multifidelity models can come in different forms

Covering a range of different resolutions, scales, modeling assumptions, etc.

Simplified physics, loosened tolerance, coarse grid, data-fit, projection-based ROM, etc.

• high-fidelity model ("truth") mapping input $z$ to output $y$
  
  $$f^{(1)}: \mathcal{Z} \rightarrow \mathcal{Y}$$

• $k - 1$ lower-fidelity models mapping input $z$ to output $y$
  
  $$f^{(2)}, ..., f^{(k)}: \mathcal{Z} \rightarrow \mathcal{Y}$$

• model $f^{(i)}$ has cost $w_i$

• model $f^{(i)}$ has fidelity $f_i$

• models do not necessarily form a hierarchy
Multifidelity methods for outer-loop problems

- **Outer-loop**: computational applications that form outer loops around a model
  - overall outer-loop result is obtained at the termination of the outer loop
  - examples: optimization, uncertainty propagation, inverse problems, data assimilation, control, sensitivity analysis

- **Multifidelity methods**: goal is to solve the outer-loop problem at high fidelity
  - invoke multiple models to reduce computational cost
  - maintains guarantees on outer-loop result
Multifidelity methods for outer-loop problems

- **Multifidelity methods**: goal is to solve the outer-loop problem at high fidelity
  - invoke multiple models to reduce computational cost
  - maintains guarantees on outer-loop result

- Key questions
  - how to combine model estimates?
  - how to balance evaluations among models?
  - how to guarantee outer-loop result?
Multifidelity strategies for the outer loop

- adaptation
  - model correction
  - online adaptive model reduction
  - EGO
  - ...

- fusion
  - control variates
  - co-kriging
  - partial replacement
  - multilevel stoch. collocation
  - ...

- filtering
  - importance sampling
  - multi-stage sampling
  - ...

Examples of multifidelity strategies for the outer loop

- **optimization** – trust regions
  Alexandrov & Lewis, 1999; Eldred et al., 2004

- **forward propagation of uncertainty** – control variates

- **failure probability estimation** – adaptive sampling
  Bichon et al, 2008; Li & Xiu, 2010; Peherstorfer et al., 2016; Peherstorfer et al., 2017

- **optimization under uncertainty** – control variates
  Ng, Huynh, W., 2012; Ng & W., 2014, 2016

- **statistical inverse problems** – adaptive delayed acceptance
  Fox & Christensen, 2008; Efendiev & Hou, 2009; Cui et al., 2014
3. Multifidelity Optimization

\[
\begin{align*}
\min_{x} & \quad f(x) \\
\text{s.t.} & \quad g(x) \leq 0 \\
& \quad h(x) = 0
\end{align*}
\]
Design optimization formulation

\[
\begin{align*}
\min_{x} & \quad f(x) \\
\text{s.t.} & \quad g(x) \leq 0 \\
& \quad h(x) = 0
\end{align*}
\]

- Design variables: \( x \)
- Objective: \( f(x) \)
- Constraints: \( g(x), h(x) \)

- Interested in optimization of systems governed by PDEs (constraints and objective evaluation is expensive)
Multifidelity optimization formulation

\[
\begin{align*}
\min_{x} & \quad f(x) \\
\text{s.t.} & \quad g(x) \leq 0 \\
& \quad h(x) = 0
\end{align*}
\]

Design variables \( x \)
Objective \( f(x) \)
Constraints \( g(x), h(x) \)

\[
\begin{align*}
x & \quad \rightarrow \quad f_{hi} \\
& \quad \rightarrow \quad g_{hi} \\
& \quad \rightarrow \quad h_{hi} \\
& \quad \rightarrow \quad \text{hi-fi model}
\end{align*}
\]

\[
\begin{align*}
x & \quad \rightarrow \quad f_{lo} + \alpha \\
& \quad \rightarrow \quad g_{lo} + \beta \\
& \quad \rightarrow \quad h_{lo} + \gamma \\
& \quad \rightarrow \quad \text{lo-fi model} \quad \text{correction}
\end{align*}
\]

\[
\begin{align*}
x_j & \quad \downarrow \quad f_{hi} g_{hi} h_{hi} \\
& \quad \downarrow \quad \text{hi-fi model}
\end{align*}
\]
Multifidelity optimization

Defining the surrogate model

• Denote a surrogate model of \( f_{\text{high}}(\mathbf{x}) \) as \( m(\mathbf{x}) \)

• The surrogate model could be:

  1. The **low-fidelity function** (reduced model)
     \[
     m(\mathbf{x}) = f_{\text{low}}(\mathbf{x}) \approx f_{\text{high}}(\mathbf{x})
     \]

