Simulation optimization via bootstrapped Kriging: Survey

Jack P.C. Kleijnen

Department of Information Management / Center for Economic Research (CentER)

Tilburg School of Economics & Management (TiSEM)
Tilburg University, Tilburg, Netherlands

Workshop "Stochastic and noisy simulators"
Organizer "GDR MASCOT-NUM“, Paris, 17 May 2011

Overview

Deterministic & *random* simulation models

Focus: Optimization via *Kriging* metamodel

Analysis of Kriging: *Bootstrapping*

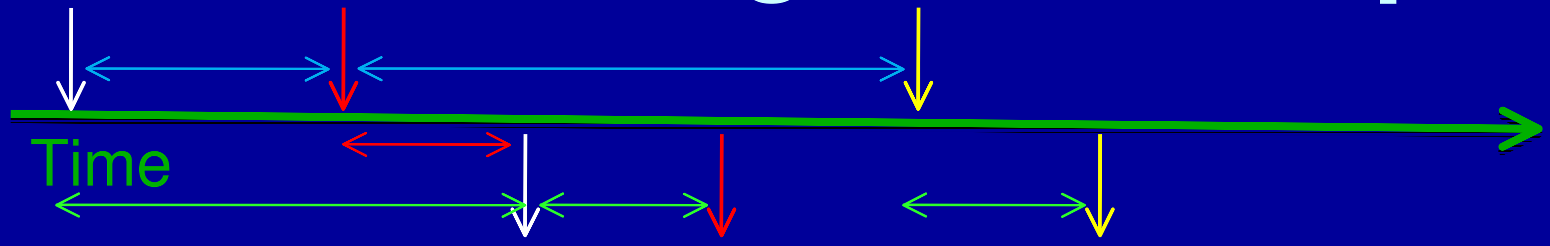
- Random simulation: replicates → *distribution-free* bootstrap
- Deterministic simulation: no replicates → *parametric* bootstrap (multivariate Gaussian); parameters estimated from simulator's I/O

Introduction

Workshop theme: '*Stochastic & noisy* simulators'
3 interpretations:

- *Deterministic* simulation with *numerical noise*
- *Deterministic* simulation with *random input* distribution: 'Uncertainty propagation'
  Epistemic uncertainty
- *Pseudo-Random Numbers* (PRN) in 'Discrete Event Dynamic Systems (DEDS)'
  Aleatory uncertainty
  See next slide

DEDS: Single-server queue



w(i + 1) = w(i) + s(i) – a(i + 1) if positive; else 0

M/M/1 (s & a times Markov / Poisson / exponential):

s(i) = -ln r(2i – 1)] / μ with

E(s) = 1/μ & pseudo-random number 0 < r < 1

a(i + 1) = -ln r(2i) / λ

Start with "empty" system: w(1) = 0

Stop after (say) 100,000 simulated jobs: i = 100,000

Extensions: Servers in parallel or sequence, feedbacks, priorities

Conclusion:

DEDS: Discrete Event Dynamic Systems
(other type: nonlinear difference equations)
Start & stop states of system
Inputs: μ, λ (plus "seed" r(0))

Kriging: Random simulation

1. *Geo*statistics: *Nugget* / *Measurement* error

2. *Deterministic* simulator: Numerical noise

Sub 1 & 2: Y(x) = μ + Z(x) + *e* with GP Z(x) & 'white noise' *e* ~ NIID[0, σ(e)]

3. *Random* simulator: σ(e) → σ[x(i)] & CRN

Predictor for new point x(n+1):

$Y\hat{}[x(n+1)] = \mu + \sigma^2 \Sigma'(n+1)[\Sigma + \Sigma(\bar{e})]^{-1}(\bar{y} - \mathbf{1}\mu)$

with *ybar* average of m(i) replicates of point i

Σ(ebar) diagonal, unless CRN

Not an exact *interpolator* of n averages

Kriging: MLE

Plug-in MLE: *Non-linea*r Kriging predictor

$\hat{\hat{Y}}[x(n+1)] = \hat{\mu} + \hat{\sigma}^2 \hat{r}'[\hat{\Sigma} + \hat{\Sigma}(\bar{e})]^{-1}(\bar{y} - 1\hat{\mu})$

