A Scalable Approach to Gradient-Enhanced Stochastic Kriging

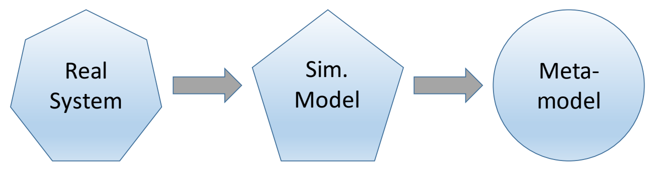
Haojun Huo[†], Xiaowei Zhang[*], and Zeyu Zheng[‡]

[†] Hong Kong University of Science and Technology, IEDA
[*] City University of Hong Kong, MS
[‡] UC Berkeley, IEOR

Table of Contents

Stochastic Kriging and Big n Problem

Metamodeling



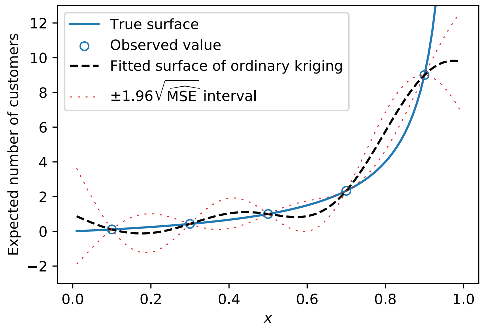
- Simulation models are often computationally expensive
- Metamodel: fast approximation of simulation model
  - Run simulation at a small number of design points
  - Predict responses based on the simulation outputs

Stochastic Kriging

- Also called Gaussian process (GP) regression
- Unknown surface is modeled as a Gaussian process

$$\mathrm{Z}(\boldsymbol{x}) = \beta + \mathrm{M}(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^d$$

- $\mathrm{M}(\boldsymbol{x})$ is characterized by covariance function $k(\boldsymbol{x}, \boldsymbol{y})$
- Leverage *spatial correlation* for prediction



Partial Literature

- Quantification of input uncertainty
    - Barton, Nelson, and Xie (2014)
    - Xie, Nelson, and Barton (2014)
- Simulation/black-box/Bayesian optimization
    - Huang et al. (2006)
    - Sun, Hong, and Hu (2014)
    - Scott, Frazier, and Powell (2011)
    - Shahriari et al. (2016)

The Big n Problem

- Response surface is observed at $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ with noise

$$z(\boldsymbol{x}_i) = \beta + \mathrm{M}(\boldsymbol{x}_i) + \varepsilon(\boldsymbol{x}_i)$$

- Best linear unbiased predictor of $\mathrm{Z}(\boldsymbol{x}_0)$

$$\widehat{\mathrm{Z}}(\boldsymbol{x}_0) = \beta + \boldsymbol{\Sigma}_{\mathrm{M}}(\boldsymbol{x}_0, \cdot)[\boldsymbol{\Sigma}_{\mathrm{M}} + \boldsymbol{\Sigma}_{\varepsilon}]^{-1}[\overline{\boldsymbol{z}} - \beta \mathbf{1}_n]$$

- Maximum likelihood estimation

$$\max_{\beta, \boldsymbol{\theta}} \Big\{ -\log[\det(\boldsymbol{\Sigma}_{\mathrm{M}} + \boldsymbol{\Sigma}_{\varepsilon})] - [\overline{\boldsymbol{z}} - \beta \mathbf{1}_n]^{\intercal}[\boldsymbol{\Sigma}_{\mathrm{M}} + \boldsymbol{\Sigma}_{\varepsilon}][\overline{\boldsymbol{z}} - \beta \mathbf{1}_n] \Big\}$$

- Slow: $[\boldsymbol{\Sigma}_{\mathrm{M}} + \boldsymbol{\Sigma}_{\varepsilon}] \in \mathbb{R}^{n \times n}$ and inverting it takes $\mathcal{O}(n^3)$ time
- Numerically unstable: $[\boldsymbol{\Sigma}_{\mathrm{M}} + \boldsymbol{\Sigma}_{\varepsilon}]$ is often nearly singular
    - Especially for the popular Gaussian covariance function
    - Usually run into trouble when $n > 100$, which can easily happen when $d \geq 3$