  2. The sum of the low-fidelity function and an additive correction
     \[
     m(\mathbf{x}) = f_{\text{low}}(\mathbf{x}) + e(\mathbf{x}) \approx f_{\text{high}}(\mathbf{x})
     \]
     where \( e(\mathbf{x}) \) is calibrated to the difference \( f_{\text{high}}(\mathbf{x}) - f_{\text{low}}(\mathbf{x}) \)

  3. The **product** of a low-fidelity function and a multiplicative correction
     \[
     m(\mathbf{x}) = \beta_c(\mathbf{x}) f_{\text{low}}(\mathbf{x}) \approx f_{\text{high}}(\mathbf{x})
     \]
     where \( \beta_c(\mathbf{x}) \) is calibrated to the quotient \( f_{\text{high}}(\mathbf{x}) / f_{\text{low}}(\mathbf{x}) \)

• Update the correction terms as the optimization algorithm proceeds and additional evaluations of \( f_{\text{high}}(\mathbf{x}) \) become available
Multifidelity optimization: Trust-region model management

At iteration $k$, define a trust region centered on iterate $x_k$ with size $\Delta_k$

$$
B_k = \{ x : \| x - x_k \| \leq \Delta_k \}
$$

$m_k$ is the surrogate model on the $k$th iteration

Determine a trial step $s_k$ at iteration $k$, by solving a subproblem of the form:

$$
\min_{s_k} m_k(x_k + s_k)
\text{s.t. } \| s_k \| \leq \Delta_k
$$

(unconstrained case)
Multifidelity optimization: Trust-region model management

• Evaluate the function at the trial point: \( f_{\text{high}}(x_k + s_k) \)

• Compute the ratio of the actual improvement in the function value to the improvement predicted by the surrogate model:

\[
\rho_k = \frac{f_{\text{high}}(x_k) - f_{\text{high}}(x_k + s_k)}{m_k(x_k) - m_k(x_k + s_k)}
\]

• Accept or reject the trial point and update trust region size according to (typical parameters):

<table>
<thead>
<tr>
<th>( \rho^k )</th>
<th>Acceptance</th>
<th>( \Delta^{k+1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \leq 0 )</td>
<td>Reject step</td>
<td>( \equiv 0.5\Delta^k )</td>
</tr>
<tr>
<td>( 0 &lt; \rho^k \leq 0.1 )</td>
<td>Accept step</td>
<td>( \equiv 0.5\Delta^k )</td>
</tr>
<tr>
<td>( 0.1 &lt; \rho^k &lt; 0.75 )</td>
<td>Accept step</td>
<td>( \equiv \Delta^k )</td>
</tr>
<tr>
<td>( 0.75 \leq \rho^k )</td>
<td>Accept step</td>
<td>( \equiv 2\Delta^k )</td>
</tr>
</tbody>
</table>
Trust-Region Algorithm for Iteration $k$

1. Compute a step, $s_k$, by solving the trust-region subproblem,
\[
\min_{s_k} m_k(x_k + s_k)
\]
\[
\text{s.t. } \|s_k\| \leq \Delta_k.
\]

2. Evaluate $f_{\text{high}}(x_k + s_k)$.

3. Compute the ratio of actual improvement to predicted improvement,
\[
\rho_k = \frac{f_{\text{high}}(x_k) - f_{\text{high}}(x_k + s_k)}{m_k(x_k) - m_k(x_k + s_k)}.
\]

4. Accept or reject the trial point according to $\rho_k$,
\[
x_{k+1} = \begin{cases} 
  x_k + s_k & \text{if } \rho_k > 0 \\
  x_k & \text{otherwise.}
\end{cases}
\]

5. Update the trust region size according to $\rho_k$,
\[
\Delta_{k+1} = \begin{cases} 
  \gamma_1 \Delta_k & \text{if } \rho_k \leq \eta_1 \\
  \Delta_k & \text{if } \eta_1 < \rho_k < \eta_2 \\
  \gamma_2 \Delta_k & \text{if } \rho_k \geq \eta_2.
\end{cases}
\]
Trust-Region Demonstration

\[ \rho = 0.993730 \]
• Provably convergent to local minimum of high-fidelity function if surrogate is first-order accurate at center of trust region [Alexandrov et al., 2001]

• Additive correction: \( a(x) = f_{\text{high}}(x) - f_{\text{low}}(x) \)

with surrogate constructed as

\[
m_k(x) = f_{\text{low}}(x) + a(x_k) + \nabla a(x_k)^T (x - x_k)
\]

