DOE for Fitting Forward and Inverse Simulation Metamodels

Russell R. Barton
The Smeal College of Business
The Pennsylvania State University

Overview

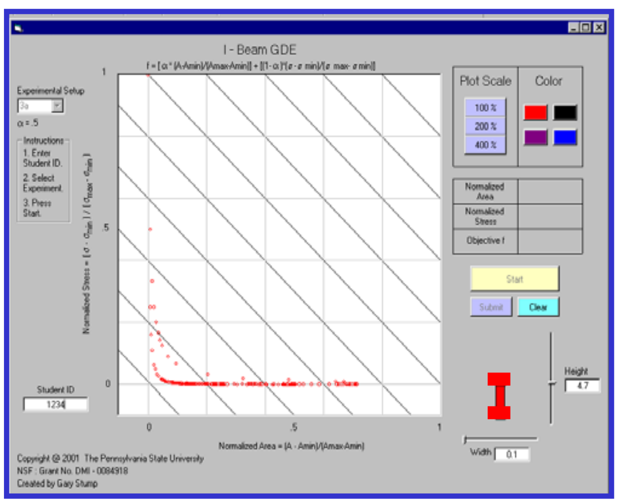
- Introduction
- Inverse Metamodeling Idea
- Simulation of Inventory Control at Freescale Semiconductor
- Key Issues in Developing Inverse Metamodels
- Model-Based Optimal Design Types
- Designs based on Optimal Spatial Characteristics
- A Maximin Combined Forward-Inverse Design Method
- Randomness and its Impact on the Phase 1 Model
- Conclusions and Future Work
- Acknowledgments

Acknowledgments

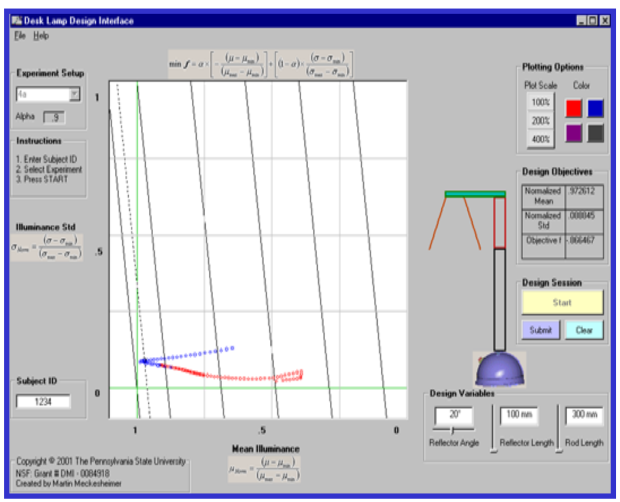
- Bruce Ankenman
- Timothy Simpson (ME, Penn State) and Martin Meckesheimer (Boeing) – collaborators on early work
- Smeal College of Business Competitive Research Award
- National Science Foundation DMI-0084918

Introduction

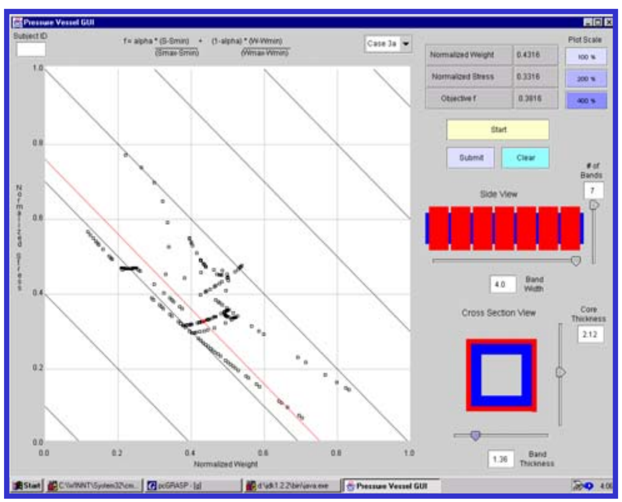
- Simulation-based design used for products and systems
- Models might be discrete-event simulation, finite element modeling or other engineering design modeling software



I-Beam



Desk Lamp



Pressure Vessel

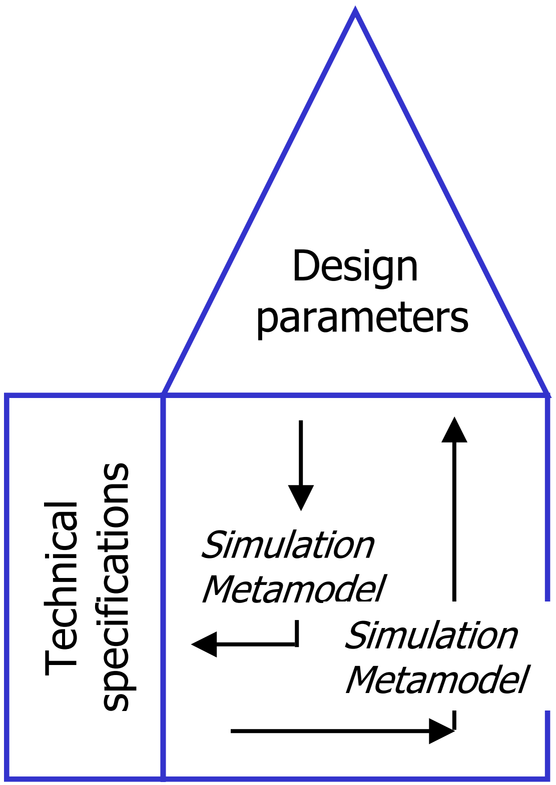
Introduction

- Why inverse models?
  - House 2 of QFD process and customer driven design
  - *Goal:* formalize the mapping

    technical specifications → design parameters
  - *Strategy:* metamodeling (approximations)

    metamodels run faster than engineering model
  - This is the *key* to customer-driven mapping:

    maps inverse to the usual engineering models can be fitted with few (or no) extra forward runs, even when no explicit inverse model exists



Metamodeling: Approximation

- A *metamodel* is a model of a model

Simulation model
$f(X) = E(Y)$

↓ ↓ ↓

Metamodel
$m_f(X) \sim f(X)$

<u>Metamodeling process for standard regression</u>

1. Identify region of interest R_x
2. Choose basis functions for $m_f(X)$
3. Select fitting design $X \in C_x$ $X \rightarrow D$
4. Estimate true response $f(X) \sim y$ or $\bar{y}$
5. Fit approximation $b = (D'D)^{-1}D'y$
6. Validate approximation (metamodel) fit over R_x

Inverse Metamodel Idea

- Given
  - $X = n \times k$ matrix of design parameters
  - $Y = n \times p$ matrix of performance measures
- Use (X, Y) data to fit p forward metamodels
- Use (X, Y) as (Y, X) to fit k inverse metamodels

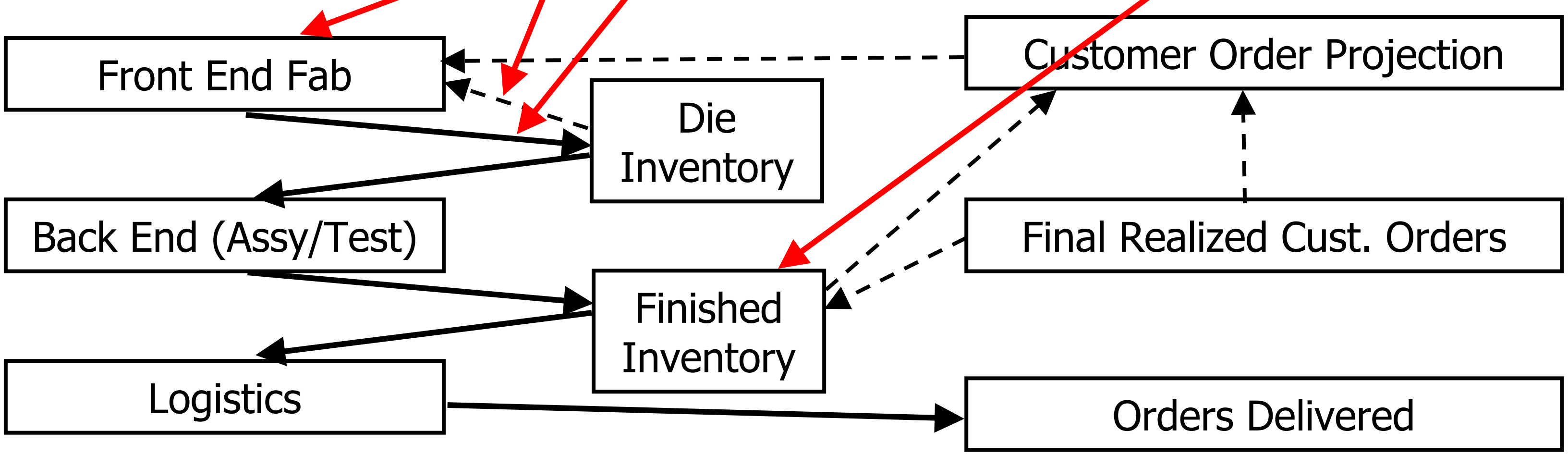
Note

Fitting an inverse metamodel *is not* generally the same as the forward metamodel's inverse

Simulation of Inventory Policy at Freescale Semiconductor

(Morrice, Valdez, Chida and Eido, WSC05)

- Objective: How does inventory policy affect on time delivery (FOTD)?
- Added: cost implications for change (COST)
- Inventory level (MaxDieQ) controls job release rate
- Front End Lead Time (FELT) a constraining factor but expensive to improve