Biased estimator of predictor var., $s^2\{\hat{\hat{Y}}[x(n+1)]\}$

Random simulation: Estimate Σ(e) from replicates

CRN: m > n (# replicates > # combinations)

$s^2\{\hat{\hat{Y}}(x)\}$ and $s^2\{\hat{Y}(x)\}$ are not *max* at same x (see EGO; slide 7)

Parametric bootstrap for $s^2\{Y\hat{\hat{}}(x)\}$

1. *Original* I/O (***X***, ***y***) gives original ***μ***^ and **Σ**^(***θ***^, σ^)
2. *Sample* (y*(1), …, y*(n), y*(n + 1))' from N(***μ***^, **Σ**^) with
   μ^: all (n + 1) elements equal to *μ*^
   Σ^: (n + 1) x (n + 1) matrix with … (see paper)
3. *Bootstrapped* I/O (***X***, ***y****) (see Step 2) gives bootstrap ***μ****^ and **Σ***^(***θ****^, σ*^) (see Step 1)
4. Compute bootstrap *predictor* y*^^(n + 1), using Step 3
5. Compute squared Error *SE* = [y*^^(n + 1) - y*(n + 1)]² (see Steps 4 resp. 2)
6. Repeat Steps 2 -5, B times: $s^{2*} = \Sigma_b SE(b) / B$

Example: Circuit simulator in Sacks et al. (1989)

EI / EGO with bootstrap variance

Local / *global* optima: Exploration / exploitation

Assume: *Deterministic* simulation; single output

1. Find y^o, minimum among n old outputs
2. Find $\mathbf{x}^o$, maximizer $\mathbf{x}$ of
   $EI(\mathbf{x}) = E[y^o - y(\mathbf{x}) \mid y(\mathbf{x}) < y^o]$ with
   $y(\mathbf{x}) \sim N(\hat{\hat{y}}, s^2)$
   Find $\mathbf{x}^o$ via candidate set or Genetic Algorithm
3. Simulate $\mathbf{x}^o$; refit Kriging; go to 1 until $EI \approx 0$