• Multiplicative correction: \( \beta(x) = \frac{f_{\text{high}}(x)}{f_{\text{low}}(x)} \)

with surrogate constructed as

\[
m_k(x) = [\beta(x_k) + \nabla \beta(x_k)^T (x - x_k)] f_{\text{low}}(x)
\]

• Only first-order corrections required to guarantee convergence; quasi-second-order corrections accelerate convergence [Eldred et al., 2004]

• Trust-region POD [Arian, Fahl, Sachs, 2000]
• Derivative-free trust region approaches
  [Conn, Scheinberg, and Vicente, 2009]

• Provably convergent under appropriate conditions if the surrogate model is “fully linear”

\[ \left\| \nabla f_{\text{high}}(x) - \nabla m_k(x) \right\| \leq \kappa_g \Delta_k \]

\[ \left| f_{\text{high}}(x) - m_k(x) \right| \leq \kappa_f \Delta^2_k \]

• Achieved through adaptive corrections or adaptive calibration e.g., radial basis function calibration with sample points chosen to make surrogate model fully linear by construction
  [Wild, Regis and Shoemaker, 2011; Wild and Shoemaker, 2013]

• Key: never need gradients wrt the high-fidelity model
4. Multifidelity Monte Carlo (MFMC)

Efficient uncertainty propagation leveraging multiple models

Ng & W., Multifidelity approaches for optimization under uncertainty, *IJNME*, 2014
Estimating QoI statistics via Monte Carlo sampling

- **uncertain input** $z \in \mathcal{Z}$
- **output** quantity of interest $y \in \mathcal{Y}$
- **high-fidelity model** $f^{(1)}: \mathcal{Z} \rightarrow \mathcal{Y}$
  - with cost $w_1 > 0$ ("truth")

- **Goal:** given random input variable $Z$, estimate statistics $s$ of $f^{(1)}(Z)$

- **Example:** expected value
  $$ s = E[f^{(1)}(Z)] $$

- Monte Carlo estimator for $s$ using $n$ realizations $z_1, \ldots, z_n$ of $Z$ has costs $nw_1$:
  $$ \hat{s} = \bar{y}_{n}^{(1)} = \frac{1}{n} \sum_{i=1}^{n} f^{(1)}(z_i) $$
Multifidelity Monte Carlo

leverages multiple approximate models to estimate statistics of the high-fidelity model

- high-fidelity model
  \[ f^{(1)}: \mathcal{Z} \rightarrow \mathcal{Y} \text{ (“truth”) } \]

- \( k - 1 \) surrogate models
  \[ f^{(2)}, \ldots, f^{(k)}: \mathcal{Z} \rightarrow \mathcal{Y} \]

- model \( f^{(i)} \) has cost \( w_i \)

- \( m_i \) evaluations for model \( i \), with
  \[ m_1 \leq m_2 \leq \ldots \leq m_k \]

- Models do not necessarily form a hierarchy (cf. multi-level Monte Carlo)
  - How to combine models?
  - How to balance evaluations among them?
Multifidelity Monte Carlo

leveraging multiple approximate models to estimate statistics of the high-fidelity model

- Draw $m_k$ realizations $z_1, ..., z_{m_k}$ of $Z$ and evaluate $f^{(i)}$:
  
  $f^{(i)}(z_1), ..., f^{(i)}(z_{m_i})$

- Compute mean estimators
  
  $\bar{y}_{m_1}^{(1)}, ..., \bar{y}_{m_k}^{(k)}$ and $\hat{\bar{y}}_{m_1}, ..., \hat{\bar{y}}_{m_k-1}^{(k)}$

- MFMC estimator:
  
  $\hat{s} = \bar{y}_{m_1}^{(1)} + \sum_{i=2}^{k} \alpha_i \left( \bar{y}_{m_i}^{(i)} - \hat{\bar{y}}_{m_i-1}^{(i)} \right)$

  MFMC estimate for the mean, mean estimate using $m_1$ evaluations of truth model, mean estimate using $m_i$ evaluations of model $i$, mean estimate using $m_{i-1}$ evaluations of model $i$
Multifidelity Monte Carlo

Example for intuition: two models \( k = 2 \)

\[ z_i = \text{samples of } Z \]
\[ a_i = f^{(1)}(z_i) = \text{samples of } A \]
\[ b_i = f^{(2)}(z_i) = \text{samples of } B = a_i + \text{error} \]

\( s_A = \text{statistics of } A \) (e.g., mean, variance)
\( \hat{s}_A = \text{estimator of } s_A \)