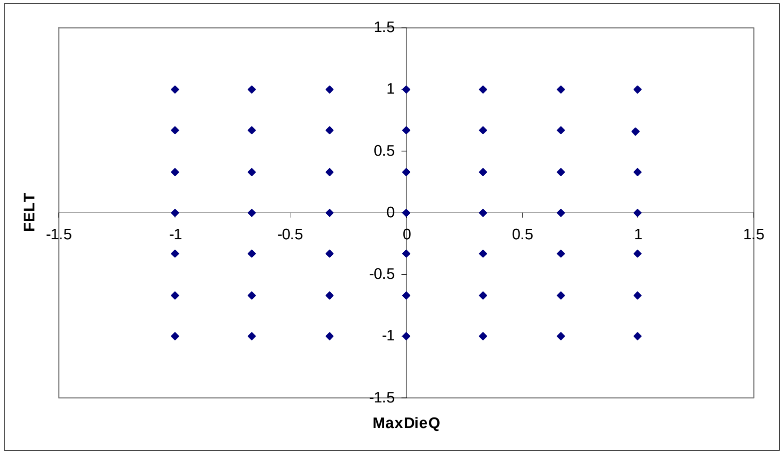


Simulation of Inventory Policy at Freescale Semiconductor
(Morrice, Valdez, Chida and Eido, WSC05)

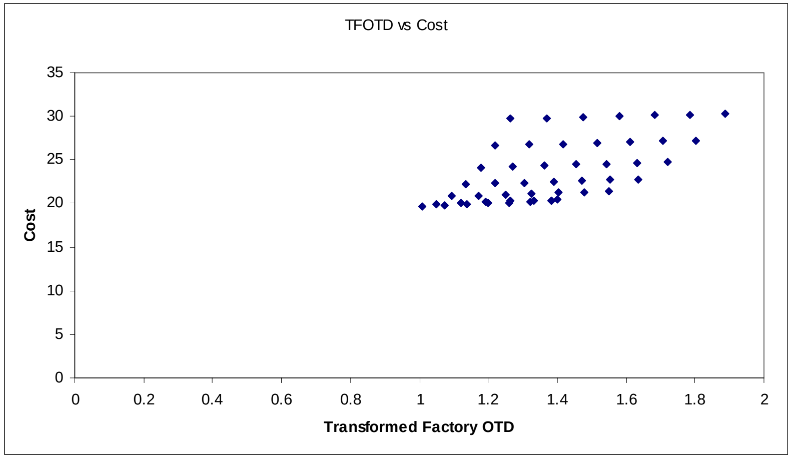
- Simulation used to model effects of MaxDieQ and FELT on performance (FOTD, COST)
- Simulation metamodels allow one to explore how design changes affect performance
- Ideally, would rather explore 2-D performance space and have corresponding design identified
- How do spaces relate?

Simulation of Inventory Policy at Freescale Semiconductor
(Morrice, Valdez, Chida and Eido, WSC05)

- Design (Domain) and Performance (Range) Spaces



X space



Y space

Forward and Inverse Metamodels: Key Issues

Key Issues

- When the target point $y_{desired}$ occurs at the local minimum or maximum of one or more elements of $\boldsymbol{f}$, the function will not be (locally) invertible.
- When the dimension of $\boldsymbol{y}$ (i.e., p) does not match the dimension of $\boldsymbol{x}$ (i.e., $p \neq k$), how can an inverse function be established?
- How can one find an 'optimal' experiment design for simultaneously fitting $\boldsymbol{m}_f$ and $\boldsymbol{m}_{f^{-1}}$?
- What is the relationship between $(\boldsymbol{m}_f)^{-1}$ and $\boldsymbol{m}_{f^{-1}}$?
- How can constraints on $\boldsymbol{x}$ and $\boldsymbol{y}$ be included in the experiment design methodology?

Forward and Inverse Metamodels: Key Issues

Today We'll Focus on One Key Issue

- How can one find an 'optimal' experiment design for simultaneously fitting $\boldsymbol{m}_f$ and $\boldsymbol{m}_{f^{-1}}$?

Model-Based Measures of Design Optimality

Consider the linear model plus interaction:

$$f(X) = y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_{12}$$

Run Matrix

$$X = \begin{pmatrix} x_1 & x_2 \\ 1 & -1 \\ 1 & 1 \\ -1 & -1 \\ -1 & 1 \end{pmatrix}$$

Design Matrix

$$D = \begin{pmatrix} 1 & x_1 & x_2 & x_{12} \\ 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}$$

- *D*-optimality minimizes the variance of the parameter estimates ***b***
  - choose X to *maximize* $\det(D'D)$

Model-Based Measures of Design Optimality

- *G*-optimality minimizes the variance of a predicted value at $\boldsymbol{x}_0$ based on $\boldsymbol{b}$
  - choose *X* to *minimize* $\max\{f(\boldsymbol{x}_0)'(D'D)^{-1}f(\boldsymbol{x}_0)\}$
- *A*-optimality minimizes the average variance of the components of $\boldsymbol{b}$
  - choose X to *minimize* $\Sigma(1/\lambda_q)$, where $1/\lambda_q$, $q = 1, \ldots, r$ are the eigenvalues of $(D'D)$
- *E*-optimality minimizes the variance of any combination $\boldsymbol{a}'\boldsymbol{b}$ with $\Sigma \boldsymbol{a}_q=0$ and $\boldsymbol{a}'\boldsymbol{a}=1$
  - choose *X* to *minimize* $\max\{1/\lambda_q\}$, where the maximum is over $q = 1, \ldots, r$

Model-Based Measures of Design Optimality

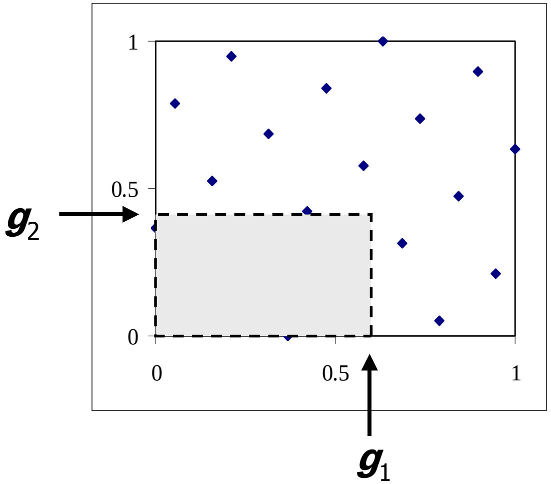
- Combined Forward-Inverse D-Optimal Designs were developed by Barton, Meckesheimer and Simpson using a two-phase approximation to generate x-optimal optimal designs for the first phase and y-optimal designs for the second phase
- ***These are Model-Based optimality measures, but need know only the form, not the model coefficients***
- This means that simultaneous selection of the design points is fine – observing responses that give more information about the unknown coefficients does not give an opportunity to improve the design
- Applicable for standard regression metamodels, not for more general metamodels such as spatial correlation (kriging), radial basis functions, neural networks, smoothing splines
- For more general metamodels, sequential designs based on reducing prediction variance have advantages

Other Model-Based Designs

- Sequential Minimum Prediction Variance Designs: van Beers and Kleijnen use cross-validation/jackknife methods and bootstrapping methods to estimate prediction variance of metamodels at untested points, then choose the untested point with the highest prediction variance for the next experiment

Designs based on Optimal Spatial Characteristics

- **Uniform Designs based on Discrepancy** minimizes the difference between the percentage of points falling in a particular region on a unit cube and the percentage of volume occupied by this region



- **Orthogonal Arrays** are fractions of multi-level factorial designs with good projection properties (projection dimension depends on 'strength')
- **Latin Hypercube** designs are orthogonal arrays of strength 1, easy to construct but must be evaluated against specific metrics (e.g. D-optimality) to get good designs

Designs based on Spatial Characteristics (cont.)

- Minimax distance designs provide low bias by minimizing the maximum distance between each sample point and every point in the design space
- Maximin distance designs provide low variance by maximizing the minimum distance between any two points in the design space, and D-optimal under general conditions

- A Two-Phase Combined Maximin Forward-Inverse design will be described next

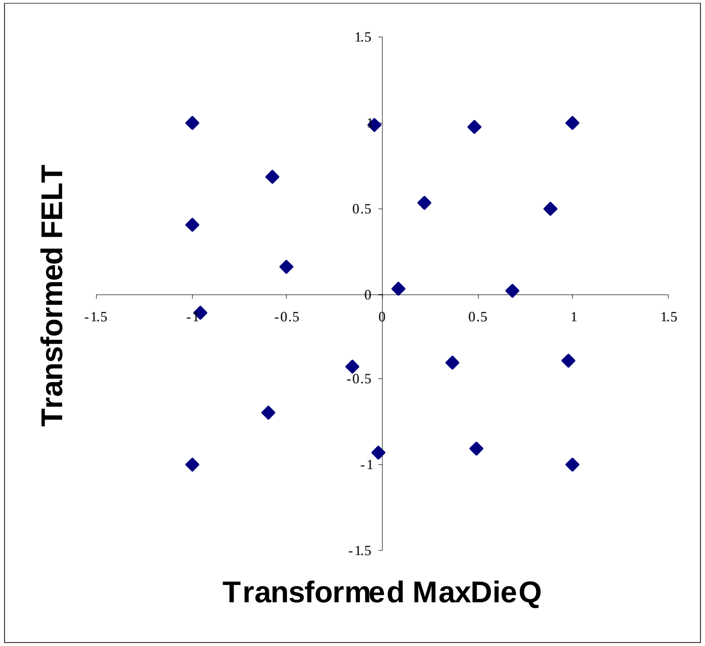
A Combined Forward-Inverse Maximin Strategy