Alternative: *Bootstrap* estimator s^{2*}

Result: Better in 3 of 4 test functions; one tie

Constrained optimization in random simulation

Goal output y(0): Min E[y(0, **x)]**

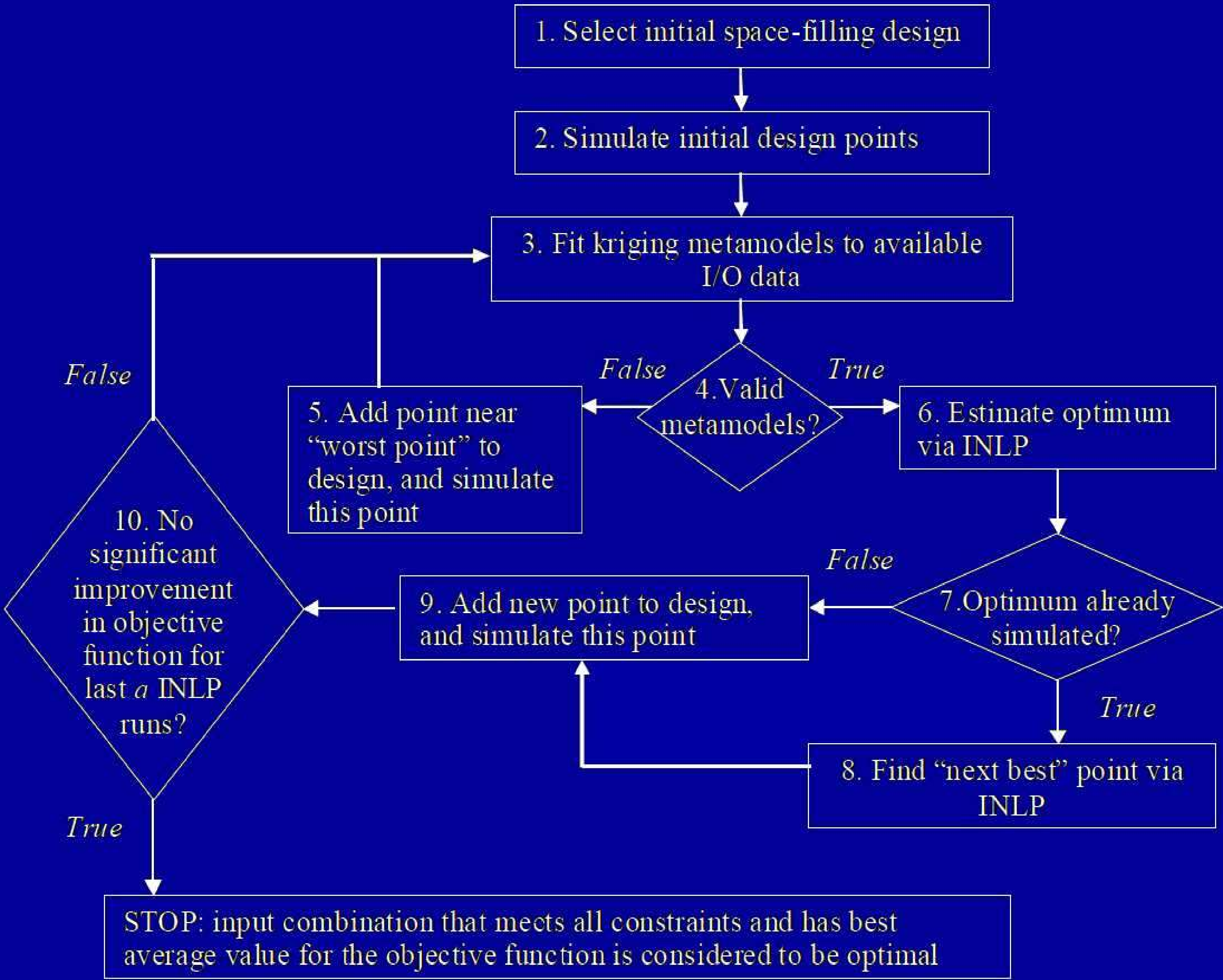
Other r – 1 constrained outputs: E[y(h, **x**)] ≥ c(h)

s constraints for d inputs: f(g)[x(1),…,x(d)] ≥ c(g)

Non-negative integer inputs: x(j) ∈ N

Solution combines (see next slide)

- Sequential DOE (like EGO)
- Kriging (like EGO)
- Integer Non-Linear Programming (INLP)


1. Select initial space-filling design
2. Simulate initial design points
3. Fit kriging metamodels to available I/O data
4.Valid metamodels?
False
True
5. Add point near "worst point" to design, and simulate this point
6. Estimate optimum via INLP
7.Optimum already simulated?
False
True
8. Find "next best" point via INLP
9. Add new point to design, and simulate this point
10. No significant improvement in objective function for last a INLP runs?
False
True
STOP: input combination that meets all constraints and has best average value for the objective function is considered to be optimal

Robust optimization

Taguchi's *worldview*:

Decision inputs (e.g., service rate, order quantity)

Uncertain environmental inputs (e.g., demand)

Taguchi's *methodology* adapted:

Kriging replaces polynomial regression

Bootstrap to quantify Kriging's variability

NLP to estimate *Pareto* frontier

Example:

Deterministic simulator with uncertain demand rate

Minimize expected cost E(C)

such that standard deviation σ(C) ≤ T

Robust optimization: Methodology

1. *Design*: Cross
   Space-filling design for decision inputs *d*
   LHS for environmental inputs *e* with prob. F(e)

n(d) conditional means: wbar(i) = ∑ w(ij) / n(e)

Conditional variance: $s(i)^2$ = ∑ $[(w(ij) – wbar(i)]^2$ / [n(e) -1]

2. *Metamodel*: Kriging metamodel for μ resp. σ
3. Min μ s.t. $\sigma \leq T$: *Mathematical Programming*
4. Vary *T*: *Pareto frontier*
5. Quantify frontier's variability: *Distribution-free bootstrap* (resample n(e) times **w** (n(d)-dimen.)