Enhancing SK with Gradient Information

- j-th run of the simulation model at $\boldsymbol{x}_i$ produces
    - response estimate $z_j(\boldsymbol{x}_i)$
    - gradient estimate $\boldsymbol{g}_j(\boldsymbol{x}_i) = (g_j^1(\boldsymbol{x}_i), \ldots, g_j^d(\boldsymbol{x}_i))^\intercal$

      $$g_j^r(\boldsymbol{x}_i) = G^r(\boldsymbol{x}_i) + \delta_j^r(\boldsymbol{x}_i), \quad r = 1, \ldots, d,$$

      where $G^r(\boldsymbol{x}_i)$ is the true r-th partial derivative
- Predict $\mathsf{Z}(\boldsymbol{x}_0)$ using both response estimates and gradient estimates
    - Qu and Fu (2014): *gradient extrapolated stochastic kriging* (GESK); simple, using gradients indirectly
    - Chen, Ankenman, and Nelson (2013): *stochastic kriging with gradient estimators* (SKG); sophisticated, using gradients directly

GESK (Qu and Fu, 2014)

- Use gradient estimates to create "pseudo" response estimates

$$z_j(\tilde{\boldsymbol{x}}_i) \approx z_j(\boldsymbol{x}_i) + \boldsymbol{g}_j(\boldsymbol{x}_i)^\intercal \Delta \boldsymbol{x}_i,$$

  where $\tilde{\boldsymbol{x}}_i = \boldsymbol{x}_i + \Delta \boldsymbol{x}_i$

- $\Delta \boldsymbol{x}_i$: the direction and step size of the linear extrpolation
- Predict $\mathsf{Z}(\boldsymbol{x}_0)$ using the augmented data

$$(\bar{z}(\boldsymbol{x}_1), \ldots, \bar{z}(\boldsymbol{x}_n),\ \bar{z}(\tilde{\boldsymbol{x}}_1), \ldots, \bar{z}(\tilde{\boldsymbol{x}}_n))$$

- The size of the covariance matrix now becomes $2n \times 2n$
- One could create d pseudo response estimates at each $\boldsymbol{x}_i$, resulting in inverting a matrix of size $(d+1)n \times (d+1)n$
- Similar problem for SKG

Approximation Schemes

- Well developed in spatial statistics and machine learning
  - Banerjee et al. (2015)
  - Rasmussen and Williams (2006)
- Reduced-rank approximations: emphasize long-range dependences
- Sparse approximations: emphasize short-range dependences

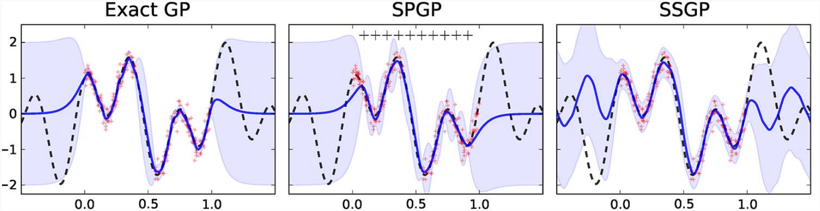


Figure 1: Posterior means and variances. Source: Shahriari et al. (2016)

Approximation-free?

Markovian Covariance Functions

Gaussian Markov Random Field (GMRF)

- M is multivariate normal with sparsity specified on $\boldsymbol{\Sigma}_{\mathrm{M}}^{-1}$
- A discrete model, using graph to describe Markovian structure
  - Given all its neighbors, node i is *conditionally independent* of its non-neighbors
  - E.g., $\mathrm{M}(\boldsymbol{x}_2) \perp (\mathrm{M}(\boldsymbol{x}_0), \mathrm{M}(\boldsymbol{x}_4))$, given $(\mathrm{M}(\boldsymbol{x}_1), \mathrm{M}(\boldsymbol{x}_3))$
  - $\boldsymbol{\Sigma}_{\mathrm{M}}^{-1}(i, j) \neq 0 \iff i$ and j are neighbors

- The sparsity can reduce necessary computation to $\mathcal{O}(n^2)$

Disadvantages

- Has no explicit expression for the covariances
- Cannot predict locations "off the grid"

$$\widehat{\mathsf{Z}}(\boldsymbol{x}_0) = \beta + \underbrace{\boldsymbol{\Sigma}_{\mathsf{M}}(\boldsymbol{x}_0, \cdot)}_{\text{unknown}}[\boldsymbol{\Sigma}_{\mathsf{M}} + \boldsymbol{\Sigma}_{\varepsilon}]^{-1}[\overline{\boldsymbol{z}} - \beta \mathbf{1}_n]$$

Markovian Covariance Function: Best of Two Worlds?