Ng & W., AIAA 2012, IJNME 2014, J. Aircraft 2015
Monte Carlo variance reduction

Classical control variate approach

- Regular MC estimator for \( s_A = \mathbb{E}[A] \) using \( n \) samples of \( A \):
  \[
  \bar{a}_n = \frac{1}{n} \sum_{i=1}^{n} a_i \quad \text{Var}[\bar{a}_n] = \frac{\sigma_A^2}{n}
  \]

- Control variate estimator of \( s_A \):
  - Additional random variable \( B \) with known \( s_B = \mathbb{E}[B] \)
    \[
    \hat{s}_A = \bar{a}_n + \alpha(s_B - \bar{b}_n)
    \]
    \[
    \text{Var}[\hat{s}_A] = \frac{\sigma_A^2 + \alpha^2 \sigma_B^2 - 2\alpha \rho_{AB} \sigma_A \sigma_B}{n}
    \]

- Minimize \( \text{Var}[\hat{s}_A] \) with respect to \( \alpha \)
  \[
  \text{Var}[\hat{s}_A^*] = \frac{(1 - \rho_{AB}^2) \sigma_A^2}{n} \leq 1
  \]

Definitions:
- \( \sigma_A^2 = \text{Var}[A] \)
- \( \sigma_B^2 = \text{Var}[B] \)
- \( \rho_{AB} = \text{Corr}[A,B] \)
Multifidelity estimator of $s_A$ based on control variate method:
- $A = \text{random output of high-fidelity model}$
- $B = \text{random output of low-fidelity model}$ ($s_B$ unknown)

$$\hat{s}_{A,p} = \bar{a}_n + \alpha (\bar{b}_m - \bar{b}_n) \quad \text{with} \quad m \gg n$$

$$\text{Var}[\hat{s}_{A,p}] = \frac{\sigma_A^2 + \alpha^2 \sigma_B^2 - 2\alpha \rho_{AB} \sigma_A \sigma_B}{n} - \frac{\alpha^2 \sigma_B^2 - 2\alpha \rho_{AB} \sigma_A \sigma_B}{m}$$

- Using difference $(\bar{b}_m - \bar{b}_n)$ as correction to $\bar{a}_n$
- Leveraging correlation between $A$ and $B$
  - Correlation captured in $\alpha$

Definitions:
- $\sigma_A^2 = \text{Var}[A]$
- $\sigma_B^2 = \text{Var}[B]$
- $\rho_{AB} = \text{Corr}[A,B]$
Multifidelity Monte Carlo
optimally allocate computational budget across \( k \) models


- MFMC estimator

\[
\hat{s} = \bar{y}_{m_1}^{(1)} + \sum_{i=2}^{k} \alpha_i \left( \bar{y}_{m_i}^{(i)} - \bar{y}_{m_{i-1}}^{(i)} \right)
\]

- MFMC estimator is **unbiased**, even with no error bounds for surrogates:

\[ E[\hat{s}] = s \]

- The costs of the MFMC estimator are

\[
c(\hat{s}) = \sum_{i=1}^{k} w_i m_i
\]

- Distinguishing features of MFMC method:
  - optimal selection of the number of model evaluations
    \( m_1 \leq m_2 \leq \ldots \leq m_k \) and of coefficients \( \alpha_2, \ldots, \alpha_k \)
  - applicable to general information sources
    (e.g., any type of surrogate model, database curve fits, etc.)
Minimize the MSE of the MFMC estimator for a given computational budget $p$

MFMC estimate $\hat{s}$ is unbiased; MSE is given by $\text{Var}[\hat{s}]$

$$\text{Var}[\hat{s}] = \frac{\sigma_1^2}{m_1} + \sum_{i=2}^{k} \left( \frac{1}{m_{i-1}} - \frac{1}{m_i} \right) \left( \alpha_i^2 \sigma_i^2 - 2 \alpha_i \rho_i \sigma_i \sigma_1 \right)$$

- $\sigma_i^2$ is variance of $f^{(i)}(Z)$
- $\rho_i$ is correlation coefficient between $f^{(1)}(Z)$ and $f^{(i)}(Z)$

Leads to optimization problem

$$\min \limits_{m \in \mathbb{R}^k, \alpha_2, \ldots, \alpha_k \in \mathbb{R}} \text{Var}[\hat{s}]$$

such that

- $m_{i-1} \leq m_i, i = 2, \ldots, k$
- $0 \leq m_1$
- $c(\hat{s}) = w^T m = p$
Balancing the number of information source (IS) evaluations