Two-Phase Maximin Method: Assume use $N > N_{min}$ runs

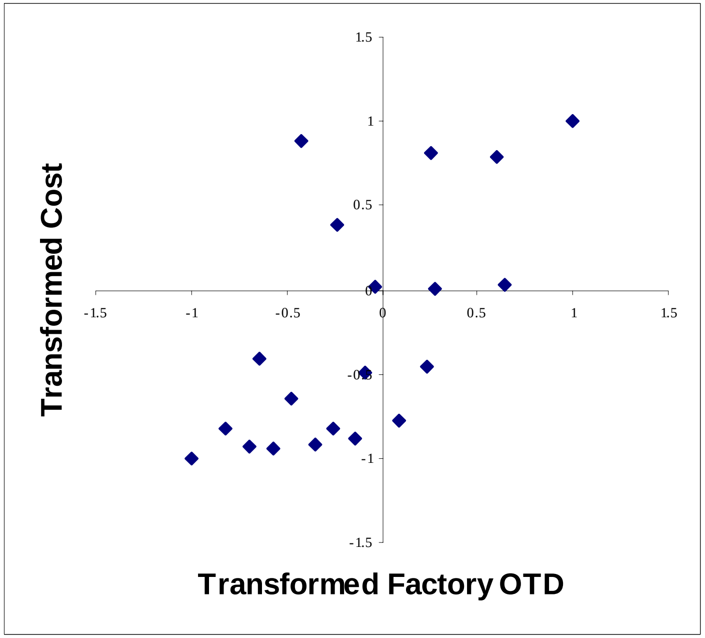
1. Use $N_0 \geq N_{min}$ *maximin* (in terms of x) first phase forward model runs ($\boldsymbol{X}^{\mathbf{Mmx}}$) to generate image points in *y*-space ($\boldsymbol{Y}^{\mathbf{1}}$).
2. Scale $\boldsymbol{X}$ and $\boldsymbol{Y}$ to +/-1 for each coordinate. Keep this scaling through the rest of the process.
3. Fit the phase 1 forward metamodel, $\boldsymbol{m}_{\mathbf{f1}}$.
4. In second phase, select $N - N_0$ design points ($\boldsymbol{X}^{\mathbf{Mmxy}}$) that are maximin *both* in terms of $\boldsymbol{X}$ (direct calculation) and $\boldsymbol{Y}$ (by computing the distances for candidate image points $\boldsymbol{Y}^{\mathbf{2}}$ using $\boldsymbol{m}_{\mathbf{f1}}$).
5. Evaluate the models at $\boldsymbol{X}^{\mathbf{Mmxy}}$ to get the true $\boldsymbol{Y}^{\mathbf{2}}$
6. Fit the final forward metamodel ($\boldsymbol{m}_{\mathbf{f}}$) with {$\boldsymbol{X}^{\boldsymbol{Mmx}}$, $\boldsymbol{Y}^{\mathbf{1}}$} U {$\boldsymbol{X}^{\mathbf{Mmxy}}$, $\boldsymbol{Y}^{\mathbf{2}}$}
7. Fit the final inverse metamodel ($\boldsymbol{m}_{\mathbf{finv}}$) with {$\boldsymbol{Y}^{\mathbf{1}}$, $\boldsymbol{X}^{\mathbf{Mmx}}$} U {$\boldsymbol{Y}^{\mathbf{2}}$, $\boldsymbol{X}^{\boldsymbol{Mmxy}}$}, for *y* in $\boldsymbol{Y}^{\mathbf{1}}$ and *y* in $\boldsymbol{Y}^{\mathbf{2}}$ satisfying $y \in C_y$

Inventory Example with Forward Maximin

20 points, Maximin on X for All: Xmin = 0.50, Ymin = 0.13
(scaled units: +/-1)



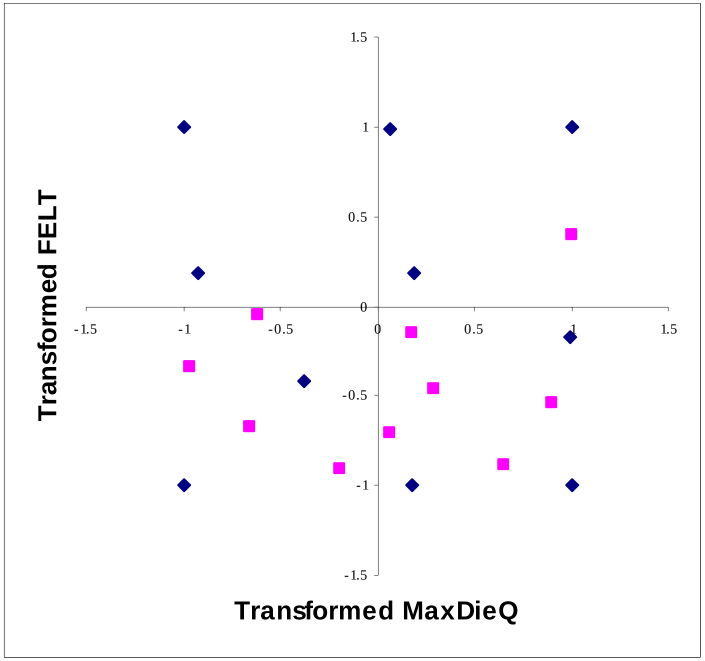
X Space



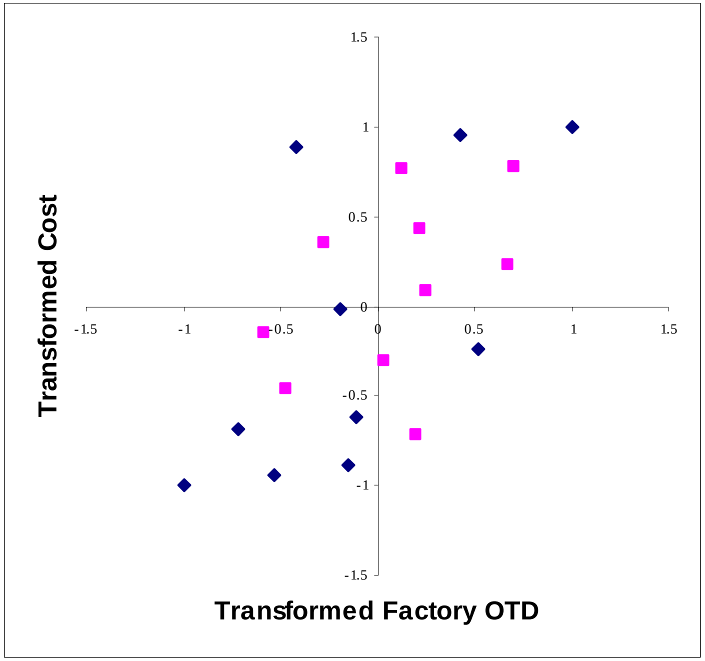
Y Space

Inventory Example with Forward Maximin

20 points, Maximin on X for First 10, X and Y for Next 10:
Xmin = 0.32, Ymin = 0.32 (Y distance same as X)



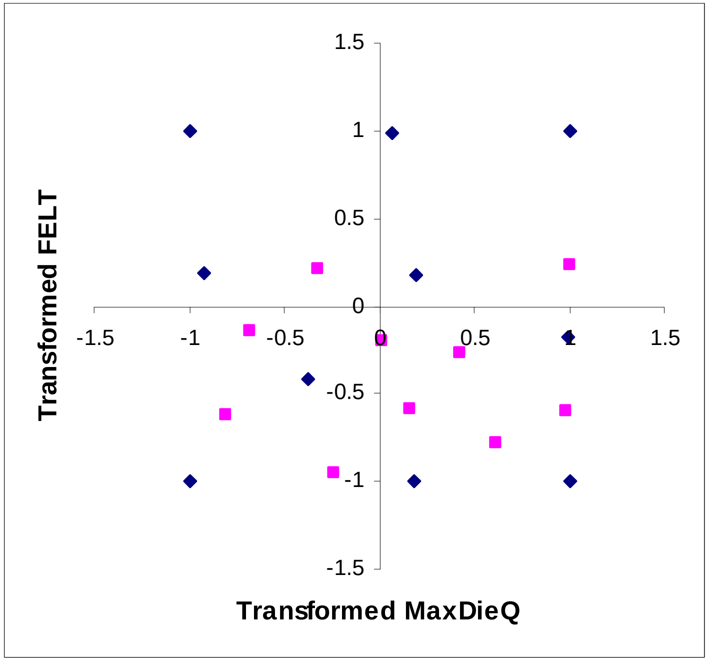
X Space



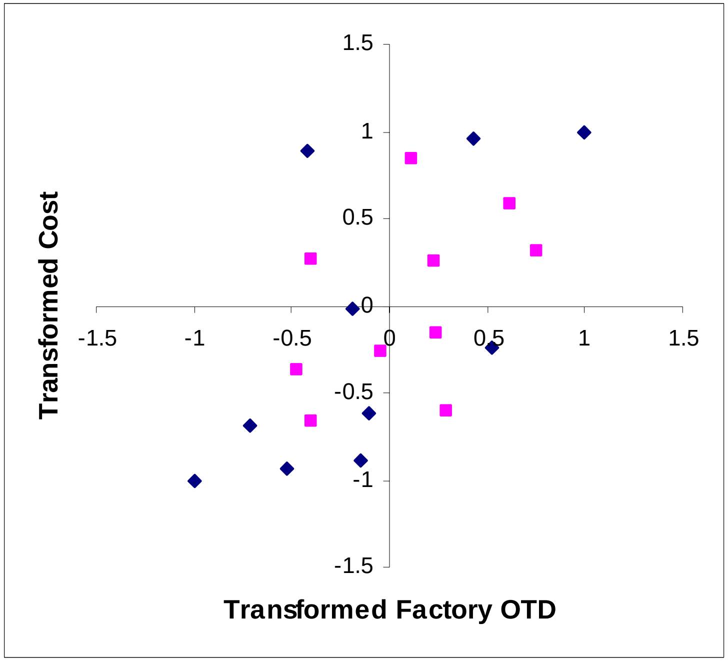
Y Space

Inventory Example with Forward Maximin

20 points, Maximin on X for First 10, X and Y for Next 10:
Xmin = 0.40, Ymin = 0.29 (Y distance sccaled by 1.4)