Monotonic bootstrapped Kriging

Practice: I/O function known to be *monotonic*
Example: Queuing simulation's mean & quantile

Random simulation: *Replication*

Distribution-free bootstrapped Kriging (next slide)

Result: *Confidence intervals* with higher 'coverage' and similar width

Future research:

- Replace classic Kriging by *stochastic* Kriging
- Preserve *convexity* or *nonnegativity*
- *Deterministic* simulator: Parametric bootstrap

Procedure for monotonic Kriging

1. *Resample* -- with replacement -- the m IID original w(i, r): *w*(i, r)* [i = 1, …, n; r = 1, …, m]
2. From w*(i, r) compute the *average* wbar*(i)
3. From (**X**, **wbar***) compute *Kriging* y(θ*)
4. Accept only *monotonically* increasing Kriging: $\nabla y(i)^* > 0$ (i = 1,…,n): Positive *gradients*
5. Repeat B times; *sort* B' predictions y*[**x**(n + 1)]

Point estimator: *Median* of B' predictions

90% confidence interval (CI):

Lower limit: *5% quantile* of B' predictions
Upper limit: *95% quantile* of B'
Asymmetric CI; *positive* lower limit

Conclusions: General

Topic:

Simulation optimization via bootstrap Kriging

Bootstrapping:

1. *Distribution-free*:
   Random simulation with replicates
2. *Parametric*:
   Deterministic simulation (no replicates)
   Multivariate Gaussian with MLE of parameters

Conclusions: Specific topics

- *EGO*: Parametric bootstrap estimator of variance of Kriging predictor with random par.
- *Constrained* opt. in *random* simulation: Distribution-free bootstrap for validation of Kriging model (giving opt. via INLP)
- *Robust* opt. for uncertain environment: Distribution-free bootstrap for variability of Kriging model (giving Pareto frontier via NLP)
- *Monotonic* bootstrapped Kriging

The End

Kriging: Basics

Kriging: *Global* model

$Y(x) = \mu + Z(x)$

with stationary GP $Z(x)$ with zero mean

$corr[Y\{x(i)\}, Y\{x(j)\}] = \prod exp[-\theta(k)\{x(ik) - x(jk)\}^2)$

Linear predictor for point $x(n+1)$:

$Y\hat{}[x(n+1)] = \mu + r'R^{-1}(y-1\mu)$

Exact interpolator: $y\hat{}[x(i)] = y[x(i)]$ with $i = 1, \ldots, n$

Predictor variance:

$$\sigma^2[1- r'R^{-1}r + \{(1 -1'R^{-1}r)^2)/(1'R^{-1}1)\}]$$

Parametric bootstrap: Basics

Data driven statistical method

Examples: Give n IID observations y(i) (i = 1, …,n)

a. Mean E(y) of y(i) ~ Exp(λ)

b. Skewness: ∑_(y(i) – ybar)^3/[(n - 1)s^3]

Sub a:

1. Estimate λ^ = 1/ybar
2. Sample y* from Exp(λ^): *Parametric* bootstrap
3. Estimate mean: y*bar = ∑ y*(i)}/n
4. Repeat Steps 2-3: y*bar(b) (b = 1, …, B)
5. Sort y*bar(b): y*bar(1) < …< y*bar(B)
6. 90% CI for mean: y*bar(0.05B), y*bar(0.95B)

Distribution-free bootstrap: Basics

1. Resample with replacement $y(i)$ gives $y^*(i)$
   Example: $y(1)$ is sampled, 0, 1, …, n times
2. Estimate mean: $y^*bar = \sum y^*(i)\}/n$
3. Repeat Steps 2-3: $y^*bar(b)$ ($b = 1, \ldots, B$)
4. Sort $y^*bar(b)$: $y^*bar(1) < \ldots < y^*bar(B)$
5. 90% CI for mean: $y^*bar(0.05B)$, $y^*bar(0.95B)$

Bootstrap: Applications

Bootstrap: simple idea; yet, “art” of modeling

1. CI for estimated skewness (Example 2)
2. Validation of simulation models
3. Ranking of journals on quality (citations)
4. See next slides