- Optimization problem has unique (analytic) solution if
  \[ \rho_1^2 > \rho_2^2 > \ldots > \rho_k^2 \]
  and
  \[ \frac{w_{i-1}}{w_i} > \frac{\rho_{i-1}^2 - \rho_i^2}{\rho_i^2 - \rho_{i+1}^2}, \quad i = 2, \ldots, k \]

- The costs/correlation ratio establishes a relationship between
  - preceding IS \( f^{(i-1)} \),
  - current IS \( f^{(i)} \), and
  - succeeding IS \( f^{(i+1)} \)

- If \( f^{(1)}, \ldots, f^{(k)} \) violate conditions, can construct a subset of IS’s that satisfy conditions

The interactions between the IS's impact the behavior of the MFMC estimator; not the properties of the IS’s alone
MFMC variance reduction

- Let $\bar{y}_n^{(1)}$ be the (benchmark) Monte Carlo estimator with computational budget $p$
- Ratio of MSE of MFMC estimator $\hat{s}$ and MSE of $\bar{y}_n^{(1)}$ is
  \[
  \frac{e(\hat{s})}{e(\bar{y}_n^{(1)})} = \left( \sum_{i=1}^{k} \sqrt{\frac{w_i}{w_1}} \left( \rho_i^2 - \rho_{i+1}^2 \right) \right)^2
  \]
- The MFMC estimator has lower MSE than the Monte Carlo estimator if
  \[
  \sum_{i=1}^{k} \sqrt{\frac{w_i}{w_1}} \left( \rho_i^2 - \rho_{i+1}^2 \right) < 1
  \]
- Condition on the collective whole of the models (sum), not on properties of each model separately
- The interaction between the models is what drives the MFMC estimator, not model properties alone
Example: three information sources ($k = 3$)

- truth model $f^{(1)}$, cost $w_1$
- surrogate model $f^{(2)}$, cost $w_2$, correlation with truth $\rho_2$
- surrogate model $f^{(3)}$, cost $w_3$, correlation with truth $\rho_3$

- feasibility conditions
  
  \[
  1 > \rho_2^2 > \rho_3^2, \quad \frac{w_1}{w_2} > \frac{\rho_1^2 - \rho_2^2}{\rho_2^2 - \rho_3^2}, \quad \frac{w_2}{w_3} > \frac{\rho_2^2 - \rho_3^2}{\rho_3^2}
  \]

- variance reduction (low $S \rightarrow$ low variance)

\[
S(w_1, w_2, w_3, \rho_2, \rho_3) = \sum_{i=1}^{k} \sqrt{\frac{w_i}{w_1} (\rho_i^2 - \rho_{i+1}^2)}
\]

\[
= \sqrt{1 - \rho_2^2} + \sqrt{\frac{w_2}{w_1} (\rho_2^2 - \rho_3^2)} + \sqrt{\frac{w_3}{w_1} \rho_3^2}
\]

$< 1$ for the MFMC estimator to be efficient
Example: three information sources \((k = 3)\)

\[
S(w_1, w_2, w_3, \rho_2, \rho_3) = \sqrt{1 - \rho_2^2} + \frac{w_2}{w_1} (\rho_2^2 - \rho_3^2) + \frac{w_3}{w_1} \rho_3^2
\]

- Set \(w_2/w_1 = 0.1, \quad \rho_2 = 0.9\)
- Using \(f^{(1)}, f^{(2)}\) only → larger variance than MC estimator \((S > 1)\)
- Vary \(w_3/w_1\) and \(\rho_3\); plot contours of \(S\)
- If costs \(w_3\) high \( (> 0.01w_1)\): third term in \(S\) can dominate, increasing correlation \(\rho_3\) can lead to larger \(S\)
- If \(w_3\) low, then increase of \(\rho_3\) always reduces \(S\)
Example: three information sources ($k = 3$)

$$S(w_1, w_2, w_3, \rho_2, \rho_3) = \sqrt{1 - \rho_2^2} + \sqrt{\frac{w_2}{w_1} (\rho_2^2 - \rho_3^2)} + \sqrt{\frac{w_3}{w_1} \rho_3^2}$$

- Set $w_2/w_1 = 0.1$, $\rho_2 = 0.6$
- Increasing the correlation can violate feasibility condition: if $\rho_3 \approx \rho_2$ then denominator $\rho_2^2 - \rho_3^2$ in feasibility condition becomes small and condition is violated
Example: three information sources \((k = 3)\)

\[
S(w_1, w_2, w_3, \rho_2, \rho_3) = \sqrt{1 - \rho_2^2} + \frac{w_2}{w_1} (\rho_2^2 - \rho_3^2) + \frac{w_3}{w_1} \rho_3^2
\]