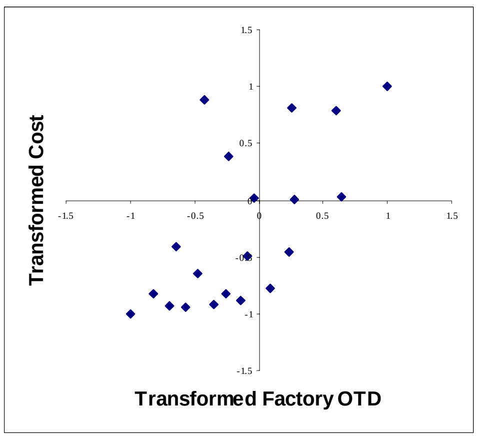
X Space



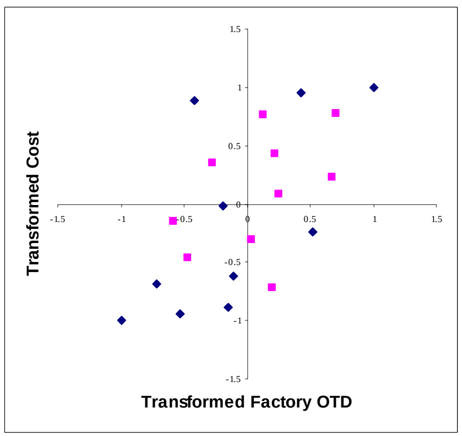
Y Space

Inventory Example with Forward Maximin

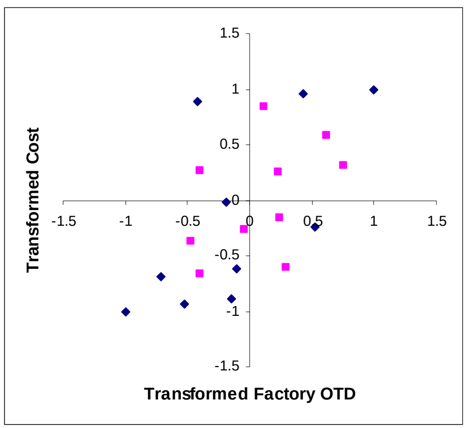
A Comparison of Y-Space Designs



Y Space – Maximin on X



Y Space – Unscaled Dist.



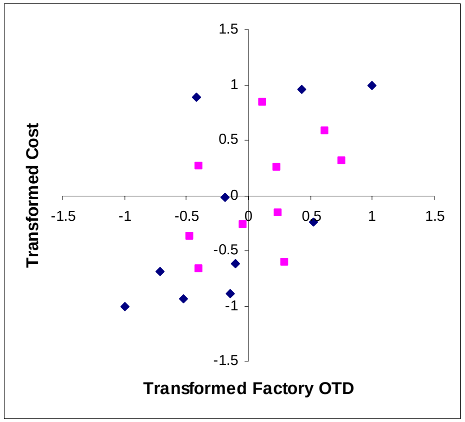
Y Space – Scaled Dist.

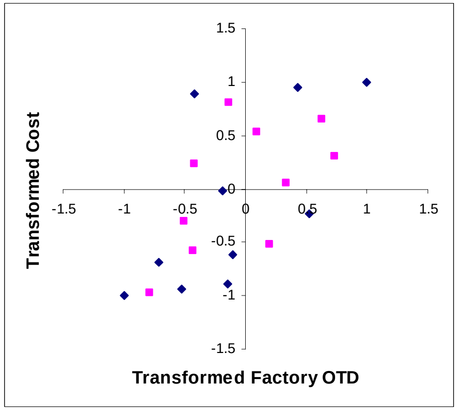
Forward Maximin with Deterioration as Response Noise Increases

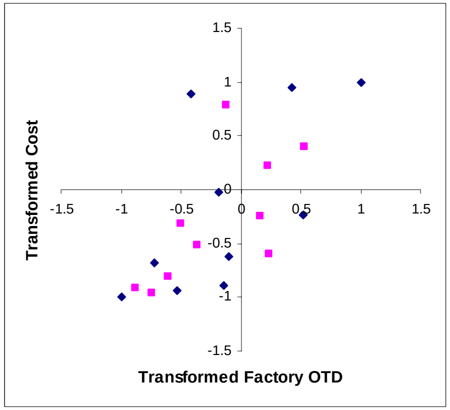
σ (multiple of range of y values)	R^2 for Phase 1 Model for y_1	R^2 for Phase 1 Model for y_2	Min. Dist. for x	Min. Dist. for y
0	1.0	1.0	0.40	0.29
0.1	0.96	0.99	0.41	0.22
0.2	0.86	0.95	0.40	0.14
0.3	0.75	0.91	0.39	0.10
Maximin X only	-	-	0.50	0.13
Maximin Y only	-	-	0.24	0.37

Forward Maximin and Deterioration as Response Noice Increases

A Comparison of Y-Space Designs with Increasing Response Standard Deviation (1.4 dist. scaling)







$\sigma = 0$, $R^2 = 1.0/1.0$ (min dist = .40, .29)

$\sigma = .1*\text{range}$, $R^2 = .96/.99$ (min dist = .41, .22)

$\sigma = .1*\text{range}$, $R^2 = .86/.95$

Conclusions Future Work

- Combined forward-inverse metamodels have a place in product and process design
- Future work: continued development and assessment of spatial designs for non-regression metamodels (splines, radial basis functions, kriging)
- Unresolved issues:
  - Multiple objectives require desirability function, utility function?
  - Fully sequential vs phases?

References

- Barton, R. R. 2005. Issues in development of simultaneous forward-inverse metamodels. *Proceedings of the 2005 Winter Simulation Conference*, ed. M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, 209-217. Available at www.informs-sim.org
- Kleijnen, J. P. C., and W. C. M. van Beers. 2004. Application-driven sequential designs for simulation experiments: kriging metamodeling. *Journal of the Operational Research Society* 55: 876-883.
- van Beers, W. C. M., and J. P. C. Kleijnen. 2005. Customized sequential designs for random simulation experiments: kriging metamodeling and bootstrapping. Tilburg University: CentER Discussion Paper No. 2004-63.
- Meckesheimer, M., R. R. Barton and T. W. Simpson. 2000. Experimental design issues for simultaneous fitting of forward and inverse metamodels. *Proceedings of DETC'00, the 2000 ASME International Design Engineering Technical Conferences*. American Society of Mechanical Engineers DETC2000/DAC-14282.

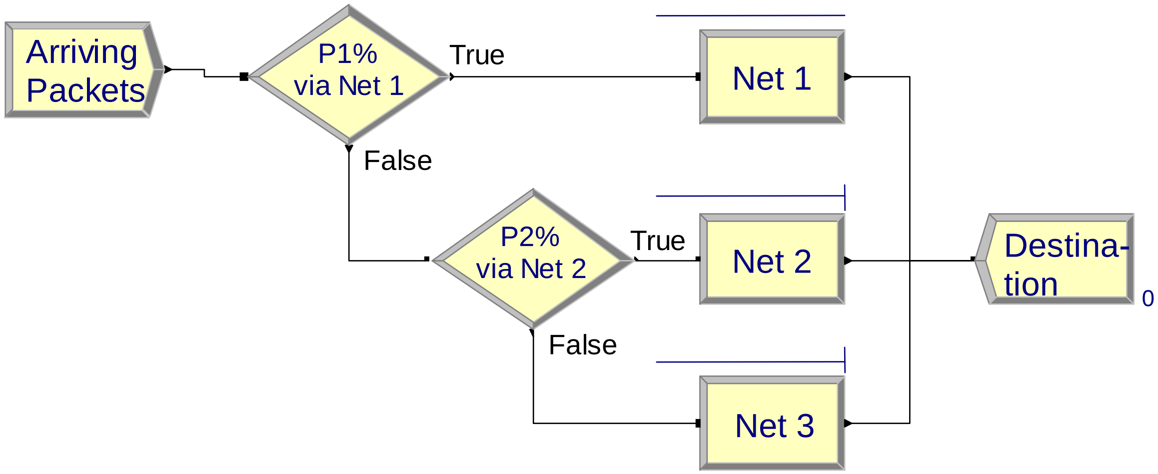
Forward and Inverse Metamodels: Key Issues

Key Issues

- When the target point $y_{desired}$ occurs at the local minimum or maximum of one or more elements of ***f***, the function will not be (locally) invertible
- Measure: To be locally invertible, the function ***f*** must be 1-1. Smooth maps will be invertible locally if and only if the matrix of first derivatives, the Jacobian matrix, $J = [\partial f_i / \partial x_j]$ evaluated at that point is invertible (full rank, i.e. have nonzero Jacobian determinant). For the mapping to be globally invertible, the Jacobian determinant must be nonzero everywhere
- Practical strategies:
  - Check Jacobian of preliminary fitted forward metamodel (randomly, via grid or via global minimization)
  - Select the coordinate functions of ***f*** carefully

