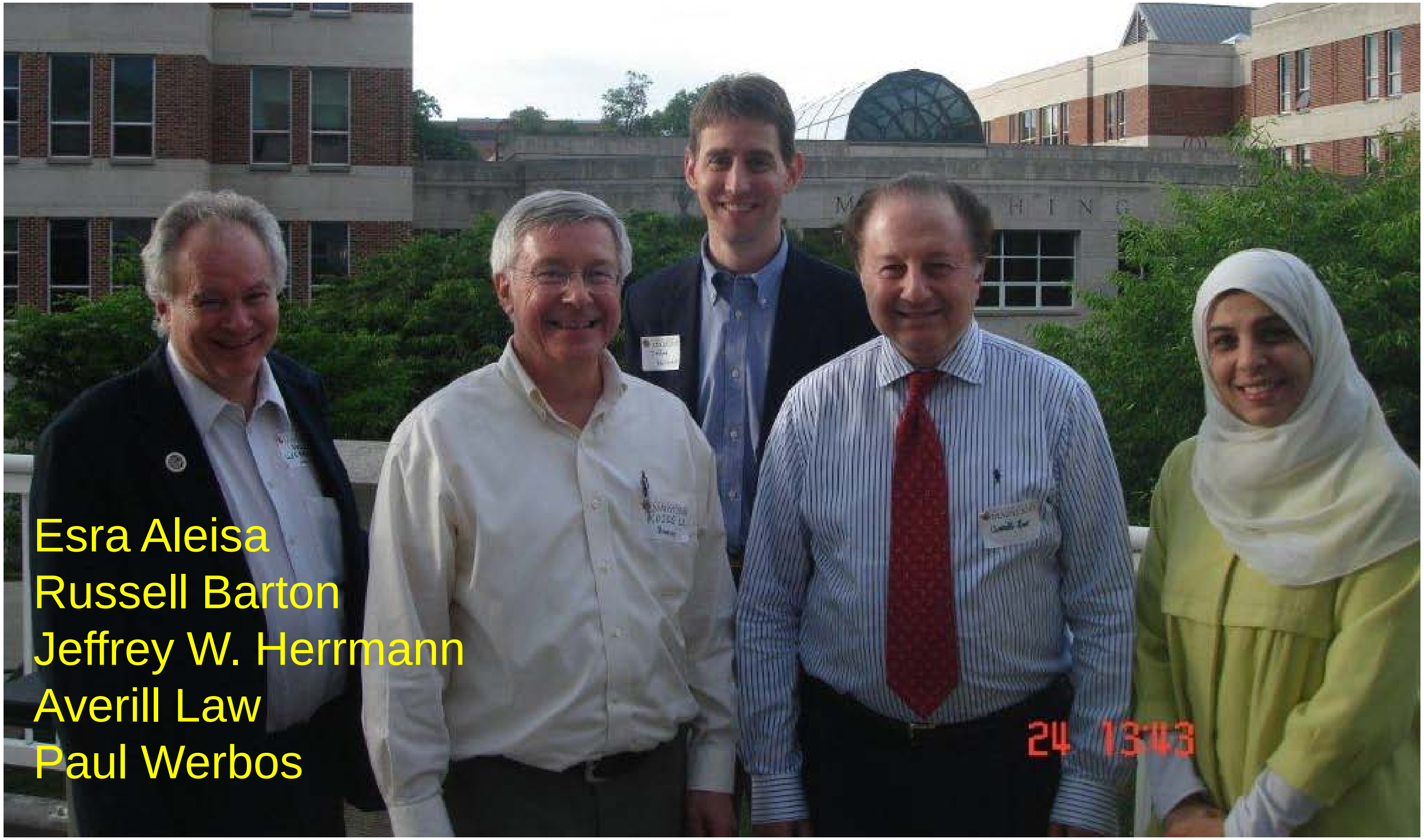
Metamodeling WHAT?



Why metamodel?

- To avoid running expensive simulation models during prediction or optimization.
- To gain insight into and understanding of system response to drivers, variable interactions, and sensitivity.