- Set \(w_2/w_1 = 0.1, \ \rho_2 = 0.4\)
- IS \(f^{(2)}\) has high costs and low correlation
- Variance cannot be improved by adding a third IS
- Any third IS will lead either to a violation of the feasibility condition or to a higher variance than the MC estimator \((S > 1)\)
- Have to remove/change IS \(f^{(2)}\) to reduce variance
Example: three information sources \((k = 3)\)

\[ S(w_1, w_2, w_3, \rho_2, \rho_3) = \sqrt{1 - \rho_2^2} + \sqrt{\frac{w_2}{w_1} (\rho_2^2 - \rho_3^2)} + \sqrt{\frac{w_3}{w_1} \rho_3^2} \]

- Set \(w_2/w_1 = 10^{-4}, \rho_2 = 0.4\)
- Decreasing the costs of \(f^{(2)}\) releases deadlock; adding a third model can improve the variance
Example: Locally damaged plate (multiple models)

- Locally damaged plate
- Inputs: nominal thickness, load, two damage parameters
- Inputs uniformly distributed in $[0.05, 0.1] \times [1, 100] \times [0, 0.2] \times (0, 0.05]$
- QoI: maximum deflection of plate

- Six models available
  - High-fidelity model: FEM, 300 dof
  - Reduced model: POD, 10 dof
  - Reduced model: POD, 2 dof
  - Reduced model: POD, 5 dof
  - Data-fit model: linear interpolation, 256 pts
  - Support vector machine: 256 pts

- Variance, correlation, runtime estimated from 100 samples
Locally damaged plate: MFMC

MFMC estimation of mean deflection achieves up to 4 orders of magnitude reduction in computational cost

- Combine high-fidelity + reduced (POD, 10) + data-fit (linear interp, 256)
- Reduced and data-fit model lead to biased estimator, MFMC is unbiased
Locally damaged plate: MFMC mean estimate

Successively add reduced (POD, 10), data-fit (linear interp, 256), and then all others

Adding data-fit model reduces variance, even though data-fit model is poor approximation of high-fidelity model

MFMC achieves almost 4 orders of magnitude improvement over standard Monte Carlo simulation with high-fidelity model only.
A broad view of multifidelity models

in many outer-loop applications, can exploit past evaluations as a low-fidelity model

Ng & W., J. Aircraft, 2015
Cook, Jarrett, W., IJNME 2018

- in optimization under uncertainty, can exploit model correlation over design space
  - use $f^{(1)}(x + \Delta x, z)$ as surrogate for $f^{(1)}(x, z)$

- at current design point $x_k$
  - Define $A = f^{(1)}(x_k, z)$
  - Want to compute $\hat{s}$ as estimator of $s = \mathbb{E}[A]$

- previously visited design point $x_\ell$ where $\ell < k$
  - Define surrogate as $C = f^{(1)}(x_\ell, z)$
  - Reuse available data: $\hat{s}_C$ as estimator of $s_C = \mathbb{E}[C]$ with error $\text{Var}[\hat{s}_C]$
**Multifidelity Monte Carlo**

- control variate formulation
- applies to general models and information sources
- need to sample and learn the correlation between models
- optimal model allocations

**Multilevel Monte Carlo** [Heinrich 2001, Giles, 2008, Cliffe 2011]

- control variate formulation
- formulated for hierarchies of grids in specific problems (elliptic PDEs)
- leverage known error rates and cost rates

→ Clear connections between the two, especially as MLMC is expanded to consider a broader range of problems

5. Multifidelity Importance Sampling (MFIS)

Efficient estimation of low probability events, leveraging multiple models

Peherstorfer, Cui, Marzouk, W., Multifidelity importance sampling, CMAME, 2016

Peherstorfer, Kramer, W., Combining multiple surrogate models to accelerate failure probability estimation with expensive high-fidelity models, J. Computational Physics, 2017

Estimating a failure probability via Monte Carlo sampling

• uncertain input $z \in \mathcal{Z}$
• output quantity of interest $y \in \mathcal{Y}$
• high-fidelity model $f^{(1)}: \mathcal{Z} \to \mathcal{Y}$ with cost $w_1 > 0$ ("truth")

• define indicator function $I^{(1)}: \mathcal{Z} \to \mathcal{Y}$ as
  $$I^{(1)}(z) = \begin{cases} 1, & \text{if } f^{(1)}(z) < 0 \\ 0, & \text{else} \end{cases}$$

• random variable $Z$ with probability density $p$

• goal: estimate failure probability $P_f = \mathbb{E}_p[I^{(1)}(Z)]$
• Monte Carlo estimation of $P_f$ using $N$ realizations $Z_1, \ldots, Z_N$:
  $$P_f^{\text{MC}} = \frac{1}{N} \sum_{i=1}^{N} I^{(1)}(Z_i)$$
Estimating a failure probability via importance sampling