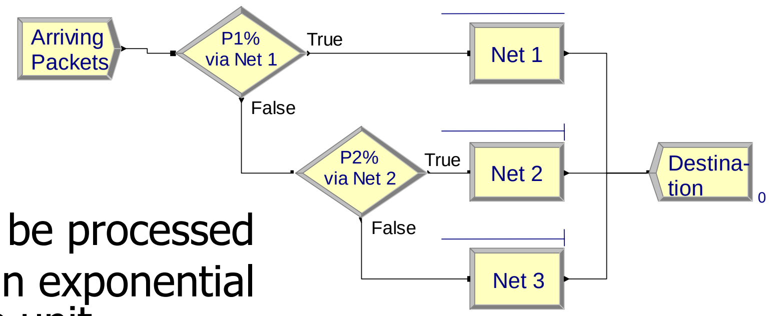
A Network Design Example

- Discrete-event simulation models for service, business or manufacturing process design
- Simple example: network design, routing percentages P_1, P_2



A Network Design Example

- Suppose delay costs at \$.005/time unit each packet is in the system.
- Suppose per-packet processing cost c_i varies by network:
  - \$.03 for network 1,
  - \$.01 for network 2 and
  - \$.005 for network 3.



- Suppose
  - 1000 information packets must be processed
  - packet interarrival times have an exponential distribution with mean = 1 time unit
  - network transit times have triangular distributions with mean E(S) and limits +/- .5
  - with E(S) = 1, 2, and 3 for networks 1, 2, and 3 respectively.
- Objective: find P_1, P_2 (the x's in this example) with desired delay and cost objectives (the y's)

Forward and Inverse Metamodels: Key Issues

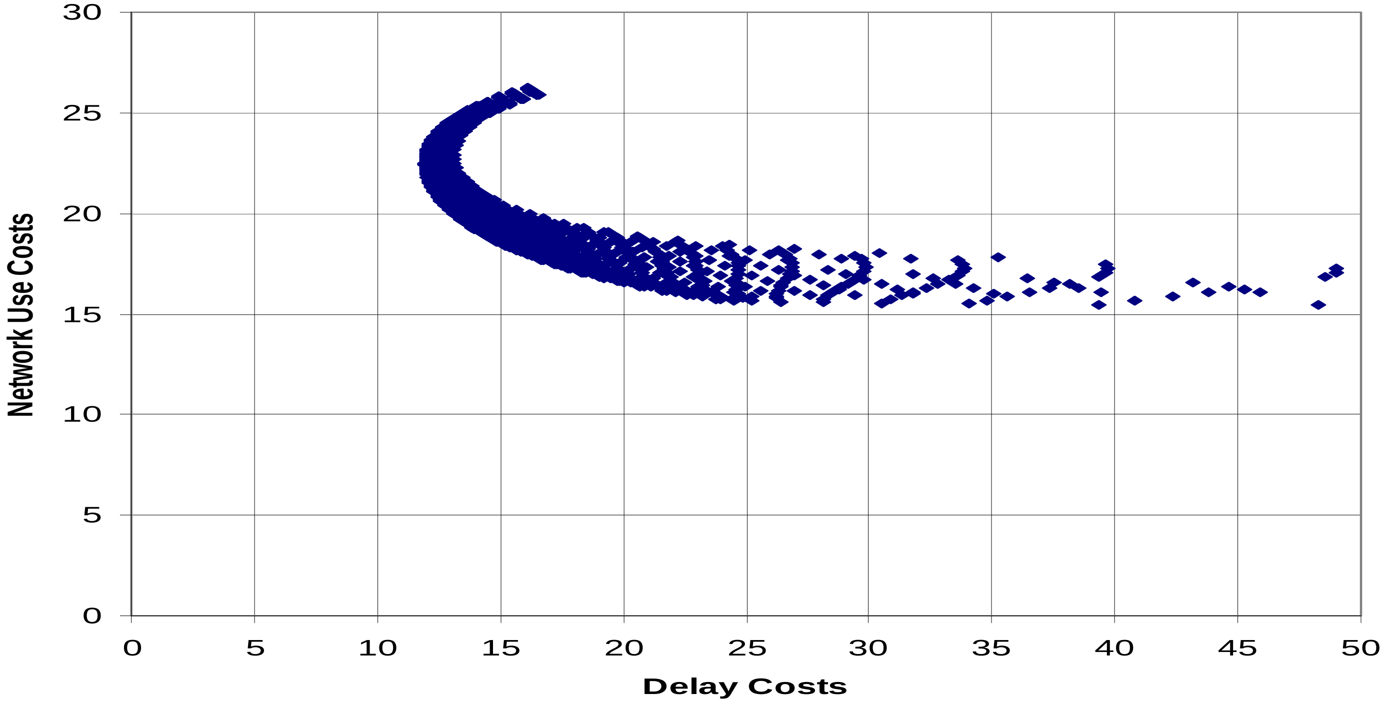
Key Issues

- Select the coordinate functions of $\boldsymbol{f}$ carefully to avoid local maxima/minima
- If each f_i is a monotonic function of the x variables on which it depends - then the map will be invertible. This argues for the decomposition of a total cost function, for example, into separate investment cost and delay cost elements
- Our example: $\boldsymbol{x} = (P_1, P_2)$, and the components of $\boldsymbol{f}$ could include delay costs and network use costs

Forward and Inverse Metamodels: Key Issues

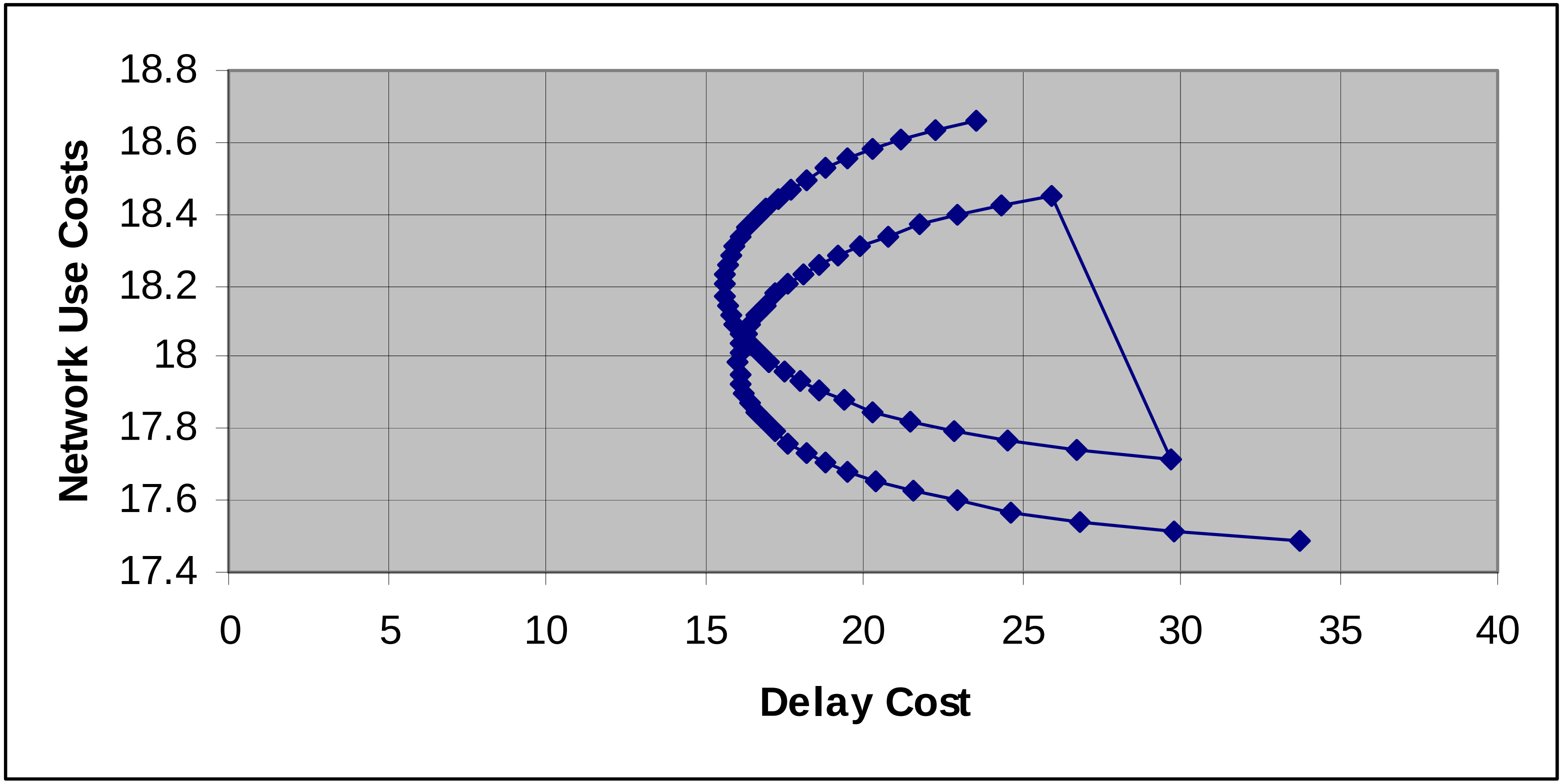
Example: f_1 = total delay cost, f_2 = total use cost

- Image of grid of points in space folds back on itself here – not invertible – why?