- Construct a class of covariance functions for which:
    1. $\boldsymbol{\Sigma}_{\mathrm{M}}$ can be inverted analytically
    2. $\boldsymbol{\Sigma}_{\mathrm{M}}^{-1}$ is sparse
- Explicit link between covariance function and sparsity

Definition (1-d MCF)

Let p and q be two positive continuous functions that satisfy $p(x)q(y) - p(y)q(x) < 0$ for all $x < y$. Then, $k(x, y) = p(x)q(y)\,\mathbb{I}_{\{x \le y\}} + p(y)q(x)\,\mathbb{I}_{\{x > y\}}$ is called a 1-d MCF.

- Brownian motion: $k_{\mathrm{BM}}(x, y) = x\,\mathbb{I}_{\{x \le y\}} + y\,\mathbb{I}_{\{x > y\}}$
- Brownian bridge: $k_{\mathrm{BR}}(x, y) = x(1 - y)\,\mathbb{I}_{\{x \le y\}} + y(1 - x)\,\mathbb{I}_{\{x > y\}}$
- OU process: $k_{\mathrm{OU}}(x, y) = e^{x}e^{-y}\,\mathbb{I}_{\{x \le y\}} + e^{y}e^{-x}\,\mathbb{I}_{\{x > y\}}$

Markovian Covariance Function

- $\{x_1, \ldots, x_n\}$ are not necessarily equally spaced

Theorem (Ding and Z. 2018)

$\boldsymbol{K}^{-1}$ is tridiagonal and its nonzero entries are

$$(\boldsymbol{K}^{-1})_{i,i} = \begin{cases} \dfrac{p_2}{p_1(p_2 q_1 - p_1 q_2)}, & \text{if } i = 1, \\ \dfrac{p_{i+1} q_{i-1} - p_{i-1} q_{i+1}}{(p_i q_{i-1} - p_{i-1} q_i)(p_{i+1} q_i - p_i q_{i+1})}, & \text{if } 2 \leq i \leq n-1, \\ \dfrac{q_{n-1}}{q_n(p_n q_{n-1} - p_{n-1} q_n)}, & \text{if } i = n, \end{cases}$$

and

$$(\boldsymbol{K}^{-1})_{i-1,i} = (\boldsymbol{K}^{-1})_{i,i-1} = \frac{-1}{p_i q_{i-1} - p_{i-1} q_i}, \quad i = 2, \ldots, n.$$

Reduction in Complexity

- Woodbury matrix identity

$$[\boldsymbol{\Sigma}_{\mathsf{M}} + \boldsymbol{\Sigma}_{\varepsilon}]^{-1} = \underbrace{\boldsymbol{\Sigma}_{\mathsf{M}}^{-1}}_{\text{known}} + \underbrace{\boldsymbol{\Sigma}_{\mathsf{M}}^{-1}}_{\text{sparse}} \left[\underbrace{\boldsymbol{\Sigma}_{\mathsf{M}}^{-1} + \boldsymbol{\Sigma}_{\varepsilon}^{-1}}_{\text{sparse}} \right]^{-1} \boldsymbol{\Sigma}_{\mathsf{M}}^{-1}$$

  - inversion: $\mathcal{O}(n^2)$
  - multiplications: $\mathcal{O}(n^2)$
  - addition: $\mathcal{O}(n^2)$

- It takes $\mathcal{O}(n^2)$ time to compute BLUP

$$\widehat{Z}(\boldsymbol{x}_0) = \beta + \underbrace{\boldsymbol{\Sigma}_{\mathsf{M}}(\boldsymbol{x}_0, \cdot)}_{\text{known}} [\boldsymbol{\Sigma}_{\mathsf{M}} + \boldsymbol{\Sigma}_{\varepsilon}]^{-1} [\overline{\boldsymbol{z}} - \beta \mathbf{1}_n]$$

  - If the noise is negligible ($\boldsymbol{\Sigma}_{\varepsilon} \approx \mathbf{0}$), then no numerical inversion is needed and computing BLUP is $\mathcal{O}(n)$!