Constrained optimization: details

- “Enough” replicates per point; see Law (2007)
- CRN
- Fit Kriging to averages

Global/local: Steps 5 /9 in next flowchart

Cross-validate n(cv) points incl. bootstrap: Step 4
Complications in bootstrap:
Multiple outputs, non-constant m(i), CRN

Studentized prediction error: Divide by √ of bootstrapped variance + replicate var. estimate

Apply *Bonferroni*’s inequality: $\alpha/[r \times n(cv)]$

Constrained optimization: details

Step 5: If Kriging is rejected, then add point halfway worst point and nearest neighbor

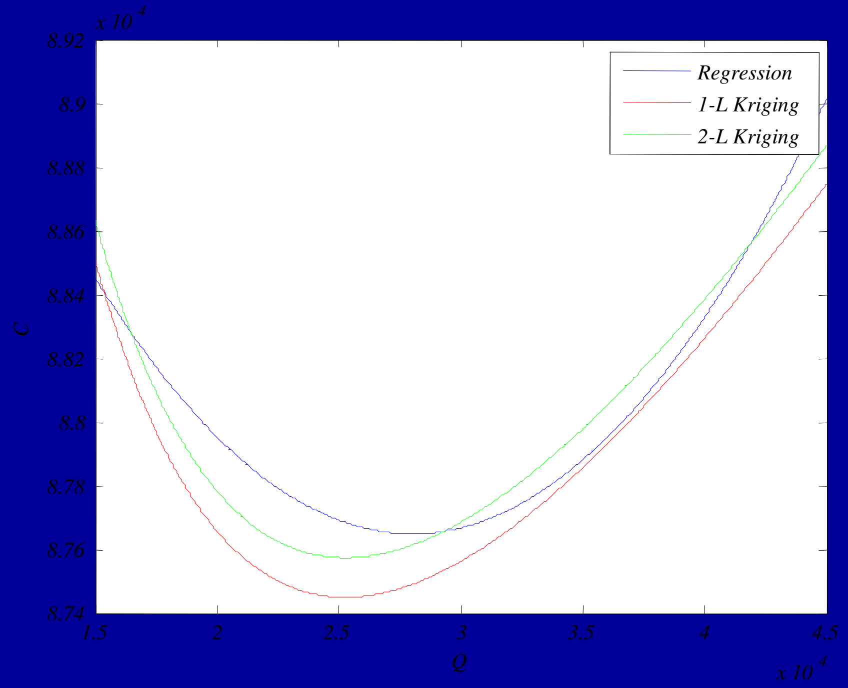
Adapt for

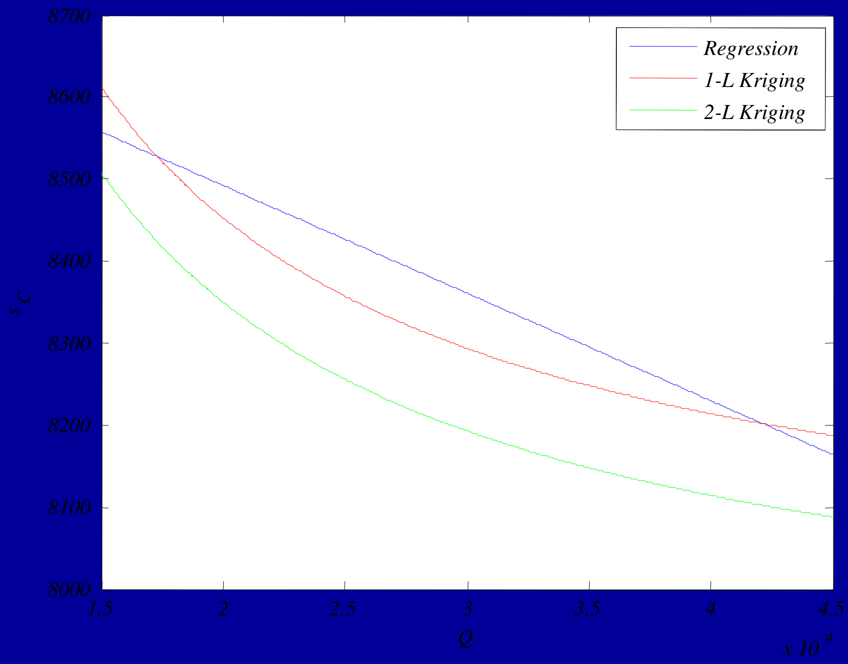
- *continuous* inputs, incl. *gradients*
- *deterministic* outputs