Forward and Inverse Metamodels: Key Issues

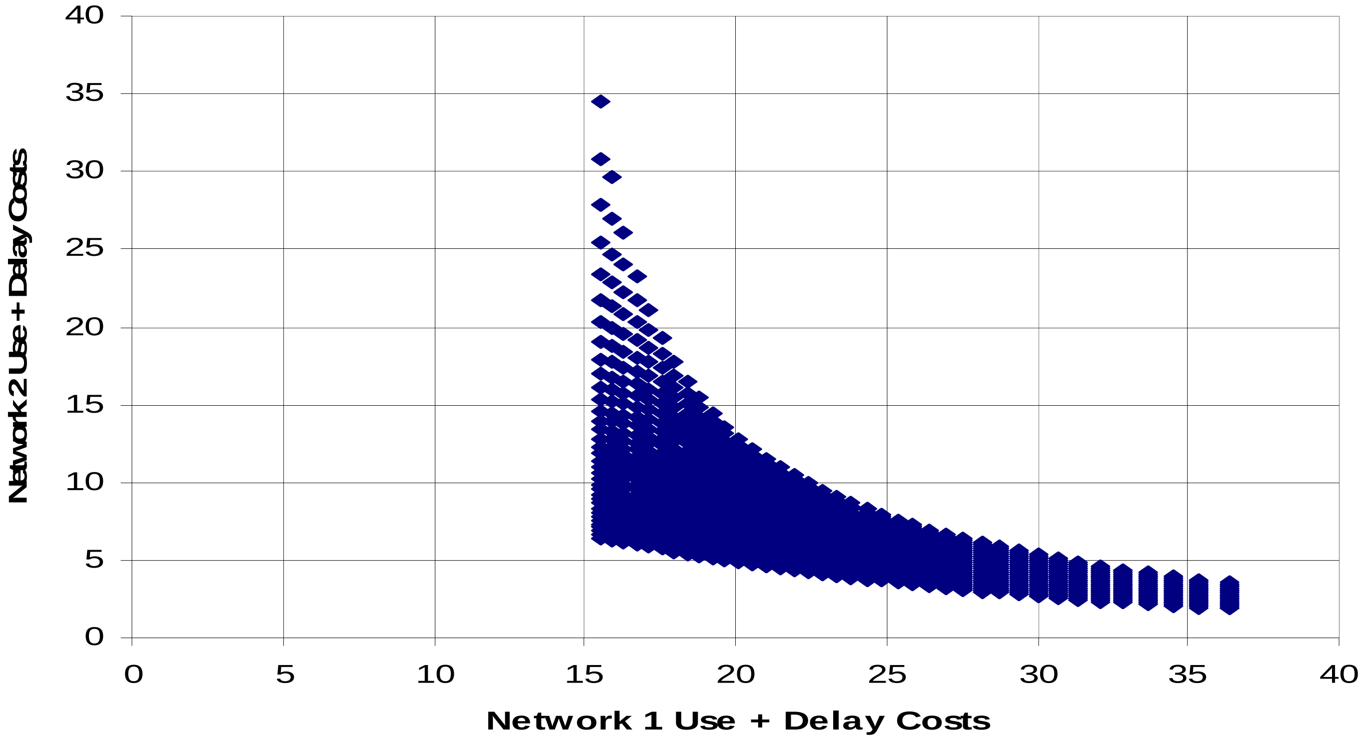
- Easiest to see by plotting values of delay, access costs for two values of P_1 (45%, 46%) as P_2 varies from 45% - 80%:



Forward and Inverse Metamodels: Key Issues

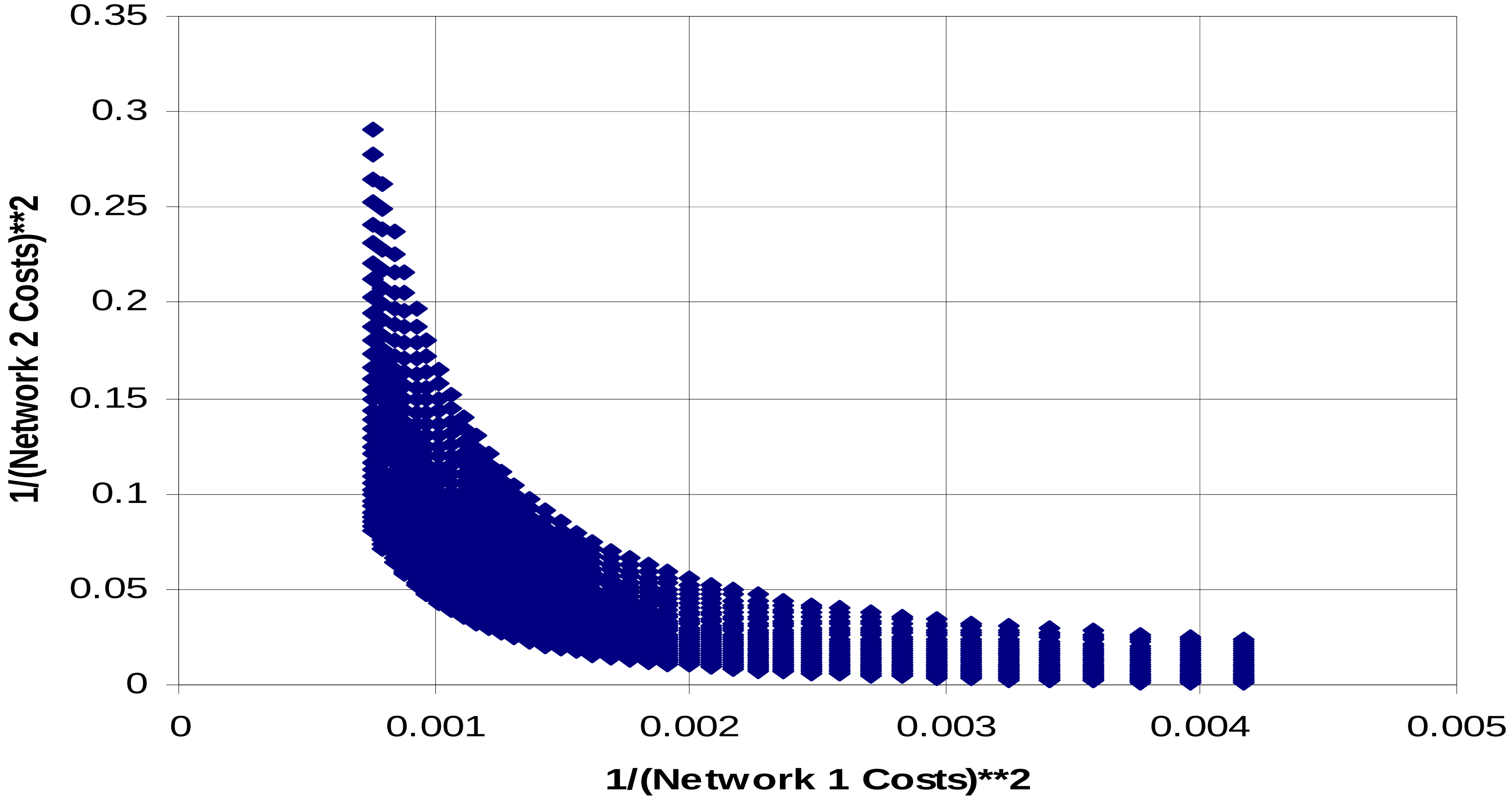
Example: f_1 = network 1 total cost, f_2 = network 2 total cost

- Functions monotonic in (P_1, P_2), resulting image of grid does not fold over itself
- Strange shape of the image region will affect optimal experiment design strategies



Forward and Inverse Metamodels: Key Issues

- Rescaling the responses for homogeneous variance: slope of log s.d. vs log mean (Montgomery) or Maximum likelihood with Box-Cox transformations



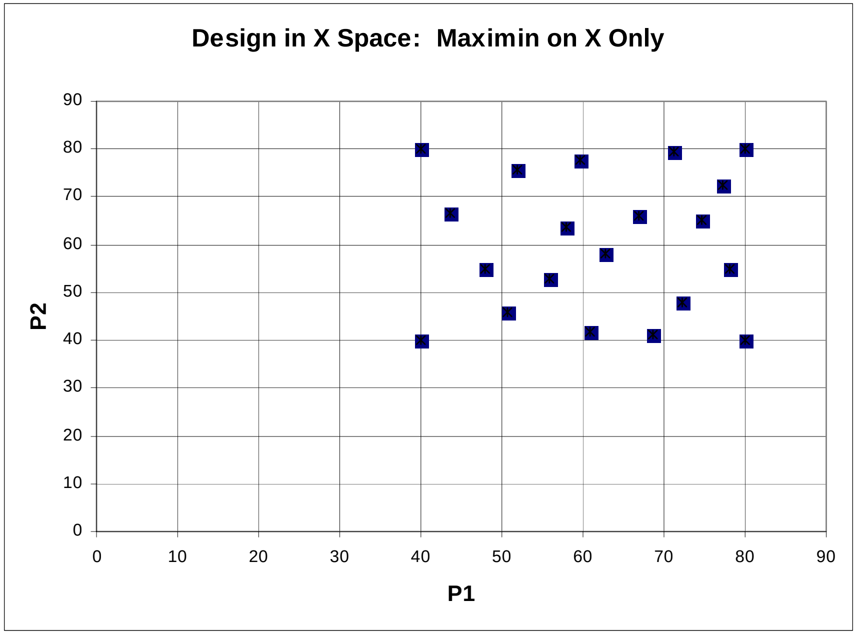
Forward and Inverse Metamodels: Key Issues