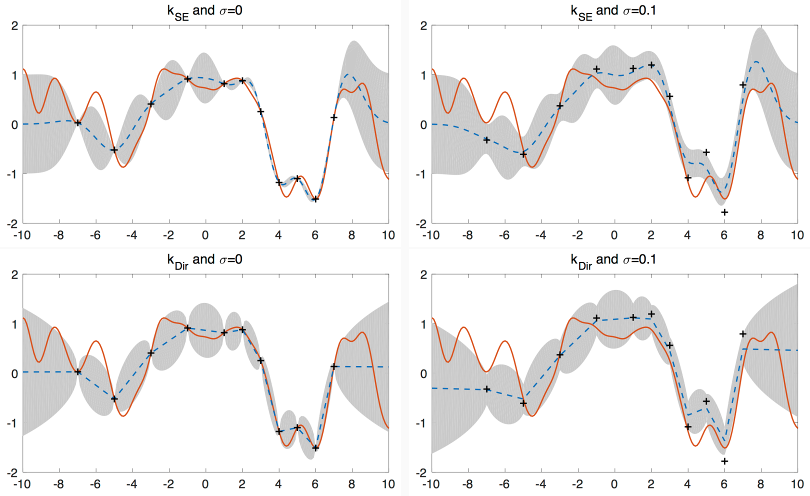
Improvement in Stability

1. $\boldsymbol{\Sigma}_{\mathsf{M}}$ can be made much better conditioned
2. Woodbury also improves numerical stability

$$[\boldsymbol{\Sigma}_{\mathsf{M}} + \boldsymbol{\Sigma}_{\varepsilon}]^{-1} = \boldsymbol{\Sigma}_{\mathsf{M}}^{-1} + \boldsymbol{\Sigma}_{\mathsf{M}}^{-1}\left[\boldsymbol{\Sigma}_{\mathsf{M}}^{-1} + \boldsymbol{\Sigma}_{\varepsilon}^{-1}\right]^{-1}\boldsymbol{\Sigma}_{\mathsf{M}}^{-1}$$

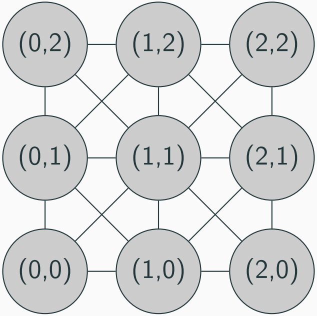
- The diagonal entries of $\boldsymbol{\Sigma}_{\varepsilon}^{-1}$ are often large

Uncertainty Quantification



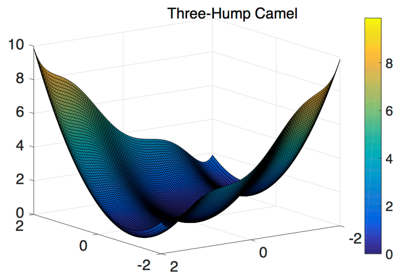
Extension for $d > 1$

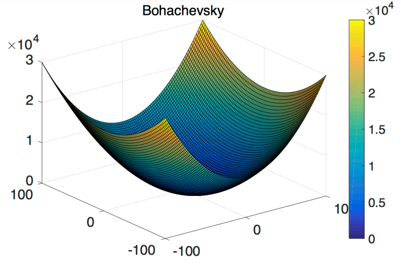
- Product form: $k(\boldsymbol{x}, \boldsymbol{y}) = \prod_{i=1}^{d} k_i(x^i, y^i)$
- Limitation: $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ must form a regular lattice
- Then, $\boldsymbol{K} = \bigotimes_{i=1}^{d} \boldsymbol{K}_i$ and $\boldsymbol{K}^{-1} = \bigotimes_{i=1}^{d} \boldsymbol{K}_i^{-1}$, preserving sparsity