Applications:

- Academic *inventory* system (s, S)
- Realistic *call center*
- 'Better' than *OptQuest* (in Arena)

Robust optimization: Example





Robust optimization: *Future* research

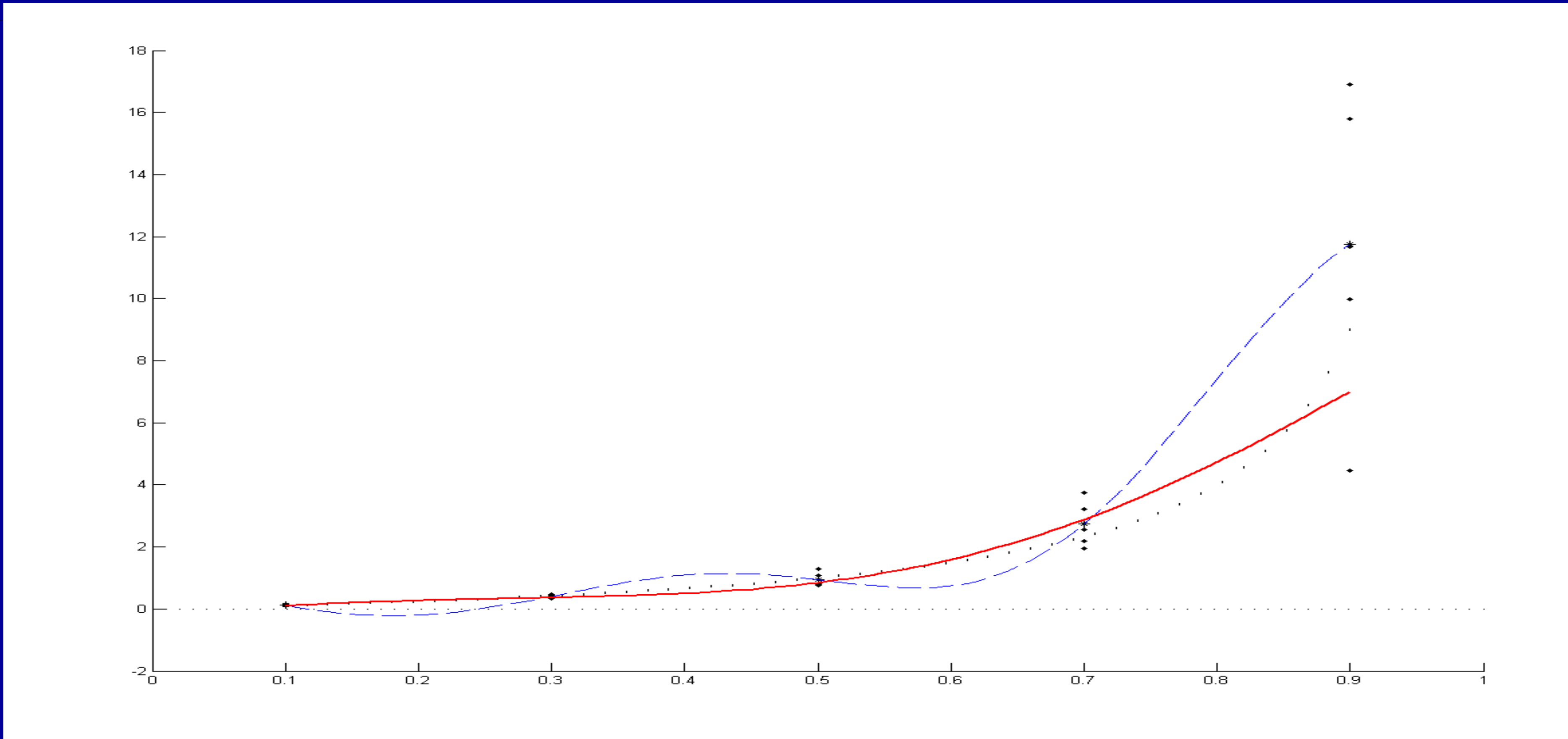
Replace Min μ s.t. $\sigma \leq T$ by quantile or CVaR