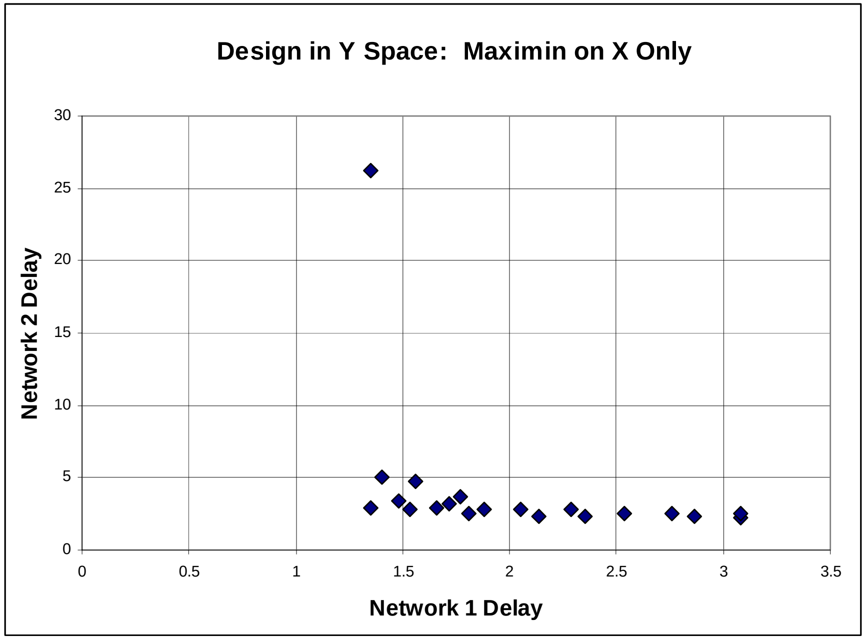
Key Issues

- When the dimension of ***y*** does not match the dimension of ***x***, how can an inverse function be established?
- For the network design example, how to include the total costs associated with traffic on network 3?
- Let $S = \{(s_1, s_2, s_3)\}$ represent the space of total network costs where s_i is the total cost for traffic using network i. Since the x-space has dimension 2 (i.e., (P_1, P_2)), the map from (x_1, x_2) to (s_1, s_2, s_3) generates a surface in S. If the functions are monotonic, any point on this surface can be identified uniquely using the pair (s_1, s_2) or any other pair of s's

Network Example with Forward Maximin

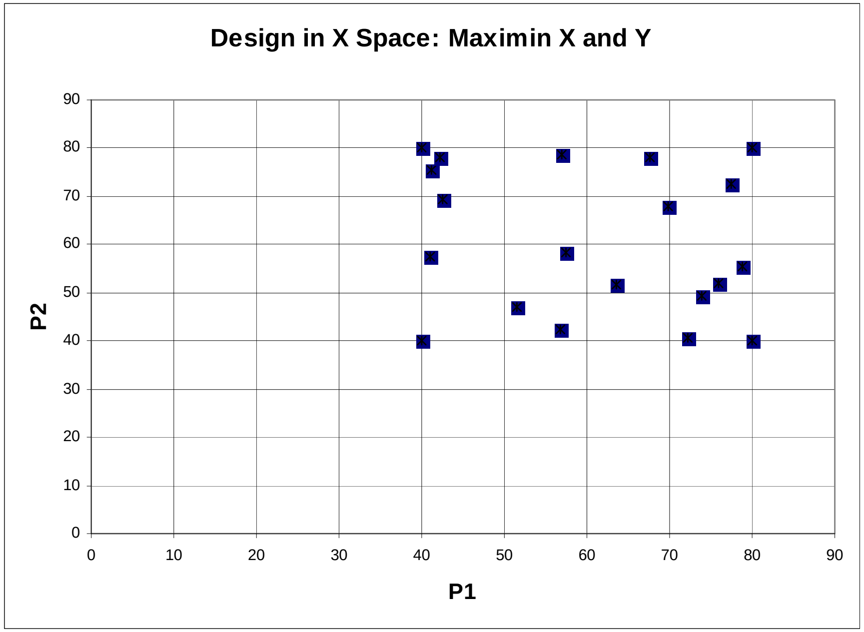
20 points, Maximin on X for All: Xmin = 0.38, Ymin = 0.08
(scaled units: +/-1)

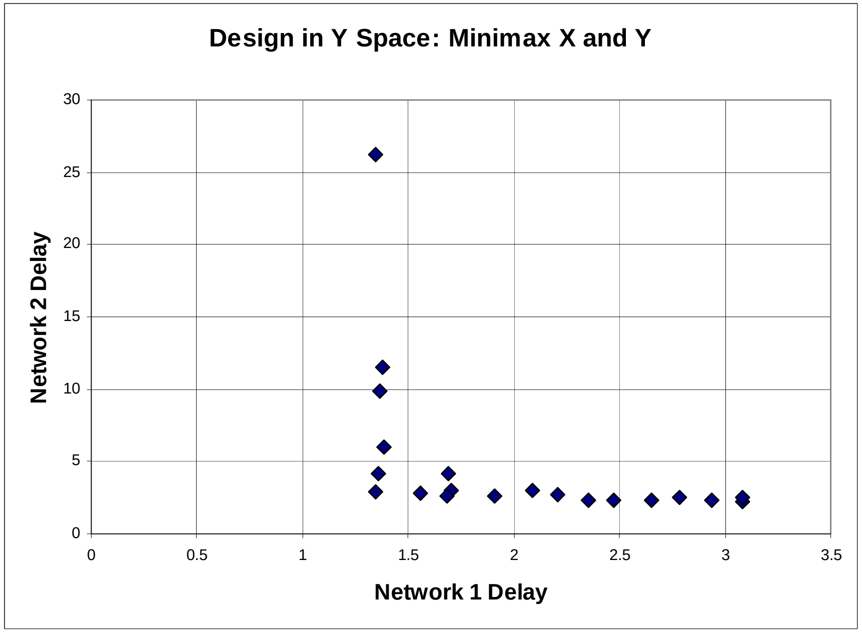




Network Example with Two-Phase Forward-Inverse Maximin Strategy

20 Points, 10 Phase 1 Maximin on X, 10 Phase 2 Maximin X and Y:
Xmin = 0.14, Ymin = 0.14 (scaled units: +/-1)

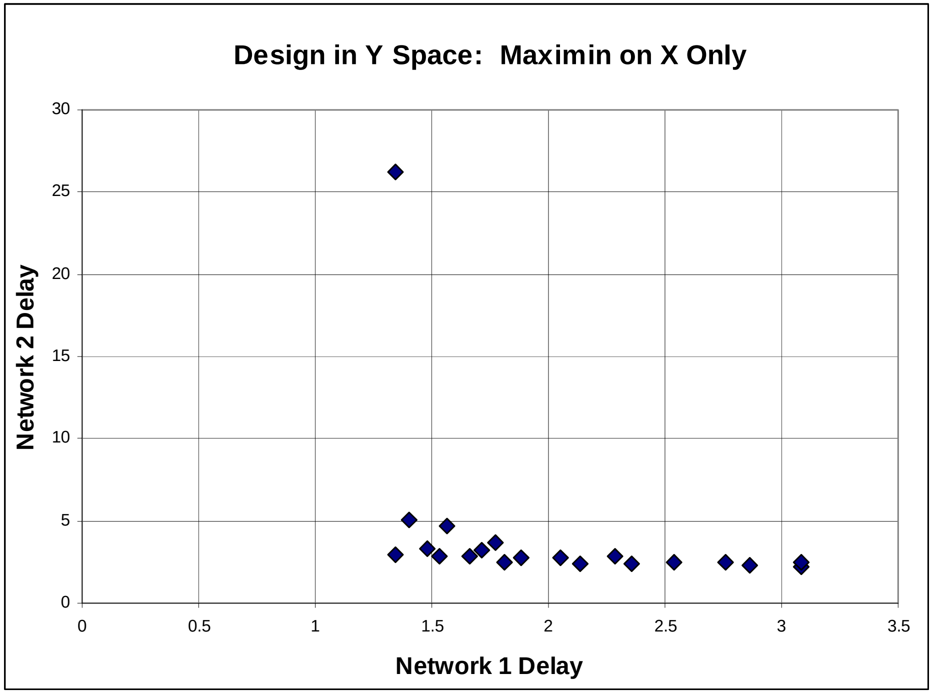


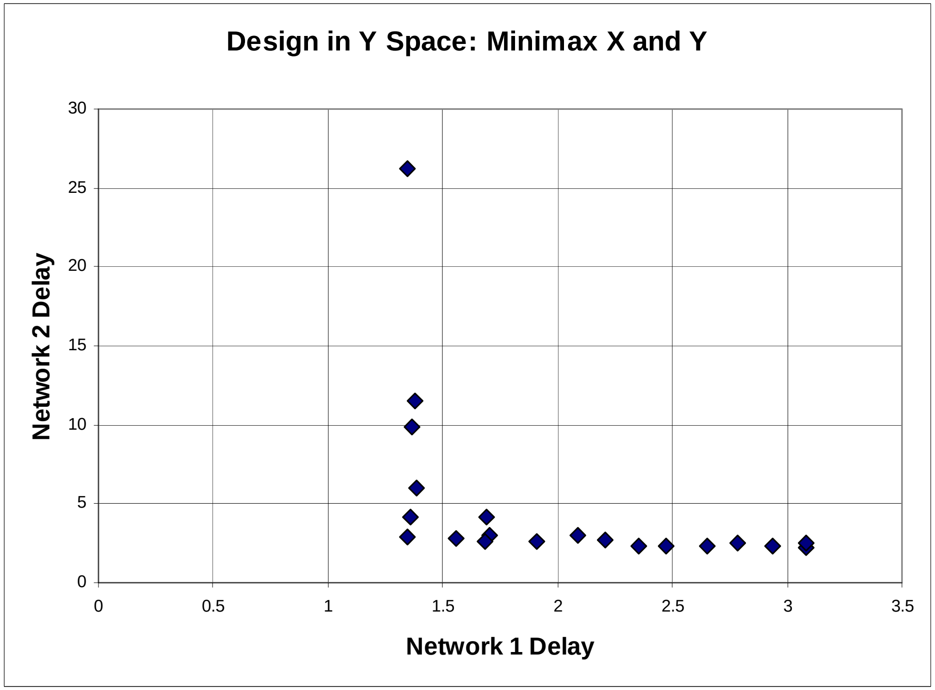


The Benefit of the Combined Design

Xmin = 0.38, Ymin = 0.08

Xmin = 0.14, Ymin = 0.14





Example D-Optimal Strategies

Barton, Meckesheimer and Simpson (2000) Developed Three Combined Forward-Inverse Methods based on D-optimality