Two-Dimensional Response Surfaces

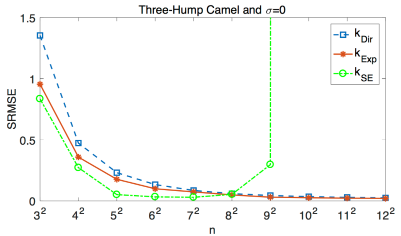
Function Name	Expression
Three-Hump Camel	$Z(x, y) = 2x^2 - 1.05x^4 + \frac{x^6}{6} + xy + y^2$
Bohachevsky	$Z(x, y) = x^2 + 2y^2 - 0.3\cos(3\pi x) - 0.4\cos(4\pi y) + 0.7$

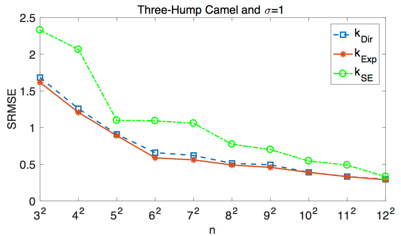


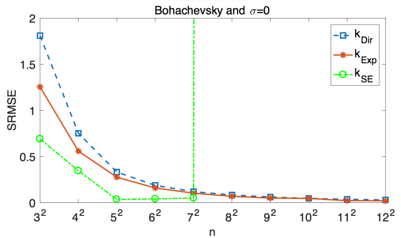


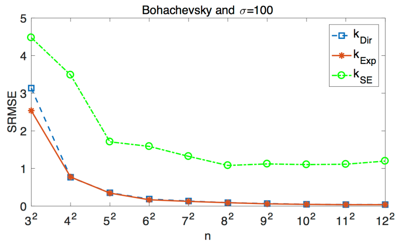
Prediction Accuracy

- Standardized RMSE $= \dfrac{\sqrt{\sum_{i=1}^{K}\left[Z(\boldsymbol{x}_i)-\hat{Z}(\boldsymbol{x}_i)\right]^2}}{\sqrt{\sum_{i=1}^{K}\left[Z(\boldsymbol{x}_i)-K^{-1}\sum_{h=1}^{K} Z(\boldsymbol{x}_h)\right]^2}}$



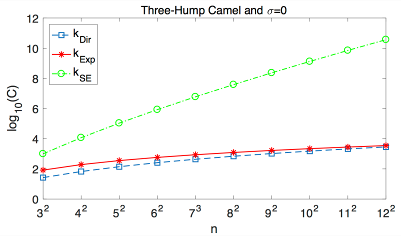


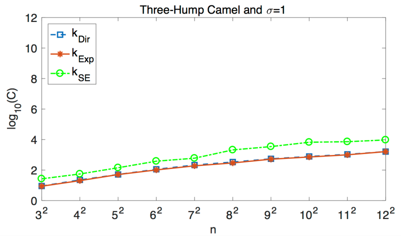




Condition Number of $\Sigma_M + \Sigma_\varepsilon$

- $C = \lambda_{\max}(\boldsymbol{K})/\lambda_{\min}(\boldsymbol{K})$ measures "closeness to singularity"

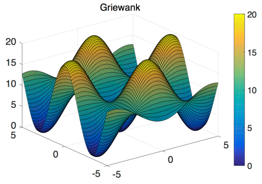


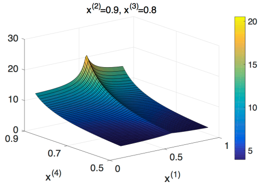


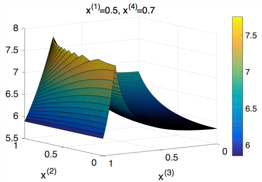
Scalability Demonstration

- 4-d Griewank func.: $Z(\boldsymbol{x}) = \sum_{i=1}^{4} \left(\frac{x^{(i)}}{20}\right)^2 - 10 \prod_{i=1}^{D} \cos\left(\frac{x^{(i)}}{\sqrt{i}}\right) + 10$
- Mean cycle time of a N-station Jackson network with D different types of arrivals (Yang et al. 2011): $N = D = 4$