- Importance sampling: create biasing density $q$ that puts more weight on failure events
- Let $\hat{Z}$ be the corresponding RV
- Introduce the weight function
  $$w(z) = \frac{p(z)}{q(z)}$$
- Reformulate failure probability as
  $$P_f = \mathbb{E}_p[I^{(1)}(Z)] = \mathbb{E}_q[I^{(1)}(\hat{Z})w(\hat{Z})]$$
- Goal: construct a biasing density $q$ such that
  $$\text{Var}_q[I^{(1)}(\hat{Z})w(\hat{Z})] < \text{Var}_p[I^{(1)}(Z)]$$
- Lower variance means fewer realizations of $\hat{Z}$ than of $Z$ are necessary to achieve the same MSE $\rightarrow$ fewer model evaluations
Multifidelity importance sampling (MFIS) with two models

We derive biasing distribution $q$ with surrogate $f^{(2)}$, and use $f^{(1)}$ to estimate $P_f$

- Step 1: Construction of biasing distribution (“speedup”)
- Step 2: Estimation of $P_f$ using $q$ (“establish accuracy guarantees”)

Multifidelity importance sampling

Step 1: construction of biasing density

- Draw many realizations $z_1, ..., z_N$ of $Z$ (nominal)
- Evaluate **surrogate model** to obtain outputs

$$f^{(2)}(z_1), ..., f^{(2)}(z_N)$$

- Fit normal dist. $q$ to realizations that correspond to failure

$$\{z_i | I^{(2)}(z_i) = 1, i = 1, ..., N\}$$

- Use Expectation-Maximization (EM) algorithm to fit density

- Derive random variable $\hat{Z}$ with distribution given by $q$
Multifidelity importance sampling

Step 2: estimation of failure probability

- Draw $M \in \mathbb{N}$ realizations $\hat{Z}_1, \ldots, \hat{Z}_M$ of $\hat{Z}$ (biasing)
- Evaluate **high-fidelity model** to obtain outputs $f^{(1)}(\hat{Z}_1), \ldots, f^{(1)}(\hat{Z}_M)$
  - typically have $M \ll N$, and therefore fewer high-fidelity model evaluations
- Derive the multifidelity importance sampling (MFIS) estimate
  
  $$P_f^{\text{MFIS}} = \frac{1}{M} \sum_{i=1}^{M} I^{(1)}(\hat{Z}_i) w(\hat{Z}_i)$$

- We can show unbiasedness of the MFIS estimator
  $$P_f = \mathbb{E}_q[P_f^{\text{MFIS}}]$$
• Given are $k - 1$ models
\[ f^{(2)}, \ldots, f^{(k)} : Z \to Y \]

• Approximation qualities of these sources unknown

• Which of these should we use for constructing $q$?

• Our approach: Mixed MFIS
  • Use each surrogate $f^{(i)}$ to construct a density $q_i$, for $i = 2, \ldots, k$
  • Sample from all these densities $q_2, \ldots, q_k$ and combine samples
  • Mixed MFIS estimator $P_f^{\text{Mixed}}$ derived as in [Owen et al, 2000]

• Known that
\[ \frac{\text{Var}[P_f^{\text{Mixed}}]}{k - 1} \leq \min_{i=2,\ldots,k} \text{Var} \left[ I^{(1)} \frac{p}{q_i} \right] \]

• Our $P_f^{\text{Mixed}}$ is up to factor $k - 1$ as good as using the surrogate that minimizes variance
Locally damaged plate: MFIS

Estimate the probability that the deflection exceeds a critical value

- Biasing density constructed from $N = 10^6$ realizations
- Using surrogate only leads to large bias
- MFIS leads to unbiased estimate of $P_f$
- If ROM available, speedup of up to $10^4$, cf. high-fidelity
6. Bayesian Optimization and Multi-Information Source Optimization

Optimal management of multiple sources of information
Bayesian optimization is a powerful and flexible foundation for multifidelity modeling.
1. **Statistical model**
   - uses data and Bayes’ Theorem to compute posterior distribution

2. **Optimization via a value of information analysis**
   - uses the posterior distribution and an acquisition function to decide what data to obtain next
• Instead of modeling $f_{\text{high}}(x)$ directly using a Gaussian process, use a Gaussian process to capture the error between $f_{\text{high}}(x)$ and $f_{\text{low}}(x)$ (Kennedy and O’Hagan, 2000)

$$G(x) = f_{\text{low}}(x) + \mu + \beta^T x + Z(x)$$

• Low-fidelity information can greatly improve convergence rates

• Model calibration also useful when high-fidelity analysis is black box – only need sample information
Efficient Global Optimization (Jones, 1998)