<u>Assumptions</u>

- k design parameters = p performance measures
- Budget = N runs
- $N_{min} = (k + 1)(k + 2)/2$ runs are required for quadratic metamodel
- Functions must be invertible over the region of interest
- For Methods 2 and 3, the inverse function cannot be evaluated directly

Example D-Optimal Strategies

Method 1 Assumptions: explicit inverse, requires $N \geq 2N_{min}$ runs

1. Use ***N*/ 2** *D*-optimal forward model runs ($\boldsymbol{X}^{\boldsymbol{D}}$) to generate image points in *y*-space ($\boldsymbol{Y}$)
2. Use ***N*/ 2** *D*-optimal inverse model runs ($\boldsymbol{Y}^{\boldsymbol{D}}$) to generate image points in *x*-space ($\boldsymbol{X}$)
3. Fit the forward metamodel ($\boldsymbol{m}_{\mathbf{f}}$) with $\{\boldsymbol{X}^{\boldsymbol{D}}, \boldsymbol{Y}\} \cup \{\boldsymbol{X}, \boldsymbol{Y}^{\boldsymbol{D}}\}$, for x in $\boldsymbol{X}$ satisfying $x \in C_x$
4. Fit the inverse metamodel ($\boldsymbol{m}_{\mathbf{finv}}$) with $\{\boldsymbol{Y}^{\boldsymbol{D}}, \boldsymbol{X}\} \cup \{\boldsymbol{Y}, \boldsymbol{X}^{\boldsymbol{D}}\}$, for y in $\boldsymbol{Y}$ satisfying $y \in C_y$

Example D-Optimal Strategies

Method 2 Assumption: requires $N \geq 2N_{min}$ runs

- Use ***N*/2** *D*-optimal first phase forward model runs ($\boldsymbol{X}^D$) to generate image points in *y*-space ($\boldsymbol{Y}^1$)
- Fit a first phase inverse approximation ($\boldsymbol{m}_{\mathbf{finv1}}$) with {$\boldsymbol{Y}^1$, $\boldsymbol{X}^D$}
- Use a *D*-optimal design in *y*-space ($\boldsymbol{Y}^D$) to identify the image points ($\boldsymbol{X}^1$) with $\boldsymbol{m}_{\mathbf{finv1}}$
- Use ***N*/2** additional forward model runs ($\boldsymbol{X}^1$) to generate image points in *y*-space ($\boldsymbol{Y}^2$) which replace (and should approximate) the $\boldsymbol{Y}^D$ values
- Fit the final forward metamodel ($\boldsymbol{m}_{\mathbf{f}}$) with {$\boldsymbol{X}^D$, $\boldsymbol{Y}^1$} U {$\boldsymbol{X}^1$, $\boldsymbol{Y}^2$}, for x in $\boldsymbol{X}^1$ satisfying $x \in C_x$
- Fit the final inverse metamodel ($\boldsymbol{m}_{\mathbf{finv}}$) with {$\boldsymbol{Y}^2$, $\boldsymbol{X}^1$} U {$\boldsymbol{Y}^1$, $\boldsymbol{X}^D$}, for y in $\boldsymbol{Y}^1$ and y in $\boldsymbol{Y}^2$ satisfying $x \in C_x$

Example D-Optimal Strategies

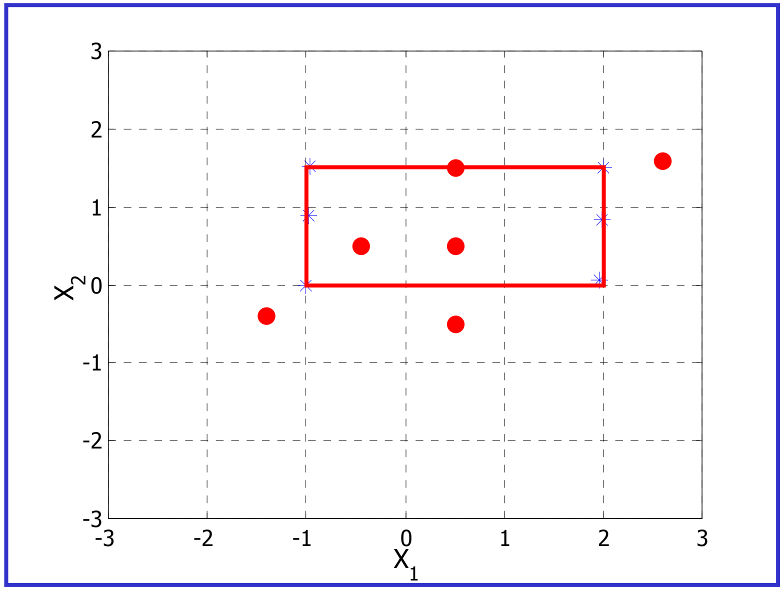
Method 3 Assumption: requires $N \geq N_{min}$ runs

1. Use $N_0 \geq N_{min}$ D-optimal first phase forward runs ($\boldsymbol{X}^D$), generate image points in y-space ($\boldsymbol{Y}^1$)
2. Fit a first phase inverse approximation ($\boldsymbol{m}_{\mathbf{finv1}}$) with {$\boldsymbol{Y}^1$, $\boldsymbol{X}^D$}
3. Construct a $N - N_0$ run D-optimal augmentation to the design in y-space ($\boldsymbol{Y}^D$), using all y in $\boldsymbol{Y}^1$ satisfying $y \in C_y$
4. Approximate the image points ($\boldsymbol{X}^1$) of the augmentation ($\boldsymbol{Y}^D$) with $\boldsymbol{m}_{\mathbf{finv1}}$
5. Use $N - N_0$ additional forward model runs ($\boldsymbol{X}^1$) to generate image points in y-space ($\boldsymbol{Y}^2$) which replace (and should approximate) the $\boldsymbol{Y}^D$ values
6. Fit the final forward metamodel ($\boldsymbol{m}_{\mathbf{f}}$) with {$\boldsymbol{X}^D$, $\boldsymbol{Y}^1$} U {$\boldsymbol{X}^1$, $\boldsymbol{Y}^2$}, for x in $\boldsymbol{X}^1$ satisfying $x \in C_x$
7. Fit the final inverse metamodel ($\boldsymbol{m}_{\mathbf{finv}}$) with {$\boldsymbol{Y}^2$, $\boldsymbol{X}^1$} U {$\boldsymbol{Y}^1$, $\boldsymbol{X}^D$}, for y in $\boldsymbol{Y}^1$ and y in $\boldsymbol{Y}^2$ satisfying $y \in C_y$

Method 3 for $k = p = 2$ and a Quadratic Regression Metamodel

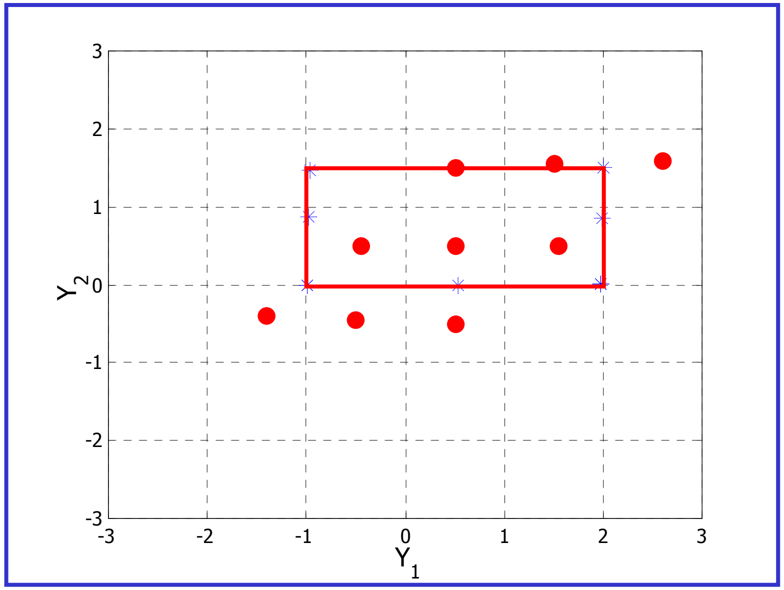
Method 3

12 run case



Optimal inverse metamodel design (6**+3** runs)

18 run case



Optimal inverse metamodel design (7**+4** runs)