Forward Metamodels

- $X \rightarrow$ simulation $\rightarrow$ $Y \rightarrow$ function $\rightarrow Z$
- $Y = f(X)$ is an object.
- $Z = g(Y) = g(f(X)) = h(X)$.
- Metamodel $Y \approx \hat{f}(X)$
- Metamodel $Z \approx \hat{h}(X)$
  – Metamodels depend upon model type, experiment design (samples), fitting technique

Inverse Metamodels

- Problem instance I
- I $\rightarrow$ optimal objective function value v*(I)
  - Metamodel $v^* \approx \hat{a}(I)$
- I $\rightarrow$ optimal solution x*(I)
  - Metamodel $x^* \approx \hat{b}(I)$
- Target T
- I, T $\rightarrow$ satisfactory solution x(I,T)
  - Metamodel $x \approx \hat{c}(I,T)$

Research priorities

- Determining when it is better to create metamodel of simulation output Y or better to create metamodel of objective (constraint) function Z.
- Ability to prescribe metamodeling technique based on problem characteristics.
- Better understanding of screening strategies that can identify most important design variables.
- Exploring link between sampling to create metamodels and sampling to generate next point in a search algorithm.
- Improving global fit of metamodels used to cover a range of scenarios $\hat{f}(X_1, X_2)$, where X_1 is the design and X_2 is the scenario.

Research priorities

- Creating useful inverse metamodels to find good points quickly and to generate one or more good initial points for search algorithms.
- Creating universal approximators (metamodels with continuous, discrete/integer, and categorical variables).
  - Learning appropriate distance metrics.
- Developing good hybrids on global metamodels and local metamodels.
- Expand expected improvement to techniques beyond spatial correlation; improve theoretical and empirical understanding of expected improvement techniques.

Research priorities

- Defining and measuring robustness of metamodels for prediction, optimization, and sensitivity
  - For example, linear interpolation is robust (“fit for use”) across many classes of responses.
- Developing metamodels for complex objects: schedules, networks, probability distributions, e.g.
  - How to parameterize an object (e.g., using the dimensions of an object instead of the thousands of nodes in a finite element model).
- Expand expected improvement to techniques beyond spatial correlation; improve theoretical and empirical understanding of expected improvement techniques.

Barriers to Practice

- Understanding metamodels
  - Managers’ comprehension
  - Analysts’ comprehension of assumptions of metamodeling procedure, avoid using metamodels outside of range of sample points.
  - When to use which type of metamodel
- High-dimensional problems require many sample points.
- Technology transfer
  - No software developers to build robust implementation of new methods to extend commercial software
  - Lack of good examples to show applicability to other problems

Next steps

- Create examples of creating and using metamodels for a variety of problem/model types
  - Supply chain management, engineering design, transportation networks, health care operations, call centers, and others.
- WSC Berlin sessions that include papers on simulation optimization, machine learning, and evolutionary computation.

Metamodeling Uncertainty

- $Y = f(X)$, the response of a stochastic simulation model, is a random variable.
- Given X, we would like to have a metamodel : $X \rightarrow (\hat{f}(X), \hat{\sum}_1(X), \hat{\sum}_2(X))$
  - $\hat{f}(X)$ = estimated response
  - $\hat{\sum}_1(X)$ = estimate of noise (aleatory uncertainty)
  - $\hat{\sum}_2(X)$ = estimate of ignorance (epistemic uncertainty)
- Such a metamodel would provide a better understanding of how to pick the next point to sample.

Inverse Metamodels

- Problem instance I, target performance T
- Want $X : f(X) \geq T$ (for maximization)
- $I, T \rightarrow$ satisfactory solution $X(I,T)$
  - Metamodel $X \approx \hat{c}(I,T)$
- Example
- Challenges
  - Design of experiments over problem instances and targets
  - Metamodels that output values for discrete variables depend upon structure; cf. nonlinear discriminant analysis
  - Metamodels that output complicated structures: schedules, network designs, e.g.

Inverse Metamodels

- Problem min c^T: x: $Ax \leq b$, $x \geq 0$
- instance $I = (A, b, c)$

Inverse Metamodels

- Challenges
  - Under certain conditions, the samples used to generate $\hat{f}(X)$ can be used to fit $\hat{c}(I,T)$ (Barton, 2006).
  - Design of experiments over problem instances and targets
  - Metamodels that output values for discrete variables depend upon structure; cf. nonlinear discriminant analysis
  - Metamodels that output complicated structures: schedules, network designs, e.g.