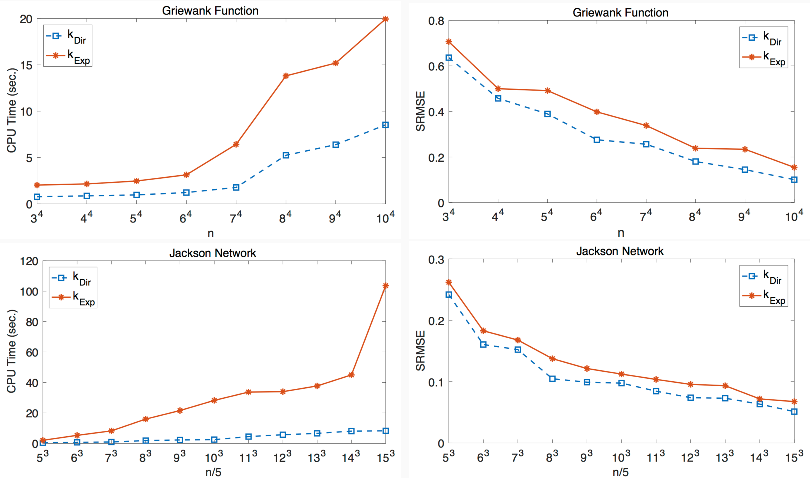
$$\mathbb{E}[\mathrm{CT}_1] = \sum_{j=1}^{N} \frac{\delta_{1j}}{\mu_j \left[1 - \rho\left(\frac{\sum_{i=1}^{D} \alpha_i \delta_{ij}/\mu_j}{\max_h \sum_{i=1}^{D} \alpha_i \delta_{ih}/\mu_h}\right)\right]}$$







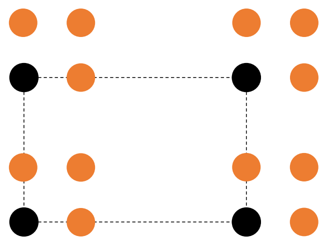
Computational Efficiency



Scalable Gradient Extrapolated Stochastic Kriging

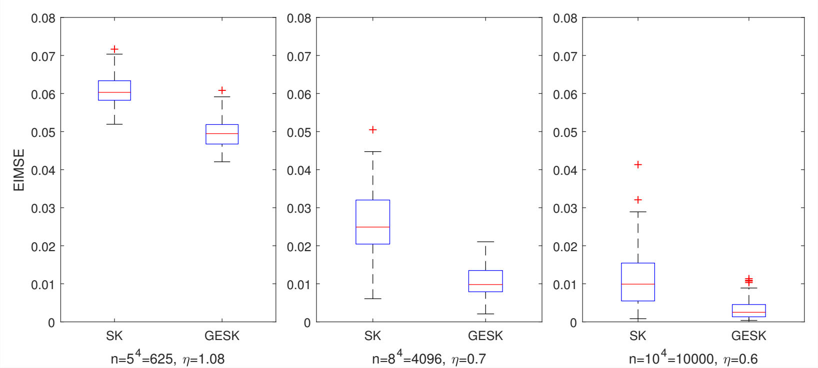
Enhancing Scalability of GESK with MCFs

- GESK creates an augmented set of response estimates for SK
- MCFs can be applied if the design points form a regular lattice of size $n = n_1 \times n_2 \times \cdots n_d$



- Result in $2^d n$ points in the augmented dataset
- Σ_{M} has size $2^d n \times 2^d n$ but we can leverage the Kronecker product to reduce its inversion to inverting d much smaller matrices, each having size $2n_r \times 2n_r$

Numerical Illustration



- 4-dimensional Griewank function
- Can manage $n = 10^4$ design points

Conclusions

Remarks on MCFs

- Allow modeling association directly, while retaining sparsity in the precision matrix
- Improve the scalability of SK so that it can be used for simulation models with a high-dimensional design space
  - Reduce computational cost from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$ without approx.
  - Further reduce to $\mathcal{O}(n)$ if observations are noise-free
  - Enhance numerical stability substantially
- Limitation: design points must form a regular lattice, though not necessarily equally spaced

Remarks on Gradient Enhanced SK

- GESK (Qu and Fu, 2014) can easily benefit from MCFs
- But there are two issues
  - Extrapolation error is hard to characterize
  - Each design point needs $(2^d - 1)$ pseudo response estimates, a great deal of redundancy in using gradient info
- SKG (Chenn, Ankenman, and Nelson, 2013) does not incur such computational overhead, but requires calculating the gradient surface of the Gaussian process (on-going work)

Markovian covariances without approx.

v.s.

Good approx. for all covariances