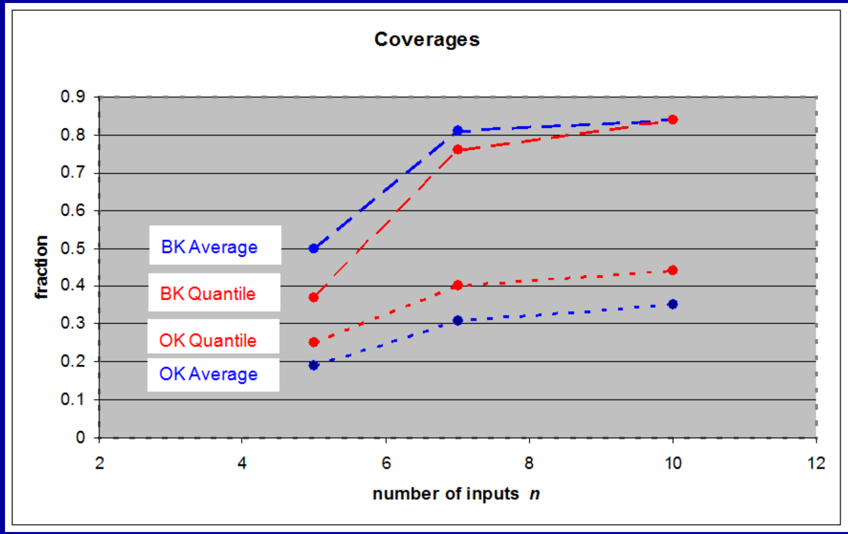
Random (s, S): aleatory & epistemic uncertainty

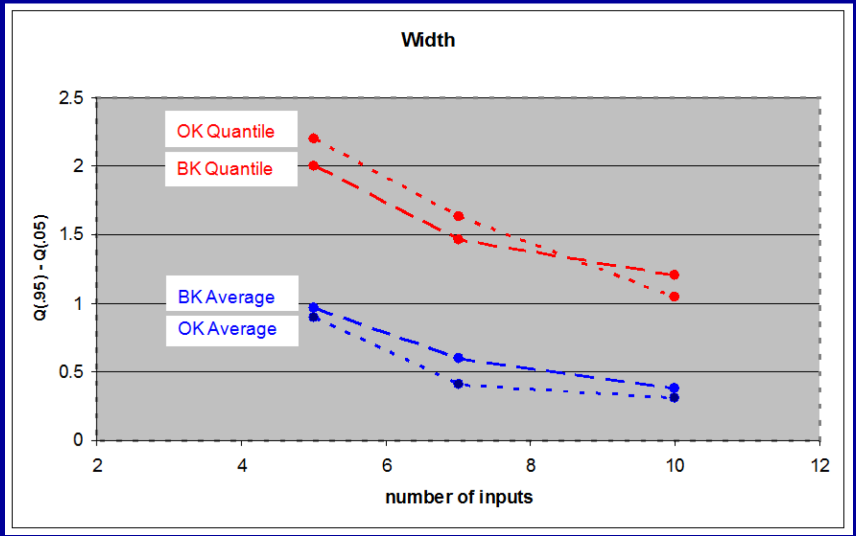
Multiple constrained outputs

M/M/1 example: Wiggling versus monotonic Kriging

m = 5; no CRN; wbar(i) < wbar(I + 1);
Gaussian correlation function




Coverages
fraction
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
BK Average
BK Quantile
OK Quantile
OK Average
2
4
6
8
10
12
number of inputs n


Width
Q(.95) - Q(.05)
2.5
2
1.5
1
0.5
0
OK Quantile
BK Quantile
BK Average
OK Average
2
4
6
8
10
12
number of inputs