- Define $f_{\text{min}}$ as the lowest observed value of the high-fidelity function

$$f_{\text{min}} = \min_{j=1,\ldots,M} f_{\text{high}}(x^j)$$

- Improvement would be finding a value of $f_{\text{high}}$ that is lower than $f_{\text{min}}$

- Using Gaussian process model, improvement at point $x$ is

$$I(x) = \max(f_{\text{min}} - G(x), 0)$$

- Maximum expected improvement is

$$E[I(x)] = (f_{\text{min}} - m(x)) P \left( \frac{f_{\text{min}} - m(x)}{\sigma(x)} \right) + \sigma p \left( \frac{f_{\text{min}} - m(x)}{\sigma(x)} \right)$$

where $P(\cdot)$ is the standard normal cdf and $p(\cdot)$ is the standard normal pdf
EGO Algorithm

EGO algorithm from Jones (1998):

- Sample the high-fidelity function at $M$ design points

- Find the point that globally maximizes the expected improvement in $f_{\text{high}}(x)$

- do
  - Calculate $f_{\text{high}}(x)$ at the point maximizing the expected improvement and construct a new Gaussian process model
  - (Optional) Validate the new Gaussian process model using cross-validation
  - Find the point that globally maximizes the expected improvement in $f_{\text{high}}(x)$

while the maximum expected improvement is larger than a tolerance
Multifidelity surrogates can also embed estimation of information source fidelity

A standard Gaussian process model encodes uncertainty due to training point locations

Our fidelity function encodes confidence in underlying info source (model discrepancy)

Our multifidelity surrogate
- combines training uncertainty and fidelity
  \[ \sigma_t = \sqrt{\sigma_{GP}^2 + \sigma_f^2} \]
- fuses information from multiple sources

A particularly important source of uncertainty in early-stage design

[Lam, Allaire, Willcox, AIAA 2015]
Multi-information source optimization with general model discrepancies

misoKG samples to maximize expected information gain per unit cost

[Poloczek, Wang, Frazier, NIPS 2016]

Two kinds of uncertainty

I) Model discrepancy
The internal model of each information source deviates from reality. This results in an unknown bias, e.g., due to an incomplete implementation of physical laws.

II) Noise and numerical errors
when observing the output of an information source.

→ More general than traditional “multifidelity”; no hierarchy that limits correlations. Indeed, misoKG exploits correlations among information sources.

• GP regression: a unified treatment of the various (latent) quantities
• Value of information analysis to decide optimally in a single, coordinated step what information source to query and for what design. Balances expected information gain and cost for every sampling decision.

Assemble-To-Order (8d)
Gaussian process regression naturally accommodates derivative observations.

Exploiting availability of adjoints gives better scalability.

- Value of information analysis must account for the expected information gain and the cost of sampling both \( f(x) \) and \( \nabla f(x) \).
- Derivative observations can be incomplete and/or noisy.

[Wu, Poloczek, Wilson, Frazier, NIPS 2017]
Multifidelity stability boundary characterization

- Combines many cheap ROM evaluations and few expensive FOM evaluations
- New Bayesian optimization formulation that adaptively chooses when and where in parameter space to evaluate a ROM

[Marques, Lam, W. NIPS 2018]

"Brute force" grid search

Multi-fidelity approach

- Rigorous multi-fidelity framework combines accuracy of HFM with computational efficiency of ROM
- Use Bayesian optimization formulation: Gaussian process model + new acquisition function to drive adaptive sampling based on Knowledge of Boundary (KoB)

Tubular reactor test problem (LCO instability): algorithm explores using ROM + handful of carefully placed HFM evaluations
7. Conclusions
Multifidelity strategies for the outer loop

Leverage approximate models but maintain guarantees on outer-loop result

- Multifidelity optimization (MFO) via trust region model management
- Multifidelity Monte Carlo (MFMC): a control variate formulation for estimating means
- Multifidelity Importance Sampling (MFIS): an importance sampling formulation for estimating probabilities
- Bayesian optimization and Multi-Information Source Management (MISO)
- Many more topics to cover!
  - Many methods to construct multifidelity surrogates
  - Bayesian optimization in a variety of settings
  - MFMC for global sensitivity analysis
  - Optimization under uncertainty (control variate, importance sampling formulations)
  - Adaptive delayed acceptance MCMC
References


References


References


References


References


